Quantum Information & Computation
ISSN 1533-7146 Volume 25 (2025) pp. 773–783
©Cerebration Science Publishing

Tutorial

Quantum Information and Computing for Beginners: Basics of Qubit Transformations and Quantum Algorithms

Archit Dhingra*

FinEstBeAMS, MAX IV Laboratory, Fotongatan 2, 22484 Lund, Sweden
* Corresponding author: archit.dhingra@maxiv.lu.se

Received date: 29 August 2025; Accepted date: 31 October 2025; Published online: 8 December 2025

Abstract: Quantum information and quantum computing are based on the concept and manipulation of qubits. In this pedagogical Tutorial–written with the aim of assisting self-motivated undergraduate mathematics, physics, and engineering students–transformation of these qubits, especially the Hadamard transformation (H) and the phase shifter transformation (Φ), is illustrated. Two-qubit gates and their application in production of entangled states, along with quantum algorithms (including Deutsch–Jozsa algorithm), are included as well. With 2025 being declared as the International Year of Quantum Science and Technology (IYQ) by the United Nations General Assembly (UNGA), under the leadership of United Nations Educational, Scientific and Cultural Organization (UNESCO), the timeliness and relevance of this Tutorial, in order to nudge the budding researchers in the "quantum" direction, cannot be emphasized enough.

Keywords: Hilbert space; Bloch sphere; quantum entanglement; quantum algorithm; quantum parallelism; succinct quantum computing tutorial for university students

1. Introduction

1.1. Basics of Quantum Information

Quantum information is the information of the state of a quantum system. Here, a state refers to a physical state of the system. State is a set of variables describing a system, without any reference to its history. Simply put, a quantum state refers to any element $|\psi\rangle \in H$ of Hilbert space. Technically, a quantum state provides a probability distribution for the value of each observable (a measurable physical quantity). Therefore, knowledge of the quantum state, along with the interpretation, and applications of rules for the system's evolution in time can help us predict the system's behavior.

The fundamental concept of classical information (or computation) is a *bit*. Likewise, the fundamental concept of quantum information is a *quantum bit*, or *qubit* (for short). The difference between the *classical bit* and the *qubit*, is that the former one has two possibilities (or possible states); namely: 0 or 1. However, the latter one can have any state: either $|0\rangle$ or $|1\rangle$ or any linear combination of those two states. This linear combination is written mathematically as

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \tag{1}$$

The special states, $|0\rangle$ and $|1\rangle$, in equation 1 are known as the computational basis states and the linear combination is called the superposition (of the aforementioned basis states). Also, the numbers in Equation (1), α and β, are complex numbers [1]. The state of a qubit, $|\psi\rangle$, given by Equation (1) can be represented as a vector on the *Bloch sphere* (Figure 1). Bloch sphere (named after Felix Bloch [2]) is, hence, a geometrical representation of the pure state (a vector in Hilbert space) space of a qubit.

DOI: 10.2478/qic-2025-0040

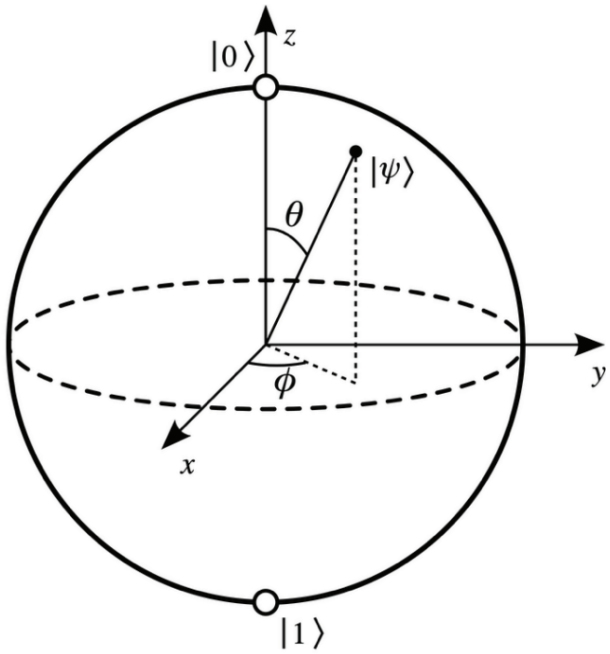

Figure 1. Representation of $|\psi\rangle$ as a vector on the Bloch sphere.

Another difference between the classical bit and the qubit lies in the fact that one can always examine whether the former is in the state 0 or 1, whereas examining a qubit to determine its quantum state cannot be achieved. However, quantum physics dictates that measuring a qubit will result in 0, with probability $|\alpha|^2$, or it will result in 1, with probability $|\beta|^2$. And, of course, the normalization condition for equation 1 indicates

$$|\alpha|^2 + |\beta|^2 = 1 \tag{2}$$

On a more fundamental level, measuring classical information is carried out using Shannon entropy [3] whereas measurement of its quantum counterpart involves application of Von Neumann entropy [4,5]. This difference arises because, of course, classical information follows the classical statistical mechanics while quantum information follows the quantum statistical mechanics model.

Shannon entropy is mathematically described as [6]

$$S = -\sum_i P_i \log P_i \tag{3}$$

here, P_i is the probability mass function.

And, Von Neumann entropy is represented as [7]

$$S = -\mathrm{Tr}(\rho \ln \rho) \tag{4}$$

here, ρ is the density matrix of the statistical ensemble of quantum mechanical systems.

1.2. Quantum Computation

Quantum computation comprises of the techniques and algorithms involved in manipulation (processing) of quantum information. However, there are some theorems that invoke some limitations on the way quantum information (or qubits) can be manipulated [1]. These theorems are:

- No-teleportation theorem: states that a qubit cannot be converted into classical bits, which implies that cannot be read [8,9].
- No-cloning theorem: states that an arbitrary bit cannot be copied [8,10].
- No-deleting theorem: states that an arbitrary bit cannot be deleted [11].
- No-broadcasting theorem: states that even though a qubit is allowed to be transported from one place to another, it cannot be delivered to multiple recipients (simultaneously) [12].

Abiding by the limitations invoked by the above-mentioned theorems, we can manipulate quantum information by performing quantum Fourier transform (QFT) [13]. Here it is worth mentioning that the prospective reader is only expected to understand that QFT is a linear transformation on qubits and is analogous to, its classical counterpart, inverse discrete Fourier transform (IDFT). As far as the message of this tutorial is concerned, understanding of the proper mathematical formalism of QFT is not as crucial as it is to know its function, i.e.,: implementing QFT to qubits is how one gets basic quantum circuits operating on qubits – called quantum logic gates – to perform manipulation of information.

1.2.1. Single-Qubit Transformations

Hadamard gate, Pauli-X gate, Pauli-Y gate, Pauli-Z gate, Square root of NOT gate, Phase shift gate, etc. are some of the examples of single-qubit gates which can be used to perform transformations. The two single-qubit transformations (or gates) that deserve special attention, as far as this paper is concerned, are the Hadamard transformation (H) and the phase shifter transformation (Φ).

Hadamard transformation (H) [1,14]: represented as H gate (see Figure 2), is a single-qubit rotation that maps the basis states $|0\rangle$ and $|1\rangle$ to two superposition states with equal weights.

In Dirac notation, H is represented as

$$H = \frac{|0\rangle + |1\rangle}{\sqrt{2}}\langle 0| + \frac{|0\rangle - |1\rangle}{\sqrt{2}}\langle 1| \tag{5}$$

Above equation for H (Equation (5)) gives us the transformation matrix H' (i.e., just the 2×2 matrix operator without the computational basis states $|0\rangle$ and $|1\rangle$) corresponding to the Hadamard gate H (which performs the single-qubit Hadamard transformation on the computational basis states $|0\rangle$ and $|1\rangle$) as [15]

$$H' = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{6}$$

Coefficients of $\langle 0|$ and $\langle 1|$ in Equation (5), together, form the polar basis in the realm of quantum computation. Coefficient of $\langle 0|$, $\frac{|0\rangle+|1\rangle}{\sqrt{2}}$, is represented as $|+\rangle$ whereas the coefficient of $\langle 1|$, $\frac{|0\rangle-|1\rangle}{\sqrt{2}}$, is represented as $|-\rangle$.

Figure 2. Pictorial representation of Hadamard gate in a circuit.

Phase shifter transformation (Φ) [14]: represented by Φ gate (see Figure 3), is a single-qubit transformation that modifies the phase of the qubit but not the probability of measuring its basis states ($|0\rangle$ and $|1\rangle$).

In Dirac notation, Φ can be represented as

$$\Phi = |0\rangle\langle 0| + e^{i\phi}|1\rangle\langle 1| \tag{7}$$

While its corresponding transformation matrix Φ' is given as

$$\Phi' = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{bmatrix} \tag{8}$$

From above Equations (7) and (8), it is easy to see how Φ gate only shifts the phase of $|1\rangle$ state by ϕ as it is changed to $e^{i\phi}|1\rangle$. However, $|0\rangle$ is left unaltered.

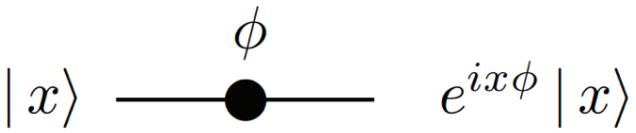

Figure 3. Pictorial representation of the action of Φ gate on a qubit (or a quantum state). Here, in accordance with Equations (7) and (8), $x = 0$ or 1.

Applications of, both, the H gate and the Φ gate can be seen when their corresponding transformations are applied to the problem at hand.

Problem:

Assume that the input state is a general qubit

$$|\mathcal{Q}\rangle = \alpha|0\rangle + \beta|1\rangle \tag{9}$$

Find the state of the output qubit after the transformation $H\Phi H$.

Solution:

There are two ways to solve the given problem: either by using Dirac notation (Equations (5) and (7)) for the H gate and the Φ gate or by using the transformation matrices (Equations (6) and (8)) corresponding to the respective gates.

$H\Phi H$ applied to Equation (9), gives

$$H(\Phi(H|\mathcal{Q}\rangle)) = H(\Phi(H(\alpha|0\rangle + \beta|1\rangle))) \tag{10}$$

Now by using Equations (5) and (9) and the fact that basis states are orthonormal (i.e.,$\langle i|j\rangle = \delta_{ij}$, where $i, j \in \{0,1\}$), we get

$$\begin{aligned} H|\mathcal{Q}\rangle &= \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\langle 0| + \frac{|0\rangle - |1\rangle}{\sqrt{2}}\langle 1|\right)(\alpha|0\rangle + \beta|1\rangle) \\ \Rightarrow H|\mathcal{Q}\rangle &= \frac{|0\rangle + |1\rangle}{\sqrt{2}}\alpha + \frac{|0\rangle - |1\rangle}{\sqrt{2}}\beta \end{aligned} \tag{11}$$

Now, by using Equations (7) and (11), we get

$$\begin{aligned} \Phi(H|\mathcal{Q}\rangle) &= (|0\rangle\langle 0| + e^{i\Phi}|1\rangle\langle 1|)\left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\alpha + \frac{|0\rangle - |1\rangle}{\sqrt{2}}\beta\right) \\ \Rightarrow \Phi(H|\mathcal{Q}\rangle) &= \frac{(\alpha+\beta)}{\sqrt{2}}|0\rangle + \frac{(\alpha-\beta)e^{i\Phi}}{\sqrt{2}}|1\rangle \end{aligned} \tag{12}$$

Finally, by using equations (5) and (12), we get

$$\begin{aligned} H(\Phi(H|\mathcal{Q}\rangle)) &= \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}}\langle 0| + \frac{|0\rangle - |1\rangle}{\sqrt{2}}\langle 1|\right)\left(\frac{(\alpha+\beta)}{\sqrt{2}}|0\rangle + \frac{(\alpha-\beta)e^{i\Phi}}{\sqrt{2}}|1\rangle\right) \\ \Rightarrow H(\Phi(H|\mathcal{Q}\rangle)) &= \left(\frac{(\alpha+\beta)}{2} + \frac{(\alpha-\beta)e^{i\Phi}}{2}\right)|0\rangle + \left(\frac{(\alpha+\beta)}{2} - \frac{(\alpha-\beta)e^{i\Phi}}{2}\right)|1\rangle \end{aligned} \tag{13}$$

Equation (13) is, therefore, the solution to our problem.

1.2.2. Two-Qubit Gates and Entangled States

Building up on our knowledge about the fundamentals and functionality of single-qubit transformations/gates, we can now proceed to multiple qubit gates. The scope of this article, however, is limited to two-qubit gates.

Swap (*SWAP*) gate, Square root of Swap gate ($\sqrt{SWAP}$), Controlled (*CNOT*) gate, etc. are some examples of two-qubit gates with *CNOT* being the most important among them all [16,17].

SWAP gate: it swaps two qubits. Its matrix (with respect to the basis $|00\rangle$, $|01\rangle$, $|10\rangle$, $|11\rangle$) is represented as:

$$SWAP = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$\sqrt{SWAP}$ gate: it performs a "half-way" two-qubit swap, and its matrix can be represented as:

$$\sqrt{SWAP} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{2}(1+i) & \frac{1}{2}(1-i) & 0 \\ 0 & \frac{1}{2}(1-i) & \frac{1}{2}(1+i) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Here, the Dirac notation of the SWAP and $\sqrt{\text{SWAP}}$ is not shown, but rather left as a trivial exercise for the readers (hint: since the transformation matrix for these gates are given in Figure 4 and Figure 5, and the basis vectors are provided as $|00\rangle$, $|01\rangle$, $|10\rangle$, $|11\rangle$, their corresponding Dirac notations can be derived by working along the lines of what is shown in Equations (7) and (8).

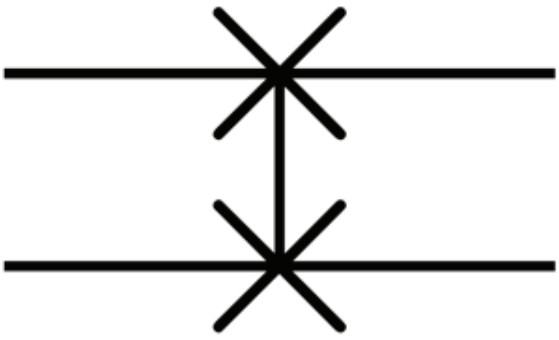

Figure 4. Pictorial representation of *SWAP* gate in a circuit.

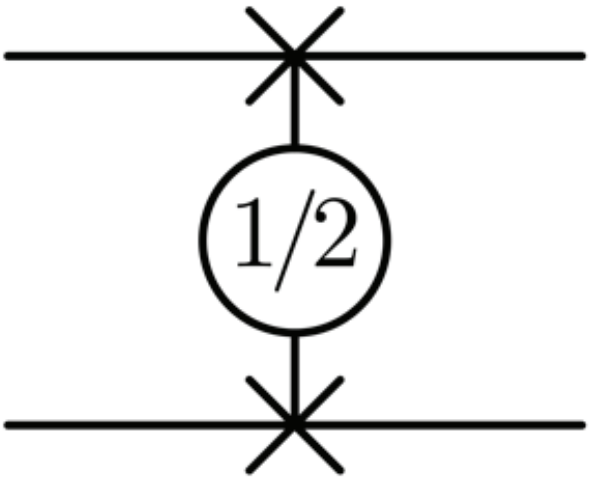

Figure 5. Pictorial representation of $\sqrt{SWAP}$ gate in a circuit.

CNOT gate: has two input qubits with one of them acting as the "control" qubit, while the other qubit is the "target" qubit. If the control qubit is $|0\rangle$, then the target qubit is unchanged. But, if the control qubit is $|1\rangle$ then the target is flipped [1,16,17]. Its operation can be summarized as,

$$|\mathrm{A},\mathrm{B}\rangle \rightarrow |\mathrm{A},\mathrm{B}\oplus\mathrm{A}\rangle \tag{14}$$

What makes controlled-NOT gate the most important amongst all the multiple qubit gates (some pertinent examples are represented in the left panel of Figure 6) is the *universality* result, which states: Any multiple qubit logic gate may be composed from *CNOT*(right panel of Figure 6) and single qubit gates [1].

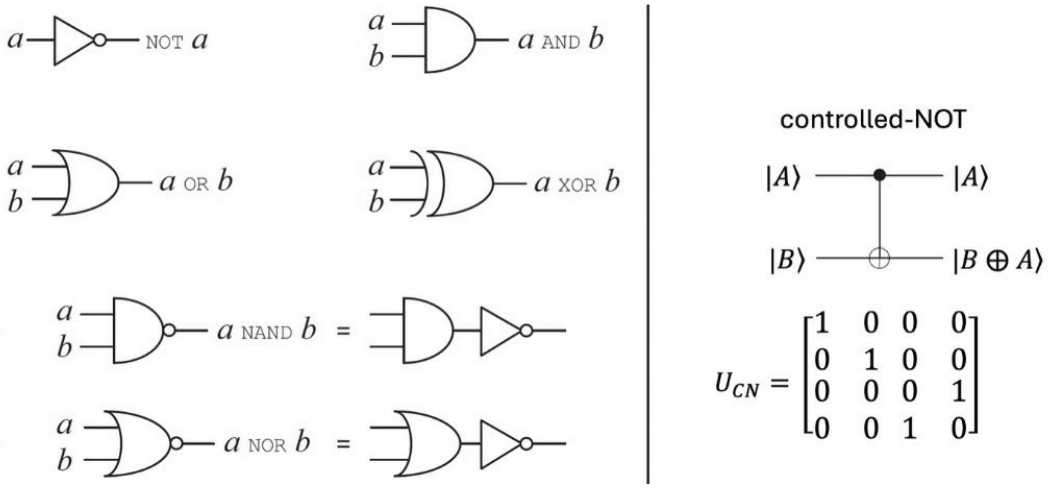

Figure 6. Some standard single and multiple bit gates (**left**), and the controlled-NOT (*CNOT*) gate along with its matrix representation (**right**). Adapted in part from ref. [1].

Application of these gates lies in the production (or creation) of Bell (or entangled[1]) states [18]–[21], which are defined as

$$|\Phi^{+}\rangle = \frac{1}{\sqrt{2}}(|0\rangle_A \otimes |0\rangle_B + |1\rangle_A \otimes |1\rangle_B) \tag{B1}$$

$$|\Phi^{-}\rangle = \frac{1}{\sqrt{2}}(|0\rangle_A \otimes |0\rangle_B - |1\rangle_A \otimes |1\rangle_B) \tag{B2}$$

$$|\Psi^{+}\rangle = \frac{1}{\sqrt{2}}(|0\rangle_A \otimes |1\rangle_B + |1\rangle_A \otimes |0\rangle_B) \tag{B3}$$

$$|\Psi^{-}\rangle = \frac{1}{\sqrt{2}}(|0\rangle_A \otimes |1\rangle_B - |1\rangle_A \otimes |0\rangle_B) \tag{B4}$$

There are numerous ways in which quantum circuits can be used to create the entangled Bell states (states described by Equations (B1)–(B4)), and using the combination of a Hadamard gate and a controlled-NOT gate is one of them. The mechanism of this combinational circuit can be understood with the following example:

The combinational quantum circuit (Figure 7) takes $|00\rangle$ as the two-qubit input and produces the first Bell state (Equation (B1)) as the output. Manifestly, H gate acts on $|00\rangle$ and gives $|+\rangle|0\rangle$[2] as its output. H gate's output will now act as the control qubit (input) for the *CNOT* gate, which will invert the target qubit when the control qubit is $|1\rangle$. Therefore, the resultant qubit will be the first Bell state ($|\Phi^{+}\rangle$). In a similar fashion, one can produce the remaining Bell states as well. The aforementioned production of Bell states (given by Equations (B1)–(B4)) can be, mathematically, summarized as

$$|\mathrm{B}(x,y)\rangle = \frac{1}{\sqrt{2}}(|0,y\rangle + (-1)^{x}|1,Y\rangle) \tag{15}$$

[1]An entangled state is an inseparable state, which means it cannot be represented as a product of its constituent states. In other words: an entangled state cannot be represented as a tensor product of its constituent qubits.

[2]The following Dirac notations: $|\mathrm{AB}\rangle$, $|\mathrm{A},\mathrm{B}\rangle$, and $|\mathrm{A}|\mathrm{B}\rangle$, are all equivalent but are used interchangeably to avoid any possible confusion.

Here, $|B(x,y)\rangle$ stands for Bell states and Y is the *negation* of y, where $x, y \epsilon \{0,1\}$.

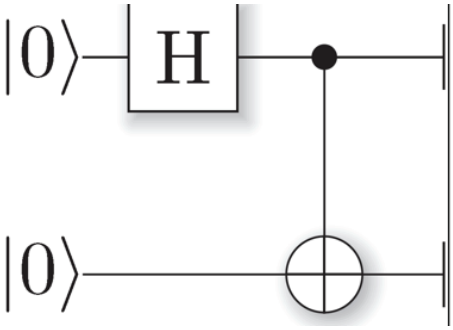

Figure 7. *H–CNOT* combinational quantum circuit to produce $|\Phi^+\rangle$ when the input is $|00\rangle$.

1.2.3. Quantum Algorithms

A quantum algorithm is a finite sequence of well-defined instructions for solving a problem by using quantum circuits for computation [22]. Quantum algorithms are based on two phenomena of quantum computation: *quantum interference and quantum entanglement* [23]. Quantum algorithms can be categorized under three classes. The first is: the class of quantum algorithms bases on quantum Fourier transform. This class includes, *Deutsch-Jozsa* algorithm [24] (its simpler case being *Deutsch's* algorithm [25]), *Bernstein-Vazirani* algorithm [26], *Simon's* algorithm, and *Shor's* algorithms for discrete logarithms and factoring [27], to name a few. Second class consists of quantum search algorithms whose basic principles are given by *Grover's* algorithm [28]. And, third class of quantum algorithms comprises of quantum simulation by a quantum computer to simulate a quantum system.
This paper talks about *Deutsch–Jozsa* algorithm (member of the first class of quantum algorithms) in detail while slight description of *Grover's* algorithm (an example of second class of quantum algorithms) is also included.[3]

1.2.4. Deutsch–Jozsa problem statement and algorithm

Problem statement: We are given a black box quantum computer that implements some function $f : \{0,1\}^n \longrightarrow \{0,1\}$. That is to say, the function takes n-digit binary values as input and gives either 0 or 1 as the output for each value. Moreover, we are promised that $f(x)$ is either *constant* (0 on all outputs or 1 on all outputs) for all values of values of x or *balanced* (equal to 1 on exactly half of the domain and 0 for the other half of the domain). Now, the problem reduces to determining if f is constant or balanced by using the *oracle*.

2. Algorithm

This algorithm combines *quantum parallelism* and *interference* to solve the aforementioned problem.

Here, let's digress for a bit to briefly describe what *quantum parallelism* truly is. It is a fundamental feature of many quantum algorithms [1], as it allows quantum computers to evaluate a function $f(x)$ for several different values of x simultaneously. Its working can be understood by considering a function that has one-qubit domain and range, like the one considered in *Deutsch's* algorithm[4] (presented on page 12, with a sample circuit representation in Figure 9).

Now, coming back to *Deutsch–Jozsa* algorithm:[5] The steps of this algorithm are shown in Figure 8. The input state is given as

$$|\psi_0\rangle = |0\rangle^{\otimes n}|1\rangle \tag{D1}$$

here the query register describes the state of n qubits prepared in state $|0\rangle$, whereas $|1\rangle$ is the answer register. Here, as a general note, it is worth mentioning that, regardless of the algorithm, both query and answer registers are equally important and, therefore, must be read at all times. Hadamard transform (a generalized $2^n \times 2^n$ matrix applied to an

[3] *Deutsch-Jozsa* algorithm and *Deutsch's* algorithm are what we are interested in as far as the scope of this paper lies, and hence description of other algorithms is discarded.

[4] Description of *Deutsch's* algorithm is followed by detailed description of *quantum parallelism*.

[5] Nota bene: despite my attempts to present this algorithm in simpler terms, its description is mathematically heavy. The reason being that some algorithms are purely mathematical and, hence, cannot be simplified any further without compromising the quality of the article.

n-qubit register) is applied to the query register, while Hadamard gate (which performs the single-qubit Hadamard transformation on the computational basis states $|0\rangle$ and $|1\rangle$) is applied to the answer register to produce

$$|\psi_1\rangle = \sum_{x\epsilon\{0,1\}^n} \frac{|x\rangle}{\sqrt{2^n}} \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right] \tag{D2}$$

Equation (D2) leaves us with query register in a superposition of all values, while the answer register is now an evenly weighted superposition of 0 and 1. Now, the function f is evaluated using $U_f : |x,y\rangle \longrightarrow |x, y \bigoplus f(x)\rangle$, resulting in

$$|\psi_2\rangle = \sum_x \frac{(-1)^{f(x)}|x\rangle}{\sqrt{2^n}} \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right] \tag{D3}$$

We now have a set of qubits in which the result of U_f is stored in the amplitude of the qubit superposition state. Now, we have to *interfere* terms in the superposition using Hadamard transformation on the query register. We should, first, calculate $H|x\rangle$. By checking for $x = 0$ and $x = 1$, we see that for a single qubit

$$H|x\rangle = \sum_z (-1)^{xz}|z\rangle/\sqrt{2} \tag{D4}$$

Therefore, we have

$$H^{\otimes n}|x_1, \ldots, x_n\rangle = \sum_{z_1,z_2,z_3,\ldots,z_n} (-1)^{x_1z_1+..+x_nz_n}|z_1,\ldots,z_n\rangle/\sqrt{2^n} \tag{D5}$$

Equation (D5) can be rewritten as

$$H^{\otimes n}|x\rangle = \sum_z (-1)^{x\cdot z}|z\rangle/\sqrt{2^n} \tag{D6}$$

here, $x \cdot z$ is bitwise inner product of x and z, modulo 2, which is (explicitly) written as

$$x \cdot z = x_1z_1 \oplus x_2z_2 \oplus x_3z_3 \oplus \cdots \oplus x_nz_n \tag{D7}$$

Now, $|\psi_3\rangle$ can be calculated using (D2) and (D6) and is given as

$$|\psi_3\rangle = \sum_z \sum_x \frac{(-1)^{x\cdot z+f(x)}|z\rangle}{2^n} \left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right] \tag{D8}$$

We can now observe the query register and realize that there are two possible cases – f is constant, and f is balanced – to see what happens. For the former case, the amplitude for $|0\rangle^{\otimes n}$ is $+1$ or -1, depending on the constant value that $f(x)$ bears. Given the fact that $|\psi_3\rangle$ is of unit length, it's trivial to conclude that all other amplitudes must be zero, and a measurement will lead 0s for all qubits in the query register. Now, for the latter case (where f is balanced) the negative and positive contributions to the amplitude for $|0\rangle^{\otimes n}$ cancel each other thus leaving a zero amplitude. This would imply that a measurement should give a result different from 0 on at least one of the qubits in the query register. In other words: if measurements result in all 0s then f is constant, and balanced otherwise.

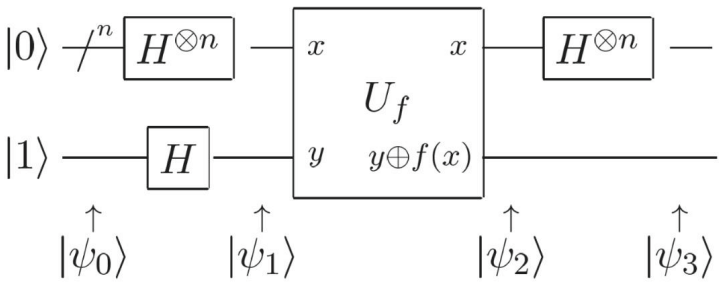

Figure 8. Representation of *Deutsch–Jozsa* algorithm as a quantum circuit, where '/^{n}' represents a set of n qubits. Adapted in part from ref. [1].

2.1. Deutsch's algorithm

It combines quantum parallelism and interference, just like Deutsch – Jozsa does, and is actually a simpler case of (more general) Deutsch – Jozsa algorithm. The input state for Deutsch algorithm is given as

$$|\psi_0\rangle = |01\rangle \tag{D1′}$$

which, on passing through two H gates, gives

$$|\psi_1\rangle = \left[\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right]\left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right] \tag{D2′}$$

Now, applying U_f to a state $|x\rangle\left[\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right]$ gives us the state $(-1)^{f(x)}|x\rangle\left[\frac{|0\rangle-|1\rangle}{\sqrt{2}}\right]$. Therefore, $U_f|\psi_1\rangle$ gives us

$$|\psi_2\rangle = \pm\left[\frac{|0\rangle + |1\rangle}{\sqrt{2}}\right]\left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right] \quad \text{if } f(0) = f(1) \tag{D3′a}$$

$$|\psi_2\rangle = \pm\left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right]\left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right] \quad \text{if } f(0) \neq f(1) \tag{D3′b}$$

Now, finally applying H gate on $|\psi_2\rangle$, we get

$$|\psi_3\rangle = \pm|0\rangle\left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right] \quad \text{if } f(0) = f(1) \tag{D4′a}$$

$$|\psi_3\rangle = \pm|1\rangle\left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right] \quad \text{if } f(0) \neq f(1) \tag{D4′b}$$

Equations (D4′a) and (D4′b) can be combined and written as

$$|\psi_3\rangle = \pm|f(0) \oplus f(1)\rangle\left[\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right] \tag{D5′}$$

Therefore, by measuring the first qubit, we can determine the global property of $f(x)$, i.e., $f(0) \oplus f(1)$, by using a single evaluation of $f(x)$. This is quite interesting because its classical counterpart would require at least two evaluations. Thus, making it faster than its classical counterpart.

Now, let's have a transformation, U_f, acting on a state $|x,y\rangle$ such that it takes $|x,y\rangle \rightarrow |x, y \oplus f(x)\rangle$. Here, the first register is called the *data* register while the second register is called the *target* register. If we have $y = 0$, then it is trivially concluded that the final state of the second qubit will be given by the value of the function $f(x)$. Applying U_f to a superposition state $|x\rangle = \frac{|0\rangle+|1\rangle}{\sqrt{2}}$, while keeping $|y\rangle = |0\rangle$, we get

$$U_f|x,0\rangle = |\psi\rangle = \frac{|0, f(0)\rangle + |1, f(1)\rangle}{\sqrt{2}} \tag{16}$$

Here, $|x\rangle = \frac{|0\rangle+|1\rangle}{\sqrt{2}}$ can be produced by applying H gate to $|0\rangle$. Figure 9 (below) summarizes the discussion about quantum parallelism.

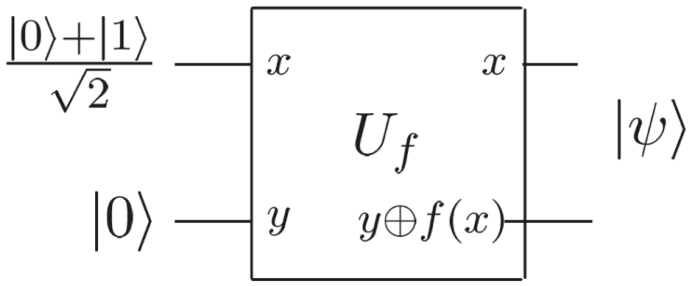

Figure 9. A sample circuit representing quantum parallelism, where states on the left represent the input and $|\psi\rangle$ is given by Equation (16).[1]

2.2. Deutsch's Algorithm V/s Classical Algorithm

Comparison between Deutsch's algorithm with a classical algorithm from the standpoint of number of steps involved can be carried out as follows:

Deutsch–Jozsa (or Deutsch's) algorithm for n qubits: Recalling that for f to be balanced we must have at least one of the measurements resulting in a non-zero value. Otherwise, f will be constant. Moreover, $x \cdot z = x_1z_1 \oplus x_2z_2 \oplus x_3z_3 \oplus \cdots \oplus x_nz_n$ (Equation (D7)) denotes that there will be at least n steps involved in solving the problem using this algorithm as one needs to evaluate all x_iz_i for $i \in \{1,2,3,4,\ldots,n\}$. That is to say, Deutsch's algorithm requires $O(n)$ steps to determine f.

Classical algorithm: Exhaustive search is the fastest known classical algorithm [29] we have at our disposal for comparing the efficiency of classical algorithms with Deutsch's algorithm. Following exhaustive search algorithm, one will have to carry out (at most) '$2^{n-1}+1$' measurements to determine the nature of f. This is because for every bit we'll have two possibilities, thus there are 2^n possibilities for n bits. Now, the worst-case scenario would be that the first half of the measurements (i.e., 2^{n-1} measurements) are all the same, which means result of the following $((2^{n-1}+1)^{\text{th}})$ measurement will be mandatory to determine nature of f. Therefore, even the most efficient classical algorithm requires $O(2^n)$ steps to achieve the same task as performed by Deutsch's algorithm.

2.3. Grover's Algorithm

It is a quantum search algorithm that solves the problem of finding an element satisfying a known property in a search size of N, with no prior knowledge about the structure of information in the given search space. Classical solution to this problem would require $O(N)$ steps, whereas *Grover's* algorithm requires $O(\sqrt{N})$ steps [1].

3. Conclusions

Quantum information and quantum computing are based on the concept and manipulation of qubits, as opposed to classical bits. QFT based quantum circuits (quantum logic gates) are used to manipulate quantum information. An input state $|\mathcal{Q}\rangle = \alpha|0\rangle + \beta|1\rangle$ going through a Hadamard-Phase shifter-Hadamard transformation ($H\Phi H$) is examined, and the result of $H\Phi H|\mathcal{Q}\rangle$ is given by equation 13. Two-qubit gates can be used (along with single qubit gates) for production of Bell (entangled) states. For example, in this paper, a combination of *C-NOT* gate and H gate is used to produce all four Bell states (described by Equations (B1)–(B4)).

Moreover, a direct comparison between Deutsch–Jozsa (or Deutsch's) algorithm and the fastest available classical algorithm (exhaustive search algorithm) concludes that the former is way more efficient in comparison with the latter one. Quantitatively speaking, (for n-qubits) Deutsch's algorithm requires $O(n)$ steps whereas, on the other hand, any classical algorithm requires $O(2^n)$ steps (for n-bits) to perform the same task.

Finally, I would like to add that while there is a myriad of existing literature on the topics contained herein, what makes this tutorial stand out is its concise-yet-comprehensive nature in comparison with other great pieces of work in this field [30–33]. In other words, this tutorial is ideal for the readers who would like to take the first step towards understanding the marvels of quantum physics, applied to quantum information and computing, but do not have enough hours in a day to dedicate to the cause.

Funding
This work received no external funding.

Conflicts of Interest Statement
The author declares no conflict of interest.

Data Availability Statement
No new data were generated or analyzed in this tutorial.

References

1. M. A. Nielsen and I. L. Chuang (2010). *Quantum Computation and Quantum Information*, Cambridge University Press, Cambridge.
2. F. Bloch (1951). *Physica*, 17, 272–281.
3. C. E. Shannon (1948). *Bell System Technical Journal*, 27, 379–423.
4. I. Bengtsson and K. Zyczkowski (2017). *Geometry of Quantum States*, Cambridge University Press, Cambridge.
5. J. Von Neumann (1996). *Mathematical Foundations of Quantum Mechanics*, Princeton University Press.

6. R. K. Pathria and P. D. Beale (2011). In *Statistical Mechanics (Third Edition)*, eds. R. K. Pathria and P. D. Beale, Academic Press, Boston, Third Edit., pp. 39–90.
7. I. Bengtsson and K. Zyczkowski (2017). *Geometry of Quantum States*, Cambridge University Press, Cambridge.
8. C. H. Bennett, G. Brassard, C. Crépeau, R. Jozsa, A. Peres and W. K. Wootters (1993). *Physical Review Letters*, **70**, 1895–1899.
9. D. Bouwmeester, J.-W. Pan, K. Mattle, M. Eibl, H. Weinfurter and A. Zeilinger (1997). *Nature*, 390, 575–579.
10. F. Grosshans and P. Grangier (2001). *Physical Review A - Atomic, Molecular, and Optical Physics*, 64, 4.
11. A. K. Pati and S. L. Braunstein (2000). *Nature*, 404, 164–165.
12. A. Kalev and I. Hen (2008). *Physical Review Letters*, 100, 210502.
13. D. Coppersmith, *arXiv* (2002). DOI: https://doi.org/10.48550/arXiv.quant-ph/0201067.
14. A. Ekert, P. M. Hayden and H. Inamori (2007). In *Coherent atomic matter waves*, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 661–701.
15. J. Wen, S. Cong and X. Zou (2012). *Proceedings of the World Congress on Intelligent Control and Automation (WCICA)*, 5096–5101.
16. D. Deutsch, A. Barenco and A. Ekert (1995). *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 449, 669–677.
17. A. Barenco, C. H. Bennett, R. Cleve, D. P. Divincenzo, N. Margolus, P. Shor et al. (1995). *Physical Review A*, 52, 3457–3467.
18. S. L. Braunstein, A. Mann and M. Revzen (1992). *Physical Review Letters*, 68, 3259–3261.
19. C. H. Bennett, H. J. Bernstein, S. Popescu and B. Schumacher (1996). *Physical Review A*, **53**, 2046–2052.
20. H. J. Briegel and R. Raussendorf (2001). *Physical Review Letters*, 86, 910–913.
21. D. Sych and G. Leuchs (2009). *New Journal of Physics*, 11, 013006.
22. M. Mosca (2012). *Computational Complexity: Theory, Techniques, and Applications*, 9781461418, 2303–2333.
23. R. Cleve, A. Ekert, C. Macchiavello and M. Mosca (1998). *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 454, 339–354.
24. D. Deutsch and R. Jozsa (1992). *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 439, 553–558.
25. D. Deutsch (1985). *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 400, 97–117.
26. E. Bernstein and U. Vazirani (1997). *SIAM Journal on Computing*, 26, 1411–1473.
27. P. W. Shor (2002). In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, IEEE Comput. Soc. Press, pp. 124–134.
28. L. K. Grover (1997). *Physical Review Letters*, 79, 325–328.
29. E. Grumbling and M. Horowitz (2019). Eds., *Quantum Computing*, National Academies Press, Washington, D.C.
30. R. Müller and F. Greinert (2023). *Quantum Technologies: For Engineers*, 1–216.
31. C. Bernhardt (2019). *Quantum Computing for Everyone*, The MIT Press.
32. C. Hughes, J. Isaacson, A. Perry, R. F. Sun and J. Turner. *Quantum Computing for the Quantum Curious*, DOI:https://doi.org/10.1007/978-3-030-61601-4.
33. J. Bley, E. Rexigel, A. Arias, N. Longen, L. Krupp, M. Kiefer-Emmanouilidis et al. (2024). *Phys Rev Res*, 6, 023077.