

Shor's Quantum Factoring Algorithm

Samuel J. Lomonaco, Jr.

ABSTRACT. This paper is a written version of a one hour lecture given on Peter Shor's quantum factoring algorithm. It is based on [4], [6], [7], [9], and [15].

CONTENTS

1. Preamble to Shor's algorithm
2. Number theoretic preliminaries
3. Overview of Shor's algorithm
4. Preparations for the quantum part of Shor's algorithm
5. The quantum part of Shor's algorithm
6. Peter Shor's stochastic source \mathcal{S}
7. A momentary digression: Continued fractions
8. Preparation for the final part of Shor's algorithm
9. The final part of Shor's algorithm
10. An example of Shor's algorithm

References

1. Preamble to Shor's algorithm

2000 *Mathematics Subject Classification*. Primary 81P68, 81-01, 81-02, 68Q25, 68W40; Secondary 68Q05.

Key words and phrases. Shor's algorithm, factoring, quantum computation, quantum algorithms, algorithms, quantum mechanics.

This work was partially supported by Army Research Office (ARO) Grant #P-38804-PH-QC, by the National Institute of Standards and Technology (NIST), by the Defense Advanced Research Projects Agency (DARPA) and Air Force Materiel Command USAF under agreement number F30602-01-0522, and by the L-O-O-P Fund. The author gratefully acknowledges the hospitality of the University of Cambridge Isaac Newton Institute for Mathematical Sciences, Cambridge, England, where some of this work was completed. I would also like to thank the other AMS Short Course lecturers, Howard Brandt, Dan Gottesman, Lou Kauffman, Alexei Kitaev, Peter Shor, Umesh Vazirani and the many Short Course participants for their support. (Copyright 2000.)

©0000 (copyright holder)

There are cryptographic systems (such as RSA¹) that are extensively used today (e.g., in the banking industry) which are based on the following questionable assumption, i.e., conjecture:

Conjecture(Assumption). *Integer factoring is computationally much harder than integer multiplication. In other words, while there are obviously many polynomial time algorithms for integer multiplication, there are no polynomial time algorithms for integer factoring. I.e., integer factoring computationally requires super-polynomial time.*

This assumption is based on the fact that, in spite of the intensive efforts over many centuries of the best minds to find a polynomial time factoring algorithm, no one has succeeded so far. As of this writing, the most asymptotically efficient *classical* algorithm is the number field sieve [10], [11], which is heuristically estimated² to factor an integer N in time $O\left(\exp\left[c(\lg N)^{1/3}(\lg \lg N)^{2/3}\right]\right)$. Thus, this is a super-polynomial time algorithm in the number $O(\lg N)$ of digits in N .

However, ... Peter Shor suddenly changed the rules of the game.

Hidden in the above conjecture is the unstated, but implicitly understood, assumption that all algorithms run on computers based on the principles of classical mechanics, i.e., on **classical computers**. But what if a computer could be built that is based not only on classical mechanics, but on quantum mechanics as well? I.e., what if we could build a **quantum computer**?

Shor, starting from the works of Benioff, Bennett, Deutsch, Feynman, Simon, and others, created an algorithm to be run on a quantum computer, i.e., a **quantum algorithm**, that factors integers in polynomial time! Shor's algorithm takes asymptotically $O\left((\lg N)^2(\lg \lg N)(\lg \lg \lg N)\right)$ steps on a quantum computer, which is polynomial time in the number of digits $O(\lg N)$ of N .

2. Number theoretic preliminaries

Since the time of Euclid, it has been known that every positive integer N can be uniquely (up to order) factored into the product of primes. Moreover, it is a computationally easy (polynomial time) task to determine whether or not N is a prime or composite number. For the primality testing algorithm of Miller-Rabin[14] makes such a determination at the cost of $O(s \lg N)$ arithmetic operations [$O(s \lg^3 N)$ bit operations] with probability of error $Prob_{Error} \leq 2^{-s}$.

¹RSA is a public key cryptographic system invented by Rivest, Shamir, Adleman. Hence the name. For more information, please refer to [17].

²This heuristic estimate is based on unproven theorems in number theory. This estimate appears to match reasonably well with experimentally observed running times.

However, once an odd positive integer N is known to be composite, it does not appear to be an easy (polynomial time) task on a classical computer to determine its prime factors. As mentioned earlier, so far the most asymptotically efficient *classical* algorithm known is the number field sieve [10], [11], which is heuristically estimated to factor an integer N in time $O\left(c \exp\left[(\lg N)^{1/3} (\lg \lg N)^{2/3}\right]\right)$.

Prime Factorization Problem. *Given a composite odd positive integer N , find its prime factors.*

It is well known[14] that factoring N can be reduced to the task of choosing at random an integer m relatively prime to N , and then determining its modulo N multiplicative order P , i.e., to finding the smallest positive integer P such that

$$m^P = 1 \pmod{N} .$$

It was precisely this approach to factoring that enabled Shor to construct his factoring algorithm.

3. Overview of Shor's algorithm

But what is Shor's quantum factoring algorithm?

Let $\mathbb{N} = \{0, 1, 2, 3, \dots\}$ denote the set of natural numbers.

Shor's algorithm provides a solution to the above problem. His algorithm consists of the five steps (**steps 1** through **5**), with only **STEP 2** requiring the use of a quantum computer. The remaining four other steps of the algorithm are to be performed on a classical computer.

We begin by briefly describing all five steps. After that, we will then focus in on the quantum part of the algorithm, i.e., **STEP 2**.

Step 1. Choose a random positive integer m . Use the polynomial time Euclidean algorithm³ to compute the greatest common divisor $\gcd(m, N)$ of m and N . If the greatest common divisor $\gcd(m, N) \neq 1$, then we have found a non-trivial factor of N , and we are done. If, on the other hand, $\gcd(m, N) = 1$, then proceed to **STEP 2**.

³The Euclidean algorithm is $O(\lg^2 N)$. For a description of the Euclidean algorithm, see for example [3] or [2].

STEP 2. Use a QUANTUM COMPUTER to determine the unknown period P of the function

$$\begin{array}{ccc} \mathbb{N} & \xrightarrow{f_N} & \mathbb{N} \\ a & \mapsto & m^a \bmod N \end{array}$$

Step 3. If P is an odd integer, then go to **Step 1**. [The probability of P being odd is $(\frac{1}{2})^k$, where k is the number of distinct prime factors of N .] If P is even, then proceed to **Step 4**.

Step 4. Since P is even,

$$(m^{P/2} - 1)(m^{P/2} + 1) = m^P - 1 = 0 \bmod N .$$

If $m^{P/2} + 1 = 0 \bmod N$, then go to **Step 1**. If $m^{P/2} + 1 \neq 0 \bmod N$, then proceed to **Step 5**. It can be shown that the probability that $m^{P/2} + 1 = 0 \bmod N$ is less than $(\frac{1}{2})^{k-1}$, where k denotes the number of distinct prime factors of N .

Step 5. Use the Euclidean algorithm to compute $d = \gcd(m^{P/2} - 1, N)$. Since $m^{P/2} + 1 \neq 0 \bmod N$, it can easily be shown that d is a non-trivial factor of N . Exit with the answer d .

Thus, the task of factoring an odd positive integer N reduces to the following problem:

Problem. Given a periodic function

$$f : \mathbb{N} \longrightarrow \mathbb{N} ,$$

find the period P of f .

4. Preparations for the quantum part of Shor's algorithm

Choose a power of 2

$$Q = 2^L$$

such that

$$N^2 \leq Q = 2^L < 2N^2 ,$$

and consider f restricted to the set

$$S_Q = \{0, 1, \dots, Q - 1\}$$

which we also denote by f , i.e.,

$$f : S_Q \longrightarrow S_Q .$$

In preparation for a discussion of **STEP 2** of Shor's algorithm, we construct an L -qubit quantum registers REGISTER1 and a $\lceil \lg N \rceil$ -qubit quantum REGISTER2 to hold respectively the arguments and the values of the function f , i.e.,

$$|\text{REG1}\rangle |\text{REG2}\rangle = |a\rangle |f(a)\rangle = |a\rangle |b\rangle = |a_0 a_1 \cdots a_{L-1}\rangle |b_0 b_1 \cdots b_{\lceil \lg N \rceil - 1}\rangle$$

In doing so, we have adopted the following convention for representing integers in these registers:

Notation Convention. In a quantum computer, we represent an integer a with radix 2 representation

$$a = \sum_{j=0}^{L-1} a_j 2^j,$$

as a quantum register consisting of the 2^n qubits

$$|a\rangle = |a_0 a_1 \cdots a_{L-1}\rangle = \bigotimes_{j=0}^{L-1} |a_j\rangle$$

For example, the integer 23 is represented in our quantum computer as n qubits in the state:

$$|23\rangle = |10111000 \cdots 0\rangle$$

Before continuing, we remind the reader of the classical definition of the Q -point Fourier transform.

DEFINITION 1. Let ω be a primitive Q -th root of unity, e.g., $\omega = e^{2\pi i/Q}$. Then the Q -point Fourier transform is the map

$$\begin{aligned} \text{Map}(S_Q, \mathbb{C}) &\xrightarrow{\mathcal{F}} \text{Map}(S_Q, \mathbb{C}) \\ [f : S_Q \rightarrow \mathbb{C}] &\mapsto [\hat{f} : S_Q \rightarrow \mathbb{C}] \end{aligned}$$

where

$$\hat{f}(y) = \frac{1}{\sqrt{Q}} \sum_{x \in S_Q} f(x) \omega^{xy}$$

We implement the Fourier transform \mathcal{F} as a unitary transformation, which in the standard basis

$$|0\rangle, |1\rangle, \dots, |Q-1\rangle$$

is given by the $Q \times Q$ unitary matrix

$$\mathcal{F} = \frac{1}{\sqrt{Q}} (\omega^{xy}).$$

This unitary transformation can be factored into the product of $O(\lg^2 Q) = O(\lg^2 N)$ sufficiently local unitary transformations. (See [15], [6].)

5. The quantum part of Shor's algorithm

The quantum part of Shor's algorithm, i.e., **STEP 2**, is the following:

STEP 2.0 Initialize registers 1 and 2, i.e.,

$$|\psi_0\rangle = |\text{REG1}\rangle |\text{REG2}\rangle = |0\rangle |1\rangle = |00 \cdots 00\rangle |00 \cdots 01\rangle$$

STEP 2.1 ⁴Apply the Q -point Fourier transform \mathcal{F} to REGISTER1.

$$|\psi_0\rangle = |0\rangle |1\rangle \xrightarrow{\mathcal{F} \otimes I} |\psi_1\rangle = \frac{1}{\sqrt{Q}} \sum_{x=0}^{Q-1} \omega^{0 \cdot x} |x\rangle |0\rangle = \frac{1}{\sqrt{Q}} \sum_{x=0}^{Q-1} |x\rangle |1\rangle$$

REMARK 1. Hence, REGISTER1 now holds all the integers

$$0, 1, 2, \dots, Q-1$$

in superposition.

STEP 2.2 Let U_f be the unitary transformation that takes $|x\rangle |0\rangle$ to $|x\rangle |f(x)\rangle$. Apply the linear transformation U_f to the two registers. The result is:

$$|\psi_1\rangle = \frac{1}{\sqrt{Q}} \sum_{x=0}^{Q-1} |x\rangle |1\rangle \xrightarrow{U_f} |\psi_2\rangle = \frac{1}{\sqrt{Q}} \sum_{x=0}^{Q-1} |x\rangle |f(x)\rangle$$

REMARK 2. The state of the two registers is now more than a superposition of states. In this step, we have quantum entangled the two registers.

STEP 2.3. Apply the Q -point Fourier transform \mathcal{F} to REGISTER1. The resulting state is:

$$\begin{aligned} |\psi_2\rangle = \frac{1}{\sqrt{Q}} \sum_{x=0}^{Q-1} |x\rangle |f(x)\rangle &\xrightarrow{\mathcal{F} \otimes I} |\psi_3\rangle = \frac{1}{Q} \sum_{x=0}^{Q-1} \sum_{y=0}^{Q-1} \omega^{xy} |y\rangle |f(x)\rangle \\ &= \frac{1}{Q} \sum_{y=0}^{Q-1} \|\Upsilon(y)\rangle \cdot |y\rangle \frac{\|\Upsilon(y)\rangle}{\|\Upsilon(y)\rangle}, \end{aligned}$$

where

$$|\Upsilon(y)\rangle = \sum_{x=0}^{Q-1} \omega^{xy} |f(x)\rangle.$$

⁴In this step we could have instead applied the Hadamard transform to REGISTER1 with the same result, but at the computational cost of $O(\lg N)$ sufficiently local unitary transformations. The phrase "sufficiently local unitary transformation" is defined in the last part of section 7.7 of [13].

STEP 2.4. Measure REG1, i.e., perform a measurement with respect to the orthogonal projections

$$|0\rangle\langle 0| \otimes I, |1\rangle\langle 1| \otimes I, |2\rangle\langle 2| \otimes I, \dots, |Q-1\rangle\langle Q-1| \otimes I,$$

where I denotes the identity operator on the Hilbert space of the second register REG2.

As a result of this measurement, we have, with probability

$$Prob(y_0) = \frac{\|\Upsilon(y_0)\|^2}{Q^2},$$

moved to the state

$$|y_0\rangle \frac{|\Upsilon(y_0)\rangle}{\|\Upsilon(y_0)\|}$$

and measured the value

$$y_0 \in \{0, 1, 2, \dots, Q-1\}.$$

If after this computation, we ignore the two registers REG1 and REG2, we see that what we have created is nothing more than a classical probability distribution \mathcal{S} on the sample space

$$\{0, 1, 2, \dots, Q-1\}.$$

In other words, the sole purpose of executing STEPS 2.1 to 2.4 is to create a classical finite memoryless stochastic source \mathcal{S} which outputs a symbol $y_0 \in \{0, 1, 2, \dots, Q-1\}$ with the probability

$$Prob(y_0) = \frac{\|\Upsilon(y_0)\|^2}{Q^2}.$$

(For more details, please refer to section 8.1 of [13].)

As we shall see, the objective of the remainder of Shor's algorithm is to glean information about the period P of f from the just created stochastic source \mathcal{S} . The stochastic source was created exactly for that reason.

6. Peter Shor's stochastic source \mathcal{S}

Before continuing to the final part of Shor's algorithm, we need to analyze the probability distribution $Prob(y)$ a little more carefully.

PROPOSITION 1. *Let q and r be the unique non-negative integers such that $Q = Pq + r$, where $0 \leq r < P$; and let $Q_0 = Pq$. Then*

$$Prob(y) = \begin{cases} \frac{r \sin^2\left(\frac{\pi P y}{Q} \cdot \left(\frac{Q_0}{P} + 1\right)\right) + (P-r) \sin^2\left(\frac{\pi P y}{Q} \cdot \frac{Q_0}{P}\right)}{Q^2 \sin^2\left(\frac{\pi P y}{Q}\right)} & \text{if } Py \neq 0 \pmod{Q} \\ \frac{r(Q_0+P)^2 + (P-r)Q_0^2}{Q^2 P^2} & \text{if } Py = 0 \pmod{Q} \end{cases}$$

PROOF. We begin by deriving a more usable expression for $|\Upsilon(y)\rangle$.

$$\begin{aligned}
|\Upsilon(y)\rangle &= \sum_{x=0}^{Q-1} \omega^{xy} |f(x)\rangle = \sum_{x=0}^{Q_0-1} \omega^{xy} |f(x)\rangle + \sum_{x=Q_0}^{Q-1} \omega^{xy} |f(x)\rangle \\
&= \sum_{x_0=0}^{P-1} \sum_{x_1=0}^{\frac{Q_0}{P}-1} \omega^{(Px_1+x_0)y} |f(Px_1+x_0)\rangle + \sum_{x_0=0}^{r-1} \omega^{[P(\frac{Q_0}{P})+x_0]y} |f(Px_1+x_0)\rangle \\
&= \sum_{x_0=0}^{P-1} \omega^{x_0y} \cdot \left(\sum_{x_1=0}^{\frac{Q_0}{P}-1} \omega^{Pyx_1} \right) |f(x_0)\rangle + \sum_{x_0=0}^{r-1} \omega^{x_0y} \cdot \omega^{Py(\frac{Q_0}{P})} |f(x_0)\rangle \\
&= \sum_{x_0=0}^{r-1} \omega^{x_0y} \cdot \left(\sum_{x_1=0}^{\frac{Q_0}{P}} \omega^{Pyx_1} \right) |f(x_0)\rangle + \sum_{x_0=r}^{P-1} \omega^{x_0y} \cdot \left(\sum_{x_1=0}^{\frac{Q_0}{P}-1} \omega^{Pyx_1} \right) |f(x_0)\rangle
\end{aligned}$$

where we have used the fact that f is periodic of period P .

Since f is one-to-one when restricted to its period $0, 1, 2, \dots, P-1$, all the kets

$$|f(0)\rangle, |f(1)\rangle, |f(2)\rangle, \dots, |f(P-1)\rangle,$$

are mutually orthogonal. Hence,

$$\langle \Upsilon(y) | \Upsilon(y) \rangle = r \left| \sum_{x_1=0}^{\frac{Q_0}{P}} \omega^{Pyx_1} \right|^2 + (P-r) \left| \sum_{x_1=0}^{\frac{Q_0}{P}-1} \omega^{Pyx_1} \right|^2.$$

If $Py = 0 \pmod{Q}$, then since ω is a Q -th root of unity, we have

$$\langle \Upsilon(y) | \Upsilon(y) \rangle = r \left(\frac{Q_0}{P} + 1 \right)^2 + (P-r) \left(\frac{Q_0}{P} \right)^2.$$

On the other hand, if $Py \neq 0 \pmod{Q}$, then we can sum the geometric series to obtain

$$\begin{aligned}
\langle \Upsilon(y) | \Upsilon(y) \rangle &= r \left| \frac{\omega^{Py \cdot (\frac{Q_0}{P} + 1)} - 1}{\omega^{Py} - 1} \right|^2 + (P-r) \left| \frac{\omega^{Py \cdot (\frac{Q_0}{P})} - 1}{\omega^{Py} - 1} \right|^2 \\
&= r \left| \frac{e^{\frac{2\pi i}{Q} \cdot Py \cdot (\frac{Q_0}{P} + 1)} - 1}{e^{\frac{2\pi i}{Q} \cdot Py} - 1} \right|^2 + (P-r) \left| \frac{e^{\frac{2\pi i}{Q} \cdot Py \cdot (\frac{Q_0}{P})} - 1}{e^{\frac{2\pi i}{Q} \cdot Py} - 1} \right|^2
\end{aligned}$$

where we have used the fact that ω is the primitive Q -th root of unity given by

$$\omega = e^{2\pi i/Q}.$$

The remaining part of the proposition is a consequence of the trigonometric identity

$$|e^{i\theta} - 1|^2 = 4 \sin^2 \left(\frac{\theta}{2} \right).$$

□

As a corollary, we have

COROLLARY 1. *If P is an exact divisor of Q , then*

$$\text{Prob}(y) = \begin{cases} 0 & \text{if } Py \neq 0 \pmod{Q} \\ \frac{1}{P} & \text{if } Py = 0 \pmod{Q} \end{cases}$$

7. A momentary digression: Continued fractions

We digress for a moment to review the theory of continued fractions. (For a more in-depth explanation of the theory of continued fractions, please refer to [5] and [12].)

Every positive rational number ξ can be written as an expression in the form

$$\xi = a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{a_3 + \frac{1}{\dots + a_N}}}}$$

where a_0 is a non-negative integer, and where a_1, \dots, a_N are positive integers. Such an expression is called a (finite, simple) **continued fraction**, and is uniquely determined by ξ provided we impose the condition $a_N > 1$. For typographical simplicity, we denote the above continued fraction by

$$[a_0, a_1, \dots, a_N] .$$

The continued fraction expansion of ξ can be computed with the following recurrence relation, which always terminates if ξ is rational:

$$\boxed{\begin{cases} a_0 = \lfloor \xi \rfloor \\ \xi_0 = \xi - a_0 \end{cases}, \text{ and if } \xi_n \neq 0, \text{ then } \begin{cases} a_{n+1} = \lfloor 1/\xi_n \rfloor \\ \xi_{n+1} = \frac{1}{\xi_n} - a_{n+1} \end{cases}}$$

The n -th **convergent** ($0 \leq n \leq N$) of the above continued fraction is defined as the rational number ξ_n given by

$$\xi_n = [a_0, a_1, \dots, a_n] .$$

Each convergent ξ_n can be written in the form, $\xi_n = \frac{p_n}{q_n}$, where p_n and q_n are relatively prime integers ($\gcd(p_n, q_n) = 1$). The integers p_n and q_n are determined by the recurrence relation

$$\boxed{\begin{aligned} p_0 &= a_0, & p_1 &= a_1 a_0 + 1, & p_n &= a_n p_{n-1} + p_{n-2}, \\ q_0 &= 1, & q_1 &= a_1, & q_n &= a_n q_{n-1} + q_{n-2} . \end{aligned}}$$

8. Preparation for the final part of Shor's algorithm

DEFINITION 2. ⁵For each integer a , let $\{a\}_Q$ denote the **residue** of a modulo Q of **smallest magnitude**. In other words, $\{a\}_Q$ is the unique integer such that

$$\begin{cases} a = \{a\}_Q \bmod Q \\ -Q/2 < \{a\}_Q \leq Q/2 \end{cases} .$$

PROPOSITION 2. Let y be an integer lying in S_Q . Then

$$\text{Prob}(y) \geq \begin{cases} \frac{4}{\pi^2} \cdot \frac{1}{P} \cdot \left(1 - \frac{1}{N}\right)^2 & \text{if } 0 < |\{Py\}_Q| \leq \frac{P}{2} \cdot \left(1 - \frac{1}{N}\right) \\ \frac{1}{P} \cdot \left(1 - \frac{1}{N}\right)^2 & \text{if } \{Py\}_Q = 0 \end{cases}$$

PROOF. We begin by noting that

$$\begin{aligned} \left| \frac{\pi\{Py\}_Q}{Q} \cdot \left(\frac{Q_0}{P} + 1\right) \right| &\leq \frac{\pi}{Q} \cdot \frac{P}{2} \cdot \left(1 - \frac{1}{N}\right) \cdot \left(\frac{Q_0+P}{P}\right) \leq \frac{\pi}{2} \cdot \left(1 - \frac{1}{N}\right) \cdot \left(\frac{Q+P}{Q}\right) \\ &\leq \frac{\pi}{2} \cdot \left(1 - \frac{1}{N}\right) \cdot \left(1 + \frac{P}{Q}\right) \leq \frac{\pi}{2} \cdot \left(1 - \frac{1}{N}\right) \cdot \left(1 + \frac{N}{N^2}\right) < \frac{\pi}{2} , \end{aligned}$$

where we have made use of the inequalities

$$N^2 \leq Q < 2N^2 \quad \text{and} \quad 0 < P \leq N .$$

It immediately follows that

$$\left| \frac{\pi\{Py\}_Q}{Q} \cdot \frac{Q_0}{P} \right| < \frac{\pi}{2} .$$

As a result, we can legitimately use the inequality

$$\frac{4}{\pi^2} \theta^2 \leq \sin^2 \theta \leq \theta^2, \quad \text{for } |\theta| < \frac{\pi}{2}$$

to simplify the expression for $\text{Prob}(y)$.

Thus,

$$\begin{aligned} \text{Prob}(y) &= \frac{r \sin^2\left(\frac{\pi\{Py\}_Q}{Q} \cdot \left(\frac{Q_0}{P} + 1\right)\right) + (P-r) \sin^2\left(\frac{\pi\{Py\}_Q}{Q} \cdot \frac{Q_0}{P}\right)}{Q^2 \sin^2\left(\frac{\pi Py}{Q}\right)} \\ &\geq \frac{r \cdot \frac{4}{\pi^2} \cdot \left(\frac{\pi\{Py\}_Q}{Q} \cdot \left(\frac{Q_0}{P} + 1\right)\right)^2 + (P-r) \cdot \frac{4}{\pi^2} \cdot \left(\frac{\pi\{Py\}_Q}{Q} \cdot \frac{Q_0}{P}\right)^2}{Q^2 \left(\frac{\pi\{Py\}_Q}{Q}\right)^2} \\ &\geq \frac{4}{\pi^2} \cdot \frac{P \cdot \left(\frac{Q_0}{P}\right)^2}{Q^2} = \frac{4}{\pi^2} \cdot \frac{1}{P} \cdot \left(\frac{Q-r}{Q}\right)^2 \\ &= \frac{4}{\pi^2} \cdot \frac{1}{P} \cdot \left(1 - \frac{r}{Q}\right)^2 \geq \frac{4}{\pi^2} \cdot \frac{1}{P} \cdot \left(1 - \frac{1}{N}\right)^2 \end{aligned}$$

⁵ $\{a\}_Q = a - Q \cdot \text{round}\left(\frac{a}{Q}\right) = a - Q \cdot \left\lfloor \frac{a}{Q} + \frac{1}{2} \right\rfloor$.

The remaining case, $\{Py\}_Q = 0$ is left to the reader. \square

LEMMA 1. *Let*

$$Y = \left\{ y \in S_Q \mid \left| \{Py\}_Q \right| \leq \frac{P}{2} \right\} \quad \text{and} \quad S_P = \{d \in S_Q \mid 0 \leq d < P\} .$$

Then the map

$$\begin{aligned} Y &\longrightarrow S_P \\ y &\longmapsto d = d(y) = \text{round} \left(\frac{P}{Q} \cdot y \right) \end{aligned}$$

is a bijection with inverse

$$y = y(d) = \text{round} \left(\frac{Q}{P} \cdot d \right) .$$

Hence, Y and S_P are in one-to-one correspondence. Moreover,

$$\{Py\}_Q = P \cdot y - Q \cdot d(y) .$$

REMARK 3. *Moreover, the following two sets of rationals are in one-to-one correspondence*

$$\left\{ \frac{y}{Q} \mid y \in Y \right\} \longleftrightarrow \left\{ \frac{d}{P} \mid 0 \leq d < P \right\}$$

As a result of the measurement performed in STEP 2.4, we have in our possession an integer $y \in Y$. We now show how y can be used to determine the unknown period P .

We now need the following theorem⁶ from the theory of continued fractions:

THEOREM 1. *Let ξ be a real number, and let a and b be integers with $b > 0$. If*

$$\left| \xi - \frac{a}{b} \right| \leq \frac{1}{2b^2} ,$$

then the rational number a/b is a convergent of the continued fraction expansion of ξ .

As a corollary, we have:

COROLLARY 2. *If $\left| \{Py\}_Q \right| \leq \frac{P}{2}$, then the rational number $\frac{d(y)}{P}$ is a convergent of the continued fraction expansion of $\frac{y}{Q}$.*

PROOF. Since

$$Py - Qd(y) = \{Py\}_Q ,$$

we know that

$$|Py - Qd(y)| \leq \frac{P}{2} ,$$

which can be rewritten as

$$\left| \frac{y}{Q} - \frac{d(y)}{P} \right| \leq \frac{1}{2Q} .$$

⁶See [5, Theorem 184, Section 10.15].

But, since $Q \geq N^2$, it follows that

$$\left| \frac{y}{Q} - \frac{d(y)}{P} \right| \leq \frac{1}{2N^2}.$$

Finally, since $P \leq N$ (and hence $\frac{1}{2N^2} \leq \frac{1}{2P^2}$), the above theorem can be applied. Thus, $\frac{d(y)}{P}$ is a convergent of the continued fraction expansion of $\xi = \frac{y}{Q}$. \square

Since $\frac{d(y)}{P}$ is a convergent of the continued fraction expansion of $\frac{y}{Q}$, it follows that, for some n ,

$$\frac{d(y)}{P} = \frac{p_n}{q_n},$$

where p_n and q_n are relatively prime positive integers given by a recurrence relation found in the previous subsection. So it would seem that we have found a way of deducing the period P from the output y of STEP 2.4, and so we are done.

Not quite!

We can determine P from the measured y produced by STEP 2.4, only if

$$\begin{cases} p_n = d(y) \\ q_n = P \end{cases},$$

which is true only when $d(y)$ and P are relatively prime.

So what is the probability that the $y \in Y$ produced by STEP 2.4 satisfies the additional condition that

$$\gcd(P, d(y)) = 1?$$

PROPOSITION 3. *The probability that the random y produced by STEP 2.4 is such that $d(y)$ and P are relatively prime is bounded below by the following expression*

$$\text{Prob} \{y \in Y \mid \gcd(d(y), P) = 1\} \geq \frac{4}{\pi^2} \cdot \frac{\phi(P)}{P} \cdot \left(1 - \frac{1}{N}\right)^2,$$

where $\phi(P)$ denotes Euler's totient function, i.e., $\phi(P)$ is the number of positive integers less than P which are relatively prime to P .

The following theorem can be found in [5, Theorem 328, Section 18.4]:

THEOREM 2.

$$\liminf \frac{\phi(N)}{N/\ln \ln N} = e^{-\gamma},$$

where γ denotes Euler's constant $\gamma = 0.57721566490153286061\dots$, and where $e^{-\gamma} = 0.5614594836\dots$.

As a corollary, we have:

COROLLARY 3.

$$\text{Prob}\{y \in Y \mid \gcd(d(y), P) = 1\} \geq \frac{4}{\pi^2 \ln 2} \cdot \frac{e^{-\gamma} - \epsilon(P)}{\lg \lg N} \cdot \left(1 - \frac{1}{N}\right)^2,$$

where $\epsilon(P)$ is a monotone decreasing sequence converging to zero. In terms of asymptotic notation,

$$\text{Prob}\{y \in Y \mid \gcd(d(y), P) = 1\} = \Omega\left(\frac{1}{\lg \lg N}\right).$$

Thus, if STEP 2.4 is repeated $O(\lg \lg N)$ times, then the probability of success is $\Omega(1)$.

PROOF. From the above theorem, we know that

$$\frac{\phi(P)}{P/\ln \ln P} \geq e^{-\gamma} - \epsilon(P).$$

where $\epsilon(P)$ is a monotone decreasing sequence of positive reals converging to zero. Thus,

$$\frac{\phi(P)}{P} \geq \frac{e^{-\gamma} - \epsilon(P)}{\ln \ln P} \geq \frac{e^{-\gamma} - \epsilon(P)}{\ln \ln N} = \frac{e^{-\gamma} - \epsilon(P)}{\ln \ln 2 + \ln \lg N} \geq \frac{e^{-\gamma} - \epsilon(P)}{\ln 2} \cdot \frac{1}{\lg \lg N}$$

□

REMARK 4. $\Omega\left(\frac{1}{\lg \lg N}\right)$ denotes an asymptotic lower bound. Readers not familiar with the big-oh $O(*)$ and big-omega $\Omega(*)$ notation should refer to [2, Chapter 2] or [1, Chapter 2].

REMARK 5. For the curious reader, lower bounds $LB(P)$ of $e^{-\gamma} - \epsilon(P)$ for $3 \leq P \leq 841$ are given in the following table:

P	$LB(P)$
3	0.062
4	0.163
5	0.194
7	0.303
13	0.326
31	0.375
61	0.383
211	0.411
421	0.425
631	0.435
841	0.468

Thus, if one wants a reasonable bound on the $\text{Prob}\{y \in Y \mid \gcd(d(y), P) = 1\}$ before continuing with Shor's algorithm, it would pay to first use a classical algorithm to verify that the period P of the randomly chosen integer m is not too small.

9. The final part of Shor's algorithm

We are now prepared to give the last step in Shor's algorithm. This step can be performed on a classical computer. (We remind the reader that ξ_n denotes the n -th convergent of $\frac{y}{Q}$.)

Step 2.5 Compute the period P from the integer y produced by **STEP 2.4**.

- LOOP FOR EACH n FROM $n = 1$ UNTIL $\xi_n = 0$.
- – Use the recurrence relations given in subsection 13.7, to compute the p_n and q_n of the n -th convergent $\xi_n = \frac{p_n}{q_n}$ of $\frac{y}{Q}$.
- – Test to see if $q_n = P$ by computing⁷

$$m^{q_n} = \prod_i \left(m^{2^i}\right)^{q_{n,i}} \bmod N ,$$

where $q_n = \sum_i q_{n,i} 2^i$ is the binary expansion of q_n .
If $m^{q_n} = 1 \bmod N$, then exit with the answer $P = q_n$, and proceed to **Step 3**. If not, then continue to the loop.

- END OF LOOP
- If you happen to reach this point, you are a very unlucky quantum computer scientist. You must start over by returning to **STEP 2.0**. But don't give up hope! The probability that the integer y produced by **STEP 2.4** will lead to a successful completion of Step 2.5 is bounded below by

$$\frac{4}{\pi^2 \ln 2} \cdot \frac{e^{-\gamma} - \epsilon(P)}{\lg \lg N} \cdot \left(1 - \frac{1}{N}\right)^2 > \frac{0.232}{\lg \lg N} \cdot \left(1 - \frac{1}{N}\right)^2 ,$$

provided the period P is greater than 3. [γ denotes Euler's constant.]

⁷The indicated algorithm for computing $m^{q_n} \bmod N$ requires $O(\lg q_n)$ arithmetic operations.

10. An example of Shor's algorithm

Let us now show how $N = 91 (= 7 \cdot 13)$ can be factored using Shor's algorithm.

We choose $Q = 2^{14} = 16384$ so that $N^2 \leq Q < 2N^2$.

Step 1 Choose a random positive integer m , say $m = 3$. Since $\gcd(91, 3) = 1$, we proceed to **STEP 2** to find the period of the function f given by

$$f(a) = 3^a \pmod{91}$$

REMARK 6. *Unknown to us, f has period $P = 6$. For,*

a	0	1	2	3	4	5	6	7	...
$f(a)$	1	3	9	27	81	61	1	3	...
\therefore Unknown period $P = 6$									

STEP 2.0 Initialize registers 1 and 2. Thus, the state of the two registers becomes:

$$|\psi_0\rangle = |0\rangle |1\rangle$$

STEP 2.1 Apply the Q -point Fourier transform \mathcal{F} to register #1, where

$$\mathcal{F} |k\rangle = \frac{1}{\sqrt{16384}} \sum_{x=0}^{16383} \omega^{0 \cdot x} |x\rangle ,$$

and where ω is a primitive Q -th root of unity, e.g., $\omega = e^{\frac{2\pi i}{16384}}$. Thus the state of the two registers becomes:

$$|\psi_1\rangle = \frac{1}{\sqrt{16384}} \sum_{x=0}^{16383} |x\rangle |1\rangle$$

STEP 2.2 Apply the unitary transformation U_f to registers #1 and #2, where

$$U_f |x\rangle |\ell\rangle = |x\rangle |f(x) - \ell \pmod{91}\rangle .$$

(Please note that $U_f^2 = I$.) Thus, the state of the two registers becomes:

$$\begin{aligned}
|\psi_2\rangle &= \frac{1}{\sqrt{16384}} \sum_{x=0}^{16383} |x\rangle |3^x \bmod 91\rangle \\
&= \frac{1}{\sqrt{16384}} (\quad |0\rangle |1\rangle + |1\rangle |3\rangle + |2\rangle |9\rangle + |3\rangle |27\rangle + |4\rangle |81\rangle + |5\rangle |61\rangle \\
&\quad + |6\rangle |1\rangle + |7\rangle |3\rangle + |8\rangle |9\rangle + |9\rangle |27\rangle + |10\rangle |81\rangle + |11\rangle |61\rangle \\
&\quad + |12\rangle |1\rangle + |13\rangle |3\rangle + |14\rangle |9\rangle + |15\rangle |27\rangle + |16\rangle |81\rangle + |17\rangle |61\rangle \\
&\quad + \dots \\
&\quad + |16380\rangle |1\rangle + |16381\rangle |3\rangle + |16382\rangle |9\rangle + |16383\rangle |27\rangle \\
&\quad)
\end{aligned}$$

REMARK 7. *The state of the two registers is now more than a superposition of states. We have in the above step quantum entangled the two registers.*

STEP 2.3 Apply the Q -point \mathcal{F} again to register #1. Thus, the state of the system becomes:

$$\begin{aligned}
|\psi_3\rangle &= \frac{1}{\sqrt{16384}} \sum_{x=0}^{16383} \frac{1}{\sqrt{16384}} \sum_{y=0}^{16383} \omega^{xy} |y\rangle |3^x \bmod 91\rangle \\
&= \frac{1}{16384} \sum_{x=0}^{16383} |y\rangle \sum_{x=0}^{16383} \omega^{xy} |3^x \bmod 91\rangle \\
&= \frac{1}{16384} \sum_{x=0}^{16383} |y\rangle |\Upsilon(y)\rangle,
\end{aligned}$$

where

$$|\Upsilon(y)\rangle = \sum_{x=0}^{16383} \omega^{xy} |3^x \bmod 91\rangle$$

Thus,

$$\begin{aligned}
|\Upsilon(y)\rangle &= |1\rangle + \omega^y |3\rangle + \omega^{2y} |9\rangle + \omega^{3y} |27\rangle + \omega^{4y} |81\rangle + \omega^{5y} |61\rangle \\
&\quad + \omega^{6y} |1\rangle + \omega^{7y} |3\rangle + \omega^{8y} |9\rangle + \omega^{9y} |27\rangle + \omega^{10y} |81\rangle + \omega^{11y} |61\rangle \\
&\quad + \omega^{12y} |1\rangle + \omega^{13y} |3\rangle + \omega^{14y} |9\rangle + \omega^{15y} |27\rangle + \omega^{16y} |81\rangle + \omega^{17y} |61\rangle \\
&\quad + \dots \\
&\quad + \omega^{16380y} |1\rangle + \omega^{16381y} |3\rangle + \omega^{16382y} |9\rangle + \omega^{16383y} |27\rangle
\end{aligned}$$

STEP 2.4 Measure REG1. The result of our measurement just happens to turn out to be

$$y = 13453$$

Unknown to us, the probability of obtaining this particular y is:

$$0.3189335551 \times 10^{-6} .$$

Moreover, unknown to us, we're lucky! The corresponding d is relatively prime to P , i.e.,

$$d = d(y) = \text{round}\left(\frac{P}{Q} \cdot y\right) = 5$$

However, we do know that the probability of $d(y)$ being relatively prime to P is greater than

$$\frac{0.232}{\lg \lg N} \cdot \left(1 - \frac{1}{N}\right)^2 \approx 8.4\% \quad (\text{provided } P > 3) .$$

We also know that

$$\frac{d(y)}{P}$$

is a convergent of the continued fraction expansion of

$$\xi = \frac{y}{Q} = \frac{13453}{16384}$$

So with a reasonable amount of confidence, we proceed to **Step 2.5**.

Step 2.5 Using the recurrence relations found in subsection 13.7 of this paper, we successively compute (beginning with $n = 0$) the a_n 's and q_n 's for the continued fraction expansion of

$$\xi = \frac{y}{Q} = \frac{13453}{16384} .$$

For each non-trivial n in succession, we check to see if

$$3^{q_n} = 1 \pmod{91} .$$

If this is the case, then we know $q_n = P$, and we immediately exit from **Step 2.5** and proceed to **Step 3**.

- In this example, $n = 0$ and $n = 1$ are trivial cases.
- For $n = 2$, $a_2 = 4$ and $q_2 = 5$. We test q_2 by computing

$$3^{q_2} = 3^5 = \left(3^{2^0}\right)^1 \cdot \left(3^{2^1}\right)^0 \cdot \left(3^{2^2}\right)^1 = 61 \neq 1 \pmod{91} .$$

Hence, $q_2 \neq P$.

- We proceed to $n = 3$, and compute

$$a_3 = 1 \text{ and } q_3 = 6.$$

We then test q_3 by computing

$$3^{q_3} = 3^6 = (3^{2^0})^0 \cdot (3^{2^1})^1 \cdot (3^{2^2})^1 = 1 \pmod{91}.$$

Hence, $q_3 = P$. Since we now know the period P , there is no need to continue to compute the remaining a_n 's and q_n 's. We proceed immediately to **Step 3**.

To satisfy the reader's curiosity we have listed in the table below all the values of a_n , p_n , and q_n for $n = 0, 1, \dots, 14$. But it should be mentioned again that we need only to compute a_n and q_n for $n = 0, 1, 2, 3$, as indicated above.

n	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
a_n	0	1	4	1	1	2	3	1	1	3	1	1	1	1	3
p_n	0	1	4	5	9	23	78	101	179	638	817	1455	2272	3727	13453
q_n	1	1	5	6	11	28	95	123	218	777	995	1772	2767	4539	16384

Step 3. Since $P = 6$ is even, we proceed to **Step 4**.

Step 4. Since

$$3^{P/2} = 3^3 = 27 \not\equiv -1 \pmod{91},$$

we go to **Step 5**.

Step 5. With the Euclidean algorithm, we compute

$$\gcd(3^{P/2} - 1, 91) = \gcd(3^3 - 1, 91) = \gcd(26, 91) = 13.$$

We have succeeded in finding a non-trivial factor of $N = 91$, namely 13. We exit Shor's algorithm, and proceed to celebrate!

We leave the reader with the following instructive exercise:

EXERCISE 1 (Exercise for the reader). *Go through the above example assuming that **Step. 1** produces $m = 5$ instead of $m = 3$.*

References

- [1] Brassard, Gilles, and Paul Bratley, "**Algorithmics: Theory and Practice**," Prentice-Hall, (1988).
- [2] Cormen, Thomas H., Charles E. Leiserson, and Ronald L. Rivest, "**Introduction to Algorithms**," McGraw-Hill, (1990).
- [3] Cox, David, John Little, and Donal O'Shea, "**Ideals, Varieties, and Algorithms**," (second edition), Springer-Verlag, (1996).
- [4] Ekert, Artur K. and Richard Jozsa, "**Quantum computation and Shor's factoring algorithm**," Rev. Mod. Phys., 68,(1996), pp 733-753.
- [5] Hardy, G.H., and E.M. Wright, "**An Introduction to the Theory of Numbers**," Oxford Press, (1965).
- [6] Hoyer, Peter, "**Efficient quantum transforms**, (1997), <http://xxx.lanl.gov/abs/quant-ph/9702028>.

- [7] Jozsa, Richard, **Quantum algorithms and the Fourier transform**, (1997), <http://xxx.lanl.gov/abs/quant-ph/9707033>.
- [8] Jozsa, Richard, Proc. Roy. Soc. London Soc., Ser. A, 454, (1998), 323 - 337.
- [9] Kitaev, A., **Quantum measurement and the abelian stabiliser problem**, (1995), <http://xxx.lanl.gov/abs/quant-ph/9511026>.
- [10] Lenstra, A.K., and H.W. Lenstra, Jr., eds., **"The Development of the Number Field Sieve,"** Lecture Notes in Mathematics, Vol. 1554, Springer-Verlag, (1993).
- [11] Lenstra, A.K., H.W. Lenstra, Jr., M.S. Manasse, and J.M. Pollard, **The number field sieve.** Proc. 22nd Annual ACM Symposium on Theory of Computing ACM, New York, (1990), pp 564 - 572. (See expanded version in Lenstra & Lenstra, (1993), pp 11 - 42.)
- [12] LeVeque, William Judson, **"Topics in Number Theory: Volume I,"** Addison-Wesley, (1958).
- [13] Lomonaco, Samuel J., Jr., **A Rosetta Stone for quantum mechanics with an introduction to quantum computation**, this AMS Proceedings of Symposia in Applied Mathematics (PSAPM). (<http://xxx.lanl.gov/abs/quant-ph/0007045>).
- [14] Miller, G. L., **Riemann's hypothesis and tests for primality**, J. Comput. System Sci., 13, (1976), pp 300 - 317.
- [15] Shor, Peter W., **Polynomial time algorithms for prime factorization and discrete logarithms on a quantum computer**, SIAM J. on Computing, 26(5) (1997), pp 1484 - 1509. (1995), (<http://xxx.lanl.gov/abs/quant-ph/9508027>)
- [16] Shor, Peter W., **Introduction to quantum algorithms**, this AMS Proceedings of Symposia in Applied Mathematics (PSAPM). (<http://xxx.lanl.gov/abs/quant-ph/0005003>)
- [17] Stinson, Douglas R., **"Cryptography: Theory and Practice,"** CRC Press, Boca Raton, (1995).

UNIVERSITY OF MARYLAND BALTIMORE COUNTY, BALTIMORE, MD 21250

E-mail address: Lomonaco@UMBC.EDU

URL: <http://www.csee.umbc.edu/~lomonaco>