Shor's Algorithm Report - DSA Final Project

Sparsh Gupta

August 3, 2025

1 Introduction

Shor's algorithm is a polynomial time quantum algorithm for integer factorization formulated by Peter Shor in 1994. Integer factorization consists of finding the prime factors of any integer by decomposing it into a product of these factors. For example, the prime factor decomposition of integer 30 will be 2 x 3 x 5.

In terms of time complexity, the best classical integer factoring algorithm, known as the General Number Field Sieve (GNFS), has a super-polynomial but sub-exponential time complexity of $O(e^{1.9(log\ N)^{\frac{1}{3}}(log\ log\ N)^{\frac{2}{3}}})$. No other algorithms are known to humans presently that are faster than this in the classical world. However, in the quantum world, Shor's algorithm beats this algorithm by having a polynomial time complexity using quantum gates of order $O((log\ N)^2(log\ log\ N)(log\ log\ log\ N))$ to factor an integer N.

In classical computing, information only exists in bits, which are 0 or 1, whereas quantum computing introduces the concept of qubits that can have quantum basis states $|0\rangle$ and $|1\rangle$, which can be represented as column vectors $\begin{bmatrix}1\\0\end{bmatrix}$ and $\begin{bmatrix}0\\1\end{bmatrix}$ respectively, at the same time taking advantage of the concept of superposition, introducing features such as quantum parallelism.

2 Background

2.1 Qubit

A qubit can store information as a linear combination of both the basis states such that the sum of the probabilities of measuring any states sums up to 1:

$$\psi = \alpha\,|0\rangle + \beta\,|1\rangle$$

where α and β are the complex ($\mathbb{C}$) probability amplitudes for each basis state.

2.2 Quantum Superposition

Quantum superposition is fundamentally important to quantum computing to perform certain calculations more efficiently, making sure that the qubit measurements have a balanced probability of being measured as $|0\rangle$ and $|1\rangle$, and to attain the quantum advantage. We can use a quantum gate, known as the Hadamard (H) gate to attain an equal superposition of the two basis states. The H gate maps the basis state $|0\rangle$ to $\frac{|0\rangle+|1\rangle}{\sqrt{2}}$ and the basis state $|1\rangle$ to $\frac{|0\rangle-|1\rangle}{\sqrt{2}}$. It is represented by the matrix $H = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}$.

2.3 Modular Exponentiation

Modular exponentiation is the process of finding a sequence of integers by repeated exponentiation of the integer x and applying a modulus N on every value such that:

$$x^a\ mod\ N = x^1\ mod\ N, x^2\ mod\ N, x^3\ mod\ N, ...$$

For example, if $x = 2$ and $N = 15$,

$$\begin{aligned} 2^a \ mod\ 15 &= 2^1 \ mod\ 15, 2^2 \ mod\ 15, 2^3 \ mod\ 15, 2^4 \ mod\ 15, 2^5 \ mod\ 15, ... \\ &= 2 \ mod\ 15, 4 \ mod\ 15, 8 \ mod\ 15, 16 \ mod\ 15, 32 \ mod\ 15, ... \\ &= 2, 4, 8, 1, 2, ... \end{aligned}$$

2.4 Quantum Fourier Transform (QFT)

A fourier transform is used to transform signals from one domain into another domain. To understand QFT better, let's start with the classical Discrete Fourier Transform (DFT). A DFT takes in an input vector $(x_0, ..., x_{N-1}) \in \mathbb{C}^N$, and transforms it into an output of vector $(y_0, ..., y_{N-1}) \in \mathbb{C}^N$ such that:

$$y_k = \sum_{j=0}^{N-1} x_j e^{-2\pi i \frac{jk}{N}}$$

Similarly in QFT, we want to only transform the amplitudes of the state. Therefore, it takes in a quantum state input $|x\rangle = \sum_{i=0}^{N-1} x_i |i\rangle$, and transforms it into an output quantum state of $|y\rangle = \sum_{i=0}^{N-1} y_i |i\rangle$ such that:

$$y_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j e^{2\pi i \frac{jk}{N}}$$

2.4.1 QFT on a quantum circuit

The QFT has the potential to encode an N-dimensional into only $n = log_2 N$ qubits. Therefore, for simplicity, we will use $N = 2^n$ in the following calculations and hence, QFT applied on the state $|x\rangle = |x_1 ... x_n\rangle$ looks like:

$$\begin{aligned} QFT\,|x\rangle &= \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} e^{2\pi i \frac{xy}{2^n}} |y\rangle \\ &= \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} e^{2\pi i (\sum_{k-1}^{n} \frac{y_k}{2^k}) x} |y_1 ... y_n\rangle \\ &= \frac{1}{\sqrt{N}} \sum_{y=0}^{N-1} \prod_{k=0}^{n} e^{2\pi i \frac{x y_k}{2^k}} |y_1 ... y_n\rangle \\ &= \frac{1}{\sqrt{N}} \bigotimes_{k=1}^{n} (|0\rangle + e^{2\pi i \frac{x}{2^k}} |1\rangle) \\ &= \frac{1}{\sqrt{N}} (|0\rangle + e^{2\pi i [0.x_n]} |1\rangle) \otimes \cdots \otimes (|0\rangle + e^{2\pi i [0.x_1 . x_2 ... x_n]} |1\rangle) \end{aligned}$$

3 Algorithm

The algorithm breaks the factoring problem into two major parts: Period finding and Factor derivation.
The algorithm takes two coprime integers, N and x, and finds the period r of $\mathcal{F}(a) = x^a \ mod \ N$.

3.1 Period finding

3.1.1 Initialization

We initialize two quantum registers first: the argument register with t qubits and the function register with $n = log_2 N$ qubits. Then, we choose $T = 2^t$ such that $N^2 \leq T \leq 2N^2$ where t is the number of qubits in the argument registers.

The initial quantum state of the registers is:

$$|\psi_0\rangle = |0\rangle \, |0\rangle$$

3.1.2 Superposition

We apply a Hadamard gate on each qubit of the argument register to attain superposition of all integers from 0 to T:

$$|\psi_1\rangle = \frac{1}{\sqrt{T}} \sum_{a=0}^{T-1} |a\rangle \, |0\rangle$$

3.1.3 Modular Exponentiation

We then implement the modular exponentiation function on the function register yielding the state:

$$|\psi_2\rangle = \frac{1}{\sqrt{T}} \sum_{a=0}^{T-1} |a\rangle \, |x^a \ mod \ N\rangle$$

This causes the quantum state $|\psi_2\rangle$ to be highly entangled and it demonstrates quantum parallelism with the modular exponentiation function.

3.1.4 QFT

We perform a quantum fourier transform on the argument register to obtain:

$$|\psi_3\rangle = \frac{1}{T} \sum_{a=0}^{T-1} \sum_{z=0}^{T-1} e^{(2\pi i)(az/T)} \, |z\rangle \, |x^a \ mod \ N\rangle$$

This causes constructive interference in the qubits and only the terms $|z\rangle$ with $z = \frac{qT}{r}$ where q is a random integer ranging from 0 to $r - 1$ have significant amplitudes.

3.1.5 Measurement of the Argument Register

Finally, we measure the qubits from the argument register to obtain classical output z. Using continued fraction approximation of $\frac{T}{z}$ (or simply T modulo z), the result gives us the value of the period r.

3.2 Factor derivation

To derive the factors from the obtained period, we use the Euclidean algorithm such that the factors are $p = GCD(x^{\frac{r}{2}} + 1, N)$ and $q = GCD(x^{\frac{r}{2}} - 1, N)$.

As this is a probabilistic algorithm due to its quantum nature, we get different pairs of factors for an integer that has more than two factors. This is based on the different measurement possibilities of the argument register after each run.

4 Example

We can look at an example for factorizing $N = 21$ with coprime integer $x = 2$.

4.1 Period finding

4.1.1 Initialization

We choose $T = 2^t = 2^9 = 512$ such that $N^2 \leq T \leq 2N^2$ $(21^2 \leq 2^9 \leq 2 \cdot 21^2)$.

Now, we initialize the two quantum registers: the argument register with $t = 9$ qubits and the function register with $n = log_2 N = 5$ qubits:

$$|\psi_0\rangle = |0\rangle\,|0\rangle$$

4.1.2 Superposition

We apply a Hadamard gate on each qubit of the argument register:

$$|\psi_1\rangle = \frac{1}{\sqrt{512}} \sum_{a=0}^{511} |a\rangle\,|0\rangle$$

4.1.3 Modular Exponentiation

Then, implement the modular exponentiation function on the function register:

$$\begin{aligned} |\psi_2\rangle &= \frac{1}{\sqrt{512}} \sum_{a=0}^{511} |a\rangle\,|2^a \ mod\ 21\rangle \\ &= \frac{1}{\sqrt{512}}(|0\rangle\,|1\rangle + |1\rangle\,|2\rangle + |2\rangle\,|4\rangle + |3\rangle\,|8\rangle + |4\rangle\,|16\rangle + |5\rangle\,|11\rangle + |6\rangle\,|1\rangle + |7\rangle\,|2\rangle + \cdots) \end{aligned}$$

Here, we can observe that the states of the second (function) register start repeating and therefore, we can group the terms which have a common state:

$$\begin{aligned} |\psi_2\rangle = \frac{1}{\sqrt{512}}[&(|0\rangle + |6\rangle + \cdots + |504\rangle + |510\rangle)\,|1\rangle + \\ &(|1\rangle + |7\rangle + \cdots + |505\rangle + |511\rangle)\,|2\rangle + \\ &(|2\rangle + |8\rangle + \cdots + |506\rangle)\,|4\rangle + \\ &(|3\rangle + |9\rangle + \cdots + |507\rangle)\,|8\rangle + \\ &(|4\rangle + |10\rangle + \cdots + |508\rangle)\,|16\rangle + \\ &(|5\rangle + |11\rangle + \cdots + |509\rangle)\,|11\rangle] \end{aligned}$$

Now, we measure the function register before performing a QFT to simplify the equations. The possible values of the states of the function register are $1, 2, 4, 6, 8, 16, 11$ as you can see in the above equation. When we measure this

register, we will obtain one of these numbers with equal probability. Let's say the result of the measurement was 4, then:

$$|\psi_3\rangle = \frac{1}{\sqrt{512}}(|2\rangle + |8\rangle + |14\rangle + \cdots + |506\rangle)\,|4\rangle$$

One thing to note here is that it does not matter what the result of the measurement is because we are trying to extract the periodic pattern. This is captured by the values of the first (argument) register, and we can use QFT to obtain the period.

4.1.4 QFT

We perform a QFT on the argument register:

$$\begin{aligned}
|\psi_4\rangle &= QFT(|\psi_3\rangle \\
&= QFT(\frac{1}{\sqrt{85}}\sum_{a=0}^{84}|6a+2\rangle)\,|4\rangle \\
&= \frac{1}{\sqrt{512}}\sum_{j=0}^{511}([\frac{1}{\sqrt{85}}\sum_{a=0}^{84}e^{-2\pi i\frac{6ja}{512}}]e^{-2\pi i\frac{j}{512}}\,|j\rangle)\,|4\rangle
\end{aligned}$$

4.1.5 Measurement of the Argument Register

The measurement performed on a quantum circuit differs from how we are doing it classically for this example. Therefore, the probability of measuring a result j from the argument register can be calculated classically as:

$$Probability(j) = \frac{1}{512\cdot 85}\left|\sum_{a=0}^{84}e^{-2\pi i\frac{6ja}{512}}\right|^2$$

Let us say that the probabilities peak at $j = 0, 85, 171, 256, 341, 427$ for this measurement. Then, let us suppose that the result of the measurement results in $j = 85$. Using continued fraction approximation of $\frac{T}{j} = \frac{512}{85}$ (or 512 modulo 85), we obtain the **period r = 6**.

4.2 Factor derivation

Using the Euclidean algorithm, the factors are $p = GCD(x^{\frac{r}{2}}+1, N) = GCD(2^3+1, 21)$ and $q = GCD(x^{\frac{r}{2}}-1, N) = GCD(2^3-1, 21)$. Therefore, we finally obtain the **factors p = 1 and q = 7 for N = 21** for this measurement.

We can obtain other factors with different probabilistic results from the measurement of the argument register.

5 References

[1] https://github.com/qiskit-community/qiskit-community-tutorials/blob/master/algorithms/shor_algorithm.ipynb

[2] https://arxiv.org/abs/quant-ph/0303175