An introduction to quantum computing

Carlos Cotrini
Department of Computer Science
ETH Zürich

September 14, 2019

1 Introduction

Quantum computing is the use of quantum phenomena like entanglement and superposition to do efficient computations that, in some cases, cannot be efficiently done by a classical computer. For example, assume we are given a boolean array of even length that is either constant —all entries are zero— or balanced —exactly half of its entries are zero—. A classical computer needs to access at least half of this array's entries to decide if the array is constant or balanced. However, a quantum computer can decide this by "accessing only one entry". We use quotation marks because this is not entirely accurate yet, but our goal at this point is just to illustrate the potential of quantum computers.

These notes give a simplified introduction to quantum computation. No knowledge of quantum mechanics or complex algebra is required. Only knowledge from a second year in a bachelor of computer science is required.

2 Preliminaries

2.1 Linear algebra

Definition 1. A *vector* is an ordered sequence of complex numbers[1]. A vector's *length* is the total of numbers occurring in it (repetitions of a number are also counted). We let $\mathbb{C}^n$ be the set of all vectors of length n.

We write a vector of length n as follows $(a_0, a_1, \ldots, a_n)$, where a_0, a_1, …, a_n are the numbers in the vector. We also sometimes write vectors as follows:

$$\begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix}.$$

Definition 2. A *matrix* is a rectangular array of complex numbers. A matrix's *size* is the expression $N \times M$, where N and M are the number of rows and columns in the matrix, respectively. For a matrix A, we denote the number in the i-th row and j-th column as A_{ij}.

[1] If you are not familiar with complex numbers, do not worry. Just read along replacing "real" with "complex" throughout these notes.

We write a matrix A of size $N \times M$ as follows:

$$\begin{pmatrix} A_{11} & A_{12} & \dots & A_{1M} \\ A_{21} & A_{22} & \dots & A_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ A_{N1} & A_{N2} & \dots & A_{NM} \end{pmatrix}$$

Observe that a vector of length N can be seen as a matrix of size $N \times 1$ or also as a matrix of size $1 \times N$.

Definition 3. If A and B are matrices of the same size. Then $A + B$ is the matrix such that $(A + B)_{ij} = A_{ij} + B_{ij}$.

Example 1.

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} + \begin{pmatrix} 100 & 200 \\ 300 & 400 \end{pmatrix} = \begin{pmatrix} 101 & 202 \\ 303 & 404 \end{pmatrix}.$$

In particular, the sum of two vectors of size n can be described as follows:

$$(a_0, a_1, \dots, a_n) + (b_0, b_1, \dots, b_n) = (a_0 + b_0, a_1 + b_1, \dots, a_n + b_n).$$

Definition 4. Let A be a matrix and y be a complex number, then the matrix yA is the matrix such that $(yA)_{ij} = yA_{ij}$.

Example 2.

$$7\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} = \begin{pmatrix} 7 & 14 \\ 21 & 28 \end{pmatrix}.$$

Example 3.

$$7(1, 2, 3, 4) = (7, 14, 21, 28).$$

Definition 5. Let A be a matrix of size $N \times M$. The *transpose of* A is the matrix $A^\top$ of size $M \times N$ such that $\left(A^\top\right)_{ij} = A_{ji}$.

Example 4.

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}^\top = \begin{pmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{pmatrix}$$

Definition 6. Let A and B be matrices of size $N \times M$ and $M \times R$, respectively. Then the *product of* A *and* B is the matrix AB of size $N \times R$ such that

$$(AB)_{ij} = \sum_{j=1}^{M} A_{ik} B_{kj}.$$

Example 5. Let A and B be the following matrices

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \qquad B = \begin{pmatrix} B_{11} & B_{12} & B_{13} \\ B_{21} & B_{22} & B_{23} \end{pmatrix}.$$

Then the product of A and B can be depicted as follows:

$$\begin{pmatrix} B_{11} & B_{12} & B_{13} \\ B_{21} & B_{22} & B_{23} \end{pmatrix}$$

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \quad \begin{pmatrix} A_{11}B_{11} + A_{12}B_{21} & A_{11}B_{12} + A_{12}B_{22} & A_{11}B_{13} + A_{12}B_{23} \\ A_{21}B_{11} + A_{22}B_{21} & A_{21}B_{12} + A_{22}B_{22} & A_{21}B_{13} + A_{22}B_{23} \end{pmatrix}$$

Definition 7. A *linear transformation* is a function $T : \mathbb{C}^N \to \mathbb{C}^N$ such that for any $\lambda_1, \lambda_2 \in \mathbb{C}$ and any two vectors $\psi_1, \psi_2 \in \mathbb{C}^N$

$$T(\lambda_1\psi_1 + \lambda_2\psi_2) = \lambda_1 T(\psi_1) + \lambda_2 T(\psi_2).$$

2.2 Exercises on linear algebra

Let A, B and C be matrices of sizes $M \times N$, $N \times Q$, and $N \times Q$, respectively. Let $\lambda \in \mathbb{C}$. Prove (or at least convince yourself) of the following facts.

1. $AB \neq BA$, in general.
2. $(AB)^\top = B^\top A^\top$.
3. $A(\lambda B) = (\lambda A) B = \lambda (AB)$.
4. $A(B + C) = AB + AC$.
5. $(B + C)A = BA + CA$.

3 Quantum computation

3.1 Qubits

A qubit is the fundamental data unit in quantum computing, analogous to a bit in classical computing.

Definition 8. A qubit is a pair (a_0, a_1) of complex numbers such that $|a_0|^2 + |a_1|^2 = 1$.

If you are not familiar with complex numbers, do not worry. Simply replace "complex" with "real" and replace any complex notion with its analogous real counterpart. For example, you can imagine a qubit as a pair (a_0, a_1) of real numbers such that $|a_0|^2 + |a_1|^2 = 1$, where $|a_0|$ and $|a_1|$ are a_0 and a_1's absolute values. However, bear in mind that this analogy is just limited to these notes and if you want to understand more advanced topics like Shor's algorithm you must become familiar with complex numbers.

Intuitively, a qubit (a_0, a_1) can be interpreted as a bit that is simulatenously 1 and 0, just like Schrödinger's cat. When we measure or observe this qubit, then we obtain a value of 0 with probability $|a_0|^2$ and a value of 1 with probability $|a_1|^2$. Once a qubit has been measured, it loses its uncertainty. If a 0 was obtained, then the qubit becomes $(1, 0)$ and $(0, 1)$ otherwise.

3.2 Qubit arrays

A classical bit array of length n is simply an ordered sequence of n bits. One would therefore think that a qubit array of length n is also an ordered sequence of n qubits, but this is inaccurate, as it ignores the fact that qubits in the array may become "entangled" after manipulating them. We will understand better quantum entanglement later in Section 3.6. For the moment, we ask the reader to bear with us and accept the definition we give next.

Definition 9. A qubit array of length n is a vector $|\psi\rangle = (a_0, a_1, \ldots, a_{N-1}) \in \mathbb{C}^N$, with $N = 2^n$, and such that $\sum_{x<N} |a_x|^2 = 1$. For $x \leq N$, let $[x]_2$ be the bit array of length n describing x in base 2.

The value $|a_x|^2$ indicates the probability that we observe the classical bit array $[x]_2 \in \{0,1\}^n$ after measuring $|\psi\rangle$. For simplicity, we won't distinguish between x and $[x]_2$.

Example 6. The qubit array $(0, 1/\sqrt{2}, 0, -1/\sqrt{2})$ can, after being measured, be either the bit array 01 or the bit array 11, each with probability 1/2.

Example 7. The qubit array $(1/\sqrt{2}, 0, 0, 0, 1/2, 0, 0, -1/2)$ can, after being measured, be either 000, 100, or 111, each with probabilities 1/2, 1/4, or 1/4.

We distinguish a useful set of qubit arrays. For a classical bit array $x \in \{0,1\}^n$, we define the qubit array $|x\rangle$ as the vector whose entry for x is 1 and zero for all other entries. We define

$$\mathcal{B}_n := \{|x\rangle \mid x \in \{0,1\}^n\}\,. \tag{1}$$

We usually denote qubit arrays with $|\psi\rangle$, with ψ a Greek letter. Qubit arrays in $\mathcal{B}_n$ are denoted with $|x\rangle$, with x a Latin letter. One can show that any qubit $|\psi\rangle = (a_0, a_1, \ldots, a_{N-1})$ can be rewritten as

$$|\psi\rangle = \sum_{x\in\{0,1\}^n} a_x\,|x\rangle\,. \tag{2}$$

This algebraic representation of $|\psi\rangle$ facilitates the computation with quantum logic gates. It also contains all the information that determines $|\psi\rangle$.

A famous qubit array that we will encounter often is

$$|?\rangle := \sum_{x\in\{0,1\}^n} \frac{1}{\sqrt{2^n}}\,|x\rangle\,. \tag{3}$$

Observe that this is one of the most "uncertain" qubit arrays. Any bit array is equally likely to be observed.

Those familiar with linear algebra will recognize that qubit arrays are vectors in the unit sphere of $\mathbb{C}^N$ and that $\mathcal{B}_n$ is a basis of $\mathbb{C}^N$.

3.3 Superpositions

We model a qubit array $|\psi\rangle$ of length n with a vector of 2^n complex numbers. However, $|\psi\rangle$ is a storage unit with n bits of capacity. The array $|\psi\rangle$ is not some sort of data structure or device that stores all 2^n bit arrays of length n in some clever way. It is physically analogous to a classical bit array with n bits of capacity. All 2^n bit arrays coexist at the same time on the same n bits. This unusual phenomenon is called *superposition*.

3.4 Measuring quantum bits and quantum bit arrays

In quantum mechanics, it is possible to construct mechanisms that "measure" quantum bits, called *measurement operators*. In these notes, we restrict ourselves to a set of very simple measurement operators. These operators receive as input a quantum bit and output a classical bit. Intuitively, a measurement operator "looks into" the quantum bit

$a_0 |0\rangle + a_1 |1\rangle$ to see what value it contains. A measurement operator outputs 0 and 1 with probabilities $|a_0|^2$ and $|a_1|^2$, respectively. After performing a measuring operation, all uncertainty carried by a quantum bit is lost and the quantum bit has become essentially a classical bit.

We also consider here simple measurement operators for quantum bit arrays. If a measurement operator receives as input the quantum bit array $\sum_{x\in\{0,1\}^n} a_x |x\rangle$, then the operator outputs the bit array x with probability $|a_x|^2$. Just as in the case of quantum bits, after performing a measuring operations, the quantum bit array becomes the observed classical bit array.

3.5 Quantum gates

Just as a classical logical gates transforms a bit array, a quantum gate is a quantum mechanism that transforms qubit arrays of length n into qubit arrays of the same length.

Quantum gates can be modeled as *unitary transformations on the vector space* $\mathbb{C}^N$, the space where qubit arrays reside. A unitary transformation is a function $G : \mathbb{C}^N \to \mathbb{C}^N$ with some special properties that we describe later in Section 6. What we only need to know for the moment is that such transformations are *linear*. This means that for any two qubit arrays $|\psi_1\rangle$ and $|\psi_2\rangle$ and any two α_1, $\alpha_2 \in \mathbb{C}$,

$$G(\alpha_1 |\psi_1\rangle + \alpha_2 |\psi_2\rangle) = \alpha_1 G |\psi_1\rangle + \alpha_2 G |\psi_2\rangle . \tag{4}$$

This means that for a quantum gate G and any qubit array $\sum_x a_x |x\rangle$

$$G\left(\sum_x a_x |x\rangle\right) = \sum_x a_x G |x\rangle . \tag{5}$$

Hence, a quantum gate is determined by only how it transforms the qubit arrays in $\mathcal{B}_n$.

We now present the quantum gates that constitute the circuits presented in these notes.

Hadamard gate. This gate, usually denoted with the letter H, is defined for $|x\rangle \in \mathcal{B}_n$ as follows:

$$H |x\rangle := \sum_{y\in\{0,1\}^n} \frac{(-1)^{x^\top y}}{\sqrt{2^n}} |y\rangle , \tag{6}$$

where $x^\top y := \sum_{i\le n} x[i]y[i]$ is the classical inner product. In particular, for $\mathbf{0} := (0, 0, \ldots, 0)$,

$$H |\mathbf{0}\rangle = \sum_{y\in\{0,1\}^n} \frac{1}{\sqrt{2^n}} |y\rangle = |?\rangle . \tag{7}$$

Observe that we defined H only for $\mathcal{B}_n$. How is $H |\psi\rangle$ defined in general for any qubit array $|\psi\rangle = \sum_x a_x |x\rangle$? Recall that quantum gates are linear transformations. Therefore,

$$H |\psi\rangle = H\left(\sum_x a_x |x\rangle\right) = \sum_x a_x H |x\rangle = \sum_{x,y} \frac{a_x (-1)^{x^\top y}}{\sqrt{2^n}} |y\rangle \tag{8}$$

$$= \sum_y \left(\sum_x \frac{a_x (-1)^{x^\top y}}{\sqrt{2^n}}\right) |y\rangle . \tag{9}$$

Emulation gate. Quantum gates happen to be able to efficiently emulate any classical Boolean gate. If $f : \{0,1\}^n \to \{0,1\}$ is a classical Boolean circuit, then it is possible to construct a quantum gate U_f that works on $|x\rangle \in \mathcal{B}_n$ as follows:

$$U_f |x\rangle := (-1)^{f(x)} |x\rangle . \tag{10}$$

More generally, by recalling that quantum gates are linear transformations, we get that for $\psi = \sum_x a_x |x\rangle$,

$$U_f |\psi\rangle = \sum_x a_x U_f |x\rangle = \sum_x (-1)^{f(x)} a_x |x\rangle . \tag{11}$$

We remark that if a classical Boolean circuit f runs in $O(K)$-time, then U_f also runs in $O(K)$-time.

Reflection gate. This is a simple quantum gate F that works on $\mathcal{B}_n$ as follows:

$$F |x\rangle \begin{cases} |\mathbf{0}\rangle & \text{if } x = \mathbf{0} \text{ and} \\ -|x\rangle & \text{otherwise.} \end{cases} \tag{12}$$

3.6 Quantum entanglement

We now present another counterintuitive property of quantum bit arrays, called *quantum entanglement*. For this, we introduce the CNOT gate. We do not give the full specification of this gate. For this discussion, it is enough to say that $\text{CNOT} |00\rangle := 1/\sqrt{2} |00\rangle + 1/\sqrt{2} |11\rangle$.

Observe that if we measure $|\varphi\rangle = \text{CNOT} |00\rangle$, then we can only obtain either 00 or 11, nothing else. In quantum terminology, we say that the two qubits in $|\varphi\rangle$ are *entangled*. This means that if you observe $|\varphi\rangle$'s first qubit and you see a 0, then you know with complete certainty, that $|\varphi\rangle$'s second qubit will be 0, after measuring it.

To emphasize the nature of quantum entanglement, suppose that Alice takes the qubit array $|00\rangle$ and computes $\text{CNOT} |00\rangle$. She then keeps $|\varphi\rangle$'s first qubit and gives $|\psi'\rangle$'s second qubit to Bob. Bob then travels to Alpha Centauri, which is at least 4 light years away from Earth. Suppose now that Alice measures her qubit and observes a 0. Then Bob's qubit will also yield 0 after measurement, even if he performs the measurement one second after Alice's measurement.

4 The Deutsch-Jozsa algorithm

4.1 Overview and intuition

Definition 10. A computable function $f : \{0,1\}^n \to \{0,1\}$ is *balanced* if $f(x) = 0$ for exactly half of the elements of its domain. The function f is *constant* if $f(x) = 0$ for all elements of its domain.

Definition 11. Given a function $f : \{0,1\}^n \to \{0,1\}$, known to be balanced or constant, decide if it is balanced.

Assume that a classical circuit takes $O(K)$-time to evaluate $f(x)$, for any x. As of today, classical computers can solve the problem above in $O(2^{n-1}K)$-time, as they must

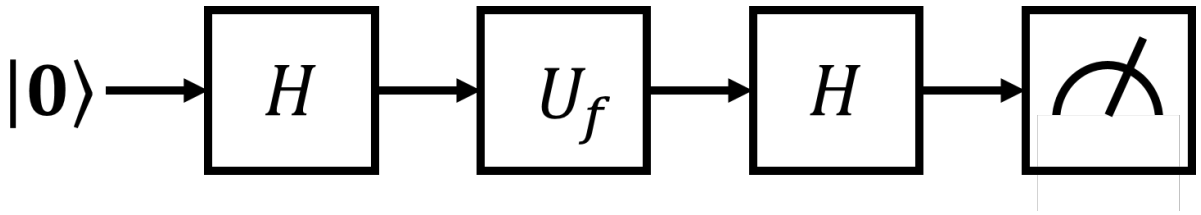

Figure 1: Quantum circuit for the Deutsch-Jozsa algorithm.

evaluate f on at least half of f's domain. Figure 1 presents a quantum circuit that solves the problem above in time $O(K)$-time.

The circuit receives as input the quantum bit array $|\mathbf{0}\rangle$, where $\mathbf{0} = (0, 0, \ldots, 0)$. The circuit consists of a Hadamard transformation, followed by an emulator U_f of f, then by another Hadamard transformation, and finally by a measurement.

5 Exercises on the Deutsch-Jozsa algorithm

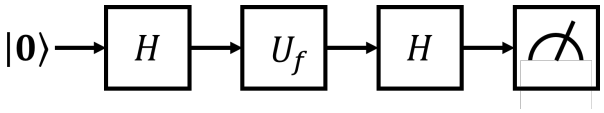

Figure 2: Quantum circuit for the Deutsch-Jozsa algorithm.

Figure 2 shows the circuit for the Deutsch-Jozsa algorithm. Recall that, for a qubit array $|x\rangle \in \mathcal{B}_n$,

$$H|x\rangle = \sum_{y\in\{0,1\}^n} \frac{(-1)^{x^\top y}}{\sqrt{2^n}} |y\rangle \text{ and } U_f|x\rangle = (-1)^{f(x)}|x\rangle .$$

1. Let $|\psi'\rangle$ be the qubit array obtained after applying H, U_f, and H, in that order, to $|00\ldots0\rangle$, as dictated by the Deutsch-Jozsa algorithm. Prove that

$$|\psi'\rangle = \sum_{z\in\{0,1\}^n} \left(\sum_{y\in\{0,1\}^n} \frac{(-1)^{z^\top y + f(y)}}{\sqrt{2^n}} \right) |z\rangle .$$

2. If f is constant, what is the probability that we get the bit array $00\ldots0$ when we measure $|\psi'\rangle$? What if f is balanced?

3. Conclude that the Deutsch-Jozsa algorithm decides whether a function f is constant or balanced.

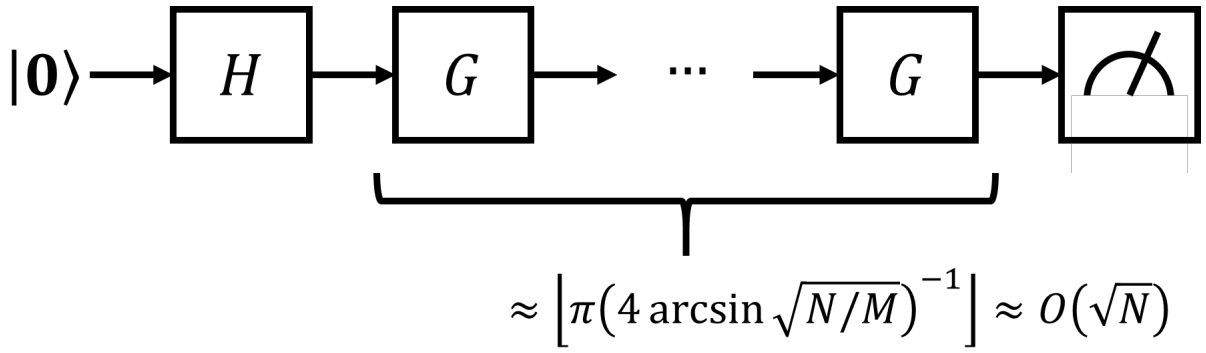

Figure 3: Quantum circuit for Grover's algorithm.

6 Grover's algorithm

6.1 Unitary transformations

A *linear transformation* is a function $T : \mathbb{C}^N \to \mathbb{C}^N$ such that for any $a_1, a_2 \in \mathbb{C}$ and any two $|\psi_1\rangle, |\psi_2\rangle \in \mathbb{C}^N$,

$$T(a_1 |\psi_1\rangle + a_2 |\psi_2\rangle) = a_1 T |\psi_1\rangle + a_2 T |\psi_2\rangle . \tag{13}$$

A popular result in linear algebra is that every linear transformation is identified with a unique matrix $[\![T]\!] \in \mathbb{C}^{N\times N}$ such that $T|\psi\rangle = [\![T]\!]|\psi\rangle$, the product of the matrix $[\![T]\!]$ and the vector $|\psi\rangle$. $[\![T]\!]$ is the matrix whose x-th column is $T|x\rangle$, for $x \in \{0,1\}^n$.

Another popular result states that if T_1 and T_2 are linear transformations, then $T_1(T_2|x\rangle) = [\![T_1]\!][\![T_2]\!]|x\rangle$. From now on, we identify T with $[\![T]\!]$.

A quantum gate G is a unitary transformation. A linear transformation $G : \mathbb{C}^N \to \mathbb{C}^N$, with $N = 2^n$, is a *unitary* if $G^\dagger G = I$. If you are not familiar with complex algebra, then you can think of $G^\dagger$ as $G^\top$, G's transpose.

Recall that the transpose of a squared matrix G is the matrix $G^\top$ obtained by "mirroring" G through its diagonal. More precisely, for $i, j \leq N$, we have that $\left(G^\top\right)_{ij} = G_{ji}$.

6.2 Overview and intuition

Definition 12. Given a function $f : \{0,1\}^n \to \{0,1\}$, find an element $x \in \{0,1\}^n$ such that $f(x) = 1$. We call such an element *a solution of* f. Let $N := 2^n$ and assume that there are $M \ll N$ solutions of f.

We assume M to be known. There are more involved quantum algorithms that can solve the problem above when M is unknown.

Figure 3 shows a quantum circuit implementing Grover's algorithm, a quantum algorithm that computes a solution of f in $O\left(\sqrt{N}K\right)$-time, where $O(K)$ is the time complexity for computing $f(x)$, for any $x \in \{0,1\}^n$. The algorithm consists of a Hadamard transformation followed by a sequence of ω *Grover rotations*, with $\omega \approx \lfloor \frac{\pi}{2} \arcsin \sqrt{\frac{M}{N}} \rfloor$. At the end a measurement is performed.

The Hadamard transformation yields $|?\rangle$. Let $|\sigma\rangle := \frac{1}{\sqrt{M}} \sum_{x:f(x)=1} |x\rangle$ and $|\sigma\rangle^\perp := \frac{1}{\sqrt{N-M}} \sum_{x:f(x)=0} |x\rangle$. One can show that $|\sigma\rangle$ and $|\sigma\rangle^\perp$ are orthogonal (i.e., they form an angle of 90 degrees) and have the same length as $|?\rangle$. Moreover, $|?\rangle$ lies in the plane

defined by these two vectors. Therefore, we can visualize $|\sigma\rangle$, $|\sigma\rangle^{\perp}$, and $|?\rangle$ as shown in Figure 4a.

After obtaining $|?\rangle$, we apply a series of Grover rotations transforms $|?\rangle$ into a vector $|\psi^{\omega}\rangle$ that is very close to $|\sigma\rangle^{\perp}$. As a result, a measurement of $|\psi^{\omega}\rangle$ will yield, with high probability, a solution of f.

Recall that we assume $M \ll N$. Hence, $|?\rangle$ stands very close to $|\sigma\rangle^{\perp}$. Let $\theta/2$ be the angle between $|?\rangle$ and $|\sigma\rangle^{\perp}$. It can be shown that $\theta/2 = \arcsin\sqrt{M/N}$. We will see later in Section 6.3, that the Grover rotation G rotates $|\sigma\rangle^{\perp}$ towards $|\sigma\rangle$ by θ, as shown in Figure 4b.

Our goal is to apply several rotations to $|?\rangle$ until the result $|\psi^{\omega}\rangle$ is very close to $|\sigma\rangle$. In this way, if we perform a measurement, then we get a solution of f with high probability. If we do some arithmetic, we can conclude that after $\omega = \lfloor\frac{\pi-\theta}{2\theta}\rfloor = O(\lfloor\frac{\pi}{2\theta}\rfloor)$ rotations, we can expect $|\psi^{\omega}\rangle$ to be very close to $|\sigma\rangle$.

The algorithm's running time consists of the running time of a Hadamard transformation, which is constant, plus ω times performing the Grover rotation. Each Grover rotation takes $O(K)$-time, because this is the time that takes to evaluate f via the emulator U_f. So Grover's algorithm takes $O(\omega K)$-time. Now, since $M \ll N$, we have that $\frac{\theta}{2} = \arcsin\sqrt{\frac{M}{N}}$ is very small. For such values, we have that $\arcsin\varphi \approx \varphi$. Therefore, $\omega = \lfloor\frac{\pi}{2\theta}\rfloor \approx \lfloor\frac{\pi}{4}\sqrt{\frac{N}{M}}\rfloor \leq \lfloor\frac{\pi}{4}\sqrt{N}\rfloor = O\left(\sqrt{N}\right)$.

6.3 The Grover rotation

We now explain in detail how the Grover rotation works. Figure 5 illustrates the circuit that implements a Grover rotation. It consists of an emulator U_f of f, a Hadamard transformation H, a reflection F through the line spanned by $|\mathbf{0}\rangle$, and another Hadamard transformation H.

To understand why the Grover rotation rotates $|?\rangle$ by θ towards $|\sigma\rangle$, we compute first its matrix representation. The matrix representation of a composition of linear transformations happens to be the product of the matrix representation of each transformation. Thus, the Grover rotation's matrix representation is

$$G = HFHU_f. \tag{14}$$

Observe that $F = 2\,|\mathbf{0}\rangle\,|\mathbf{0}\rangle^{\top} - I$, where I is the identity matrix. Therefore,

$$G = H\left(2\,|\mathbf{0}\rangle\,|\mathbf{0}\rangle^{\top} - I\right)HU_f \tag{15}$$

$$= \left(2H\,|\mathbf{0}\rangle\,|\mathbf{0}\rangle^{\top} H - H^2\right)U_f. \tag{16}$$

Recall that $H\,|\mathbf{0}\rangle = |?\rangle$. Using linear algebra, we can show that H is symmetric; that is, $H^{\top} = H$. Thus, $|\mathbf{0}\rangle^{\top} H = |\mathbf{0}\rangle^{\top} H^{\top} = (H\,|\mathbf{0}\rangle)^{\top} = |?\rangle^{\top}$. Hence,

$$G = \left(2\,|?\rangle\,|?\rangle^{\top} - H^2\right)U_f. \tag{17}$$

Using the fact that H is unitary and symmetric, we get that $H^{-1} = H$, so $H^2 = H^{-1}H = I$. This means that

$$G = \left(2\,|?\rangle\,|?\rangle^{\top} - I\right)U_f. \tag{18}$$

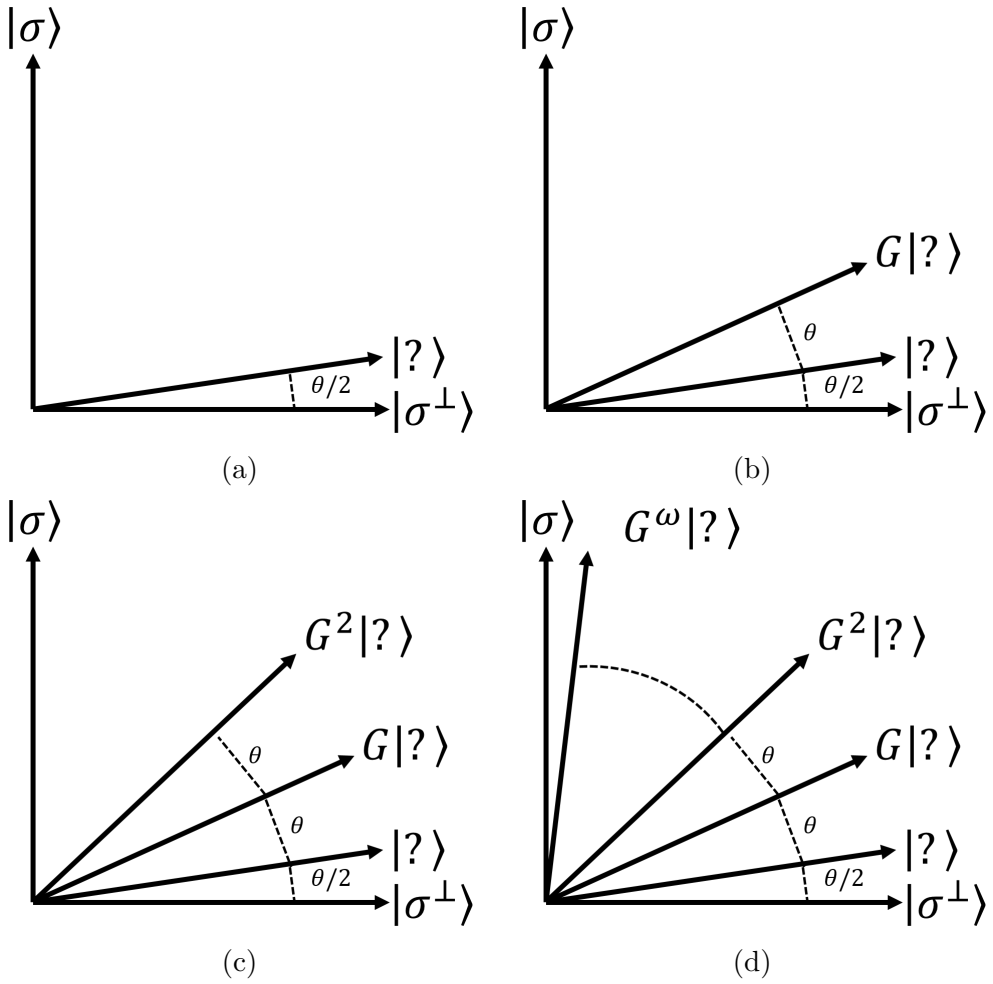

Figure 4: An illustration of how Grover's algorithm works. The algorithm first takes the qubit array $|\mathbf{0}\rangle$ and passes it through a Hadamard gate, yielding $|?\rangle$ (Figure 4a). Afterwards, it applies Grover's rotation $\omega = \lfloor \frac{\pi}{2} \arcsin \sqrt{M/N} \rfloor$ times (Figures 4b–4d). The resulting qubit array $G^{\omega}|?\rangle$ is very close to $|\sigma\rangle$. Therefore, a measurement of $G^{\omega}|?\rangle$ is very likely to yield a solution of f.

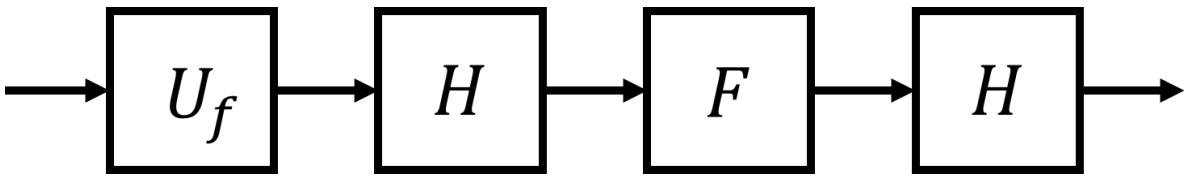

Figure 5: Quantum circuit for Grover's rotation.

Observe now that U_f is a reflection through $\left|\sigma^\top\right\rangle$. Similarly, $\left(2\left|?\right\rangle\left|?\right\rangle^\top - I\right)$ is another reflection. Using linear algebra, we can show that the product of two reflections is a rotation. In this case, the rotation is by an angle of θ towards $|\sigma\rangle$. We conclude then that the Grover rotation indeed rotates vectors in the span of $|\sigma\rangle$ and $\left|\sigma^\top\right\rangle$ by θ towards $|\sigma\rangle$.

7 Exercises on Grover's algorithm

For the following exercises, the following properties of matrix algebra are useful. Let A, B and C be matrices of adequate sizes and let $\lambda \in \mathbb{C}$.

- $AB \neq BA$, in general.
- $(AB)^\top = B^\top A^\top$.
- $A\,(\lambda B) = (\lambda A)\,B = \lambda\,(AB)\,.$
- $A(B + C) = AB + AC$.
- $(B + C)A = BA + CA$.

1. Let $|\sigma\rangle := \sum_{x:f(x)=1} \sqrt{\frac{1}{M}}\,|x\rangle$ and $\left|\sigma^\perp\right\rangle := \sum_{x:f(x)=0} \sqrt{\frac{1}{N-M}}\,|x\rangle$.

   (a) Prove that $|\sigma\rangle$ and $\left|\sigma^\perp\right\rangle$ are normal and orthogonal.

   (b) Prove that $|?\rangle$ lies in the span of these two vectors.

2. The Grover rotation is implemented as $G \equiv U_f \to H \to F \to H$.

   (a) Show that F's matrix representation is $2\,|00\ldots0\rangle\,|00\ldots0\rangle^\top - I$.

   (b) Show that $H^\top = H$.

   (c) Show that the matrix representation of HFH is $2\,|?\rangle\,|?\rangle^\top - I$.

   (d) Show that U_f is a reflection through the qubit array $\left|\sigma^\perp\right\rangle$. Recall that a linear transformation is a reflection through a vector v if its matrix representation is $2vv^\top - I$.

   (e) Conclude that G performs a rotation. It can be shown that this rotation is done by an angle of $\theta = 2\arcsin\sqrt{M/N}$ towards $|\sigma\rangle$.

3. Write down the circuit implementing Grover's algorithm and argue why it computes a solution of f with high probability.