Quantum Computation – CS 7805 Lecture Notes

Artem Pelenitsyn and Max Daniels

April 7, 2020

Contents

Introduction

The works of Paul Benioff, Richard Feynman, and Yuri Manin from early 1980-s presented the idea of using the effects of quantum mechanics in computation. By 1994, Peter Shor came up with an idea and design of a quantum polynomial-time algorithm for factoring integer numbers — the problem without an efficient (probabilistic or deterministic) solution to this date.

The essence of the speedup achieved lies in so called *quantum parallelism* — an observation that the computational space increases exponentially with the size of the quantum system. One more (quantum) bit of input allows for computation over two times more of states simultaneously.

Quantum parallelism does not come for free, though: at the end of computation you have to "measure" its result; this will collapse all parallel states into a single one with a certain probability. Only certain algorithm designs can benefit from this style of parallel probabilistic computation.

1 Computational Model

1.1 Quantum Information: Qbits

We define the representation of information we are going to compute over as follows.

Postulate 1 (State Space Postulate)**.** The state of a system is described by a unit vector in a Hilbert space $\mathcal{H}$.

The word "system" here refers to some physical reality used to encode information, akin to transistors used to encode classical bits. A quantum system can be implemented with some elementary particle and its property. E.g. an electron and its spin, a photon and its polarization, etc.

Recall that a Hilbert space is a complex linear space with a scalar product structure on it. Hilbert space can have infinite dimensions and that is often the case when working out quantum mechanics but for the purpose of quantum computing it suffices to employ the finite-dimensional case.

For a finite-dimensinal $\mathcal{H}$: there is a natural number n such that $\dim(\mathcal{H}) = n$. The dimensionality depends on the degree of freedom of the concrete system. We will assume that we have at our disposal an unbounded number of replicas of a 2-dimensional system, which (for the reasons explained later) we will call *qubits*.

A state of a qubit, therefore, is described by a pair of complex numbers (a member of the coordinate space $\mathbb{C}^2$) and a choice of basis in the state space. We assume that there exists the standard basis which we denote $|0\rangle$, $|1\rangle$. E.g. if our concrete realization of qubits is the spin characteristic of an electron, the usual choice of the basis would be: $|0\rangle$ for spin-up and $|1\rangle$ for spin-down.

The algebraic form describing a state of a qubit is:

$$\alpha_0 |0\rangle + \alpha_1 |1\rangle , \qquad \alpha_i \in \mathbb{C} : |\alpha_0|^2 + |\alpha_1|^2 = 1.$$

We usually call α_i the *amplitudes*. They are complex numbers although there is a theoretical result stating that we can limit ourselves to real numbers without a loss of generality. In that case, the state of a qubit can be pictured as a point on the unit circle.

Takeaway Our elementary piece of information, qubit, is a unit vector in a complex two-dimensional space.

1.2 Quantum Computation: Unitary Transformations

Given that qubits live in a sort of linear space, one would expect possible manipulations with them to preserve the linear structure. This is, indeed, the case.

One essential constraint attached to qubits is the unit length of the corresponding vectors. Therefore, the said transformations should preserve lengths.

Combining the two ideas we get to the Evolution Postulate.

Postulate 2 (Evolution Postulate)**.** The time-evolution of the state of a closed quantum system is described by a *unitary operator*.

In the finite-dimensional case (which is of main interest for us) unitary operators simply correspond to isometries (a linear transformation preserving lengths). In general, the definition is somewhat more involved but proves to be useful in showing even finite-dimensional operators unitary.

Unitary operators are those that preserve not only lengths but also angles. Recall that angles are measured with scalar product, so

$$U \text{ unitary} \Leftrightarrow (Ux, Uy) = (x, y).$$

Recalling that one can shuffle around the action of an operator around the scalar product using the notion of adjoint operator,

$$(Ux, Uy) = (x, U^*Uy),$$

we can get an to an equivalent formulation:

$$U \text{ unitary} \Leftrightarrow U^* = U^{-1}.$$

This formulation is sometimes easier to check in a finite-dimensional case. Recall that in terms of matrices $U^* = \{U^*_{ji}\}$, where U_{ij} is i, j-th element of U and a^* is the complex conjugate of a. Therefore, for real matrices (which we will mostly deal with), we want the matrix of the inverse operator to be equal to the trasposition of the matrix of the operator itself. Moreover, if a real operator is an involution ($U^2 = I$, i.e. it is self-inverse), then for it to be unitary it suffices to check that its own matrix is symmetric.

The postulate means that if the state $|\varphi_0\rangle$ of a qubit evolved over time into the state $|\varphi_1\rangle$ then there exists a unitary operator U, such that

$$|\varphi_1\rangle = U\,|\varphi_0\rangle\,.$$

All our computations will have to take form of unitary transformations.

One interesting aspect of quantum computing is that it is reversible: given the outputs of a quantum circuit we can always rebuild the inputs. This should seem very restrictive and indeed it is. But we will see that with some redundancy in the inputs we can model any classical computation.

Takeaway We work with qubits by means of isometric linear transformations.

1.3 Composite Systems

The next postulate provides a way to stack up qbits, similarly to how we compose bits into bit-strings (e.g. bytes, registers, etc.) in the classical setting.

Postulate 3 (Composition of Systems Postulate)**.** If one system is in the state $|\varphi_1\rangle$ and the second system in the state $|\varphi_2\rangle$, then the state of the combined system is described by the *tensor product*:

$$|\varphi_1\rangle \otimes |\varphi_2\rangle \in \mathcal{H}_1 \otimes \mathcal{H}_2, \qquad \text{if } |\varphi_i\rangle \in \mathcal{H}_i.$$

Notation We often omit $\otimes$ and write $|\varphi_1\rangle\ |\varphi_2\rangle$ or even $|\varphi_1\varphi_2\rangle$ instead of $|\varphi_1\rangle \otimes |\varphi_2\rangle$.

It is handy to think about the tensor product as a simple pairing operation which has only one non-trivial property — *bilinearity* (that is, it is linear in both arguments). So we can hoist and push down scalar multipliers and summations through this paring operation. The following example demonstrates how this works.

Consider two qubits described by

$$|\varphi_1\rangle = 1/\sqrt{2}(|0\rangle + |1\rangle), \qquad |\varphi_2\rangle = 1/\sqrt{2}(|0\rangle - |1\rangle).$$

Considered as a composite system, their collective state is described as follows:

$$|\varphi_1\rangle\,|\varphi_2\rangle = 1/2(|0\rangle + |1\rangle)(|0\rangle - |1\rangle) = 1/2(|00\rangle - |01\rangle + |10\rangle + |11\rangle).$$

It should be intuitively clear that the resulting composite state of two qubits is described with a 4-dimensional vector. In general, n-qubit system corresponds to a vector with 2^n dimensions. This is the source of quantum parallelism mentioned in the Introduction.

1.3.1 Entangled States

A pair of qubit is a system that is described by a 4-dimensional vector. An intriguing observation first made in this context by Einstein, Podolsky, and Rosen is that not every such vector can be represented as a product of two-dimensional ones. E.g.

$$|\varphi\rangle = 1/\sqrt{2}(|00\rangle + |11\rangle)$$

Such a state is indeed achievable from a basis like $|00\rangle$ using a simple unitary transformation so it should correspond to some valid state of a pair of qubits. But what are individual states of those qubits? This is not known. Such qubits are called *entangled* (or *EPR pairs*). The set of entangled qubits is dense in the 4-dimensional Hilbert space, so entanglement is not rare: "most" pairs of qubits are entangled.

We will see that entanglement is among the most important workhorses of quantum computing.

Takeaway We stack qubits up by tensoring (qlueing up) the corresponding state vectors. Not all vectors describing a pair (n-tuple) of qubits are possible to "unglue".

1.4 Measurement

We now sched some light on the notion of closed system mentioned in the previous section and when it comes to odds with our ability to access the parameters of the system, e.g. the amplitudes.

An important limitation of the model described so far is that the amplitudes cannot be recovered by a classical observer (e. g. a human) directly. From the classical point of view the possible states of the qubit are still either $|0\rangle$ or $|1\rangle$. Recall the infamous thought experiment with the Schrödinger's cat: the cat can only be either dead or alive but not $1/\sqrt{2}$ dead + $1/\sqrt{2}$ alive. Yet this is the kind of states, called *superpositions*, quantum mechanics deals with. And a superposition can be maintained (and even transformed using the Evolution Postualte) as long as the system remains unobserved. Once this condition is broken, the system shuts down back to one of classical states.

The only way to experience amplitudes requires the act of measurement that inevitably changes the state of the system.

Postulate 4 (Measurement Postulate)**.** Assume a quantum system A and its corresponding state space $\mathcal{H}_A$. For an orthonormal basis $B = \{|\varphi_i\rangle\}$ in $\mathcal{H}_A$, there exists a description of A:

$$|\varphi\rangle = \sum_i \alpha_i |\varphi_i\rangle , \qquad \sum_i |\alpha_i|^2 = 1.$$

It is possible to perform a *(Von Neumann) measurement* on system A with respect to the basis B. With probability $|\alpha_i|^2$, this act outputs a label i and leaves the system in state $|\varphi_i\rangle$.

Consider a pair of qubits in the state

$$|\varphi\rangle = \sqrt{1/11}\,|00\rangle + \sqrt{5/11}\,|01\rangle + \sqrt{2/11}\,|10\rangle + \sqrt{3/11}\,|11\rangle .$$

We can measure the system as a whole using the standard (0/1) basis and get 00 as the output with probability $1/11$, get 01 with probability $5/11$, etc. And if we get, say, 01 as the output, then the system ends up in the state $|01\rangle$.

A measurement can be performed on individual qubits of a system. Assume we want to measure the first qubit of the above mentioned system using the standard basis. From above, we know that the probability of getting 0 for the first qubit is $1/11 + 5/11 = 6/11$. In order to figure out what happens with the whole system after this measurement with the outcome of 0, we regroup the terms using bilinearity:

$$|\varphi\rangle = \sqrt{6/11}\,|0\rangle \left(\sqrt{1/6}\,|0\rangle + \sqrt{5/6}\,|1\rangle\right) + \sqrt{5/11}\,|1\rangle \left(\sqrt{2/5}\,|0\rangle + \sqrt{3/5}\,|1\rangle\right).$$

It is clear that if we get the output 0, the system ends up in the state

$$|0\rangle \left(\sqrt{1/6}\,|0\rangle + \sqrt{5/6}\,|1\rangle \right).$$

2 Quantum Gates

An n-qubit system has 2^n basis states. If we map these states to Euclidean basis vectors, the state space is like the points on the unit sphere in $\mathbb{C}^{2^n}$. To interact with these states, we apply *unitary transformations*, which are invertible maps $S^{2^n-1} \mapsto S^{2n-1}$ (geometrically, we need unitarity because we can only map the sphere to itself by twisting it around).

2.1 Basic Gates

Here are some unitary state transformations that act like logic gates. Again, we are implicitly mapping $|0\rangle \to e_0$ and $|1\rangle \to e_1$.

1. Identity (I gate):

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \begin{cases} |0\rangle \to |0\rangle \\ |1\rangle \to |1\rangle \end{cases} \tag{1}$$

2. Swap (X gate):

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \begin{cases} |0\rangle \to |1\rangle \\ |1\rangle \to |0\rangle \end{cases} \tag{2}$$

3. Swap & shift (Y gate, $Y = ZX$):

$$Y = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \quad \begin{cases} |0\rangle \to -\,|1\rangle \\ |1\rangle \to |0\rangle \end{cases} \tag{3}$$

4. Phase shift (Z gate):

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad \begin{cases} |0\rangle \to |0\rangle \\ |1\rangle \to -\,|1\rangle \end{cases} \tag{4}$$

5. Controlled Not (C_{not} gate):

$$C_{not} = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix} \quad \begin{cases} |00\rangle \to |00\rangle \\ |01\rangle \to |01\rangle \\ |10\rangle \to |11\rangle \\ |11\rangle \to |10\rangle \end{cases} \tag{5}$$

6. Walsh-Hadamard (H gate):

$$H = \begin{pmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{pmatrix} \quad \begin{cases} |0\rangle \to (1/\sqrt{2})(|0\rangle + |1\rangle) \\ |1\rangle \to (1/\sqrt{2})(|0\rangle - |1\rangle) \end{cases} \tag{6}$$

H is a rotation by $\pi/4$ and it superimposes $|0\rangle$ equally over the two states. Applying H iteratively to each bit of $|0^n\rangle$ superimposes equally over all of $|x\rangle, x \in \{0,1\}^n$

Proposition 0.1.

$$H_n |x\rangle = \frac{1}{(\sqrt{2})^n} \sum_{y=0}^{2^n-1} (-1)^{x \odot y} |y\rangle \tag{7}$$

Proof. Idea is: For the nth bit of y, if this bit is zero then its coefficient is $1/\sqrt{2}$. If this bit is one, then its coefficient is $-1/\sqrt{n}$ iff the corresponding nth bit of x is $|1\rangle$. The states accumulate a -1 factor for every Hadamard-child $|1\rangle$ of a parent $|1\rangle$. The $x \odot y$ counts occurrences of this. Simple example:

$$H |01\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \cdot \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \tag{8}$$

□

2.2 No Cloning

Theorem 0.1 (No Cloning)**.** *There is no unitary U satisfying*

$$U |a0\rangle = |aa\rangle \text{ for all qunatum states } |a\rangle \tag{9}$$

In this sense, no unitary transformation can clone qubits.

Proof. Set $|c\rangle = (1/\sqrt{2})(|a\rangle + |b\rangle)$.
Then

$$|c0\rangle = |c\rangle \otimes |0\rangle = (1/\sqrt{2})(|a\rangle + |b\rangle) \otimes |0\rangle = (1/\sqrt{2})(|a0\rangle + |b0\rangle) \tag{10}$$

$$U |c0\rangle = |cc\rangle = (1/\sqrt{2})U(|a0\rangle + |b0\rangle) = (1/\sqrt{2})(|aa\rangle + |bb\rangle) \tag{11}$$

$$\implies |cc\rangle = (1/\sqrt{2})(|aa\rangle + |bb\rangle) \text{ (an entangled state)} \tag{12}$$

$$\tag{13}$$

Whereas

$$|cc\rangle = |c\rangle \otimes |c\rangle = (1/2)(|aa\rangle + |ab\rangle + |ba\rangle + |bb\rangle) \text{ (an independent state)} \tag{14}$$

□

Remark 2.1. The reason for no cloning is because it would permit an independent state which is also entangled. Outside of quantum computation, it is a contradiction using linearity.

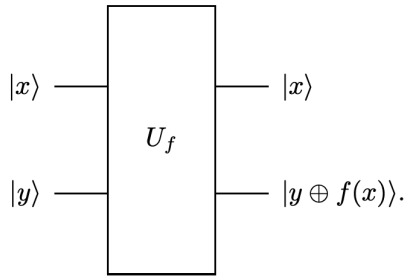

Figure 1: Circuit diagram of quantum gatearray. [3]

Remark 2.2. It is totally fine to use unitary gates to convert an entangled state into an independent state, and entanglement does not necessarily stick around forever. The impossible construction of this proof is a state which is simultaneously independent and entangled.

2.3 Completeness

Using a "double C_{not}" gate, called the Toffoli gate:

$$T = \begin{cases} |a,b,x\rangle \to |a,b,\neg x\rangle & a \wedge b = 1 \\ |a,b,x\rangle \to |a,b,x\rangle & a \wedge b = 0 \end{cases} \tag{15}$$

Then T provides both AND and NOT:

1. NOT: $T\,|1,1,x\rangle = |1,1,\neg x\rangle$
2. AND: $T\,|x,y,0\rangle = |x,y,x \wedge y\rangle$

Definition 2.1. Let f be an arbitrary classical function $\{0,1\}^m \to \{0,1\}^k$. A *quantum gate-array* U_f is defined as

$$U_f : |x,y\rangle \mapsto |x, y \oplus f(x)\rangle \tag{16}$$

This quantum operation is invertible, even if f itself is not invertible. As $f \bigoplus f = 0$ (like unitarity for additive inverses),

$$U_f U_f\,|x,y\rangle = U_f\,|x, y \oplus f(x)\rangle = |x,y\rangle \tag{17}$$

and so U_f is unitary.

3 EPR Paradox

Definition 3.1 (The Parity Game)**.** Alice and Bob are very far apart. Each receives a random bit, say x, y respectively. After seeing their random bits, each must provide a bit, say a, b. Alice and Bob win if

$$a \oplus b = x \wedge y \tag{18}$$

where $\oplus$ is XOR. To win, Alice and Bob must agree unless $x = y = 1$.

Most of the time, Alice and Bob want to agree. In the classical setting, a good strategy is to agree always. In fact, it is optimal. Assuming they are too far apart to conspire, their disagreement is essentially random. So, if Alice and Bob disagree some of the time, then conditioning on their disagreement they will lose more often than not ($3/4$ of the time). They could increase their expected score by reducing the chance of disagreement.

In the quantum case, Alice and Bob can use entanglement to conspire over long distances. To demonstrate this, we will use $|\rightarrow\rangle$ for 0 and $|\uparrow\rangle$ for 1. Because superpositions live on the unit circle, we can picture them as rotations like $|\nearrow\rangle$.

1. Construct a two qubit system in the entangled "EPR" state:

$$(1/\sqrt{2})(|\rightarrow\rightarrow\rangle + |\uparrow\uparrow\rangle) \tag{19}$$

2. Before Alice and Bob are separated, give Alice the first EPR bit, and Bob the second EPR bit.

3. Alice receives her random bit. If 1, Alice applies a rotation by $\pi/8$ to her qubit (without measuring). States are either

4. Bob receives his random bit and rotates by -22.5 if he receives a 1.

5. Alice and Bob read their qubits and output the bit they measure.

Alice and Bob will now disagree some of the time, *in a way that is correlated with* $x \wedge y$.

4 Dense Coding

Using a similar idea, Alice and Bob can deterministically communicate two bits using one qubit (and one pre-shared EPR bit).

Here is the algorithm:

1. Alice and Bob each receive one bit of a prepared EPR state $\varphi_0 = (1/\sqrt{2})(|00\rangle + |11\rangle)$.

2. Alice receives two classical bits, encoding a number 0 to 3. Depending on the value, she will perform one of $\{ I, X, Y, Z \}$:

Value	Transformation	New State
0	$\varphi_0 = (I \otimes I)\varphi_0$	$(1/\sqrt{2})(\|00\rangle + \|11\rangle)$
1	$\varphi_1 = (I \otimes X)\varphi_0$	$(1/\sqrt{2})(\|10\rangle + \|01\rangle)$
2	$\varphi_2 = (I \otimes Y)\varphi_0$	$(1/\sqrt{2})(-\|10\rangle + \|01\rangle)$
3	$\varphi_3 = (I \otimes Z)\varphi_0$	$(1/\sqrt{2})(\|00\rangle - \|11\rangle)$

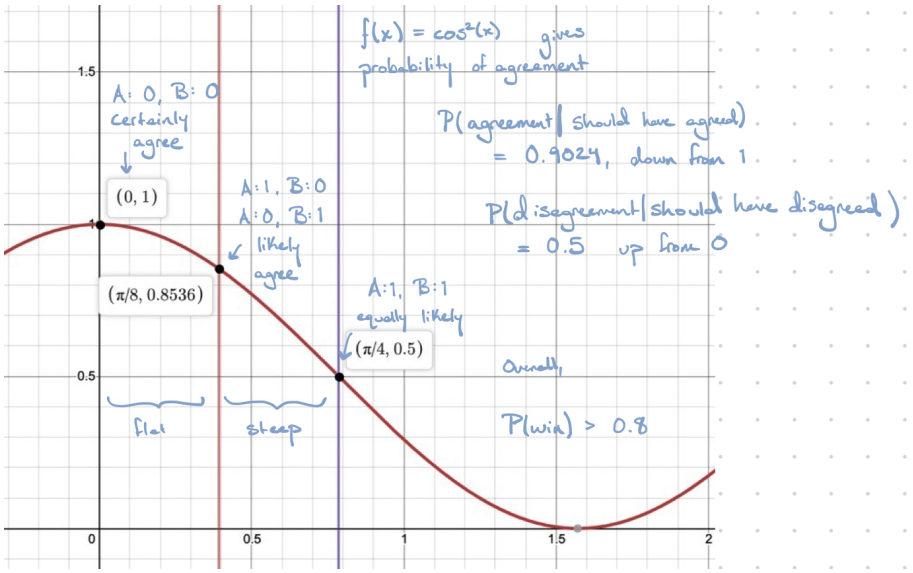

Figure 2: EPR Diagram. $(1/4)(1/2) > (3/4)(1/10)$, and the difference is 0.05, which is the payoff for using this correlation strategy.

3. Bob receives one of $\{ \varphi_0, \varphi_1, \varphi_2, \varphi_3 \}$. He applies a C_{not}, *which transforms all of these states to independent states*:

Initial State	After C_{not}	Independent State
$\varphi_0 = (1/\sqrt{2})(\|00\rangle + \|11\rangle)$	$(1/\sqrt{2})(\|00\rangle + \|10\rangle)$	$(1/\sqrt{2})(\|0\rangle + \|1\rangle) \otimes \|0\rangle$
$\varphi_1 = (1/\sqrt{2})(\|10\rangle + \|01\rangle)$	$(1/\sqrt{2})(\|11\rangle + \|01\rangle)$	$(1/\sqrt{2})(\|0\rangle + \|1\rangle) \otimes \|1\rangle$
$\varphi_2 = (1/\sqrt{2})(-\|10\rangle + \|01\rangle)$	$(1/\sqrt{2})(-\|11\rangle + \|01\rangle)$	$(1/\sqrt{2})(\|0\rangle - \|1\rangle) \otimes \|1\rangle$
$\varphi_3 = (1/\sqrt{2})(\|00\rangle - \|11\rangle)$	$(1/\sqrt{2})(\|00\rangle - \|10\rangle)$	$(1/\sqrt{2})(\|0\rangle - \|1\rangle) \otimes \|0\rangle$

4. Bob reads the second bit (without disturbing the first bit).

5. By inspection, the first bit is now in an output state of the Hadamard transformation. By unitarity, $H = H^{-1}$, so Bob recovers the first bit by applying H and measuring it.

5 BQP vs. BPP

Classically, parallelism can be used for faster computation, but one needs exponentially many parallel processes for exponential speedup. In quantum systems, the amount of parallel computation increases exponentially with the size of the system.

The catch is that access to the results is restricted. "we can only read the result of one parallel thread, and because measurement is probabilistic, we cannot even choose which one we get."

Quantum computation is readily analogous to BPP. In BPP, we have exponentially many computation paths, and choosing any one of them randomly is likely to give the correct answer. *This is physically realizable with a quantum computer*:

1. Encode the Turing Machine $M(x) \to \{ 0, 1 \}$ as a quantum circuit with input $|x0^{p(|x|)}\rangle$
2. The end padding represents the "random seed," and behavior of the circuit is deterministic based on the seed.
3. Using linearly many Hadamard transformations, map $|x0^{p(|x|)}\rangle$ to a uniform superposition over all random seeds.
4. Run the quantum circuit (polynomial time) and choose an output at random. By the BPP assumption, this output is likely to be correct.

Definition 5.1 (BQP)**.** A language L is in BQP iff there exists a polynomial-time uniform family of quantum circuits $\{ Q_n : n \in \mathbb{N} \}$ such that

1. For all $n \in \mathbb{N}$, Q_n takes n qubits as input and outputs 1 bit
2. For all $x \in L$, $Pr(Q_{|x|}(x) = 1) \geq 2/3$
3. For all $x \notin L$, $Pr(Q_{|x|}(x) = 0) \geq 2/3$

Circuit uniformity requires that the descriptions of each circuit be computable by a Turing machine of some restricted time/space. A polynomial-time uniform circuit family is computable by a polynomial time Turing machine.

6 Quantum Algorithms: Around Shor

In this section we present central ideas employed by the Shor's algorithm for factoring integers, and a couple of designs that use those ideas to solve less exciting but in some sense similar problems.

We start with giving two observations that are applied in various approaches to the problem of factorization. First, full factorization of a positive integer N can be efficiently reduced to finding a factor of N because there are at most $\log N$ such factors. Second, less trivial and the most interesting for us, finding a factor of N, by the virtue of number theory, can be reduced to finding the *order* of a random number A, e.g., such $r < N$ that $A^r \equiv 1 \pmod{N}$.

There is one particular property of the mapping

$$b \mapsto A^b \pmod{N}, \qquad b \in \mathbb{Z},$$

that makes it especially interesting target for a quantum computing design. Namely, this function is periodic (this is obvious due to finiteness of the range). The central task when factoring N is to (efficiently) find the period r of this function.

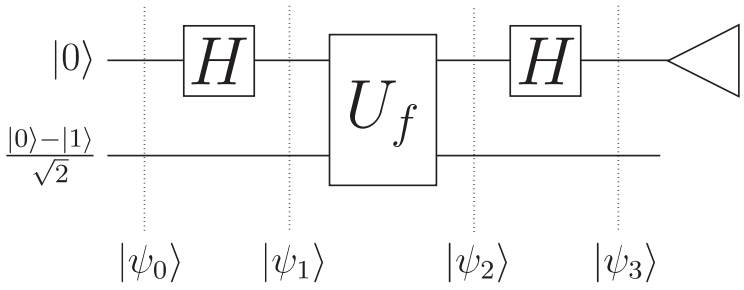

Figure 3: A circuit implementing the Deutsch algorithm.

Let us step back and think what is necessary to study periodic functions in general. First and foremost, the domain of the function should have the structure of a group. A function $f\colon G \to X$ is periodic with period $r \in G$ ($r \neq e$) if:

$$\forall n \in \mathbb{Z}\, \forall g \in G\colon f(g + rn) = f(g).$$

The group we are dealing with in the factoring problem is $\mathbb{Z}_{\varphi(N)}$. Let us consider the study of periodic functions in simpler groups: $\mathbb{Z}_2$ (Deutsch problem) and $\mathbb{Z}_2^n$ (Simon's problem).

6.1 Deutsch Problem

If a function is defined on the domain $\mathbb{Z}_2$, there is only one possibility for period: $r = 1$, and the notion of periodic function here coincides with that of constant function. We assume that the range of functions of interest is also $\mathbb{Z}_2$ for convenience. Then all functions $f\colon \mathbb{Z}_2 \to \mathbb{Z}_2$ can be divided into two groups: constant functions ($f(0) \oplus f(1) = 0$) and so called balanced functions ($f(0) \oplus f(1) = 1$).

Deutsch problem asks what it takes for a given black-box function f to tell if it is constant or balanced? Classically, this requires two calls to f. Deutsch algorithm solves this problem with just one call to (a quantum realization of) f.

Deutsch algorithm is provided by the circuit on Fig. 3. Here are all the intermediate states marked on the figure with normalizing multipliers omitted for brevity:

$$\begin{aligned}
|\psi_0\rangle &= |0\rangle\,(|0\rangle - |1\rangle),\\
|\psi_1\rangle &= \sum_{x\in\mathbb{Z}_2} |x\rangle\,(|0\rangle - |1\rangle),\\
|\psi_2\rangle &= \sum_{x\in\mathbb{Z}_2} (-1)^{f(x)}\,|x\rangle\,(\ldots), \qquad \text{// qubit 2 don't matter anymore}\\
&= (-1)^{f(0)}(|0\rangle + (-1)^{f(0)\oplus f(1)}\,|1\rangle)(\ldots),
\end{aligned}$$

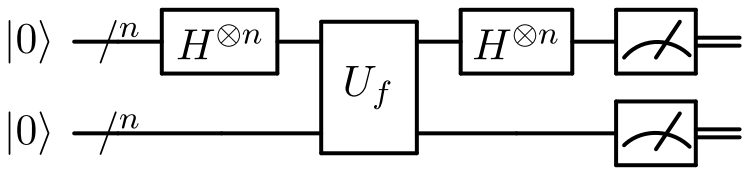

Figure 4: A circuit implementing Simon's algorithm.

$$|\psi_3\rangle = \begin{cases} |0\rangle\,(\ldots), & \text{if } f \text{ is constant,} \\ |1\rangle\,(\ldots), & \text{if } f \text{ is balanced.} \end{cases}$$

Therefore, we measure 0 on the first wire if f is constant or 1 if it is balanced.

6.2 Simon's Problem

Simon's problem studies the period of a function defined on $\mathbb{Z}_2^n$. In particular, assume a black-box for computing an unknown function $f\colon \mathbb{Z}_2^n \to X$ with the property that there exists $\bar{s}$, s.t.: $f(\bar{x}) = f(\bar{y})$ iff $\bar{x} = \bar{y}$ or $\bar{x} = \bar{y} \oplus \bar{s}$. From this, determine $\bar{s}$ by making queries to f.

There is no classical algorithm asymptotically better then brute force. This requires exponentially many queries to f. A quantum solution needs only polynomially many steps and gates. The circuit from Fig. 4 shows the main part of the algorithm. Here is the sequence of steps performed by the circuit:

$$\begin{aligned} |0^{2n}\rangle &\overset{(1)}{\longmapsto} \text{// first Hadamard} \\ \sum_{\bar{x}\in\mathbb{Z}_2^n} |\bar{x}\,0^n\rangle &\overset{(2)}{\longmapsto} \text{// } U_f \\ \sum_{\bar{x}\in\mathbb{Z}_2^n} |\bar{x}\rangle\,|f(\bar{x})\rangle &\overset{(3)}{\longmapsto} \text{// measure last } n \text{ qbits} \\ (|\bar{x}\rangle + |\bar{x}\oplus\bar{s}\rangle)\,|f(\bar{x})\rangle &\overset{(4)}{\longmapsto} \text{// second Hadamard} \ldots \end{aligned}$$

To figure out how the second Hadamard acts, we use the following fact.

Exercise.

$$H^{\oplus n}(|\bar{x}\rangle + |\bar{x}\oplus\bar{s}\rangle) = \sum_{\bar{z}\in\bar{s}^\perp} (-1)^{\bar{x}\cdot\bar{z}}\,|\bar{z}\rangle\,.$$

Therefore, the last step of the circuit will produce a random bit-string that is orthogonal to $\bar{s}$. Several applications (linear in n) of the circuit will allow us to form a non-degenerate system of linear equations that we are able to solve on the classical computer efficiently.

References

[1] Sanjeev Arora and Boaz Barak. *Computational Complexity - A Modern Approach*. 2009.

[2] Phillip Kaye, Raymond Laflamme, and Michele Mosca. *An Introduction to Quantum Computing*. 2007.

[3] Eleanor G. Rieffel and Wolfgang Polak. *An Introduction to Quantum Computing for Non-Physicists*. 1998. arXiv: `quant-ph/9809016 [quant-ph]`.