3 Quantum Algorithms

3.1 The Deutsch-Jozsa algorithm

Suppose that your task is to decide whether a function \( f : \{0, 1\}^n \to \{0, 1\} \) is either constantly equal to 0 or it is balanced, i.e. \( f(x) = 1 \) for precisely half of all inputs \( x \in \{0, 1\}^n \) (either one of these two cases is guaranteed to hold). If you decide correctly, you are awarded 1000€. On the other hand, a false answer is fatal. To help you find the right answer, you can repeatedly ask for the value of \( f \) for a given input \( x \). Each such query will set you back 2€.

Classically, there is a good chance to find the right answer by drawing an input \( x \) uniformly at random. Clearly, if \( f(x) = 1 \), you can be sure that \( f \) is balanced. On the other hand, if \( f \) is balanced, then the probability that \( f(x) = 0 \) for \( k \) inputs, chosen uniformly at random, is \( 1/2^k \), which converges to 0 exponentially fast. However, unless you query more than \( 2^n - 1 \) many inputs or get the answer that \( f(x) = 1 \), you cannot be sure of your answer.

Suppose now that you may query a QGA on \( n + 1 \) qubits for computing the function \( U_f \) defined by

\[
U_f |x\rangle |j\rangle = |x\rangle |f(x) \oplus j\rangle.
\]

Clearly, QGAs are more expensive than classical circuits, so let us say that each application of \( U_f \) costs 500€. Can you get the correct answer and still make money in this case?

Surprisingly, the answer is yes since there exists a QGA that decides whether \( f \) is balanced with just one application of \( U_f \):

\(^1\)Note that \( U_f \) has to be unitary.
3.1 The Deutsch-Jozsa algorithm

Let us examine what the circuit does: First, the vector $|0^n\rangle \otimes |1\rangle$ is mapped by $H^\otimes n + 1$ to

$$\frac{1}{\sqrt{2^{n+1}}} \sum_{x \in \{0,1\}^n} |x\rangle \otimes (|0\rangle - |1\rangle).$$

Second, the QGA for $U_f$ is applied to this vector, which yields the vector

$$\frac{1}{\sqrt{2^{n+1}}} \sum_{x \in \{0,1\}^n} (|x\rangle \otimes (-1)^f(x) (|0\rangle - |1\rangle))$$

$$= \left( \sum_{x \in \{0,1\}^n} \frac{(-1)^{f(x)} |x\rangle}{\sqrt{2^n}} \right) \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

$$= \left( \sum_{x \in \{0,1\}^n} \frac{(-1)^{f(x)} |x\rangle}{\sqrt{2^n}} \right) \otimes H |1\rangle$$

$$= :|\psi_f\rangle.$$  

To see what is the result of $H^\otimes n |\psi_f\rangle$, note that for $x \in \{0,1\}$, we can write $H|x\rangle$ as follows:

$$H|x\rangle = \frac{1}{\sqrt{2}} (|0\rangle + (-1)^x |1\rangle)$$

$$= \frac{1}{\sqrt{2}} \sum_{z \in \{0,1\}} (-1)^{xz} |z\rangle.$$  

Analogously, for $x = x_1 \cdots x_n \in \{0,1\}^n$, we have

$$H^\otimes n |x\rangle = \frac{1}{\sqrt{2^n}} \sum_{z = z_1 \cdots z_n \in \{0,1\}^n} (-1)^{x_1 z_1 + \cdots + x_n z_n} |z\rangle$$

$$= \frac{1}{\sqrt{2^n}} \sum_{z \in \{0,1\}^n} (-1)^{x \cdot z} |z\rangle.$$
Hence,

\[ H^\otimes n |\psi_0\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} (-1)^{f(x)} H^\otimes n |x\rangle \]

\[ = \frac{1}{2^n} \sum_{x \in \{0,1\}^n} \sum_{z \in \{0,1\}^n} (-1)^{f(x)+x \cdot z} |z\rangle \]

In particular, the amplitude of the basis vector \(|0^n\rangle\) in \(H^\otimes n |\psi_0\rangle\) is \(\frac{1}{2^n} \sum_{x \in \{0,1\}^n} (-1)^{f(x)}\). If \(f \equiv 0\), then this amplitude is equal to 1 and, with probability 1, the final measurement yields \(|0^n\rangle\). On the other hand, if \(f\) is balanced, then the amplitude of \(|0^n\rangle\) is 0 and, with probability 1, the final measurement yields a basis vector different from \(|0^n\rangle\).

3.2 Grover’s search algorithm

While the Deutsch-Jozsa algorithm arguably solves an artificial problem, Grover’s algorithm solves a canonical search problem: This time, the task is to find, given an arbitrary Boolean function \(f : \{0,1\}^n \rightarrow \{0,1\}\), an input \(x\) with \(f(x) = 1\) (or to determine that there is no such input). Classically, there is no better way than to test each input, which requires \(2^n\) queries to \(f\) in the worst case. Grover showed that if one has access to a QGA for computing the function

\[ U_f : H_{2^{n+1}} \rightarrow H_{2^{n+1}} |x\rangle \otimes |j\rangle \mapsto |x\rangle \otimes |f(x) \oplus j\rangle, \]

then one can build a quantum algorithm that finds an \(x\) with \(f(x) = 1\) in time \(O(\sqrt{2^n})\).

Our first approach is to apply a Hadamard transformation to \(|0^n\rangle\) to obtain a superposition of all inputs and then to apply \(U_f\) on \(H^\otimes n |0^n\rangle \otimes |0\rangle\). The resulting vector is

\[ \psi := \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle \otimes |f(x)\rangle. \]
Can we measure $|\psi\rangle$ to find an input $x$ with $f(x) = 1$? For each $x$ with $f(x) = 1$, the amplitude of $|x1\rangle$ in $|\psi\rangle$ is $\frac{1}{\sqrt{2^n}}$. Hence, if for instance there is only one such $x$, then a measurement of $\psi$ will very likely not find this $x$. The idea of the algorithm is to apply a transformation on $|\psi\rangle$ that makes the amplitudes of the basis vectors $|x1\rangle$ much larger while making those of $|x0\rangle$ smaller. After this transformation, with high probability a measurement of the last results in a basis vector of the form $|x1\rangle$, i.e. $f(x) = 1$. If the measurement fails and we obtain a vector $|x0\rangle$, we just repeat the process.

It turns out that this idea can be made to work using a modified approach, where we apply $U_f$ not to $H^\otimes n |0^n\rangle \otimes |0\rangle$, but to $H^\otimes n |0^n\rangle \otimes H|1\rangle$. As in the Deutsch-Jozsa algorithm, the resulting vector is $|\psi_f\rangle \otimes H|1\rangle$, where

$$
|\psi_f\rangle = \sum_{x \in \{0,1\}^n} \frac{(-1)^{f(x)} |x\rangle}{\sqrt{2^n}}.
$$

Let $V_f$ the transformation on the first $n$ qubits defined by $U_f$, $\phi$

$$
V_f |x\rangle = (-1)^{f(x)} |x\rangle.
$$

For $|\psi\rangle = \sum_x a_x |x\rangle$, we have

$$
V_f |\psi\rangle = \sum_{x: f(x) = 0} a_x |x\rangle - \sum_{x: f(x) = 1} a_x |x\rangle.
$$

For $|\psi\rangle = \sum_x a_x |x\rangle$, let $A := 2^{-n} \sum_x a_x$ the average amplitude. Consider the transformation $D$ that maps $|\psi\rangle$ to the vector $\sum_x (2A - a_x) |x\rangle$. The corresponding matrix is

$$
D = \begin{pmatrix}
\frac{2\pi}{2^n} & \frac{2\pi}{2^n} & \ldots & \frac{2\pi}{2^n} \\
\frac{2\pi}{2^n} & \frac{2\pi}{2^n} - 1 & \frac{2\pi}{2^n} \\
\vdots & \ddots & \ddots & \vdots \\
\frac{2\pi}{2^n} & \frac{2\pi}{2^n} & \ldots & \frac{2\pi}{2^n} - 1
\end{pmatrix}.
$$

To see this, consider a basis vector $|y\rangle = \sum_x \delta_{xy} |x\rangle$ (where $\delta_{xy} = 1$ if
x = y and δ_{xy} = 0 otherwise). The average amplitude of |y⟩ is \( A = \frac{1}{2^n} \).
Hence, \( D|y⟩ = (\frac{2}{2^n} - 1)|y⟩ + \sum_{x \neq y} \frac{2}{2^n}|x⟩ \).

**Lemma 3.1.** \( D = H^{\otimes n} \cdot R_n \cdot H^{\otimes n} \) with

\[
R_n = \begin{pmatrix}
1 & -1 & -1 & \cdots & -1 \\
1 & 1 & 1 & \cdots & 1 \\
1 & 1 & 1 & \cdots & 1 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
1 & 1 & 1 & \cdots & 1
\end{pmatrix}.
\]

Note that \( R_n \) can be implemented using \( O(n) \) simple gates.

**Proof.** Consider the matrix

\[
R' = R_n + I_n = \begin{pmatrix} 2 & 0 \\ 0 & \ddots \\ \vdots & \ddots & 0 \end{pmatrix}.
\]

We claim that

\[
H^{\otimes n} \cdot R'_n \cdot H^{\otimes n} = \frac{2}{2^n} \left( \begin{array}{cccc}
1 & 1 & \cdots & 1 \\
1 & 1 & \cdots & 1 \\
\vdots & \vdots & \ddots & \vdots \\
1 & 1 & \cdots & 1
\end{array} \right),
\]

i.e. \( H^{\otimes n} \cdot R'_n \cdot H^{\otimes n} |x⟩ = \frac{2}{2^n} \sum_y |y⟩ \) for all \( x \in \{0, 1\}^n \):

\[
|x⟩ \xrightarrow{H^{\otimes n}} \frac{1}{\sqrt{2^n}} \sum_z (-1)^{x \cdot z} |z⟩
\]

\[
\xrightarrow{R'_n} \frac{1}{\sqrt{2^n}} \sum_z (-1)^{x \cdot z} R'_n |z⟩ = \frac{2}{\sqrt{2^n}} |0⟩
\]

\[
\xrightarrow{H^{\otimes n}} \frac{2}{2^n} \sum_y |y⟩.
\]

Finally,

\[
H^{\otimes n} \cdot R_n \cdot H^{\otimes n} = H^{\otimes n} (R'_n - I_n) H^{\otimes n} = H^{\otimes n} \cdot R'_n H^{\otimes n} - H^{\otimes n} \cdot I_n \cdot H^{\otimes n} = H^{\otimes n} \cdot R'_n H^{\otimes n} - I_n = D.
\]

Q.E.D.
3.2 Grover’s search algorithm

For a given function $f : \{0, 1\}^n \to \{0, 1\}$, Grover’s search algorithm iterates the Grover operator $G := D \cdot V_f$ sufficiently often on input $H^\otimes n |0\rangle$ in order to magnify the amplitudes of the basis vectors $|x\rangle$ with $f(x) = 1$. But what do we mean by sufficiently often?

Consider the sets $T = \{x : f(x) = 1\}$ and $F = \{x : f(x) = 0\}$. After $r$ iterations of $G$, the resulting vector will be of the form $|\psi_r\rangle = t_r \sum_{x \in T} |x\rangle + f_r \sum_{x \in F} |x\rangle$ with average amplitude $A_r = \frac{1}{2^n} (t_r |T| + f_r (2^n - |T|))$. Now,

$$
|\psi_{r+1}\rangle = G|\psi_r\rangle
= D V_f \left( t_r \sum_{x \in T} |x\rangle + f_r \sum_{x \in F} |x\rangle \right)
= D \left( -t_r \sum_{x \in T} |x\rangle + f_r \sum_{x \in F} |x\rangle \right)
= (2A_r + t_r) \sum_{x \in T} |x\rangle + (2A_r - f_r) \sum_{x \in F} |x\rangle.
$$

Hence,

$$
t_{r+1} = 2A + t_r = \left( 1 - \frac{2|T|}{2^n} \right) t_r + \left( 2 - \frac{2|T|}{2^n} \right) f_r;
$$

$$
f_{r+1} = 2A - f_r = -\frac{2|T|}{2^n} t_r + \left( 1 - \frac{2|T|}{2^n} \right) f_r.
$$

This means that the coefficients $t_r$ and $f_r$ satisfy the following recursion:

$$
\begin{pmatrix}
t_{r+1} \\
f_{r+1}
\end{pmatrix} =
\begin{pmatrix}
1 - \delta & 2 - \delta \\
-\delta & 1 - \delta
\end{pmatrix}
\begin{pmatrix}
t_r \\
f_r
\end{pmatrix},
$$

where $\delta = \frac{2|T|}{2^n}$.

To compute the effect of the iterated application of $G$ on $H^\otimes n |0^n\rangle$, we have to solve (3.1) under the initial condition $t_0 = f_0 = \sqrt{\frac{1}{2^n}}$. Since $G$ is unitary, we have $\|G|\psi\| = \|\psi\|$, i.e. $|T| t_r^2 + (2^n - |T|) f_r^2 = 1$ for all $r \in \mathbb{N}$. Hence, there exist $\vartheta_r$ such that $t_r = \frac{1}{\sqrt{|T|}} \sin \vartheta_r$ and $f_r = \frac{1}{\sqrt{2^n - |T|}} \cos \vartheta_r$.

The Grover operator $G$ can be interpreted geometrically as a rota-
tion in the 2-dimensional space that is generated by the vectors

\[
|\varphi^+\rangle = \frac{1}{\sqrt{|T|}} \sum_{x \in T} |x\rangle,
|\varphi^-\rangle = \frac{1}{\sqrt{2^n - |T|}} \sum_{x \in F} |x\rangle.
\]

We have

\[
|\psi_0\rangle = \frac{1}{\sqrt{2^n}} \sum_{x} |x\rangle
= \sqrt{\frac{|T|}{2^n}} |\varphi^+\rangle + \sqrt{\frac{2^n - |T|}{2^n}} |\varphi^-\rangle
= \sin \vartheta_0 |\varphi^+\rangle + \cos \vartheta_0 |\varphi^-\rangle.
\]

Now, the Grover operator applied first performs a reflection across

\[|\varphi^+\rangle\]

followed by a reflection across

\[|\psi_0\rangle\].

The resulting operation is a rotation by \(2\vartheta_0\) towards \(|\varphi^+\rangle\). Hence, \(\vartheta_r = (2r + 1)\vartheta_0\) for all \(r \in \mathbb{N}\).

In order for the final measurement to yield \(|x\rangle\) with \(x \in T\), we need that \(\vartheta_r \approx \frac{\pi}{2}\) (so that \(|\psi_r\rangle\) is close to \(|\varphi^+\rangle\)). Solving the equation

\[(2r + 1)\vartheta_0 = \frac{\pi}{2},\]

we obtain \(r = \frac{\pi}{4\vartheta_0} - \frac{1}{2}\). Hence, for \(\vartheta_0 \approx \sin \vartheta_0 = \sqrt{\frac{|T|}{2^n}}\), we can expect that \(r = \left\lfloor \frac{\pi}{4} \sqrt{\frac{2^n}{|T|}} \right\rfloor\) iterations suffice to find a solution with high probability. More precisely, we have the following theorem.

**Theorem 3.2.** Let \(f: \{0, 1\}^n \to \{0, 1\}\) and \(m := |\{x: f(x) = 1\}|\) such that \(0 < m \leq \frac{3}{4} \cdot 2^n\), and let \(\vartheta_0 < \frac{\pi}{3}\) such that \(\sin \vartheta_0 = \frac{m}{2^n}\). After \(\left\lfloor \frac{\pi}{4\vartheta_0} \right\rfloor\) iterations of \(G\) on \(|\psi_0\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0, 1\}^n} |x\rangle\), a measurement of the resulting vector yields a basis vector \(|x\rangle\) such that \(f(x) = 1\) with probability \(\geq \frac{1}{4}\).

**Proof.** For \(|\psi_r\rangle = \sin(2r + 1)\vartheta_0 |\varphi^+\rangle + \cos(2r + 1)\vartheta_0 |\varphi^-\rangle\), we denote by \(p(r) := \sin^2(2r + 1)\vartheta_0\) the probability of a projection onto \(|\varphi^+\rangle\). (This is precisely the probability with which a measurement of \(|\psi_r\rangle\) results in a basis vector \(|x\rangle\) such that \(f(x) = 1\).) Let \(\delta \in (0, \frac{1}{2}]\) such that \(\left\lfloor \frac{\pi}{4\vartheta_0} \right\rfloor = \frac{\pi}{4\vartheta_0} - \frac{1}{2} + \delta\). Since \(|2\delta\vartheta_0| \leq |\vartheta_0| \leq \frac{\pi}{4}\), we have

\[
p\left(\left\lfloor \frac{\pi}{4\vartheta_0} \right\rfloor \right) = \sin^2 \left(\left\lfloor \frac{\pi}{4\vartheta_0} \right\rfloor \right) \vartheta_0
\]
3.2 Grover’s search algorithm

\[
\sin^2 \left( \frac{\pi}{2} + 2\delta \vartheta_0 \right) \\
\geq \sin^2 \left( \frac{\pi}{2} - \frac{\pi}{3} \right) = \frac{1}{4}. \\
\]

Q.E.D.

Finally, we can state Grover’s search algorithm. Given a QGA for the operator \( V_f \) defined by \( V_f |x\rangle = (-1)^{f(x)} |x\rangle \) and for known \( m := |\{x : f(x) = 1\}| \), the algorithm determines an input \( x \) such that \( f(x) = 1 \) by the following procedure:

if \( m \geq \frac{3}{4} \cdot 2^n \) then
\[
|\psi\rangle := H^\otimes n |0^n\rangle
\]
else
\[
r := \left\lfloor \frac{\pi}{4\vartheta_0} \right\rfloor \text{ for } 0 \leq \vartheta_0 \leq \frac{\pi}{3} \text{ with } \sin^2 \vartheta_0 = \frac{m}{2^n} \\
|\psi\rangle := G^r H^\otimes n |0^n\rangle
\]
end if

measure \( |\psi\rangle \) to obtain a basis vector \( |x\rangle \)

output \( x \)

If \( m \geq \frac{3}{4} \cdot 2^n \), the algorithm finds \( x \) such that \( f(x) = 1 \) with probability \( \geq \frac{3}{4} \) since \( |\psi\rangle \) is a uniform superposition of all basis vectors. Otherwise, Theorem 3.2 applies, and the algorithm finds \( x \) such that \( f(x) = 1 \) with probability \( \geq \frac{1}{4} \).

For \( m = 1 \) and for large \( n \), we have \( \left\lfloor \frac{\pi}{4\vartheta_0} \right\rfloor \approx \frac{\pi}{4} \sqrt{2^n} \) (since \( \sin^2 \vartheta_0 \approx \vartheta_0^2 = \frac{1}{2^n} \)). Hence, in this case, \( O(\sqrt{2^n}) \) calls to \( V_f \) suffice to find an input \( x \) such that \( f(x) = 1 \) with probability \( \geq \frac{1}{4} \), whereas classical randomised algorithms need to evaluate \( f \) at \( O(2^n) \) points to find such an \( x \) with the same probability of success.

Another interesting special case is when one fourth of the inputs are positive instances, i.e. if \( m = \frac{1}{4} \cdot 2^n \). Recall that after \( r \) iterations of \( G \) the resulting state is

\[
|\psi_r\rangle = \sin(2r + 1)\vartheta_0 |\varphi^+\rangle + \cos(2r + 1)\vartheta_0 |\varphi^-\rangle.
\]

For \( m = \frac{1}{4} \cdot 2^n \), we have \( \sin^2 \vartheta_0 = \frac{1}{4} \), and therefore \( \vartheta_0 = \frac{\pi}{6} \). After one iteration of \( G \), the resulting state is \( |\psi_1\rangle = \sin \frac{\pi}{2} |\varphi^+\rangle + \cos \frac{\pi}{2} |\varphi^-\rangle = |\varphi^+\rangle \) and a measurement will surely result in a basis vector \( x \) such that \( f(x) = 1 \).
In typical applications, the number \( m \) of positive instances is not known. How can we modify the algorithm such that it also finds a solution with good probability in this case?

**Lemma 3.3.** For all \( \alpha \in \mathbb{R} \) and all \( m \in \mathbb{N} \):

\[
\sum_{r=0}^{m-1} \cos(2r + 1)\alpha = \frac{\sin 2m\alpha}{2\sin \alpha}.
\]

In particular, \( \sin 2\alpha = 2\sin \alpha \cos \alpha \) and \( \cos 2\alpha = 1 - 2\sin^2 \alpha \).

We can now state Grover’s search algorithm for unknown \( m \):

choose \( x \in \{0, 1\}^n \) uniformly at random

if \( f(x) = 1 \) then

output \( x \)

else

choose \( r \in \{0, 1, \ldots, \lfloor \sqrt{2^n} \rfloor \} \) uniformly at random

\[ |\psi\rangle := G^r H^\otimes n |0^n\rangle \]

measure \( |\psi\rangle \) to obtain a basis vector \( |x\rangle \)

output \( x \)

end if

Clearly, if \( m \geq \frac{3}{4} \cdot 2^n \), then the algorithm returns \( x \) such that \( f(x) = 1 \) with probability \( \geq \frac{3}{4} \). Hence, assume now that \( m < \frac{3}{4} \cdot 2^n \), and set \( t := \lfloor \sqrt{2^n} \rfloor + 1 \). What is the probability that the algorithm outputs a good \( x \)? We have already seen that the probability of finding a good \( x \) after \( r \) iterations of \( G \) is \( \sin^2(2r + 1)\theta_0 \). Now, since \( r \) is chosen uniformly at random from \( \{0, 1, \ldots, t - 1\} \), the probability that the algorithm outputs a good \( x \) is

\[
\frac{1}{t} \sum_{r=0}^{t-1} \sin^2(2r + 1)\theta_0
\]

\[
= \frac{1}{2t} \sum_{r=0}^{t-1} (1 - \cos(2r + 1)2\theta_0) \quad \text{(since } \sin^2 \alpha = (1 - \cos 2\alpha)/2 \text{)}
\]

\[
= \frac{1}{2} - \frac{1}{2t} \sum_{r=0}^{t-1} \cos(2r + 1)2\theta_0
\]
3.3 Fourier transformation

\[ \frac{1}{2} - \frac{\sin 4t\vartheta_0}{4t \sin 2\vartheta_0} \] (by Lemma 3.3).

For \( 0 < m \leq \frac{3}{4} \cdot 2^n \) and \( t = \lfloor \sqrt{2^n} \rfloor + 1 \), we have

\[
\sin 2\vartheta_0 = 2 \sin \vartheta_0 \cos \vartheta_0 \\
= 2 \sqrt{\frac{m}{2^n}} \cdot \sqrt{\frac{2^n - m}{2^n}} \\
\geq 2 \sqrt{\frac{m}{2^n}} \cdot \sqrt{\frac{1}{4}} = \sqrt{\frac{m}{2^n}} \\
\geq \sqrt{\frac{1}{2^n}}
\]

and therefore

\[ t \geq \frac{1}{\sin 2\vartheta_0}. \]

Hence, the algorithm finds a good \( x \) with probability

\[
\frac{1}{2} - \frac{\sin 4t\vartheta_0}{4t \sin 2\vartheta_0} \geq \frac{1}{2} - \frac{\sin 4t\vartheta_0}{4} \geq \frac{1}{2} - \frac{1}{4} = \frac{1}{4}.
\]

To sum up, we have the following theorem.

**Theorem 3.4** (Grover). Given a function \( f : \{0, 1\}^n \to \{0, 1\}, f \neq 0 \), and a QGA for \( V_f : H_{2^n} \to H_{2^n} : \ket{x} \mapsto (-1)^{f(x)} \ket{x} \), there exists a quantum algorithm that finds an \( x \) such that \( f(x) = 1 \) with probability \( \geq \frac{1}{4} \) by evaluating \( V_f \) at most \( O(\sqrt{2^n}) \) times.

3.3 Fourier transformation

In the following, let \( (G, +) \) be an abelian group, and let \( C^* = (C \setminus \{0\}, \cdot) \).

A **character** of \( (G, +) \) is a homomorphism \( \chi : (G, +) \to C^* \). For two characters \( \chi_1, \chi_2 \), their product \( \chi_1 \cdot \chi_2 \), defined by

\[
\chi_1 \cdot \chi_2 : (G, +) \to C^* : g \mapsto \chi_1(g) \cdot \chi_2(g)
\]
is also a character. In fact the set of characters of \((G, +)\) together with this operations forms a new group, called the **dual group** and denoted by \((\hat{G}, \cdot)\).

**Lemma 3.5.** Let \((G, +)\) be a finite abelian group with \(n\) elements. Then
\[
\chi(g)^n = 1 \text{ for all } g \in G, \text{ i.e. } \chi(g) \text{ is an } n\text{th root of unity. Hence, }
\]
\[
\chi(g) = e^{2i\pi k/n} \text{ for some } k \in \{0, 1, \ldots, n - 1\}.
\]

**Proof.** For \(m \in \mathbb{N}\) and \(g \in G\), let
\[
m \cdot g := g + \cdots + g. \quad (m \text{ times})
\]

The set \(\{0, g, 2 \cdot g, \ldots\} \) forms a subgroup of \((G, +)\). Let
\[
k = \min\{m > 0: m \cdot g = 0\}
\]
be the order of this subgroup. Since the order of a subgroup divides the order of the group, we have \(n \cdot g = \frac{n}{k} \cdot k \cdot g = \frac{n}{k} \cdot 0 = 0\). Hence,
\[
\chi(g)^n = \chi(n \cdot g) = \chi(0) = 1.
\]
Q.E.D.

**Example 3.6.** Consider the cyclic group \((\mathbb{Z}_n, +)\), where \(\mathbb{Z}_n = \{0, 1, \ldots, n - 1\}\), with addition modulo \(n\). For each \(y \in \mathbb{Z}_n\), define
\[
\chi_y : \mathbb{Z}_n \rightarrow \mathbb{C}^* : x \mapsto e^{2\pi i \frac{xy}{n}}.
\]

We claim that \(\chi_y\) is a character of \((\mathbb{Z}_n, +)\), i.e. a group homomorphism from \((\mathbb{Z}_n, +)\) to \((\mathbb{C}^*, \cdot)\). Let \(x, x' \in \mathbb{Z}_n\). We have:
\[
\chi_y(x + x') = e^{2\pi i \frac{xy}{n}}
\]
\[
= e^{2\pi i \frac{xy}{n}} e^{2\pi i \frac{x'y}{n}}
\]
\[
= e^{2\pi i \frac{xy}{n}}
\]
\[
= \chi_y(x) \cdot \chi_y(x')
\]

Now consider \(y \neq y' \in \mathbb{Z}_n\). We have
\[
\chi_y(1) = e^{2\pi i \frac{y}{n}} \neq e^{2\pi i \frac{y'}{n}} = \chi_{y'}(1).
\]
Hence, also \(\chi_y \neq \chi_{y'}\). On the other hand, let \(\chi\) be a character of
(\Z_n,+). By Lemma 3.5, \(\chi(1) = e^{2i\pi y/n}\) for some \(y \in \Z_n\). But then \(\chi = \chi_y\). Finally, note that \(\chi_y \cdot \chi_{y'} = \chi_{y+y'}\). Hence, the mapping \(\Z_n \to \hat{\Z}_n: y \mapsto \chi_y\) is an isomorphism between \((\Z_n,+\rangle\) and the dual group \((\hat{\Z}_n,\cdot)\), i.e. \((\Z_n,+\rangle \cong (\hat{\Z}_n,\cdot)\).

More generally, we have the following theorem.

**Theorem 3.7.** Let \((G,+\rangle\) be a finite abelian group. Then \((G,+\rangle \cong (\hat{G},\cdot)\).

**Proof.** Every abelian group is (isomorphic to) a direct sum (or a direct product if the group operation is understood as multiplication) of cyclic groups:

\[(G,+\rangle = (\Z_{n_1},+\rangle \oplus \cdots \oplus (\Z_{n_k},+\rangle).\]

We already know that \((\Z_n,+\rangle \cong (\hat{\Z}_n,\cdot)\rangle\) and therefore also

\[(G,+\rangle \cong (\hat{\Z}_{n_1},\cdot) \times \cdots \times (\hat{\Z}_{n_k},\cdot).\]

To establish that \((G,+\rangle \cong (\hat{G},\cdot)\), it remains to show that there exists an isomorphism

\[\varphi: (\hat{\Z}_{n_1},\cdot) \times \cdots \times (\hat{\Z}_{n_k},\cdot) \to (\hat{G},\cdot).\]

For each \(g \in G\) there exists a unique decomposition into its components: \(g = g_1 + \cdots + g_k\) with \(g_i \in \Z_{n_i}\). For \(\chi_1 \in \hat{\Z}_{n_1}, \ldots, \chi_k \in \hat{\Z}_{n_k}\), we define 

\[(\varphi(\chi_1, \ldots, \chi_k))(g) := \chi_1(g_1) \cdot \cdots \cdot \chi_k(g_k).\]

Clearly, \(\varphi\) is a homomorphism. It remains to show that \(\varphi\) is a bijection.

Let us first prove that \(\varphi\) is injective: Let \((\chi_1, \ldots, \chi_k) \neq (\chi_1', \ldots, \chi_k')\), 
\(\chi = \varphi(\chi_1, \ldots, \chi_k)\), and \(\chi' = \varphi(\chi_1', \ldots, \chi_k')\). There exists \(i\) with \(\chi_i \neq \chi_i'\); in particular, there exists \(g_i \in \Z_{n_i}\) with \(\chi_i(g_i) \neq \chi_i'(g_i)\). We have 
\(\chi(g_i) = \chi_i(g_i) \neq \chi_i'(g_i) = \chi'(g_i)\) and therefore also \(\chi \neq \chi'\).

It remains to prove that \(\varphi\) is surjective: Let \(\chi \in \hat{G}\). For each \(i = 1, \ldots, k\), \(\chi\) induces a character \(\chi_i \in \hat{\Z}_{n_i}\) by setting \(\chi_i(g_i) = \chi(g_i)\) for all \(g_i \in \Z_{n_i}\). For all \(g \in G\), we have:

\[\chi(g) = \chi(g_1 + \cdots + g_k) = \chi(g_1) \cdots \chi(g_k)\]
\[ \chi = \chi_1(g_1) \cdots \chi_k(g_k) = (\varphi(\chi_1, \ldots, \chi_k))(g) \]

Hence, \( \chi = \varphi(\chi_1, \ldots, \chi_k) \).

**Example 3.8.** Consider the \( m \)-fold direct sum of \( (\mathbb{Z}_2, +) \),

\[ (\mathbb{Z}_2^m, +) = (\mathbb{Z}_2, +) \oplus \cdots \oplus (\mathbb{Z}_2, +) \text{ \( m \) times} \]

We already know that \( (\mathbb{Z}_2, +) \) has two characters, namely \( \chi_0: x \mapsto 1 \) and \( \chi_1: x \mapsto e^{\pi i x} = (-1)^x \). The characters of \( (\mathbb{Z}_2^m, +) \) are of the form

\[ \chi_y: x = x_1 \ldots x_m \mapsto (-1)^{x \cdot y} = (-1)^{x_1 y_1 + \cdots + x_m y_m}, \]

where \( y = y_1 \ldots y_m \in \{0, 1\}^m \).

The set of all functions \( f: G \to \mathbb{C} \) from a finite abelian group \( (G, +) \) to \( \mathbb{C} \) naturally forms a vector space \( V \) over \( \mathbb{C} \). If \( G = \{g_1, \ldots, g_n\} \), then this vector space is isomorphic to \( \mathbb{C}^n \), where the isomorphisms maps a function \( f \) to the tuple \( (f(g_1), \ldots, f(g_n)) \), and the functions \( e_i \) defined by

\[ e_i(g_j) = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{otherwise}, \end{cases} \]

form a basis of \( V \). The vector space \( V \) can be equipped with an inner product by setting

\[ \langle f | f' \rangle := \sum_{i=1}^n f(g_i)^* \cdot f'(g_i). \]

As usual, this inner product gives rise to a norm \( \| \cdot \| \) on \( V \), namely \( \|f\| = \sqrt{\langle f | f \rangle} \). Since \( \langle e_i | e_i \rangle = 1 \) and \( \langle e_i | e_j \rangle = 0 \) for \( i \neq j \), the set \( \{e_1, \ldots, e_n\} \) is, in fact, an orthonormal basis of \( V \). The characters of \( (G, +) \) give rise to a different orthonormal basis of \( V \). For \( \hat{G} = \{\chi_1, \ldots, \chi_k\} \), set

\[ B_i := \frac{1}{\sqrt{n}} \chi_i \text{ for all } i = 1, \ldots, n. \]
Theorem 3.9. Let $(G, +)$ be a finite abelian group with characters $\chi_1, \ldots, \chi_n$, and let $B_i := 1/\sqrt{n} \cdot \chi_i$ for all $i = 1, \ldots, n$. The vectors $B_1, \ldots, B_n$ form an orthonormal basis of $V = C^G$, called the Fourier basis.

Proof. Since $|\{B_1, \ldots, B_n\}| = |\{e_1, \ldots, e_n\}|$, it suffices to show that

$$
\langle \chi_i | \chi_j \rangle = \begin{cases} 
  n & \text{if } i = j, \\
  0 & \text{otherwise}.
\end{cases}
$$

For each $g \in G$ and for all $\chi \in \hat{G}$, by Lemma 3.5, we have $\chi(g)^n = 1$ and therefore $|\chi(g)| = 1$. Hence, $\chi(g)^* \cdot \chi(g) = |\chi(g)|^2 = 1$ and $\chi(g)^* = \chi(g)^{-1}$. We have:

$$
\langle \chi_i | \chi_j \rangle = \sum_{k=1}^{n} \chi_i(g_k)^* \cdot \chi_j(g_k) \\
= \sum_{k=1}^{n} \chi_i(g_k)^{-1} \cdot \chi_j(g_k) \\
= \sum_{k=1}^{n} (\chi_i^{-1} \cdot \chi_j)(g_k).
$$

For $i = j$, we have $\chi^{-1} \cdot \chi_j = 1$ (the trivial character) and therefore $\langle \chi_i | \chi_j \rangle = n$. For $i \neq j$, consider the character $\chi := \chi_i^{-1} \cdot \chi_j$. Since $\chi_i \neq \chi_j$, we have $\chi \neq 1$, i.e. there exists $g \in G$ with $\chi(g) \neq 1$. Consider the mapping $h_g : G \to G : g' \mapsto g' + g$. Since $G$ is finite, this mapping is not only injective, but also surjective. Hence,

$$
\langle \chi_i | \chi_j \rangle = \sum_{k=1}^{n} \chi(g_k) \\
= \sum_{k=1}^{n} \chi(g + g_k) \\
= \chi(g) \sum_{k=1}^{n} \chi(g_k) \\
= \chi(g) \cdot \langle \chi_i | \chi_j \rangle.
$$
Since $\chi(g) \neq 1$, we must have $\langle \chi_i | \chi_j \rangle = 0$. Q.E.D.

Let $G = \{g_1, \ldots, g_n\}$, $\hat{G} = \{\chi_1, \ldots, \chi_n\}$, and consider the matrix $X = (\chi_j(g_i))_{1 \leq i, j \leq n}$ and its conjugate transpose $X^* = ((\chi_i(g_j)^*))_{1 \leq i, j \leq n}$. We claim that $X^* \cdot X = n \cdot I$. To see this, consider the entry at position $i, j$:

\[
(X^* \cdot X)_{ij} = \sum_{k=1}^{n} X^*_{ik} \cdot X_{kj}
\]

\[
= \sum_{k=1}^{n} \chi_i(g_k)^* \cdot \chi_j(g_k)
\]

\[
= \langle \chi_i | \chi_j \rangle
\]

\[
= \begin{cases} 
  n & \text{if } i = j, \\
  0 & \text{otherwise.}
\end{cases}
\]

It follows that also $X \cdot X^* = n \cdot I$, i.e.

\[
\sum_{k=1}^{n} \chi_k(g_i) \cdot \chi_k(g_j)^* = \begin{cases} 
  n & \text{if } i = j, \\
  0 & \text{otherwise.}
\end{cases}
\]

(3.2)

**Corollary 3.10.** Let $(G, +)$ be a finite abelian group, $g \in G$ and $\chi \in \hat{G}$.

(a) $\sum_{k=1}^{n} \chi(g_k) = \begin{cases} 
  n & \text{if } \chi = 1, \\
  0 & \text{otherwise.}
\end{cases}$

(b) $\sum_{k=1}^{n} \chi_k(g) = \begin{cases} 
  n & \text{if } g = 0, \\
  0 & \text{otherwise.}
\end{cases}$

**Proof.** To prove (a), note that

\[
\sum_{k=1}^{n} \chi(g_k) = \langle 1 | \chi \rangle = \begin{cases} 
  n & \text{if } \chi = 1, \\
  0 & \text{otherwise.}
\end{cases}
\]

To prove (b), it suffices to apply (3.2) with $g_i = g$ and $g_j = 0$:

\[
\sum_{k=1}^{n} \chi_k(g) = \sum_{k=1}^{n} \chi_k(g) \cdot \chi_k(0)^* = \begin{cases} 
  n & \text{if } g = 0, \\
  0 & \text{otherwise.}
\end{cases}
\]

Q.E.D.
3.3 Fourier transformation

Example 3.11. For $G = \mathbb{Z}_n$, the characters are the mappings $\chi_y$, $y \in \mathbb{Z}_n$, with $\chi_y(x) = e^{2\pi i xy/n}$. Hence,

$$\sum_{y \in \mathbb{Z}_n} e^{2\pi i \frac{xy}{n}} = \begin{cases} n & \text{if } x = 0, \\ 0 & \text{otherwise.} \end{cases}$$

For $G = \mathbb{Z}_2^m$, the characters are the mappings $\chi_y$, $y \in \mathbb{Z}_2^m$, with $\chi_y(x) = (-1)^{x \cdot y}$. Hence,

$$\sum_{y \in \mathbb{Z}_n} (-1)^{x \cdot y} = \begin{cases} 2^m & \text{if } x = 0, \\ 0 & \text{otherwise.} \end{cases}$$

Finally, we can define the Fourier transformation. By Theorem 3.9, the vectors $B_i = 1/\sqrt{n} \cdot \chi_i$ form a basis of $\mathbb{C}^G$. The discrete Fourier transform of $f$ is the function $\hat{f}$ that maps the elements of $G$ to the coefficients in the unique representation of $f$ according to this basis.

Definition 3.12. Let $(G, +)$ be a finite abelian group with elements $g_1, \ldots, g_n$, and let $B_1, \ldots, B_n$ be the Fourier basis of $\mathbb{C}^G$. Given a function $f = \hat{f}_1 \cdot B_1 + \cdots + \hat{f}_n \cdot B_n \in \mathbb{C}^G$, its discrete Fourier transform (DFT) is the function $\hat{f} : G \to \mathbb{C} : g_i \to \hat{f}_i$.

How can we compute the DFT of a given function $f$? It turns out that $\hat{f}$ can be computed via the conjugate transpose of the matrix $X = (\chi_j(g_i))_{1 \leq i, j \leq n}$ as defined above.

Theorem 3.13. Let $(G, +)$ be a finite abelian group with elements $g_1, \ldots, g_n$ and characters $\chi_1, \ldots, \chi_n$, and let $X = (\chi_j(g_i))_{1 \leq i, j \leq n}$. With respect to the standard basis, for any function $f : G \to \mathbb{C}$, we have $\hat{f} = 1/\sqrt{n} \cdot X^* \cdot f$, i.e.

$$
\begin{pmatrix}
\hat{f}(g_1) \\
\hat{f}(g_2) \\
\vdots \\
\hat{f}(g_n)
\end{pmatrix} = \frac{1}{\sqrt{n}} \cdot
\begin{pmatrix}
\chi_1(g_1)^* & \cdots & \chi_1(g_n)^* \\
\chi_2(g_1)^* & \cdots & \chi_2(g_n)^* \\
\vdots & \vdots & \vdots \\
\chi_n(g_1)^* & \cdots & \chi_n(g_n)^*
\end{pmatrix}
\begin{pmatrix}
f(g_1) \\
f(g_2) \\
\vdots \\
f(g_n)
\end{pmatrix}.
$$
Proof. Since \( \{B_1, \ldots, B_n\} \) is an orthonormal basis, we have
\[
\langle B_i | f \rangle = \sum_{j=1}^{n} \langle B_i | \hat{f}_j \cdot B_j \rangle = \sum_{j=1}^{n} \hat{f}_j \cdot \langle B_i | B_j \rangle = \hat{f}_i
\]
and therefore
\[
\hat{f}(g_i) = \hat{f}_i = \langle B_i | f \rangle = \langle 1/\sqrt{n} \cdot \chi_i | f \rangle = \frac{1}{\sqrt{n}} \sum_{k=1}^{n} \chi_i(g_k) \cdot f(g_k).
\]

Q.E.D.

Corollary 3.14 (Parseval’s theorem). Let \( f: G \to \mathbb{C} \) and \( \hat{f} \) the DFT of \( f \). Then \( \|\hat{f}\| = \|f\| \).

\[\text{Proof. Since } X^* \cdot X = n \cdot I, \text{ the matrix } 1/\sqrt{n} \cdot X^* \text{ is unitary. Hence, } \|\hat{f}\| = \|1/\sqrt{n} \cdot X^* \cdot f\| = \|f\|. \]

Q.E.D.

The mapping \( f \mapsto 1/\sqrt{n} \cdot X \cdot f \) (wrt. the standard basis) is called the inverse Fourier transform.

Example 3.15. For \( G = \mathbb{Z}_n \) the characters are \( \chi_y, y \in \mathbb{Z}_n \), with \( \chi_y(x) = e^{2\pi i xy/n} \). Hence, the Fourier transform of \( f: \mathbb{Z}_n \to \mathbb{C} \) is
\[
\hat{f}: \mathbb{Z}_n \to \mathbb{C}: x \mapsto \frac{1}{\sqrt{n}} \sum_{y \in \mathbb{Z}_n} e^{-2\pi i xy/n} f(y),
\]
and its inverse Fourier transform is the function
\[
\tilde{f}: \mathbb{Z}_n \to \mathbb{C}: x \mapsto \frac{1}{\sqrt{n}} \sum_{y \in \mathbb{Z}_n} e^{2\pi i xy/n} f(y).
\]

For \( G = \mathbb{Z}_2^m \) the characters are \( \chi_y, y \in \mathbb{Z}_2^m \), with \( \chi_y(x) = (-1)^{x \cdot y} \). The Fourier transform of \( f: \mathbb{Z}_2^m \to \mathbb{C} \) is
\[
\hat{f}: \mathbb{Z}_2^m \to \mathbb{C}: x \mapsto \frac{1}{\sqrt{2^m}} \sum_{y \in \mathbb{Z}_2^m} (-1)^{x \cdot y} f(y).
\]

The same function is also the inverse Fourier transform of \( f \).
3.4 Quantum Fourier transformation

Let \((G, +)\) be a finite abelian group with elements \(g_1, \ldots, g_n\) and characters \(\chi_1, \ldots, \chi_k\), and consider the \(n\)-dimensional Hilbert space with basis \(\{|g_1\rangle, \ldots, |g_1\rangle\}\). Every state \(|\psi\rangle\) of \(H_G\) can be described by the function \(f: G \rightarrow \mathbb{C}\) with \(|\psi\rangle = \sum_{g \in G} f(g) \cdot |g\rangle\), i.e. \(f(g) = \langle g | \psi \rangle\).

**Definition 3.16.** Let \((G, +)\) be a finite abelian group; \(G = \{g_1, \ldots, g_n\}\) and \(\hat{G} = \{\chi_1, \ldots, \chi_k\}\). The mapping

\[
\text{QFT}: H_G \rightarrow H_G: \sum_{i=1}^{n} f(g_i) \cdot |g_i\rangle \mapsto \sum_{i=1}^{n} \hat{f}(g_i) \cdot |g_i\rangle
\]

is called the quantum Fourier transformation (QFT). In particular,

\[
\text{QFT} |g\rangle = \frac{1}{\sqrt{n}} \sum_{k=1}^{n} \chi_k(g)^* \cdot |g_k\rangle
\]

for all \(g \in G\).

**Lemma 3.17.** QFT is a unitary transformation.

*Proof.* Follows from Corollary 3.14. Q.E.D.

How can we implement QFT by a QGA with elementary gates? To do this, we will follow a bottom-up process. Let \(G = \{g_1, \ldots, g_m\}\) and \(G' = \{g_1, \ldots, g'_n\}\) with dual groups \(\hat{G} = \{\chi_1, \ldots, \chi_m\}\) and \(\hat{G}' = \{\chi'_1, \ldots, \chi'_n\}\). From \(G\) and \(G'\) we can build a new group \(G \oplus G' = \{g + g': g \in G, g' \in G'\}\), the direct sum of \(G\) and \(G'\). (Formally, the domain of \(G \oplus G'\) is the cartesian product of \(G\) and \(G'\), and addition is applied componentwise). The corresponding Hilbert space is \(H_{G \oplus G'} = H_G \otimes H_{G'}\) with basis vectors \(|g\rangle \otimes |g'\rangle, g \in G, g' \in G'\).

By Theorem 3.7, the dual group of \(G \oplus G'\) is isomorphic to \(\hat{G} \times \hat{G}'\). Hence, the characters of \(G \oplus G'\) are \(\chi_{ij}, 1 \leq i \leq m, 1 \leq j \leq n\), with \(\chi_{ij}(g + g') = \chi_i(g) \cdot \chi'_j(g')\) for all \(g \in G\) and all \(g' \in G'\). How does QFT behave on \(H_{G \oplus G'}\)? For a basis vector \(|g_i\rangle |g'_j\rangle =
\[ |g_i\rangle \otimes |g'_j\rangle, \text{ we have} \]

\[
\text{QFT} |g_i\rangle |g'_j\rangle = \frac{1}{\sqrt{mn}} \sum_{k=1}^{m} \sum_{l=1}^{n} \chi_{ij}(g_k + g'_l)^* \cdot |g_k\rangle |g'_l\rangle \\
= \frac{1}{\sqrt{mn}} \sum_{k=1}^{m} \sum_{l=1}^{n} (\chi_i(g_k)^* |g_k\rangle \otimes \chi_j(g'_l)^* |g'_l\rangle) \\
= \left( \frac{1}{\sqrt{m}} \sum_{k=1}^{m} \chi_i(g_k)^* |g_k\rangle \right) \otimes \left( \frac{1}{\sqrt{n}} \sum_{l=1}^{n} \chi_j(g'_l)^* |g'_l\rangle \right) \\
= \text{QFT} |g_i\rangle \otimes \text{QFT} |g'_j\rangle
\]

**Example 3.18.** Consider the group \( G = \mathbb{Z}_2^m \) (the \( m \)-fold direct product of \( \mathbb{Z}_2 \)). Then QFT on the Hilbert space \( H_G \) is equivalent to \( H^\otimes m \) since for all \( x = x_1 \ldots x_m \in \{0, 1\}^m \) we have

\[
H^\otimes m |x\rangle = \bigotimes_{i=1}^{m} \frac{1}{\sqrt{2}} \left( |0\rangle + (-1)^{x_i} |1\rangle \right) \\
= \frac{1}{\sqrt{2^m}} \sum_{y_1 \ldots y_m \in \{0, 1\}^m} (-1)^{x_1 y_1 + \ldots + x_m y_m} \cdot |y\rangle \\
= \frac{1}{\sqrt{2^m}} \sum_{y \in \{0, 1\}^m} (-1)^{x \cdot y} \cdot |y\rangle \\
= \text{QFT} |x\rangle.
\]

We are interested in QFT for the group \( G = \mathbb{Z}_n, n \in \mathbb{N} \). For this group, we have \( \text{QFT} |x\rangle = \sum_{y=0}^{n-1} e^{-2\pi ixy/n} \cdot |y\rangle \) for all \( x \in \{0, \ldots, n-1\} \).

If \( n = p \cdot q \) with \( \gcd(p, q) = 1 \), then \( \mathbb{Z}_n \cong \mathbb{Z}_p \times \mathbb{Z}_q \), and QFT on \( \mathbb{Z}_n \) can be composed from QFT on \( \mathbb{Z}_p \) and QFT on \( \mathbb{Z}_q \). However, in most applications no factorisation of \( n \) is known, or \( n = 2^m \) and no two factors are relatively prime.

For \( G = \mathbb{Z}_{2^m} \), instead of QFT, let us look at the inverse QFT. For \( x = \sum_{i=0}^{m-1} x_i \cdot 2^i \in \mathbb{Z}_{2^m} \), we identify the basis vector \( |x\rangle \) in \( H_G \) with the corresponding basis vector in \( H_{2^m} \), i.e. \( |x\rangle = |x_{m-1} \ldots x_0\rangle \). On \( H_{2^m} \), the inverse QFT on \( G \) corresponds to the transformation
3.4 Quantum Fourier transformation

**IQFT**

\[ IQFT_m : H_{2^m} \rightarrow H_{2^m} : |x\rangle \mapsto \frac{1}{\sqrt{2^m}} \sum_{y \in \mathbb{Z}_{2^m}} e^{2\pi i \cdot xy/2^m} \cdot |y\rangle. \]

**Lemma 3.19.** IQFT\(_m\) \(|x\rangle\) is decomposable for all \(x \in \mathbb{Z}_{2^m}\) and all \(m > 0\):

\[
\sum_{y \in \mathbb{Z}_{2^m}} e^{2\pi i \cdot xy/2^m} \cdot |y\rangle = \prod_{l=0}^{m-1} (|0\rangle + e^{\pi i /2^l} \cdot |1\rangle).
\]

**Proof.** The proof is by induction on \(m\). For \(m = 1\), the statement is trivial. Hence, let \(m > 1\) and assume that IQFT\(_{m-1}\) is decomposable. For all \(x \in \mathbb{Z}_{2^m}\), we have:

\[
\sum_{y \in \mathbb{Z}_{2^m}} e^{2\pi i \cdot xy/2^m} \cdot |y\rangle = \sum_{z \in \mathbb{Z}_{2^{m-1}}} \left( e^{2\pi i \cdot x \cdot 2z/2^m} \cdot |z0\rangle + e^{2\pi i \cdot x \cdot (2z+1)/2^m} \cdot |z1\rangle \right).
\]

\[
= \sum_{z \in \mathbb{Z}_{2^{m-1}}} \left( e^{2\pi i \cdot xz/2^{m-1}} \cdot |z0\rangle + e^{2\pi i \cdot xz/2^{m-1} - e^{\pi i \cdot xz/2^{m-1}}} \cdot |z1\rangle \right).
\]

\[
= \left( \sum_{z \in \mathbb{Z}_{2^{m-1}}} e^{2\pi i \cdot xz/2^{m-1}} \cdot |z\rangle \right) \otimes \left( |0\rangle + e^{2\pi i \cdot x/2^m} \cdot |1\rangle \right)
\]

\[
= \prod_{l=0}^{m-2} \left( |0\rangle + e^{\pi i /2^l} \cdot |1\rangle \right) \otimes \left( |0\rangle + e^{\pi i /2^m} \cdot |1\rangle \right)
\]

\[
= \prod_{l=0}^{m-1} \left( |0\rangle + e^{\pi i \cdot x_/2^l} \cdot |1\rangle \right).
\]

Q.E.D.

Let \(x = \sum_{l=0}^{2^m} x_i \cdot 2^i \in \mathbb{Z}_{2^m}\) and consider the operation of IQFT\(_m\) on the \(l\)th qubit:

\[
|x_l\rangle \mapsto \frac{1}{\sqrt{2}} \left( |0\rangle + e^{\pi i \cdot x_/2^l} \cdot |1\rangle \right).
\]

We have

\[
e^{\pi i \cdot x_/2^l} = \prod_{k=0}^{m-1} e^{\pi i \cdot x_k /2^l} = \prod_{k=0}^{l} e^{\pi i \cdot x_k /2^{l-k}} = (-1)^{x_l} \prod_{k<l \atop x_k=1} e^{\pi i /2^{l-k}}.
\]
Hence, $\text{IQFT}_m$ operates on the $l$th qubit like a Hadamard transformation, followed by a phase shift that depends on the qubits $|x_k\rangle$ for $k < l$. Formally, for $j \in \mathbb{N}$ define

$$R_j = \begin{pmatrix} 1 & 0 \\ 0 & e^{\pi i / 2j} \end{pmatrix}.$$ 

In particular, $R_1 = S$ and $R_2 = T$. Then

$$\text{IQFT}_m |x\rangle = \bigotimes_{l=0}^{m-1} \left( \prod_{k < l, x_k = 1} R_{l-k} \right) H |x_l\rangle$$

for all $x \in \{0, 1\}^m$. It follows that we can implement $\text{IQFT}_m$ using $O(m^2)$ Hadamard and controlled $R_j$ gates.

**Theorem 3.20.** For all $m > 0$, $\text{IQFT}_m$ can be implemented using $O(m^2)$ Hadamard and controlled $R_j$ gates, $j = 1, \ldots, m - 1$.

**QFT and periodical functions.** Let $f : \mathbb{Z}_n \to \mathbb{C}$ be a function with period $p \in \mathbb{Z}_n$, i.e. $f(m + p) = f(m)$ for all $m \in \mathbb{Z}_n$. For all $x \in \mathbb{Z}_n$, we have

$$\hat{f}(x) = \frac{1}{\sqrt{n}} \sum_{y \in \mathbb{Z}_n} e^{-2\pi i xy/n} f(y)$$

$$= \frac{1}{\sqrt{n}} \sum_{y \in \mathbb{Z}_n} e^{-2\pi i xy/n} f(y + p)$$

$$= e^{2\pi i xp/n} \cdot \frac{1}{\sqrt{n}} \sum_{y \in \mathbb{Z}_n} e^{-2\pi i (y+p)/n} f(y + p)$$

$$= e^{2\pi i xp/n} \cdot \frac{1}{\sqrt{n}} \sum_{y \in \mathbb{Z}_n} e^{-2\pi i xy/n} f(y)$$

$$= e^{2\pi i xp/n} \cdot \hat{f}(x)$$

Hence, if $\hat{f}(x) \neq 0$, then $e^{2\pi i xp/n} = 1$ and therefore $n \mid xp$.

We conclude that the Fourier transform of a function with period $p$ can only take non-zero values on arguments $x$ of the form $x = k \cdot n/p$. 

45
We can finally turn to Shor’s algorithm for factoring a composite number \( n \), i.e. the task to find given \( n \) numbers \( p, q < n \) such that \( n = p \cdot q \).

The general idea in almost all good factorisation algorithms is to find numbers \( b, c < n \) such that

\[
\begin{align*}
   b^2 &\equiv c^2 \pmod{n}, \quad (3.3) \\
   b &\not\equiv \pm c \pmod{n}. \quad (3.4)
\end{align*}
\]

We then have \((b + c)(b - c) \equiv 0 \pmod{n}\), but \( b + c \not\equiv 0 \pmod{n} \) and \( b - c \not\equiv 0 \pmod{n} \). Hence, \( b + c \) contains a factor of \( n \), which can be extracted by computing \( \gcd(b + c, n) \) in polynomial time, e.g. using Euklid’s algorithm.

Shor’s algorithm computes

\[
r := \text{ord}_n(a) = \min\{k > 0 : a^k = 1 \pmod{n}\}
\]

for a randomly chosen \( a < n \) with \( \gcd(a, n) = 1 \). If we are lucky, then \( r \) is even and \( a^{r/2} \not\equiv -1 \pmod{n} \). In this case, \( b = a^{r/2} \) and \( c = 1 \) satisfy (3.3) and (3.4).

What is the probability that we are lucky? We can assume without loss of generality that \( n \) is neither even nor a prime power because it is easy to decide whether \( n = 2^l \cdot m \) or \( n = a^k \) and to compute suitable numbers \( l, m \) or \( a, k \) if so.

**Lemma 3.21.** Let \( n \in \mathbb{N} \) be neither even nor a prime power, and let \( \mathbb{Z}_n^* = \{a \in \mathbb{Z}_n : \gcd(a, n) = 1\} \). Then

\[
\Pr_{a \in \mathbb{Z}_n^*} \left[ \text{ord}_n(a) \text{ is even and } a^{\text{ord}_n(a)/2} \not\equiv -1 \pmod{n} \right] \geq \frac{9}{16}.
\]

To prove this lemma, we need to make a small digression into number theory.
3.5.1 Number theory in a nutshell

For \( n \in \mathbb{N} \), let \( \mathbb{Z}_n^* \) the set of all \( a \in \mathbb{Z}_n \) with \( \gcd(a, n) = 1 \); we denote by \( \varphi(n) \) the cardinality of \( \mathbb{Z}_n^* \). When equipped with multiplication \( \mod n \), the set \( \mathbb{Z}_n^* \) forms an abelian group.

For prime numbers \( p \), we have \( \mathbb{Z}_p^* = \{1, 2, \ldots, p - 1\} \) and \( \varphi(p) = p - 1 \). In this case, the group \((\mathbb{Z}_p^*, \cdot)\) is isomorphic to the cyclic group \((\mathbb{Z}_{p - 1}, +)\). More generally, if \( n = p^k \) is a prime power, then

\[ \mathbb{Z}_n^* = \{a \in \mathbb{Z}_n : a \neq 0, p, 2p, \ldots, (p^k - 1)p\} \]

and \( \varphi(n) = p^k - p^{k-1} = p^{k-1}(p - 1) \).

**Theorem 3.22.** Let \( n = p^k \) for a prime \( p > 2 \) and \( k \geq 1 \). Then the group \((\mathbb{Z}_n^*, \cdot)\) is cyclic.

**Proof.** We prove that there exists an element \( b \in \mathbb{Z}_n^* \) with \( \text{ord}_n(b) = \varphi(n) = p^{k-1}(p - 1) \). We prove this by establishing the following three facts:

1. There exists \( b \in \mathbb{Z}_n^* \) with \( \text{ord}_n(b) = p - 1 \);
2. \( \text{ord}_n(1 + p) = p^{k-1} \);
3. If \((G, \cdot)\) is an abelian group and \( g, h \in G \) with \( \text{ord}_G(g) \) and \( \text{ord}_G(h) \) being relatively prime, then \( \text{ord}_G(g \cdot h) = \text{ord}_G(g) \cdot \text{ord}_G(h) \).

It follows that \( \text{ord}_n(b \cdot (1 + p)) = \varphi(n) \).

We start by proving (1). Consider the natural homomorphism

\[ f : \mathbb{Z}_n^* \rightarrow \mathbb{Z}_p^* : a \mapsto a \pmod{p}. \]

Since \( \mathbb{Z}_p^* \) is cyclic and \( f \) is surjective, there exists \( a \in \mathbb{Z}_n^* \) with \( \text{ord}_p(f(a)) = p - 1 \). Let \( r := \text{ord}_n(a) \). Since \( a^r \equiv 1 \pmod{p}^k \), we have \( f(a)^r = q \pmod{p} \) and therefore \( r = l(p - 1) \) for some \( l \in \mathbb{N} \). Set \( b := a^l \). We have \( b^{p-1} \equiv 1 \pmod{n} \). On the other hand, whenever \( b^s \equiv 1 \pmod{n} \), then \( (p - 1) | s \) because if \( b^s \equiv 1 \pmod{n} \), then also \( a^{ls} \equiv 1 \pmod{n} \) and therefore \( r = l(p - 1) | l \cdot s \). Hence, \( \text{ord}_n(b) = p - 1 \).

To prove (2), we first prove that for all \( m > 0 \) we have \( (1 + p)^{p^m} = 1 + \lambda p^{m+1} \) for some \( \lambda \in \mathbb{N} \) such that \( p \nmid \lambda \). We prove this by induction...
over $m$. For $m = 1$, we have

\[
(1 + p)^p = \sum_{i=0}^{p} \binom{p}{i} \cdot p^i
\]

\[
= 1 + p^2 + \sum_{i=3}^{p} \binom{p}{i} \cdot p^i
\]

\[
= 1 + p^2 + p^3 \cdot \sum_{i=3}^{p} \binom{p}{i} \cdot p^{i-3}
\]

\[
= 1 + p^2(1 + l \cdot p),
\]

which proves the statement since $(1 + l \cdot p) \nmid p$.

Now let $m > 1$ and assume that the statement holds for $m - 1$. We have:

\[
(1 + p)^{pm} = (1 + p)^{p^{m-1} \cdot p}
\]

\[
= (1 + \lambda \cdot p^m)^p
\]

\[
= \sum_{i=0}^{p} \binom{p}{i} \lambda^i p^{mi}
\]

\[
= 1 + \lambda p^{m+1} + \sum_{i=2}^{p} \binom{p}{i} \lambda^i p^{mi}
\]

\[
= 1 + \lambda p^{m+1} + p^{m+2} \cdot \sum_{i=2}^{p} \binom{p}{i} \lambda^i p^{m(i-1)-2}
\]

\[
= 1 + p^{m+1}(\lambda + lp).
\]

Since $\lambda \nmid p$, we also have $(\lambda + lp) \nmid p$, which proves the statement.

It follows that there exist $\lambda_1, \lambda_2 \in \mathbb{N}$ with $p \nmid \lambda_1$ and $p \nmid \lambda_2$ such that

\[
(1 + p)^{p^{k-1}} = 1 + \lambda_1 \cdot p^k \equiv 1 \pmod{n};
\]

\[
(1 + p)^{p^{k-2}} = 1 + \lambda_2 \cdot p^{k-1} \not\equiv 1 \pmod{n}.
\]
Hence, \( \text{ord}_n(1 + p) \mid p^{k-1} \) but \( \text{ord}_n(1 + p) \nmid p^{k-2} \). Thus, \( \text{ord}_n(1 + p) = p^{k-1} \).

It remains to prove (3). Let \( r = \text{ord}_G(g) \) and \( s = \text{ord}_G(h) \) with \( \gcd(r, s) = 1 \). Clearly, \((gh)^{rs} = 1\) and therefore \( \text{ord}_G(gh) \mid rs \). On the other hand, assume that \((gh)^t = 1\). We have \( 1r = (gh)^{ts} = g^{ts} \cdot h^{ts} = g^{ts} \cdot 1^t = g^{ts} \) and therefore \( r \mid ts \). Since \( \gcd(r, s) = 1 \), this implies \( r \mid t \), and an analogous argument shows that \( s \mid t \). Hence, also \( rs \mid t \), which proves that \( \text{ord}_G(gh) = rs \).

**Remark 3.23.** Theorem 3.22 does not hold for \( p = 2 \). For instance, we have \( \mathbb{Z}^*_8 = \{1, 3, 5, 7\} \) with \( 3^2 \equiv 5^2 \equiv 7^2 \equiv 1 \pmod{n} \). Hence, the group \((\mathbb{Z}^*_8, \cdot)\) is isomorphic to \((\mathbb{Z}_2 \times \mathbb{Z}_2, +)\), the Klein four-group.

Let \( n \) be an odd prime power, i.e. \( n = p^e \) for some prime \( p > 2 \). Since \( \mathbb{Z}^*_n \) is cyclic, there exists a generator \( g \) of this group, i.e. \( \mathbb{Z}^*_n = \{g, g^2, \ldots, g^{\varphi(n)}\} \). Moreover, \( \varphi(n) = \varphi(p^e) = p^{e-1}(p - 1) = 2^d \cdot u \) for \( d \geq 1 \) and an odd number \( u \).

**Lemma 3.24.** Let \( n = p^e \), \( p > 2 \), \( \varphi(n) = 2^d \cdot u \) with \( 2 \nmid u \), and let \( g \) be a generator of \( \mathbb{Z}^*_n \). Then \( i \in \mathbb{N} \) is odd if and only if \( 2^d \mid \text{ord}_n(g^i) \).

**Proof.** (\( \Rightarrow \)) Let \( i \in \mathbb{N} \) be odd. We have \( g^{i \cdot \text{ord}_n(g^i)} \equiv 1 \pmod{n} \) and therefore \( \varphi(n) \mid i \cdot \text{ord}_n(g^i) \). Since \( \varphi(n) = 2^d \cdot u \) and \( i \) is odd, this implies that \( 2^d \mid \text{ord}_n(g^i) \).

(\( \Leftarrow \)) Let \( i \in \mathbb{N} \) be even. We have \( g^{i \cdot \varphi(n)/2} = g^{\varphi(n) \cdot i/2} \equiv 1 \pmod{n} \) and therefore \( \text{ord}_n(g^i) \mid \varphi(n)/2 \). Since \( 2^d \nmid \varphi(n)/2 \), this implies that \( 2^d \nmid \text{ord}_n(g^i) \).

**Corollary 3.25.** Let \( n = p^e \), \( p > 2 \), and \( \varphi(n) = 2^d \cdot u \) with \( 2 \nmid u \). Then

\[
\Pr_{a \in \mathbb{Z}^*_n} [2^d \mid \text{ord}_n(a)] = \frac{1}{2}.
\]

Finally, we can prove Lemma 3.21.

**Proof (of Lemma 3.21).** Let \( n \in \mathbb{N} \) be neither even nor a prime power. Hence, \( n = p_1^{e_1} \cdots p_r^{e_k}, \ k > 1 \) for primes \( p_i > 2 \) such that \( p_i \neq p_j \) for
3.5 Shor’s factorisation algorithm

$i \neq j$. The Chinese remainder theorem tells us that the mapping

$$\mathbb{Z}_n^* \rightarrow \mathbb{Z}_{p_1^{e_1}}^* \times \cdots \times \mathbb{Z}_{p_k^{e_k}}^*: a \mapsto (a \mod p_1^{e_1}, \ldots, a \mod p_k^{e_k})$$

is an isomorphism. In particular, we have

$$\varphi(n) = \prod_{i=1}^k \varphi(p_i^{e_i}) = \prod_{i=1}^k p_i^{e_i-1}(p_i-1).$$

Moreover, for $a \in \mathbb{Z}_n^*$ we have $\text{ord}_n(a) = \gcd(\text{ord}_{p_1^{e_1}}(a), \ldots, \text{ord}_{p_k^{e_k}}(a))$ because, by the Chinese remainder theorem, $a^r \equiv 1 \pmod{n}$ is equivalent to $a^r \equiv 1 \pmod{p_i^{e_i}}$ for all $i$, and the latter holds if and only if $\text{ord}_{p_i^{e_i}}(a) | r$.

By the Chinese remainder theorem, a random choice of $a \in \mathbb{Z}_n^*$ corresponds to a random choice of $a_1, \ldots, a_k$ with $a_i \in \mathbb{Z}_{p_i^{e_i}}$. For $a \in \mathbb{Z}_n^*$, let $r_i = \text{ord}_{p_i^{e_i}}(a)$. Then $\text{ord}_n(a) = \gcd(r_1, \ldots, r_k)$ is odd if and only if each $r_i$ is odd. It follows from Corollary 3.25 that $\Pr_{a \in \mathbb{Z}_n^*}[r_i \text{ is odd}] \leq \frac{1}{2}$ and $\Pr_{a \in \mathbb{Z}_n^*}[\text{ord}_n(a) \text{ is odd}] \leq \frac{1}{2^k}$.

Assume now that $r = \text{ord}_n(a)$. If $a^{r/2} \equiv -1 \pmod{n}$, then $n / a^{r/2} + 1$. But then also $p_i^{e_i} | a^{r/2} + 1$ and therefore $a^{r/2} \equiv -1 \pmod{p_i^{e_i}}$ for all $i = 1, \ldots, k$. Since $a^{r/2} \equiv 1 \pmod{p_i^{e_i}}$ and $p_i > 2$, this implies that $r_i \nmid \frac{r}{2}$ for all $i$. For $r = 2^d \cdot u$ (where $u$ is odd), this means that $2^d | r_i$ for all $i = 1, \ldots, k$. Hence,

$$\Pr_{a \in \mathbb{Z}_n^*}[a^{\text{ord}_n(a)/2} \equiv -1 \pmod{n} | \text{ord}_n(a) \text{ is even}]$$

$$\leq \Pr_{a \in \mathbb{Z}_n^*}[2^d | \text{ord}_{p_i^{e_i}}(a) \text{ for all } i]$$

$$= \frac{1}{2^k}$$

where the last equality follows from Corollary 3.25. Finally,

$$\Pr_{a \in \mathbb{Z}_n^*}[2 | \text{ord}_n(a) \text{ and } a^{\text{ord}_n(a)/2} \not\equiv -1 \pmod{n}]$$

$$= \Pr_{a \in \mathbb{Z}_n^*}[2 | \text{ord}_n(a)] \cdot \Pr_{a \in \mathbb{Z}_n^*}[a^{\text{ord}_n(a)/2} \not\equiv -1 \pmod{n} | 2 | \text{ord}_n(a)]$$

$$\geq (1 - \frac{1}{2^k}) \cdot (1 - \frac{1}{2^k})$$

50
3.5.2 Factoring and QFT

To sum up, we can reduce factoring to the problem of computing, given a number \( n \in \mathbb{N} \) that is neither odd nor a prime power, the order \( \text{ord}_n(a) \) of \( a \in \mathbb{Z}_n^* \). The number \( r = \text{ord}_n(a) \) is the period of the function

\[
f: \mathbb{Z} \to \mathbb{Z}_n: x \mapsto a^x \mod n
\]

since \( f(x + r) \equiv a^{x+r} \equiv a^x \cdot a^r \equiv a^x \pmod{n} \). We can use QFT to determine this period! However, QGAs only operate on the Hadamard space \( H_{2^m} \). Hence, we choose a sufficiently large number \( m \in \mathbb{N} \) such that the period of \( f \) occurs in \( \mathbb{Z}_{2^m} \): in fact, we can always take the unique number \( m \) such that \( n^2 \leq 2^m < 2n^2 \).

We can now give an informal description of Shor’s algorithm. First, after having randomly chosen \( a < n \), the algorithm computes the quantum state

\[
|\psi\rangle = \frac{1}{\sqrt{2^m}} \sum_{x \in \mathbb{Z}_{2^m}} |x\rangle |a^x \mod n\rangle \in H_{2^{m+k}},
\]

where \( 2^k \leq n < 2^{k+1} \). Note that the function \( x \mapsto a^x \mod n \) is computable in polynomial time (by a classical circuit) and thus also by a QGA since for \( x = \sum_{i=0}^{m-1} x_i \cdot 2^i \) we have \( a^x \equiv \prod_{i=1}^{m} x_i a_i \pmod{n} \) where \( a_0 = a \) and \( a_{i+1} = a_i^2 \pmod{n} \) for all \( i < m \).

Since \( x \mapsto a^x \mod n \) has period \( r = \text{ord}_n(a) \), we have

\[
|\psi\rangle = \frac{1}{\sqrt{2^m}} \sum_{l=0}^{r-1} \sum_{q=0}^{s_l} |qr + l\rangle |a^l \mod n\rangle,
\]

where \( s_l = \max\{s \in \mathbb{N} : sr + l < 2^m\} \).

The next step of the algorithm is to apply \( \text{IQFT}_m \) to the first \( m \) qubits of \( |\psi\rangle \). The resulting state is
3.5 Shor’s factorisation algorithm

\[ |\varphi\rangle = \frac{1}{\sqrt{2^m}} \sum_{l=0}^{r-1} \sum_{q=0}^{s_l} \frac{1}{\sqrt{2^m}} \sum_{y \in \mathbb{Z}_{2^m}} e^{2\pi i y(qr+l)/2^m} |y\rangle |a^l \mod n\rangle \]

\[ = \frac{1}{2^m} \sum_{l=0}^{r-1} \sum_{y=0}^{2^m-1} e^{2\pi i y l/2^m} \sum_{q=0}^{s_l} e^{2\pi i y r q/2^m} |y\rangle |a^l \mod n\rangle \]

Finally, the algorithm performs a measurement on the first \( m \) qubits of \( |\varphi\rangle \), which yields \( y \in \mathbb{Z}_{2^m} \). Then, with some luck, \( y \approx k \cdot 2^m / r \) and \( \gcd(k, r) = 1 \). The number \( r \) can then be extracted using the method of continued fractions (see below).

Example 3.26. Let \( n = 15 \) and \( a = 7 \). In this case, it suffices to choose \( m = 4 \) (as opposed to \( m = 8 \)). Hence,

\[ |\psi\rangle = \frac{1}{\sqrt{16}} \sum_{x=0}^{15} |x\rangle |7^x \mod 15\rangle \]

\[ = \frac{1}{4} (|0\rangle |1\rangle + |1\rangle |7\rangle + |2\rangle |4\rangle + \cdots + |15\rangle |13\rangle) \]

\[ = \frac{1}{4} \left( (|0\rangle + |4\rangle + |8\rangle + |12\rangle) |1\rangle + (|1\rangle + |5\rangle + |9\rangle + |13\rangle) |7\rangle + (|2\rangle + |6\rangle + |10\rangle + |14\rangle) |4\rangle + (|3\rangle + |7\rangle + |11\rangle + |15\rangle) |13\rangle \right) \]

\[ = \sum_{j=0}^{4} \left( \sum_{y=0}^{15} f_j(y) |y\rangle \right) |7^j \mod 15\rangle, \]

where

\[ f_j(y) = \begin{cases} 
\frac{1}{4} & \text{if } y \equiv j \pmod{4} \\
0 & \text{otherwise.} 
\end{cases} \]

Each \( f_j \) has period 4. Hence, \( \hat{f}_j(x) \neq 0 \) only for \( x \in \{0, 4, 8, 12\} \). For \( k = 0, 1, 2, 3 \), we have

\[ \hat{f}_j(4k) = \frac{1}{4} \sum_{y=0}^{15} e^{2\pi i 4k y/16} \cdot f_j(y) \]
3 Quantum Algorithms

\[
\sum_{l=0}^{3} e^{2\pi i \cdot 4k(4l+j)/16} \cdot \frac{1}{4} = \frac{1}{16} \sum_{l=0}^{3} e^{2\pi i \cdot 4k(4l+j)/16}
\]

\[
= \frac{1}{16} \cdot e^{\pi i / 2} \sum_{l=0}^{3} e^{2\pi i \cdot k l}
\]

\[
= \frac{1}{16} \cdot e^{\pi i / 2} \sum_{l=0}^{3} 1
\]

\[
= \frac{1}{4} \cdot e^{\pi i \cdot k j}.
\]

Hence,

\[
|\varphi\rangle = \frac{1}{4} \left( (|0\rangle + |4\rangle + |8\rangle + |12\rangle) |1\rangle + (|0\rangle + i|4\rangle - |8\rangle - i|12\rangle) |7\rangle + (|0\rangle - |4\rangle + |8\rangle - |12\rangle) |4\rangle + (|0\rangle - i|4\rangle - |8\rangle + i|12\rangle) |13\rangle \right).
\]

With probability \(\frac{1}{4}\) each, a measurement of the first \(m\) qubits of \(|\varphi\rangle\) yields \(|0\rangle\), \(|4\rangle\), \(|8\rangle\) or \(|12\rangle\), with probability \(\frac{1}{4}\) each. From \(|0\rangle\) and \(|8\rangle\), the period \(4 = \text{ord}_{15}(7)\) cannot be extracted. However, for \(y = 4, 12\) we have \(y = 4k\) with \(\gcd(k, 4) = 1\), and the period can be extracted.

The period \(r = 4\) is even and \(7^r/2 = 7^2 - 4 \not\equiv -1 \pmod{15}\). Hence, \(3 = 4 - 1\) and \(5 = 4 + 1\) are identified as factors of 15.

The probability that a measurement of the first \(m\) qubits of \(|\varphi\rangle\) returns \(y \in \mathbb{Z}_{2^m}\) is

\[
\Pr[y] = \frac{1}{2^{2m}} \sum_{l=0}^{r-1} \left| \sum_{q=0}^{s_l} e^{2\pi i \cdot y_l/2^m} \right|^2
\]

\[
= \frac{1}{2^{2m}} \sum_{l=0}^{r-1} \left| \sum_{q=0}^{s_l} e^{2\pi i \cdot y_r/2^m} \right|^2.
\]

If \(r \mid 2^m\), i.e. for \(r = 2^s\) with \(s \leq m\), we know that \(\Pr[y] \neq 0\) only
if \( k = \cdot 2^m/r \). Moreover, all these \( y \) occur with probability \( 1/r \) because 
\[ s_l = 2^m - s = 1 \] for all \( l < r \) and

\[
\Pr[y] = \frac{r}{2^m} \left| \sum_{q=0}^{2^m-1} e^{2\pi i yq/2^m} \right|^2 = \frac{r}{2^m} \left| \sum_{q=0}^{2^m-1} \chi_q(y) \right|^2
\]

\[
= \begin{cases} 
\frac{r}{2^m} |2^{m-s}|^2 & \text{if } y \equiv 0 \pmod{2^{m-s}}, \\
0 & \text{otherwise,}
\end{cases}
\]

\[
= \begin{cases} 
\frac{r}{2^m} \cdot \frac{2^m}{r} = \frac{1}{r} & \text{if } y = k \cdot 2^m/r, \\
0 & \text{otherwise.}
\end{cases}
\]

However, in general, we cannot assume that \( r \mid 2^m \). For \( l < r \), consider the summand \( \sum_{q=0}^{s_l} |qr+l|a^l \pmod{n} \) of \( |\psi\rangle \). This summand can be written as \( \sum_{y \in \mathbb{Z}_{2^m}} f_l(y) |y|a^l \pmod{n} \), where

\[
f_l(y) = \begin{cases} 
1 & \text{if } y \equiv l \pmod{r} \\
0 & \text{otherwise.}
\end{cases}
\]

Since \( r \mid 2^m \), the function \( f_l : \mathbb{Z}_{2^m} \to \mathbb{C} \) is not exactly periodic. Hence, the Fourier transformation and subsequent measurement does not necessarily yield \( y = k \cdot 2^m/r \). However, with high probability, it yields a \( y \in \mathbb{Z}_{2^m} \) that is sufficiently close to such an element.

**Lemma 3.27.** Let \( |\varphi\rangle \) be the quantum state obtained by Shor’s algorithm on input \( n \geq 100 \) after applying IQFT\(_m\). For all \( k < r = \text{ord}_n(a) \), a measurement of the first \( m \) qubits of \( |\varphi\rangle \) yields the unique \( y \in \mathbb{Z}_{2^m} \) such that \( |y - k \cdot 2^m/r| \leq 1/2 \) with probability \( \geq 2/5 \).

**Proof.** By an elementary, but long calculation. Q.E.D.

It follows from Lemma 3.27 that a measurement of the first \( m \) qubits of \( |\varphi\rangle \) yields \( y \in \mathbb{Z}_{2^m} \) such that \( |y - k \cdot 2^m/r| \leq 1/2 \) for some \( k \in \{0, \ldots, r-1\} \) with probability \( \geq 2/5 \). The probability that \( \gcd(k, r) = 1 \) for a randomly chosen \( k \in \{0, \ldots, r-1\} \) is \( \varphi(r)/r \).
Lemma 3.28. For all $r \geq 19$,
\[
\frac{\varphi(r)}{r} \geq \frac{1}{4 \log \log r}.
\]

Corollary 3.29. Let $|\varphi\rangle$ be the quantum state obtained by Shor’s algorithm on input $n \geq 100$ after applying $\text{IQFT}_m$. A measurement of the first $m$ qubits of $|\varphi\rangle$ yields an element $y \in \mathbb{Z}_{2^m}$ such that $|y - k \cdot 2^m / r| \leq 1/2$ for some $k < r$ with $\gcd(k, r) = 1$ with probability $\geq 1/(10 \log \log n)$.

For the obtained $y$ with $|y - k \cdot 2^m / r| \leq 1/2$, it holds that
\[
\left| \frac{y}{2^m} - \frac{k}{r} \right| \leq \frac{1}{2 \cdot 2^m} \leq \frac{1}{2n^2} < \frac{1}{2r^2}.
\]
(Recall that $m$ was chosen in a way such that $n^2 \leq 2^m$.)

It remains to show that we can extract $r$ from $y$ and $2^m$ efficiently. For this task, we will use the method of continued fractions, and we will prove that 1. we can compute all convergents of the continued fraction representation for a rational number $x$ efficiently, and 2. if $x \in \mathbb{Q}$ and $p$ and $q$ are relatively prime such that $|x - p/q| \leq 1/2q^2$, then $p/q$ is a convergent of the continued fraction representation for $x$.

3.5.3 Continued fractions

Every number $\alpha \in \mathbb{R}$ can be represented as a continued fraction
\[
[a_0, a_1, \ldots] := a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{a_3 + \cdots}}},
\]
where $a_0 \in \mathbb{Z}$ and $a_n \in \mathbb{N} \setminus \{0\}$ for all $n > 0$. If $\alpha$ is irrational, then $\alpha$ has unique continued fraction representation, which is infinite. Rational numbers, on the other hand, have a two different finite continued fraction representations.
Example 3.30. Consider the rational number \( x = \frac{31}{13} \). We have

\[
x = 2 + \frac{5}{13} = 2 + \frac{1}{\frac{13}{5}}
\]

\[
= 2 + \frac{1}{2 + \frac{3}{5}} = 2 + \frac{1}{2 + \frac{1}{\frac{5}{3}}}
\]

\[
= 2 + \frac{1}{2 + \frac{1}{1 + \frac{2}{3}}} = 2 + \frac{1}{2 + \frac{1}{1 + \frac{1}{\frac{3}{2}}}}
\]

\[
= 2 + \frac{1}{2 + \frac{1}{1 + \frac{1}{1 + \frac{1}{2}}}} = 2 + \frac{1}{2 + \frac{1}{1 + \frac{1}{1 + \frac{1}{1 + \frac{1}{2}}}}}
\]

\[
= [2, 2, 1, 1, 2] = [2, 2, 1, 1, 1, 1]
\]

We will show that a continued fraction representation of a rational number \( p/q \) with \( p, q < 2^n \) can be computed using Euclid’s algorithm in \( O(n) \) basic steps. Note that we can form the expression

\[
[a_0, a_1, \ldots, a_n] := a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{a_3 + \ldots + \frac{1}{a_{n-1} + \frac{1}{a_n}}}}}
\]

for arbitrary numbers \( a_0, a_1, \ldots, a_n \in \mathbb{R}_{>0} \). For \( \alpha = [a_0, \ldots, a_n] \) and \( j \leq n \), we call \([a_0, \ldots, a_j]\) the \( j \)th convergent of \( \alpha \).

**Theorem 3.31.** For \( \alpha = [a_0, \ldots, a_n] \in \mathbb{R} \), we have \( [a_0, \ldots, a_j] = p_j/q_j \) for all \( j \leq n \), where
\[ p_0 = a_0, \quad q_0 = 1, \quad (3.5) \]
\[ p_1 = 1 + a_0 \cdot a_1, \quad q_1 = a_1, \quad (3.6) \]
\[ p_{j+2} = a_{j+2} \cdot p_{j+1} + p_j, \quad q_{j+2} = a_{j+2} \cdot q_{j+1} + q_j. \quad (3.7) \]

**Proof.** We have
\[
[a_0] = \frac{a_0}{1} = \frac{p_0}{q_0}
\]
and
\[
[a_0, a_1] = a_0 + \frac{1}{a_1} = \frac{a_0 \cdot a_1 + 1}{a_1} = \frac{p_1}{q_2},
\]
which proves (3.5) and (3.6). We prove (3.7) by induction over \( j \): We have
\[
[a_0, a_1, a_2] = a_0 + \frac{1}{a_1 + \frac{1}{a_2}}
\]
\[= \frac{a_0 \cdot a_1 \cdot a_2 + a_0 + a_2}{a_1 \cdot a_2 + 1}
\]
\[= \frac{a_2(1 + a_0 \cdot a_1) + a_0}{a_2 \cdot a_1 + 1}
\]
\[= \frac{a_2 \cdot p_1 + p_0}{a_2 \cdot q_1 + q_0} = \frac{p_2}{q_2},
\]
which establishes the base case. Now let \( 0 \leq j \leq n - 3 \) and assume that \( p_{j+2} \) and \( q_{j+2} \) satisfy (3.7). Then
\[
[a_0, \ldots, a_{j+3}] = [a_0, \ldots, a_{j+1}, a_{j+2} + 1/a_{j+3}]
\]
\[= \frac{(a_{j+2} = \frac{1}{a_{j+3}}) p_{j+1} + p_j}{(a_{j+2} + \frac{1}{a_{j+3}}) q_{j+1} + q_j}
\]
\[= \frac{a_{j+3}(a_{j+2} \cdot p_{j+1} + p_j) + p_{j+1}}{a_{j+3}(a_{j+2} \cdot q_{j+1} + q_j) + q_{j+1}}
\]
\[= \frac{a_{j+3} \cdot p_{j+2} + p_{j+1}}{a_{j+3} \cdot q_{j+2} + q_{j+1}} = \frac{p_{j+3}}{q_{j+3}},
\]
which proves (3.7) for \( j \) replaced by \( j + 1 \). Q.E.D.
Corollary 3.32. For \( \alpha = [a_0, \ldots, a_n] \in \mathbb{R} \) such that \([a_0, \ldots, a_j] = p_j/q_j\) for \(j \leq n\), we have \(p_j \cdot q_j - p_{j-1} \cdot q_{j-1} = (-1)^j\) for all \(j \geq 1\).

It follows from Corollary 3.32 that \(\gcd(p_j, q_j) = 1\) if \(a_j \in \mathbb{N} \setminus \{0\}\) for all \(j\). Hence, Euklid’s algorithm can be used to obtain \(p_{j+1}\) and \(q_{j+1}\). Moreover, by the definition of \(p_j, q_j\), we have \(p_0 < p_1 < \cdots < p_n\) and \(q_0 < q_1 < \cdots < q_n\). More precisely,

\[
p_{j+2} = a_{j+2} \cdot p_{j+1} + p_j \geq 2p_j
\]

and analogously \(q_{j+2} \geq 2q_j\). Hence, \(p_n, q_n \geq 2^{\lfloor n/2 \rfloor}\).

This proves that any rational number \(p/q\) with \(p, q < 2^n\) has a continued fraction representation \([a_0, \ldots, a_m]\) with \(m \leq 2n\).

Theorem 3.33. Let \(p \in \mathbb{Z}\), \(q \in \mathbb{N} \setminus \{0\}\) and \(x \in \mathbb{Q}\) such that \(\gcd(p, q) = 1\) and \(|p/q - x| \leq 1/2q^2\). Then \(p/q\) is a convergent of the continued fraction representation for \(x\).

Proof. Consider the continued fraction representation \([a_0, \ldots, a_n]\) of \(p/q\) with convergents \(p_1/q_1, \ldots, p_n/q_n = p/q\). Since \([a_0, \ldots, a_n] = [a_0, \ldots, a_{n-1}, a_n - 1, 1]\), we can assume without loss of generality that \(n\) is even. Let \(\delta \in \mathbb{R}\) be defined by the equation

\[
x = \frac{p_n}{q_n} + \frac{\delta}{2q_n^2}.
\]

Since \(|p/q - x| \leq 1/2q^2\) we have \(|\delta| < 1\). Without loss of generality, \(\delta > 0\). Set

\[
\lambda := \frac{2}{\delta} \cdot (p_{n-1} \cdot q_n - p_n \cdot q_{n-1}) - \frac{q_{n-1}}{q_n}.
\]

We have

\[
\lambda p_n + p_{n-1} = \frac{2 \cdot p_n \cdot q_n \cdot (p_{n-1} \cdot q_n - p_n \cdot q_{n-1})}{\delta \cdot q_n} - \frac{\delta \cdot q_{n-1} \cdot p_n + \delta \cdot q_n \cdot p_{n-1}}{\delta \cdot q_n} = \frac{(2 \cdot p_n \cdot q_n + \delta)(p_{n-1} \cdot q_n - p_n \cdot q_{n-1})}{\delta \cdot q_n}.
\]
and
\[ \lambda \cdot q_n + q_{n-1} = \frac{2 \cdot q_n^2 (p_{n-1} \cdot q_n - p_n \cdot q_{n-1}) \delta \cdot q_n}{\delta \cdot q_n} - q_{n-1} + q_{n-1} \]
\[ = \frac{2 \cdot q_n^2 (p_{n-1} \cdot q_n - p_n \cdot q_{n-1})}{\delta \cdot q_n}. \]

Hence,
\[ \frac{\lambda p_n + p_{n-1}}{\lambda q_n + q_{n-1}} = \frac{2 \cdot p_n \cdot q_n + \delta}{2q_n^2} = \frac{p_n}{q_n} + \frac{\delta}{2q_n^2} = x. \]

By Theorem 3.31, this implies that \( x = [a_0, \ldots, a_n, \lambda] \). Since \( n \) is even, \( p_{n-1} \cdot q_n - p_n \cdot q_{n-1} = 1 \). Hence,
\[ \lambda = \frac{2}{\delta} - \frac{q_{n-1}}{q_n} > 2 - 1 = 1. \]

Since \( \lambda \) is a rational number \( > 1 \), \( \lambda \) has a finite continued fraction representation \( \lambda = [b_0, \ldots, b_m] \) with \( b_0 \geq 1 \). Hence \( x = [a_0, \ldots, a_n, b_0, \ldots, b_m] \) is a continued fraction representation of \( x \) with convergent \( p/q \).

3.5.4 Complexity

Shor’s algorithm is summarised as Algorithm 3.1. To evaluate the time complexity and success probability of Shor’s algorithm, let \( k = \lfloor \log n \rfloor + 1 \) the length of the binary representation of \( n \). Hence, \( m \leq 2k \).

Steps 1–2 of Shor’s algorithm can be performed in time \( O(k^3) \) and produce either a factor of \( n \) or confirm that \( n \) is neither even nor a prime power. Step 3 can also be performed in time \( O(k^3) \) and produces either a factor of \( n \) or a randomly chosen element \( a \in \mathbb{Z}_n^* \). As we have shown, Step 4 can be implemented by a QGA with \( O(k^3) \) gates on 1 or 2 qubits. Step 5 also takes time \( O(k^3) \) and succeeds with probability \( \Omega(1/ \log k) \) (see Corollary 3.29). Finally, Step 6 takes time \( O(k^3) \) as well and succeeds with probability \( \geq \frac{9}{16} \) (by Lemma 3.21).

**Theorem 3.34.** Shor’s algorithm computes, given a composite number \( n \in \mathbb{N} \), a non-trivial factor of \( n \) with probability \( \geq 9/(160 \log \log n) \).
Algorithm 3.1. Shor’s factorisation algorithm

*input* \( n \in \mathbb{N} \) composite

1. if \( n \) is even then output 2 end.
2. if \( n = a^k \) for some \( a \in \mathbb{N}, k \geq 2 \) then output \( a \) end.
3. randomly choose \( a \in \{1, 2, \ldots, n - 1\} \)
   \[ d := \gcd(a, n) \]
   if \( d > 1 \) then output \( d \) end.
4. compute \( m \in \mathbb{N} \) such that \( n^2 \leq 2^m < 2n^2 \)
   \[ |\varphi\rangle := \frac{1}{2^m} \sum_{l=0}^{2^m-1} \sum_{y=0}^{2^n-1} e^{2\pi i \cdot yl/2^m} \sum_{q=0}^{s_l} e^{2\pi i \cdot yrq/2^m} |y\rangle |al \mod n\rangle \]
   measure first \( m \) qubits of \( |\varphi\rangle \) to obtain \( y \in \mathbb{Z}_{2^m} \)
5. compute convergents \( p_j/q_j \) of \( y/2^m \)
   \[ i := \min\{j : a^{q_i} \equiv 1 \pmod n\} \cup \{\infty\} \]
   if \( i = \infty \) then output ? end else \( r := q_i \)
6. if \( a^r \) is odd or \( a^{r/2} \equiv -1 \pmod n \) then
   output ?
   else
   \[ d := \gcd(n, a^{r/2} - 1); \text{ output } d \]

The algorithm can be implemented using \( O(\log n^3) \) classical operations and \( O(\log n^3) \) elementary quantum gates.

By repeating the algorithm \( \log n \) times, we are able to find a factor with very high probability.