Quantum Computing

Marija Kranjčević, Filip Kiršek, Petar Kunštek

Abstract

This paper explains the basics of quantum computing and gives an overview of some of the more notable quantum algorithms, such as Grover's search algorithm and Shor's factoring algorithm.

1 Introduction

There are many formulations of the Moore's law, but they all converge upon a similar conclusion: the performance of the computer chips increases, while the size decreases exponentially over time. This growth is obviously unsustainable; the classical physical models used in the construction of current computers cease to be a satisfying approximation at atomic and subatomic levels. This alone should be enough for one to be interested in quantum computing. There is more to it than that, however. Problems unsolvable through increase of computing power alone (e.g. factoring a large number within the current cycle of the universe) can be solved on quantum computers, using quantum algorithms (e.g. Shor's factoring algorithm). The downside is that quantum physics and, by extension, quantum computing often seem esoteric. This paper hopes to provide some insight into this potentially interesting field of research. It should not be consulted as a definitive guide to quantum computing, but as an introduction to the subject, providing a brief overview of the physical principles, the mathematical model and a somewhat more in-depth overview of the more important quantum algorithms. The number theory definitions and theorems used in the explanations of these algorithms are presented in Appendix.

2 Results From Quantum Physics (by Petar Kunštek)

Studying quantum physics is probably not as hard as it is confusing and unintuitive. The main reason is that, on many levels, quantum physics differs from classical physics. The "intuitive" models from classical physics usually can not be applied to explain quantum effects.

A similar situation is encountered when comparing relativistic and non-relativistic systems. There is no need for a relativistic model if an object's speed is small in comparison with the speed of light, c[1]; the classical theory is a satisfying approximation. But, when the object's speed approaches c, classical theory differs greatly from the actual results.

The same principle exists in differing between quantum and classical approach. In this case,

[1] $c = 3 \cdot 10^8$ m/s

the constant defining the scales at which classical theory becomes inapplicable is the Planck constant[2], denoted h. Therefore, particles at the atomic scales (less than 100 nm), can be seen to exhibit some unintuitive properties.

2.1 Quantum Superposition

Mirror experiment

Perhaps the best way to demonstrate this unintuitive property is an experiment. Let's take a source of light, two sensors and one half-silvered mirror, arranged like in Figure 1. The starting assumption is that a photon may be reflected or transmitted through the half-silvered mirror with the same probability. That is indeed the result of such an experiment: both sensors, measuring the intensity of light, detect the same level of intensity.

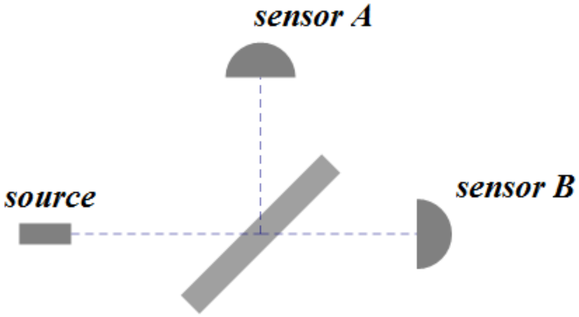

Figure 1: Half-silvered mirror

Now, with that hypothesis, complex structures of half-silvered and silvered mirrors can be constructed. Let us take two silvered mirrors (where total reflection occurs), two half-silvered mirrors, two sensors, denoted A and B, and a source of light, and arrange them as shown in Figure 2. Intuitively, it is expected that sensor A and sensor B each detect 50% of light intensity. Considering that a photon has a 50% chance to be reflected or transmitted, there should be four different possible paths, each occurring with the probability of 25%. Because two of those paths lead to sensor A, and the other two to sensor B, it could be expected that a photon has an equal chance of hitting each sensor. Yet, the experiment results are drastically different.

Sensor A constantly detects 100% of light intensity, while sensor B detects nothing. How to explain this phenomenon? Imagine that, instead of a small ball, a photon is a wave. If a single photon is studied in more detail, one could assume that, when it approached the first half-silvered mirror, the photon didn't "decide" which path to take, but reflected ant transmitted at the same time. Thus, at point S, the photon as wave is found in both of these states (created by two different paths) and, one could conclude that, toward sensor A

[2] $h = 6.626 \cdot 10^{-34}$ Js

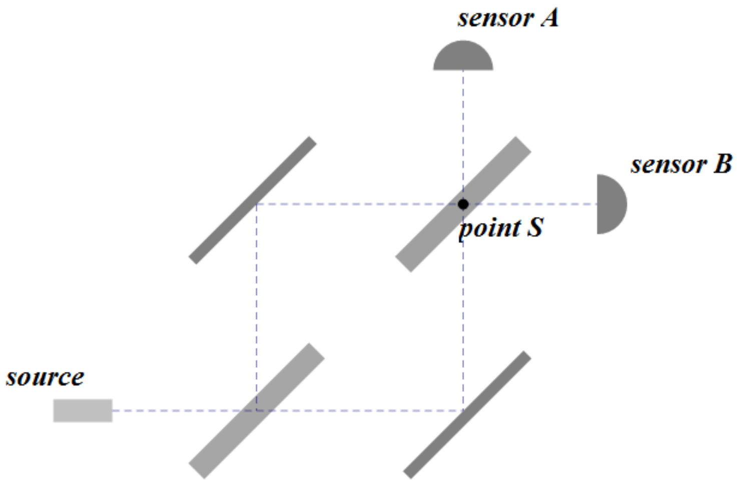

Figure 2: Mirror experiment

the photon is in a constructive interference and toward sensor B in destructive interference. Consequently, sensor A is always detecting photons, while sensor B detects nothing.
This is called superposition of waves. The real surprise is the fact that all particles can act as waves. As a result, small particles (e.g. electrons) don't posses the expected corporeal effects. The presented idea was first introduced by de Broglie[3] at the beginning of the twentieth century.

2.2 The Measuring Principle

Let us modify our experiment by placing a sensor device, which detects whether or not our photon passed through it, in one possible path (Figure 3).
The results of this modified experiment are actually the expected results of the original experiment. This demonstrates that, when a quantum system is measured (i.e. some information is obtained) quantum superposition ceases to exist. The system is then forced to "choose" between all possible results (in this case two paths) and behaves correspondingly from that point on.

2.3 Quantum Entanglement

Consider a collision between two particles. Such collisions can produce new particles, so let us assume an electron and a positron[4] were produced as a result of one such collision. There exists an experiment that determines the spin of the electron, and it can only obtain two results: up or down. If the electron is measured to have a certain spin (e.g. spin up), any measurement on the positron will produce the opposite result (in this case, down spin).

[3] Louis Victor de Broglie (1892.-1987.), French physicist
[4] antiparticle possessing positive electric charge, counterpart to electron

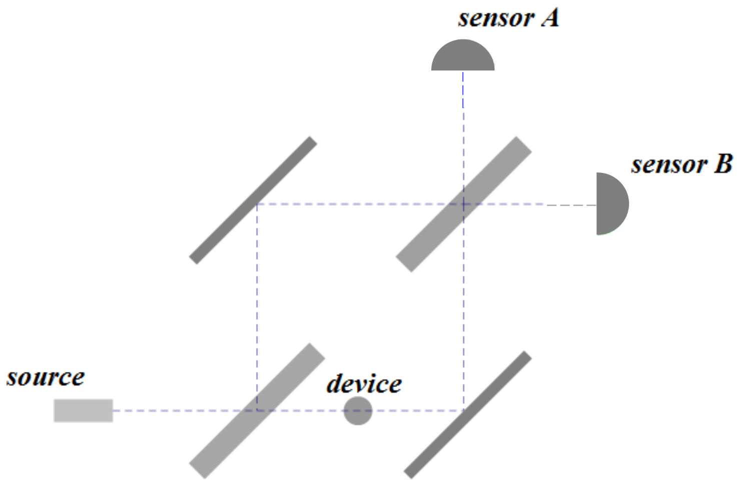

Figure 3: Mirror experiment with photon path detection

This phenomenon is one example of quantum entanglement.

To further explain quantum entanglement, let us consider another thought experiment.

Popper's experiment

Popper's[5] experiment is based on the uncertainty principle, i.e. the fact that, for a given particle, the uncertainty of momentum is inversely proportional to the uncertainty of position. In other words, if an object's position is well known, momentum is very uncertain, and vice versa. As in the previous example, let us take a source of particles, which creates them in pairs, as a result of collision. Let there also be two walls with narrow slits and a way to determine the momentum of the exiting particle (Figure 4).
To simplify, imagine a creation of a single pair where particles are moving in the opposite directions (which is actually a necessity in physics), one moving through slit A and another through slit B. In this case, both sensors would measure the same uncertainty. But, let's now change the size of the slit on wall A so that it is considerably bigger, but still small enough so that the sensor is able to detect the uncertainty in momentum. By applying the uncertainty principle, one may conclude that the uncertainty of momentum on the sensor A should be smaller than the uncertainty of momentum on the sensor B. But, the results are, rather surprisingly, the same, meaning that sensor A isn't able to detect any higher uncertainty. This can be explained through quantum entanglement. As one pair was passing through the slit on the wall B, information about the uncertainty of position became the same for both particles (although the particle moving toward wall A was passing through a bigger slit), resulting in both particles having the same uncertainty in momentum.

[5] Sir Karl Raimund Popper (1902.-1994.), Austro-British philosopher

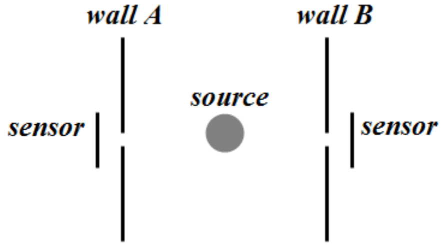

Figure 4: Popper's experiment

3 The Mathematical Model (by Filip Kiršek)

As was shown in the previous section, quantum physics deals with some unusual phenomena, most importantly, superposition, the influence of measurement on the observed system, and quantum entanglement. These properties of quantum systems are used as a basis in the construction of both the mathematical model of a quantum computer and the computer itself. A well-defined mathematical model allows us to solve problems unsolvable using the model for the classical computer.[6]

3.1 Qubits

Bit is a basic unit of information in classical computing. The entirety of classical computing is based on finite sequences of bits and manipulations thereof. Qubit, its quantum analogue, has a similar role in quantum computing.

Suppose that we know a bit has a value 1. Depending on where the bit "resides", the hard drive or the processor, that might mean that there is a non-zero current through a wire or that a part of the hard drive is magnetized. Similarly, an electron might be measured having either an up spin or a down spin. Electron, however, exhibits other properties that make it ill-fitting to be considered a bit.

Qubit is defined as a 2-dimensional complex vector, an element of $\mathbb{C}^2$. Two base states are denoted as $|0\rangle$ and $|1\rangle$, and they are represented by base state vectors, $|0\rangle = (1, 0)^t$, and $|1\rangle = (0, 1)^t$, $|0\rangle, |1\rangle \in \mathbb{C}^2$. Results from the theory of quantum physics tell us that quantum-mechanical systems rarely behave discreetly but are rather superpositions of the many possible states. Therefore, before measurement, qubit is in a superposition of $|0\rangle$ and $|1\rangle$. This means that any other qubit is a linear combination of base states, $|\phi\rangle = \alpha|0\rangle + \beta|1\rangle$, with the constraint that $|\alpha|^2 + |\beta|^2 = 1$ (where for $\alpha = a + bi$, $a, b \in \mathbb{R}$, $|\alpha| = \sqrt{a^2 + b^2}$). This normalization constraint is a result of the meaning behind α and β. $|\alpha|^2$ represents the probability of obtaining $|0\rangle$ when measuring our qubit and $|\beta|^2$, likewise, represents the probability of measuring $|1\rangle$.

The $|*\rangle$ is called a "ket", and, as was shown above, represents vectors, not just base state vectors. Generally speaking, kets represent qubits. For every ket, there is also a "bra", which

[6] most of the definitions presented in this section were obtained from [3] and [4]

can be interpreted as a row vector or, more precisely, a linear operator of scalar product. In other words, for $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, the corresponding bra is: $\langle\psi| = \alpha^*\langle 0| + \beta^*\langle 1| = (\alpha^*, \beta^*)$, or, more precisely, $\langle\psi| : \mathbb{C}^2 \to \mathbb{C}^2$, $\langle\psi|(x) = \langle\langle\psi|, x\rangle$ [7]. Later on, we will use simply $\langle x|y\rangle$, and this is called a "braket".

3.2 Measurement

Measurement of qubits is an important part of any quantum algorithm. As mentioned before, measuring quantum-mechanical systems alters them. When measuring a single qubit, we are actually collapsing the superposition and obtaining one of the base states, depending on the amplitudes (a measurement of $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ will give $|0\rangle$ with the probability $|\alpha|^2$ and $|1\rangle$ with the probability $|\beta|^2$).

In the context of quantum computing, after measuring our $|\psi\rangle$, we collapse its superposition and it becomes either $|0\rangle$ or $|1\rangle$, and, from that point on, it will behave correspondingly within the algorithm. Were we to measure the same qubit two or more times consecutively, all the measurements would give the same result as the first one. Any transformations over a qubit after it is measured will behave as if the qubit is in one of the base states - which it is. This is the reason that many quantum algorithms simply discard the measured qubit.

One might think that using qubits is a far more effective storage solution than the standard bit. The entirety of human knowledge, one might say, can be stored in the segment [0,1], so, logically, we would need but one qubit to derive wondrous amounts of information. This, however, misses one key property of qubits: during measurement, we only detect $|0\rangle$ or $|1\rangle$, depending on the probabilities associated with that qubit. To obtain just the norms of the amplitudes one must measure the same qubit infinitely many times. Even then, through measurement alone, we cannot differentiate between $|0\rangle$ and $i|0\rangle$.

The other reason why a qubit is not a better information storage than the bit is that we cannot copy qubits, or any quantum system, since copying it would involve measuring it, which would collapse the superposition and render the whole process useless.[8]

The question as to why something with two base states is chosen instead of three or more may also arise. Those systems are also a possibility - systems with three base states are called qutrits, but, like its classical origin point trit, qutrit isn't widely used, partly because of simplicity and partly because we already have a solid basis in bits. Therefore, we will only deal with qubits.

3.3 Bloch sphere

Let us observe a given qubit of the form $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ where $\alpha, \beta \in \mathbb{C}, |\alpha|^2 + |\beta|^2 = 1$. It can be easily shown that there exists a $c \in \mathbb{C}$ such that $|c| = 1$ and $c|\psi\rangle = \alpha'|0\rangle + \beta'|1\rangle$, where $\alpha' \in \mathbb{R}$, $\alpha' \geq 0$. This can be written as $c|\psi\rangle = \cos\frac{\theta}{2}|0\rangle + e^{i\phi}\sin\frac{\theta}{2}|1\rangle$, for some $\phi, \theta \in \mathbb{R}$.

Qubits $|\psi\rangle$ and $c|\psi\rangle$ are indistinguishable through measurement and any combination of quantum gates, which we will define later on. This means that we can treat any qubit as being equal to $\cos\frac{\theta}{2}|0\rangle + (\cos\phi + i\sin\phi)\sin\frac{\theta}{2}|1\rangle$, where $\theta \in [0, \pi], \phi \in [0, 2\pi]$, producing a map from the set of qubits onto a unit sphere, where we treat θ and ϕ as polar coordinates (Figure 5a).

[7] α^* represents the complex conjugate of α, $< *, * >$ is the standard scalar product in $\mathbb{C}^2$

[8] a more detailed reasoning can be found in [4]

The poles of the Bloch sphere represent $|0\rangle$ and $|1\rangle$. Any two orthogonal qubits are joined to antipodal points of the Bloch sphere (Figure 5b). Therefore, any two antipodal points represent a single orthogonal basis.

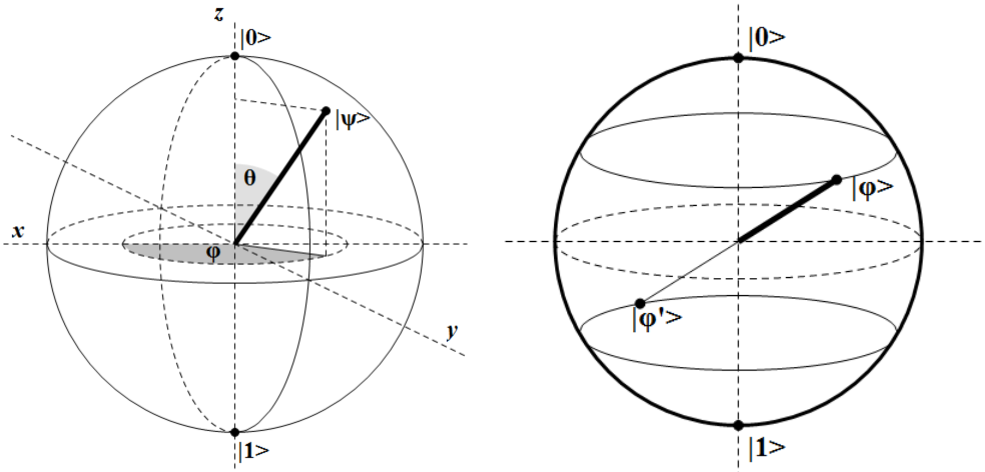

Figure 5: a) Bloch sphere, b) Orthogonal qubits on the Bloch sphere

3.4 Quantum registers

Intuitively, given two qubits x and y, since both x and y can be measured in two possible states, the quantum register containing only x and y has 4 possible states. We therefore denote the (new) base states as $|00\rangle, |01\rangle, |10\rangle, |11\rangle$[9], and all possible register states are linear combinations of these base states with complex coefficients. The two-qubit system is therefore constructed within $\mathbb{C}^4$. To do that, we use the standard tensor product. To recap (without the usual mathematical rigour), the tensor product from two 2-dimensional vector spaces is: $(x, y)^t \otimes (z, w)^t = (xz, xw, yz, yw)^t$, and the general tensor product $\otimes : \mathbb{C}^k \times \mathbb{C}^l \to \mathbb{C}^{kl}$ is defined with:

$$v \otimes w := \begin{pmatrix} v_1 w \\ \vdots \\ v_j w \\ \vdots \\ v_k w \end{pmatrix}, \, v \in \mathbb{C}^k, w \in \mathbb{C}^l,$$

where v_j is the j-th component of vector v. $\mathbb{C}^{kl}$ can, therefore, be constructed using the tensor product on all base vectors. It is obvious that our n-qubit system is contained in the

[9] base states in a two-qubit system are sometimes denoted $|0\rangle, |1\rangle, |2\rangle, |3\rangle$ (etc. for systems with more than two qubits); this plays a role in some algorithms where the input might be a natural number n represented precisely by the n-th base state of that quantum register

vector space $\mathbb{C}^{2^n}$.
The definition of the tensor product is easily extended to matrices, which plays a vital role in constructing multi-qubit quantum gates.
Using the tensor product, we can easily combine qubits into elements of the extended system. Indeed, $\{|0\rangle, |1\rangle\} \otimes \{|0\rangle, |1\rangle\}$ is the standard basis for $\mathbb{C}^4$, where $|0\rangle \otimes |0\rangle = |00\rangle, |1\rangle \otimes |0\rangle = |10\rangle$ etc., and similarly for an arbitrary quantum register
Since our quantum register also represents a quantum system and the amplitudes provide us with probabilities, it is obvious that we have to generalize the restriction regarding the amplitudes in a single qubit system: the norm of our quantum register (when we consider it to be a vector in $\mathbb{C}^n$) must be equal to 1. In other words: any given quantum register $|\psi\rangle = \alpha_1|00...0\rangle + ... + \alpha_n|11...1\rangle$ must satisfy the equation $\sum_{k=1}^{n} |\alpha_k|^2 = 1$, where $|\alpha_k|^2$ is the probability of measuring the corresponding base state when measuring the entire quantum register. As demonstrated, this is a natural generalization of a single qubit system discussed previously and many of the properties remain unchanged.

3.5 Entangled Qubits

Consider a quantum register defined with $|\psi\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{2}|10\rangle + \frac{1}{2}|11\rangle$. When measuring the first qubit of this quantum register, if the measurement gives us $|0\rangle$, the second qubit also collapses completely, since the only possible measurement result for the second qubit is obviously $|0\rangle$. On the other hand, were we to measure $|1\rangle$ on the first qubit, our quantum register would transform into $|\psi\rangle = \frac{1}{\sqrt{2}}|10\rangle + \frac{1}{\sqrt{2}}|11\rangle$, which, when measuring the second qubit, has the same probability of giving $|0\rangle$ and $|1\rangle$, while, originally, it was somewhat more probable to measure $|0\rangle$. (*Note:* we also had to resize the amplitudes to conserve the norm of our two-qubit register.)
Conversely, consider $|\phi\rangle = \frac{1}{2}|00\rangle + \frac{1}{2}|01\rangle + \frac{1}{2}|10\rangle + \frac{1}{2}|11\rangle$. It is evident that the result of any measurement of the state of the first qubit won't influence the results of measurement over the second qubit.
It can be easily shown that measurement over an individual qubit won't influence the remaining qubits if and only if the quantum register can be described as a direct tensor product of that qubit and the remaining multi-qubit vector.
When we also take into consideration that measurement changes the behaviour of that system, meaning that even a partial measurement can alter the behaviour of our quantum register, we can see that this effect corresponds to the physical effect of quantum entanglement. Indeed, this is one of the justifications for our model. This produces the following definition: A multi-qubit state is *entangled* if it cannot be factored into a direct product of individual qubits.
Entanglement is an interesting property since, given two or more entangled qubits, transformations over one qubit affect the entire register. This will play a crucial role in many quantum algorithms.

3.6 Quantum gates

Having defined qubits and quantum registers, it is now time to introduce the way we manipulate them. Firstly, it is obvious that any manipulation over a qubit or a quantum register must retain the norm. Furthermore, given that we are operating over superpositions, our

transformations will be defined over base states and the definition expanded for an arbitrary qubit through linearity of the transformation. These conditions gives us our definition:
A quantum gate is a unitary transformation.

NOT - gate

Let us consider the classical logical NOT gate. Its purpose is simple: it maps 0 to 1 and 1 to 0. Using that as a starting point for constructing the quantum NOT gate, it is obvious that we have our work cut out for us; since we already know that $|0\rangle \rightarrow |1\rangle$ and conversely $|1\rangle \rightarrow |0\rangle$, the quantum NOT - gate representation in matrix form is:

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

This gate is also often called Pauli-X gate since it corresponds to the rotation of a point on the Bloch sphere by π radians around the X-axis. There are also Pauli-Y and Pauli-Z gates represented as:

$$Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \text{ and } Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix},$$

respectively.

Hadamard gate

The Hadamard gate is deceptively simple; it is represented as

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix},$$

when dealing with a singular qubit. This allows us to transform both $|0\rangle$ and $|1\rangle$ into states that have equal probabilities of measuring both base states. We can prepare the quantum register in base states, and apply the Hadamard gate to each individual qubit to obtain a state that has equal probability of being measured in every single base state possible; a useful trick, and the reason most quantum algorithms contain the Hadamard gate.
Its generalisation on the n-dimensional quantum register is defined recursively with

$$H_n = \frac{1}{\sqrt{2}} \begin{pmatrix} H_{n-1} & H_{n-1} \\ H_{n-1} & -H_{n-1} \end{pmatrix},$$

and it can be shown that this corresponds to applying the Hadamard transform to each individual qubit (if, of course, $n = 2^m$, for $m \in \mathbb{N}$ representing the number of qubits in that register).
It should also be noted that the Hadamard transform is its own inverse - applying the Hadamard gate twice to the same quantum register would leave the register unchanged.

Phase-shift gate

For a given θ, a phase-shift gate is defined as:

$$\Theta = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix}$$

This corresponds to the mapping of $|0\rangle$ to $|0\rangle$ and $|1\rangle$ to $e^{i\theta}|1\rangle$. It leaves the probabilities of measuring $|0\rangle$ or $|1\rangle$ unchanged, but still makes a detectable change in the qubit.

Applying, consecutively, the Hadamard gate, the θ phase shift gate, the Hadamard gate (again) and the $\frac{\pi}{2} + \phi$ gate on $|0\rangle$, we obtain a qubit in the phase $\cos\frac{\theta}{2}|0\rangle + e^{i\phi}\sin\frac{\theta}{2}|1\rangle$ (Figure 6).

This, combined with the results given in the section 3.3, means that we can construct any unitary trasformation over a single qubit using only Hadamard and phase shift gates.

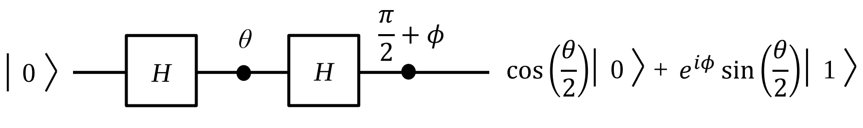
$|0\rangle$
H
θ
H
$\frac{\pi}{2} + \phi$
$\cos\left(\frac{\theta}{2}\right)|0\rangle + e^{i\phi}\sin\left(\frac{\theta}{2}\right)|1\rangle$

Figure 6: Construction of an arbitrary qubit

CU gate

Controlled gates are common in classical computing; we may or may not want to perform a transformation over a given memory register, and we control that using some other memory register, usually a single bit.
Given an arbitrary unitary transformation U over a single qubit:

$$U = \begin{pmatrix} x_1 & x_2 \\ x_3 & x_4 \end{pmatrix}$$

a CU gate is constructed as:

$$CU = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & x_1 & x_2 \\ 0 & 0 & x_3 & x_4 \end{pmatrix}$$

This transformation corresponds precisely to a transformation in which the first qubit is left unchanged and the other is unchanged when the first is $|0\rangle$ and changed by the transformation U if the first is $|1\rangle$. This transformation, of course, works even when the first qubit is in neither of these base states.

Quantum Fourier Transform

We will not discuss the importance of the Fourier transform in mathematics, nor explain its usage. This section will give but a brief definition of the Quantum Fourier transform, which will later be used in quantum algorithms.

QFT can be defined simply through its matrix as:

$$QFT = \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & \omega & \omega^2 & \dots & \omega^{N-1} \\ 1 & \omega^2 & \omega^4 & \dots & \omega^{2(N-1)} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \omega^{N-1} & \omega^{2(N-1)} & \dots & \omega^{(N-1)(N-1)} \end{pmatrix},$$

where N is the dimension of the complex vector space ($N = 2^n$ when dealing with the register of n qubits), and $\omega = e^{\frac{2\pi i}{N}}$.
This will map an arbitrary quantum register in the state $|\phi\rangle = \sum_{k=0}^{N-1} x_k|k\rangle$ to

$$\sum_{k=0}^{N-1} y_k|k\rangle, \text{where } y_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j\omega^{jk}.$$

4 Quantum algorithms (by Marija Kranjčević)

4.1 Deutsch's Algorithm

Deutsch's[10] algorithm is a very simple, deterministic[11] algorithm, which solves a somewhat contrived problem:

> Given a function $f : \{0,1\} \to \{0,1\}$, determine if f is balanced (i.e. if $f(0) \neq f(1)$) or constant ($f(0) = f(1)$).

A trivial solution, which can be carried out by any classical computer, is to compute both $f(0)$ and $f(1)$ and compare them. However, using the fact that a quantum system can be in several states simultaneously, a quantum computer produces a result with only one evaluation of the given function.
In a classical system, evaluating the function f would correspond to performing the operation which maps x to $f(x)$. However, an arbitrary function f doesn't have to be a unitary operation, whereas the operations performed on quantum states must be unitary. Therefore, to evaluate f on a quantum system, it is necessary to have a quantum gate which can be easily constructed from the function f. Such an operator can be defined with $U_f(|x\rangle|y\rangle) = |x\rangle|y \oplus f(x)\rangle$, where $\oplus$ is the exclusive operation (XOR). It is easy to see that $U_f^{-1} = U_f$ and that U_f is a unitary operator. *Note*: the input of the quantum algorithm is only the black-box[12] unitary operator U_f.
The steps of Deutsch's algorithm are as follows (Figure 7):

1. Take two qubits, in states $|0\rangle$ and $|1\rangle$, respectively. It is obvious that

$$|\varphi_0\rangle = |0,1\rangle.$$

[10] David Elieser Deutsch (1953.-), Israeli-British physicist

[11] Always gives the correct answer.

[12] A device or theoretical construct whose inputs and outputs can be known, but there is no information about its internal structure.

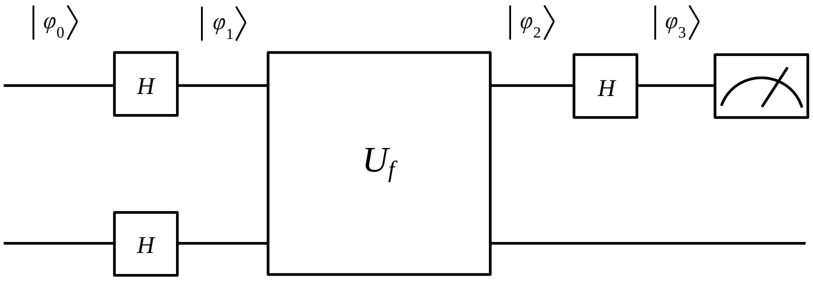

Figure 7: Deutsch's algorithm

2. Apply the Hadamard gate to both qubits to put them in a superposition of states. The state is now

$$|\varphi_1\rangle = \frac{|0\rangle + |1\rangle}{\sqrt{2}} \frac{|0\rangle - |1\rangle}{\sqrt{2}} = \frac{|0,0\rangle - |0,1\rangle + |1,0\rangle - |1,1\rangle}{2}.$$

3. Apply U_f to get

$$\begin{aligned} |\varphi_2\rangle &= U_f(\frac{|0,0\rangle - |0,1\rangle + |1,0\rangle - |1,1\rangle}{2}) \\ &= \frac{|0,0 \oplus f(0)\rangle - |0,1 \oplus f(0)\rangle + |1,0 \oplus f(1)\rangle - |1,1 \oplus f(1)\rangle}{2} \\ &= \frac{|0,f(0)\rangle - |0,\overline{f(0)}\rangle + |1,f(1)\rangle - |1,\overline{f(1)}\rangle}{2}, \end{aligned}$$

where, for $x \in \{0,1\}$, $|\overline{x}\rangle$ denotes the opposite value of $|x\rangle$.
This can be compactly written as

$$\begin{aligned} |\varphi_2\rangle &= [\frac{(-1)^{f(0)}|0\rangle + (-1)^{f(1)}|1\rangle}{\sqrt{2}}][\frac{|0\rangle - |1\rangle}{\sqrt{2}}] \\ &= \begin{cases} (\pm)\frac{|0\rangle+|1\rangle}{\sqrt{2}}\frac{|0\rangle-|1\rangle}{\sqrt{2}} & \text{if } f \text{ is constant,} \\ (\pm)\frac{|0\rangle-|1\rangle}{\sqrt{2}}\frac{|0\rangle-|1\rangle}{\sqrt{2}} & \text{if } f \text{ is balanced.} \end{cases} \end{aligned}$$

4. Apply the Hadamard gate to the first qubit. The state of the system becomes

$$|\varphi_3\rangle = \begin{cases} (\pm)|0\rangle\frac{|0\rangle-|1\rangle}{\sqrt{2}} & \text{if } f \text{ is constant,} \\ (\pm)|1\rangle\frac{|0\rangle-|1\rangle}{\sqrt{2}} & \text{if } f \text{ is balanced.} \end{cases}$$

5. Measure the state of the first qubit. The measured result gives the solution of the initial problem. If it is in the state $|0\rangle$, then f is a constant function; otherwise, f is a balanced function.

A quantum system allows for all the needed computations to be done in parallel, so multiple results can be obtained by a single evaluation. This approach shows that, with a little manipulation to obtain a useful result, quantum algorithms, and quantum computers, can solve certain problems much faster than their classical counterparts. Basically, while the solution to this problem is not a very useful one, it shows the possibility of the use of quantum computers and the main idea how the properties of quantum-mechanical systems can be used to speed-up the solutions to some problems.[13]

4.2 Grover's Algorithm

Another quantum algorithm is Grover's[14] algorithm for searching an unstructured database. Using classical computation, that problem cannot be solved in less than linear time, i.e. searching through every item. But, while classical algorithms need to successively test the indices, a quantum algorithm can test several indices at once, using quantum parallelism. Therefore, Grover's algorithm is much faster and performs the search of a database with N entries in only $O(\sqrt{N})$ steps.
The problem of searching an unstructured database can be stated as follows:

> Take an unstructured database with N entries with indices in the range from 0 to $N-1$. Suppose one of these element is tagged, i.e. let there be a black box function f_t, for some fixed t in the range from 0 to $N-1$, such that, for every index x, $f_t(x) = \delta_{xt}$. The task is to find the index t using the fewest calls to the function f_t.

For simplicity, it can be assumed that $N = 2^n$, for some natural number n (if that is not the case, a register of $n = \lceil \log_2 N \rceil$ qubits can be used). The algorithm then requires a register of n qubits to represent the indices of the database entries. (Also, a second register, consisting of a single qubit, is needed for constructing the gates.)
The idea of Grover's algorithm is to put the first register into an equal superposition of all indices, and then gradually increase the amplitude of the searched-for index t. Thus, a measurement after a certain number of steps is likely to give the index t.
The steps of the algorithm are as follows[15] (Figure 8):

1. Set the first register to $|s\rangle = \frac{1}{\sqrt{N}} \sum\limits_{x=0}^{N-1} |x\rangle$.

2. Use the function f_t to construct a black box unitary operator U_t, which acts as $U_t(|x\rangle) = (-1)^{f_t(x)}|x\rangle$ i.e. $U_t = I - 2|t\rangle\langle t|$. Also, construct a unitary operator U_s, defined with $U_s = 2|s\rangle\langle s| - I$.

3. Apply the unitary transformation $U_s U_t$ (called the Grover iterate) to the first register $\lfloor \frac{\pi\sqrt{N}}{4} \rfloor$ times.

[13] the presented explanation of Deutsch's algorithm was obtained from [9]
[14] Lov Kumar Grover (1961-), Indian-American computer scientist
[15] based mainly on [1]

4. Measure the first register. The obtained result is likely to be the index t.

As seen in Deutsch's algorithm, to evaluate an arbitrary function f_t on a quantum system, it is necessary to build a quantum gate out of it. Therefore, f_t is used to construct a unitary operator U_t, defined with $U_t(|x\rangle) = (-1)^{f_t(x)}|x\rangle$ (*Note:* $|x\rangle$ represents the state of the quantum register of n qubits). Also, considering that the indices are represented by vectors of the orthonormal basis (i.e. $\langle x|x\rangle = 1, \forall x$, and $\langle t|x\rangle = 0, \forall x \neq t$), U_t can be written as $U_t = I - 2|t\rangle\langle t|$.
The operator U_t can be constructed using a unitary operation which maps $|x\rangle|q\rangle \mapsto |x\rangle|q \oplus f(x)\rangle$ and setting the control qubit to $|q\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}}$. It is obvious from the construction that U_t is a unitary operator. Analogously, it is easy to see that U_s is also unitary.

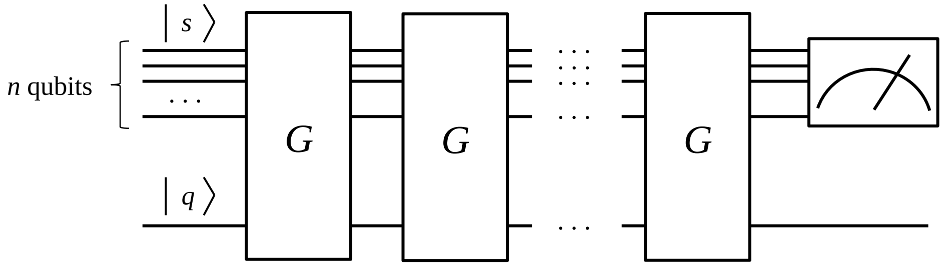

Figure 8: Grover's algorithm

The process of amplitude amplification can be demonstrated by the first iteration, which works as follows (using $\langle t|s\rangle = \langle s|t\rangle = \frac{1}{\sqrt{N}}$ and $\langle s|s\rangle = 1$):

$$U_s U_t|s\rangle = U_s(I - 2|t\rangle\langle t|)|s\rangle = (2|s\rangle\langle s| - I)(|s\rangle - \frac{2}{\sqrt{N}}|t\rangle) = \frac{N-4}{N}|s\rangle + \frac{2}{\sqrt{N}}|t\rangle.$$

Thus, by applying the operation $U_s U_t$, the probability of measuring t has increased from $|\langle t|s\rangle|^2 = \frac{1}{N}$ to $|\langle t|U_s U_t|s\rangle|^2 \approx \frac{9}{N}$.

To understand the algorithm and determine the number of iterations which have to be performed, it is easiest to consider a geometric interpretation.[16]
U_t flips the state $|t\rangle$ and does nothing to all other states. Therefore, it can be seen as a reflection across the hyperplane perpendicular to $|t\rangle$.
Also, it is easy to see that $U_s|s\rangle = |s\rangle$, and $U_s|v\rangle = -|v\rangle, \forall v \perp s$. Thus, U_s is a reflection across $|s\rangle$.
Considering that the amplitudes of the non-searched states are equal to each other, it suffices to observe the transformations in the plane spanned by $|t\rangle$ and $|u\rangle := \frac{1}{\sqrt{N-1}} \sum_{\substack{x=0 \\ x \neq t}}^{N-1} |x\rangle$.

[16] based on [6]

(*Note:* $t \perp u$.) It follows that U_t and U_s are reflections in the same plane, which means that U_sU_t is a rotation.
Let θ denote the angle between $|s\rangle$ and $|u\rangle$. Also, let $|\psi_0\rangle$ denote a state obtained by an arbitrary number of iterations. With other variables denoted as in Figure 9, an individual iteration works as follows:

$$|\psi_2\rangle := U_sU_t|\psi_0\rangle = (2|s\rangle\langle s| - I)U_t|\psi_0\rangle = 2\langle s|\psi_1\rangle|s\rangle - |\psi_1\rangle = 2\cos\beta|s\rangle - |\psi_1\rangle$$

$$\begin{aligned}\langle\psi_0|\psi_2\rangle &= 2\cos\beta\langle\psi_0|s\rangle - \langle\psi_0|\psi_1\rangle = 2\cos\beta\cos\alpha - \cos(\alpha+\beta) = \cos(\beta-\alpha)\\ &= \cos 2\theta.\end{aligned}$$

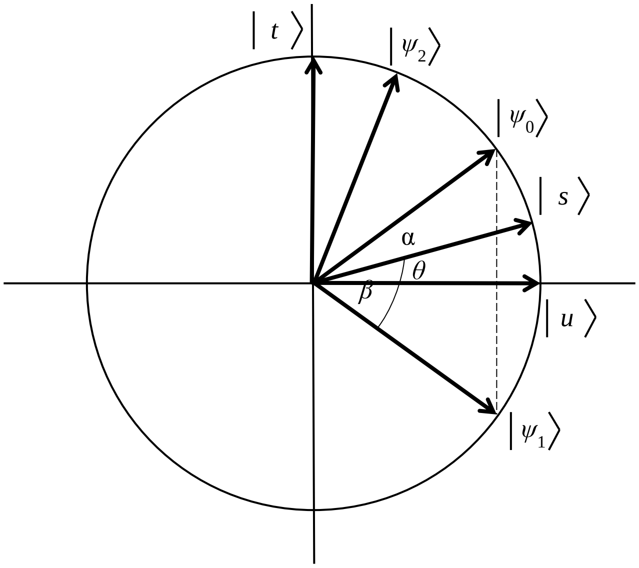

Figure 9: Geometric interpretation of Grover's algorithm

Having in mind that kets are unit vectors, it is easy to see that a single iteration rotates the state of the first register by the angle 2θ towards the index t.
Note: after a certain number of steps, the applications of the Grover iterate will start to rotate the state $|\psi\rangle$ away from $|t\rangle$ i.e the amplitude of t oscillates. Therefore, it is necessary to determine the smallest number of iterations which, when measured, give t with a probability close to 1. That number, denoted m, has to satisfy $(2m+1)\theta \approx \frac{\pi}{2}$, or, equally, $m \approx \frac{\pi}{4\theta} - \frac{1}{2}$. Therefore, $m \approx \lfloor \frac{\pi}{4\theta} \rfloor$.
Considering that $\sin\theta = \cos(\frac{\pi}{2} - \theta) = \langle t|s\rangle = \frac{1}{\sqrt{N}}$, and that for large N $\sin\theta \approx \theta$, the number of necessary iterations can be seen to be approximately $\lfloor \frac{\pi\sqrt{N}}{4} \rfloor$.

The probability of failure (taking $(\frac{\pi}{2} - (2m+1)\theta) < \theta$) is

$$p = \cos^2((2m+1)\theta) = \sin^2(\frac{\pi}{2} - (2m+1)\theta) < \sin^2\theta = \frac{1}{N}.$$

If quantum computers are built, Grover's algorithm could be used for various problems, such as finding the mean and the median of a set, and solving the collision problem or the molecular distance geometry problem. It could also have a very important role in cryptography. It is theoretically possible to use it to crack the symmetric cyphers (i.e. cyphers that use the same or trivially related private key for both the encryption and decryption of data) such as the DES (Data Encryption Standard), Triple DES, and AES (Advanced Encryption Standard). DES is one of the most popular encryption standards. It is a block cypher (i.e. the plaintext is divided into blocks and each of them is encrypted/decrypted separately) that relies on a 56-bit secret number, which is used as a key to encrypt/decrypt a block of data. It was developed in the 70's, by the IBM, on a request from NASA, but is now considered to be insecure, mostly because a 56-bit key is too small and can be found with today's computers. Its improvement, the Triple DES applies the DES three times to each data block, and is considered practically secure. AES is another symetric-key cypher, which superseded DES as FIPS[17], the official code for use in computer systems in the US, by all non military government agencies and government contractors. Those and similar cyphers, some of which are also used to protect financial transactions between banks, rely on the fact that it is practically impossible for a current, classical computer to break them by finding the secret key. A quantum computer, on the other hand, along with Grover's algorithm, would be able to do just that.

4.3 Shor's algorithm

Another cryptographic system is public-key cryptography, with the RSA[18] algorithm as one of its more important ways of encrypting/decrypting data. RSA uses a public key to encrypt data and a private key to decrypt it. The private key is obtained from two large prime numbers, and the public key is obtained from their product. The recipient chooses the private key, easily generates the public key and gives it to the sender. If anyone else finds out what the public key is, it is still of no use because a large number cannot be factored by today's computers. A quantum computer, however, would have a better chance.

While best classical factoring algorithms require $O(e^{1,9(\log N)^{\frac{1}{3}}(\log\log N)^{\frac{2}{3}}})$ time, the quantum algorithm, called Shor's factoring algorithm, requires only $O((\log N)^3)$. Devised in 1994. by Peter Shor[19], it is the most famous quantum algorithm, which prompted the design and construction of quantum computers, as well as the study of other quantum algorithms.

The factoring problem can be stated as follows:

> Given a natural number N, find its nontrivial factor.

[17] Federal Information Processing Standard

[18] Ronald L. Rivest, Adi Shamir and Leonard Max Adleman

[19] Peter Williston Shor (1959.-), American mathematician

It can be assumed that N is composite, odd and not a prime power. (There are efficient classical methods to determine if this is true, and finding the factors for N it if it is. Therefore, in those cases, there is no need for Shor's algorithm.)

Shor's algorithm can be divided into two parts: the classical part, which can be done on a classical computer, and the quantum part, the order finding subroutine, for which it is necessary to have a quantum computer.

The classical part - reducing factoring to order finding:

1. Pick a random number $1 < a < N$ and compute the greatest common divisor $\gcd(a, N)$. This can be done efficiently with Euclid's algorithm[20]. If $\gcd(a, N) \neq 1$, congratulations, you are extremely lucky and don't even need a factoring algorithm; $\gcd(a, N)$ is a nontrivial factor of N.
If that is not the case, proceed to step 2.

2. Using a quantum computer, find the order of a in the quotient group $(\mathbb{Z}/N\mathbb{Z})^{\times}$, i.e. the least natural number r such that $a^r \equiv 1 \pmod{N}$. (Equivalently, find the period of the function $f(x) = a^x \pmod{N}$.)

3. If $2 \nmid r$ or $a^{\frac{r}{2}} \equiv -1 \pmod{N}$, go back to step 1.
Otherwise, $N|(a^r - 1) = (a^{\frac{r}{2}} + 1)(a^{\frac{r}{2}} - 1)$. Considering that $a^{\frac{r}{2}} \not\equiv -1(\text{mod } N)$, and that r is the order of a in $(\mathbb{Z}/N\mathbb{Z})^{\times}$, it follows that $\gcd(a^{\frac{r}{2}} \pm 1, N)$ are nontrivial factors of N. (The existence of $a^{\frac{r}{2}}$ such that $a^{\frac{r}{2}} \not\equiv \pm 1 \pmod{N}$ is guaranteed by the Chinese remainder theorem[21]: Since N is composite, odd, and not a prime power, there exist p and q such that $p, q > 2$, $\gcd(p, q) = 1$ and $N = pq$. From the Chinese remainder theorem it follows that there exists x such that $x \equiv 1 \pmod{p}$ and $x \equiv -1 \pmod{q}$. It can be easily checked that $x^2 \equiv 1 \pmod{N}$ and $x \not\equiv \pm 1 \pmod{N}$.)

The quantum part - the order finding subroutine (Figure 10):

The order finding subroutine needs two quantum registers. The first with q qubits, where $2^q = Q$ and $N^2 \leq Q < 2N^2$, because it has to be able to store numbers from 0 to $Q - 1$, and the second with q' qubits, where q' is such that the second register can store numbers up to $N - 1$. (*Note:* it is possible to take $q' = q$.) Also, the function f is used to build a quantum gate U_f, which acts as: $U_f|x\rangle|y\rangle = |x\rangle|(y + f(x)) \mod 2^{q'}\rangle$. (*Note:* $|x\rangle$ and $|y\rangle$ represent the states of first and second register, respectively. Therefore, $|x\rangle$ and $|y\rangle$ are vectors in the complex vector spaces of dimension 2^q and $2^{q'}$, respectively.)

1. Set the registers to $|\varphi_0\rangle = |0\rangle|0\rangle$.

[20] Appendix, Theorem 1.
[21] Appendix, Theorem 2.

2. Apply the Walsh-Hadamard matrix to the first register to put it in an equal superposition of all states. The state of the registers becomes

$$|\varphi_1\rangle = (H \otimes I)|\varphi_0\rangle = \frac{1}{\sqrt{Q}} \sum_{x=0}^{Q-1} |x\rangle|0\rangle.$$

3. Apply U_f (to both registers). Thus, due to the property of superposition, the transformation $|x\rangle|0\rangle \mapsto |x\rangle|f(x)\rangle = |x\rangle|a^x \bmod N\rangle$ is applied for every x contained in the first register, in a single step. The state of the registers is now

$$|\varphi_2\rangle = U_f|\varphi_1\rangle = \frac{1}{\sqrt{Q}} \sum_{x=0}^{Q-1} |x\rangle|f(x)\rangle.$$

4. Measure the state of the second register, and call the measured value k. Then $k = f(x_0)$, for a certain x_0, uniformly random over $0, 1, \ldots, r-1$. Because of the entanglement between the registers, the first register is now in an equal superposition of all integers x such that $f(x) = k$. These integers are spaced apart by r, i.e. the first register contains a periodic superposition of period r. The state of the registers can be written as

$$|\varphi_3\rangle = \frac{1}{\sqrt{\lfloor \frac{Q}{r} \rfloor}} \sum_{j=0}^{\lfloor \frac{Q}{r} \rfloor - 1} |x_0 + jr\rangle|k\rangle.$$

5. Apply the quantum Fourier transform U_{QFT} to the first register. Denoting $\omega = e^{\frac{2\pi i}{Q}}$, this gives

$$|\varphi_4\rangle = \frac{1}{\sqrt{\lfloor \frac{Q}{r} \rfloor}} \sum_{j=0}^{\lfloor \frac{Q}{r} \rfloor - 1} (U_{QFT}|x_0 + jr\rangle)|k\rangle = \frac{1}{\sqrt{\lfloor \frac{Q}{r} \rfloor}} \sum_{j=0}^{\lfloor \frac{Q}{r} \rfloor - 1} \frac{1}{\sqrt{Q}} \sum_{b=0}^{Q-1} \omega^{(x_0+jr)b}|b\rangle|k\rangle$$

$$= \frac{1}{\sqrt{\lfloor \frac{Q}{r} \rfloor}} \frac{1}{\sqrt{Q}} \sum_{b=0}^{Q-1} \omega^{x_0 b}(\sum_{j=0}^{\lfloor \frac{Q}{r} \rfloor - 1} (\omega^{rb})^j)|b\rangle|k\rangle.$$

While the validity of the following steps can be seen approximately in the general case, it is completely straightforward in case $r|Q$. Therefore, both cases will be presented.
Suppose $r|Q$. Then the state of the registers can be written as

$$|\varphi_4\rangle = \frac{\sqrt{r}}{Q} \sum_{b=0}^{Q-1} \omega^{x_0 b}(\sum_{j=0}^{Q/r-1} (\omega^{rb})^j)|b\rangle|k\rangle.$$

Using $\sum\limits_{j=0}^{n-1} a^j = \frac{1-a^n}{1-a}$, for $a < 1$, and $\frac{1-(\omega^{rb})^{Q/r}}{1-\omega^{rb}} = \frac{1-e^{2\pi i b}}{1-e^{\frac{2\pi i r b}{Q}}} = 0$ (if $\omega^{rb} \neq 1$),

$$\sum_{j=0}^{Q/r-1} (\omega^{rb})^j = \begin{cases} Q/r & \text{if } \omega^{rb} = 1, \\ 0 & \text{if } \omega^{rb} \neq 1. \end{cases}$$

$\omega^{rb} = e^{\frac{2\pi i rb}{Q}} = 1$ iff $Q|rb$ iff b is a multiple of Q/r. Therefore, the first register is in a superposition where only the multiples of Q/r have non-zero amplitudes.
In the general case, using the fact that ω is a root of unity, the probability of measuring a state b is

$$\frac{1}{\left\lfloor \frac{Q}{r} \right\rfloor} \frac{1}{Q} \, |\omega^{x_0 b}|^2 | \sum_{j=0}^{\lfloor \frac{Q}{r} \rfloor - 1} (\omega^{rb})^j |^2,$$

which is greater the closer ω^{rb} is to 1. $\omega^{rb} = e^{\frac{2\pi i}{Q} rb}$ is 'close' to 1 iff $\frac{rb}{Q} \approx n$, for some $n \in \mathbb{N}$.

6. Measure the first register, and call the measured value y.
If $r|Q$, $y = \frac{nQ}{r}$, for some $0 \leq n < r$, i.e. $y/Q = n/r$. If $\gcd(r, n) = 1$, r is simply the denominator of y/Q written in lowest possible terms. Otherwise, it is necessary to repeat the algorithm.
In the general case, it can be shown that, with a high probability, y is 'close' to a multiple of Q/r, i.e. $\left| \frac{y}{Q} - \frac{n}{r} \right| < \frac{1}{2Q}$, for some $n \in \mathbb{N}$. In view of the Convergent approximation theorem[22] (considering $r^2 \leq N^2 \leq Q$), apply the theory of continued fractions[23] on $\frac{y}{Q}$ to obtain r' such that $r' < N$. It is easy to check if $r' = r$. If not, e.g. because $\gcd(r, n) \neq 1$, it is necessary to repeat the algorithm.

This algorithm is probabilistic[24], so it might be necessary to run it several times to obtain the period r. However, it is still more efficient than any known classical algorithm.[25]

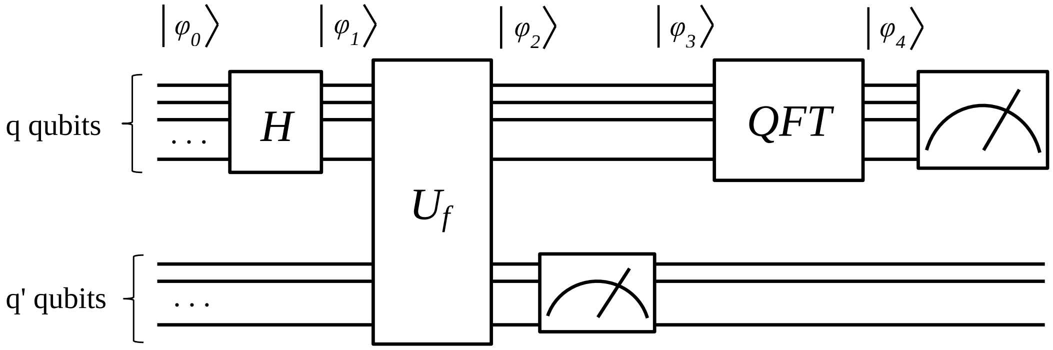

Figure 10: Quantum circuit for Shor's algorithm

[22] Appendix, Theorem 3.
[23] Appendix, Definition 3.
[24] Gives the correct answer with a certain probability $p < 1$.
[25] the explanation of Shor's algorithm is based on [5], [7], and [8]

Appendix: Basics of number theory[26]

Definition 1. *Let $d, a \in \mathbb{Z}, (d \neq 0)$. d **divides** a $(d|a)$ if there exists $c \in \mathbb{Z}$ such that $a = cd$. Otherwise, d does not divide a $(d \nmid a)$. $a \in \mathbb{N}$, $a > 1$, is called **composite** if there exists a natural number d, $1 < d < a$, s.t. $d|a$. Otherwise, a is called **prime**. If $b \in \mathbb{Z}$ and $d|b$, d is a **common divisor** of a and b. If $a \neq 0$ or $b \neq 0$, there are finitely many common divisors of a and b. The largest among them is called the **greatest common divisor** of a and b, denoted gcd(a,b). Note: $gcd(a,b) > 0$. a and b are **coprime** if gcd(a,b)=1. $a_1, a_2, \ldots, a_n \in \mathbb{Z}$ are **pairwise coprime** if gcd(a_i, a_j)=1, $\forall i, j$ s.t. $1 \leq i < j \leq n$.*

Lemma 3. *For $a \in \mathbb{N}$ and $b \in \mathbb{Z}$, there exist unique $q, r \in Z$ s.t. $b = qa + r$, $0 \leq r < b$.*

Proof. Consider the set $S := \{b - am : m \in \mathbb{Z}\}$, and denote $r := \min(S \cap \mathbb{N}_0)$. Then $0 \leq r < b$ and $\exists q \in \mathbb{Z}$ such that $b - qa = r$ i.e. $b = qa + r$.
Assume that there are q_1, r_1, which satisfy $b = q_1 a + r_1$ and $0 \leq r_1 < b$. Also, assume that $r_1 \neq r$. Without loss of generality, $r < r_1$. Then $0 < r_1 - r < a$ and $r_1 - r = (q - q_1)a \geq a$. Therefore, it must be $r_1 = r$ and $q_1 = q$. □

Lemma 4. *$gcd(a,b) = min(S := (\{ax + by : x, y \in \mathbb{Z}\} \cap \mathbb{N}))$.*

Proof. Let g:= gcd(a,b), l:= min(S). Therefore, $\exists x_0, y_0 \in \mathbb{Z}$ such that $l = ax_0 + by_0$. Assume that $l \nmid a$. From Lemma 1. it follows that $\exists q, r \in \mathbb{Z}$ such that $a = lq + r$, $0 < r < l$. Then $r = a - lq = a - (ax_0 + by_0)q = a(1 - x_0 q) + b(-y_0 q) \in S$, which is in contradiction with l=min(S). Therefore, r=0, i.e. $l|a$. Similarly, $l|b$. Therefore, $l \leq g$.
Considering g=gcd(a,b), $\exists a_0, b_0 \in \mathbb{Z}$ such that $a = a_0 g$ and $b = b_0 g$. Then $l = ax_0 + by_0 = (a_0 x_0 + b_0 y_0)g$, which means $g \leq l$. Therefore, $g = l$. □

Lemma 5. *gcd(a,b)=gcd(a,b+ax).*

Proof. Denote $d := gcd(a,b)$, $g := gcd(a, b+ax)$. From Lemma 4. it follows that $\exists x_0, y_0 \in \mathbb{Z}$ such that $d = ax_0 + by_0 = a(x_0 - xy_0) + (b + ax)y_0$. Therefore, $g|d$.
Considering that $d|a$ and $d|b$, $d|(b + ax)$. Therefore, $d|g$. Since l and g must be positive, it follows that $l = g$. □

Theorem 1. (Euclidean algorithm)
Take $a, b \in \mathbb{N}$. Assume that successive applications of Lemma 3. gave the following equations:

$$\begin{aligned} a &= bq_1 + r_1,\ 0 < r_1 < b, \\ b &= r_1 q_2 + r_2,\ 0 < r_2 < r_1, \\ r_1 &= r_2 q_3 + r_3,\ 0 < r_3 < r_2, \\ &\ldots \\ r_{i-2} &= r_{i-1} q_i + r_i,\ 0 < r_i < r_{i-1}, \\ r_{i-1} &= r_i q_{i+1}. \end{aligned}$$

Then $gcd(a,b) = r_i$. Furthermore, $i < 2log_2 b$.

[26] translated from [2]

Proof. According to Lemma 5,

$$\begin{aligned}\gcd(a,b) &= \gcd(a-bq_1,b) = \gcd(r_1,b) = \gcd(r_1,b-r_1q_2) = \gcd(r_1,r_2)\\ &= \ldots = \gcd(r_{i-1},r_i) = \gcd(r_i,0) = r_i.\end{aligned}$$

Consider step j. Either $r_j \leq \frac{r_{j-1}}{2}$ or $\frac{r_{j-1}}{2} < r_j < r_{j-1}$. If the second is true, $q_{j+1} = 1$ and $r_{j+1} = r_{j-1} - r_j < \frac{r_{j-1}}{2}$. In both cases $r_{j+1} < \frac{r_{j-1}}{2}$.
If i is even, $1 \leq r_i < \frac{r_{i-2}}{2} < \frac{r_{i-4}}{4} < \cdots < \frac{b}{2^{i/2}}$. If i is odd, $2 \leq r_{i-1} < \frac{r_{i-3}}{2} < \cdots < \frac{b}{2^{(i-1)/2}}$.
In both cases, $b > 2^{i/2}$ i.e. $i < 2\log_2 b$. □

Definition 2. *If $m \in \mathbb{Z}, m \neq 0$, divides $a-b$, a and b are* **congruent modulo** m*. Notation: $a \equiv b$ (mod m). Otherwise, a and b are not congruent modulo m, denoted $a \not\equiv b$ (mod m).*

Proposition 1. *Take $a,b,c,d,m \in \mathbb{Z}$, $m \neq 0$. If $a \equiv b$ (mod m) and $c \equiv d$ (mod m), then $a+c \equiv b+d$ (mod m), $a-c \equiv b-d$ (mod m), and $ac \equiv bd$ (mod m).*

Proof. Denote $a-b = mk$ and $c-d = ml$. Then $(a+c)-(b+d) = m(k+l)$, $(a-c)-(b-d) = m(k-l)$, and $ac-bd = a(c-d)+d(a-b) = m(al+dk)$. Therefore, $a+c \equiv b+d$ (mod m), $a-c \equiv b-d$ (mod m), and $ac \equiv bd$ (mod m). □

Theorem 2. (Chinese remainder theorem)
For $k \in \mathbb{N}$, $m_1, m_2, \ldots, m_k \in \mathbb{N}$, pairwise coprime, and $a_1, a_2, \ldots, a_k \in \mathbb{Z}$, there exists $x \in \mathbb{Z}$ which solves the system of simultaneous congruences:

$$x \equiv a_1 \ (mod\ m_1), \qquad x \equiv a_2 \ (mod\ m_2), \qquad \ldots, \qquad x \equiv a_k \ (mod\ m_k) \qquad (*)$$

Furthermore, all solutions are given with $y \equiv x$ (mod $m_1m_2\cdots m_k$).

Proof. Let $m = m_1m_2\cdots m_k$, and let $n_j = \frac{m}{m_j}$, for $j = 1,2,\ldots,k$. Then, for $j = 1,2,\ldots,k$, $\gcd(m_j,n_j) = 1$, so there exists x_j such that $n_jx_j \equiv a_j$ (mod m_j). Let

$$x = n_1x_1 + n_2x_2 + \cdots + n_kx_k.$$

Then for $j = 1,2,\ldots,k$

$$x \equiv 0 + \cdots + 0 + n_jx_j + 0 + \cdots + 0 \equiv a_j \ (\text{mod}\ _j).$$

Let x and y both satisfy (*). Then $y \equiv x$ (mod m_j), for $j = 1,2,\ldots,k$. Since m_j are pairwise coprime, $y \equiv x$ (mod m). □

Definition 3. *For a real number α_0, the* **continued fraction** *is defined as follows: for $i = 0,1,2,\ldots$, let $a_i = \lfloor \alpha_i \rfloor$, and $\alpha_i = a_i + \frac{1}{\alpha_{i+1}}$. If $a_n = \alpha_n$, for some natural number n, the procedure stops. Notation: $\alpha = [a_0, a_1, a_2, \ldots]$ or*

$$\alpha_0 = a_0 + \cfrac{1}{a_1 + \frac{1}{a_2 + \ldots}}.$$

For $n \in \mathbb{N}_0$, the fraction $\frac{p_n}{q_n} = [a_0, a_1, \ldots, a_n]$ is called the n-th convergent of α_0.

Theorem 3. (Convergent approximation theorem)
If $p, q \in \mathbb{Z}$*,* $q \geq 1$*, and* $\left|\alpha - \frac{p}{q}\right| < \frac{1}{2q^2}$*, then* $\frac{p}{q}$ *is a convergent of* α*.*

Proof. If $\alpha = \frac{p}{q}$, the statement is trivially true. Otherwise, $\alpha - \frac{p}{q} = \frac{\epsilon\theta}{q^2}$, where $\epsilon = \pm 1$, and $0 < \theta < \frac{1}{2}$. Let $\frac{p}{q} = [a_0, a_1, \ldots, a_{n-1}]$ such that $(-1)^n = \epsilon$. (This can always be achieved because $[a_0, a_1, \ldots, a_m] = [a_0, a_1, \ldots, a_m - 1, 1]$.) Also, let $\alpha = \frac{\omega p_{n-1} + p_{n-2}}{\omega q_{n-1} + q_{n-1}}$. Then $\alpha = [a_0, a_1, \ldots, a_{n-1}, \omega]$. Also,

$$\frac{\epsilon\theta}{q^2} = \alpha - \frac{p}{q} = \frac{1}{q_{n-1}}(\alpha q_{n-1} - p_{n-1}) = \frac{1}{q_{n-1}}(\frac{\omega p_{n-1} + p_{n-2}}{\omega q_{n-1} + q_{n-1}}\ q_{n-1} - p_{n-1})$$
$$= \frac{1}{q_{n-1}}(\frac{(-1)^{n-1}}{\omega q_{n-1} + q_{n-2}}).$$

Therefore, $\omega = \frac{1}{\theta} - \frac{g_{n-2}}{q_{n-1}} > 2 - 1 = 1$. Then, denoting $\omega = [a_n, a_{n+1}, \ldots]$, $a_j \in \mathbb{N}$, $\forall j \geq n$, which means that $\alpha = [a_0, a_1, \ldots, a_{n-1}, a_n, a_{n+1}, \ldots]$, and that $\frac{p}{q}$ is a convergent of α. □

References

[1] D. Aharonov, *Introduction to quantum computation: Grover's algorithm*, lecture notes for Quantum Computing (2001)

[2] A. Dujella, *Uvod u teoriju brojeva (Introduction to number theory)*

[3] A. Edalat, *Quantum Computing*, lecture notes

[4] A. Ekert, P. Hayden, H. Inamori, *Basic concepts in quantum computation*, arXiv:quant-ph/0011013v1, (2000)

[5] M. Hayward, *Quantum computing and Shor's algorithm*

[6] R. Solcà, *Grover's algorithm* (2008)

[7] C. P. Williams, *Explorations in quantum computing*, Springer-Verlag London Ltd, 2nd edt. (2011)

[8] R. de Wolf, *Quantum computation and Shor's factoring algorithm*, (1999)

[9] N. S. Yanovsky, *An introduction to quantum computing*, arXiv:0708.0261v1, (2007)