A Note On Deutsch-Jozsa Algorithm

Zhengjun Cao[1], Jeffrey Uhlmann[2], Lihua Liu[3,*]

Abstract. We remark that Deutsch-Jozsa algorithm has confused two unitary transformations: one is performed on a pure state, the other is performed on a superposition. In the past decades, no constructive specifications on the essential unitary operator performed on the superposition have been found. We think the Deutsch-Jozsa algorithm needs more constructive specifications so as to check its correctness.

Keywords: quantum computing, Deutsch-Jozsa algorithm, Shor's algorithm, superposition.

1 Introduction

Deutsch-Jozsa algorithm [5] is one of the first examples of a quantum algorithm that is exponentially faster than any possible deterministic classical algorithm. The algorithm has become the cornerstone for quantum computation and inspired Grover's algorithm [7] and Shor's algorithm [13]. In this note, we want to point out that Deutsch-Jozsa algorithm has confused two unitary transformations: one is performed on a pure state, the other is performed on a superposition. So far, no constructive specifications on the essential unitary transformation performed on a superposition have been found. This fact renders the algorithm somewhat dubious.

2 Preliminaries

A qubit is a quantum state $|\Psi\rangle$ of the form $|\Psi\rangle = a|0\rangle + b|1\rangle$, where the amplitudes $a, b \in \mathbb{C}$ such that $|a|^2 + |b|^2 = 1$, $|0\rangle$ and $|1\rangle$ are basis vectors of the Hilbert space. Two quantum mechanical systems are combined using the tensor product. For example, a system of two

[1]Department of Mathematics, Shanghai University, Shanghai, 200444, China.
[2]Department of Computer Science, University of Missouri, Columbia, USA.
[3]Department of Mathematics, Shanghai Maritime University, Shanghai, 201306, China. * liulh@shmtu.edu.cn

qubits $|\Psi\rangle = a_1|0\rangle + a_2|1\rangle$ and $|\Phi\rangle = b_1|0\rangle + b_2|1\rangle$ can be written as

$$|\Psi\rangle|\Phi\rangle = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix} \otimes \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} = \begin{pmatrix} a_1 b_1 \\ a_1 b_2 \\ a_2 b_1 \\ a_2 b_2 \end{pmatrix}$$

Its shorthand notation is $|\Psi, \Phi\rangle$.

Operations on a qubit are described by 2×2 unitary matrices. Of these, the most important is the Hadamard gate $H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$. Clearly, $H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, $H^2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = I_2$.

3 Deutsch-Jozsa algorithm

Let $f : \{0,1\}^n \to \{0,1\}$. The Deutsch-Jozsa algorithm needs a quantum oracle computing $f(x)$ from x which doesn't decohere x. It begins with the $n+1$ bit state $|0\rangle^{\otimes n}|1\rangle$. That is, the first n qubits are each in the state $|0\rangle$ and the final qubit is in the state $|1\rangle$.

A Hadamard gate is applied to each qubit to obtain the following state

$$H^{\otimes(n+1)} : \quad |0\rangle^{\otimes n}|1\rangle \longrightarrow \frac{1}{\sqrt{2^{n+1}}} \sum_{x=0}^{2^n-1} |x\rangle(|0\rangle - |1\rangle). \tag{1}$$

Suppose that the oracle $\mathcal{U}_f : |x\rangle|y\rangle \longrightarrow |x\rangle|y \oplus f(x)\rangle$ is available, where $\oplus$ is addition modulo 2. Applying the quantum oracle, it gives

$$\mathcal{W} : \quad \frac{1}{\sqrt{2^{n+1}}} \sum_{x=0}^{2^n-1} |x\rangle(|0\rangle - |1\rangle) \longrightarrow \frac{1}{\sqrt{2^{n+1}}} \sum_{x=0}^{2^n-1} |x\rangle(|f(x)\rangle - |1 \oplus f(x)\rangle). \tag{2}$$

For each x, $f(x)$ is either 0 or 1. The state can be written as $\frac{1}{\sqrt{2^{n+1}}} \sum_{x=0}^{2^n-1} (-1)^{f(x)}|x\rangle(|0\rangle - |1\rangle)$.

Ignoring the last qubit and applying the Hadamard gate to each of the first n qubits, it gives

$$H^{\otimes n} : \quad \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} (-1)^{f(x)}|x\rangle \longrightarrow \frac{1}{2^n} \sum_{x=0}^{2^n-1} (-1)^{f(x)} \left[\sum_{y=0}^{2^n-1} (-1)^{x \cdot y}|y\rangle \right] \tag{3}$$

where $x \cdot y = x_0 y_0 \oplus x_1 y_1 \oplus \cdots \oplus x_{n-1} y_{n-1}$ is the sum of the bitwise product. The above new superposition can be written as

$$\frac{1}{2^n} \sum_{y=0}^{2^n-1} \left[\sum_{x=0}^{2^n-1} (-1)^{f(x)} (-1)^{x \cdot y} \right] |y\rangle.$$

The probability for measuring the state $|0\rangle^{\otimes n}$ is $|\frac{1}{2^n} \sum_{x=0}^{2^n-1} (-1)^{f(x)}|^2$.

4 Analysis of Deutsch-Jozsa algorithm

The process of Deutsch-Jozsa algorithm can be described as follows

$$
\begin{aligned}
|\underbrace{00\cdots 0}_{n}\rangle|1\rangle \quad & \xrightarrow{H^{\otimes(n+1)}} \frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}|x\rangle(|0\rangle-|1\rangle)\\
& \xrightarrow{\mathcal{W}} \frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}|x\rangle(|f(x)\rangle-|1\oplus f(x)\rangle)\\
& \xrightarrow[\text{and obtaining the state}]{\text{ignoring the last qubit}} \frac{1}{\sqrt{2^n}}\sum_{x=0}^{2^n-1}(-1)^{f(x)}|x\rangle\\
& \xrightarrow{H^{\otimes n}} \frac{1}{2^n}\sum_{x=0}^{2^n-1}(-1)^{f(x)}\left[\sum_{y=0}^{2^n-1}(-1)^{x\cdot y}|y\rangle\right]\\
& \xrightarrow[\text{obtaining its probability}]{\text{observing the state and}} |\underbrace{00\cdots 0}_{n}\rangle.
\end{aligned}
$$

4.1 How to practically construct the oracle performed on a pure state

In Deutsch-Jozsa algorithm, the quantum oracle $\mathcal{U}_f : |x\rangle|y\rangle \longrightarrow |x\rangle|y\oplus f(x)\rangle$ must be of the form

$$\mathcal{U}_f = I_2^{\otimes n}\otimes\mathcal{V}_f,$$

where I_2 is the 2×2 identity matrix and $\mathcal{V}_f$ is a 2×2 unitary matrix.

Suppose that $\mathcal{V}_f = \begin{bmatrix} X_1 & X_2\\ X_3 & X_4\end{bmatrix}$. We have $\mathcal{V}_f|y\rangle = \begin{bmatrix} X_1 & X_2\\ X_3 & X_4\end{bmatrix}|y\rangle = |y\oplus f(x)\rangle$. If $y=0$, then $|0\rangle = \binom{1}{0}$. It gives $\binom{X_1}{X_3} = |f(x)\rangle$. Since $f(x)\in\{0,1\}$, we obtain $X_1, X_3\in\{0,1\}$. If $y=1$, then $|1\rangle = \binom{0}{1}$. It gives $\binom{X_2}{X_4} = |1\oplus f(x)\rangle$. Since $f(x)\in\{0,1\}$, we obtain $X_2, X_4\in\{0,1\}$. Thus, $\mathcal{V}_f$ is in the set

$$\left\{\begin{bmatrix}1&0\\0&1\end{bmatrix},\begin{bmatrix}0&1\\1&0\end{bmatrix},\begin{bmatrix}1&1\\0&1\end{bmatrix},\begin{bmatrix}1&1\\1&0\end{bmatrix},\begin{bmatrix}0&1\\1&1\end{bmatrix},\begin{bmatrix}1&0\\1&1\end{bmatrix}\right\}.$$

Clearly, *to determine $\mathcal{V}_f$, one has to invoke the classical computational result* $f(x)$. That means the unitary matrix $\mathcal{V}_f$ should be further specified as $\mathcal{V}_{f(x)}$. The notation is very useful because it indicates the constructive specification of the involved unitary matrix. So it is better to rewrite the quantum oracle as

$$\mathcal{U}_{f(x)} = I_2^{\otimes n}\otimes\mathcal{V}_{f(x)}.$$

Note that the construction of the oracle depends essentially on the classical computational result $f(x)$. Besides, the oracle is performed on the pure state $|x\rangle|y\rangle$.

4.2 Is it possible to construct the wanted oracle performed on the superposition

The unitary operator $\mathcal{W}$ is performed on the superposition $\frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}|x\rangle(|0\rangle-|1\rangle)$ and keeps the states of the first n qubits. Hence, it can be decomposed as $\mathcal{W}=I_2^{\otimes n}\otimes\Gamma$, where Γ is a 2×2 unitary matrix.

By the description of Deutsch-Jozsa algorithm, we have

$$\mathcal{W}=I_2^{\otimes n}\otimes\Gamma=\mathcal{U}_{f(x)}=I_2^{\otimes n}\otimes\mathcal{V}_{f(x)}.$$

That means one has to extract a classical computational result $f(x)$ from the superposition $\frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}|x\rangle(|0\rangle-|1\rangle)$ in order to construct the operator $\mathcal{W}$ practically. Since x runs through all values $0,1,\cdots,2^n-1$, one has to measure the superposition so as to obtain a value $\hat{x}$.

Once the value $\hat{x}$ is measured, applying $\mathcal{W}=I_2^{\otimes n}\otimes\mathcal{V}_{f(\hat{x})}$ to $\frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}|x\rangle(|0\rangle-|1\rangle)$ will produce one state of the following

$$\frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}|x\rangle\begin{bmatrix}1&0\\0&1\end{bmatrix}(|0\rangle-|1\rangle),\ \text{or}\ \frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}|x\rangle\begin{bmatrix}0&1\\1&0\end{bmatrix}(|0\rangle-|1\rangle),$$

$$\text{or}\ \frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}|x\rangle\begin{bmatrix}1&1\\0&1\end{bmatrix}(|0\rangle-|1\rangle),\ \text{or}\ \frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}|x\rangle\begin{bmatrix}1&1\\1&0\end{bmatrix}(|0\rangle-|1\rangle),$$

$$\text{or}\ \frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}|x\rangle\begin{bmatrix}0&1\\1&1\end{bmatrix}(|0\rangle-|1\rangle),\ \text{or}\ \frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}|x\rangle\begin{bmatrix}1&0\\1&1\end{bmatrix}(|0\rangle-|1\rangle),$$

not the wanted state $\frac{1}{\sqrt{2^{n+1}}}\sum_{x=0}^{2^n-1}|x\rangle(|f(x)\rangle-|1\oplus f(x)\rangle)$.

All in all, Deutsch and Jozsa have confused a quantum oracle performed on a pure state with a quantum oracle performed on a superposition. We now want to ask: "is it possible to construct the wanted oracle performed on the superposition?"

Finally, we would like to stress that only the Hadamard gate H is applied to each of the first n qubits twice. Since $H^2=I_2$, we find Deutsch-Jozsa algorithm always produces

$$|\underbrace{00\cdots0}_{n}\rangle|\chi\rangle$$

where $\chi\in\{0,1\}$. Their claim that the probability for the state $|0\rangle^{\otimes n}$ is $|\frac{1}{2^n}\sum_{x=0}^{2^n-1}(-1)^{f(x)}|^2$, is incorrect.

5 Conclusion

We point out that there are some flaws in Deutsch-Jozsa algorithm. We would like to stress that the construction of a unitary operator performed on a superposition must be compatible with tensor product [2], which describes the combination of two quantum systems. Some physical experiments [4, 8, 10, 11, 12, 14] on Shor's algorithm are criticized for using less qubits in the second register and other deficiencies [1, 3]. So far, those so-called quantum computers, D-wave [6] and IBM [9], have been reported to optimize some combinatoric problems only, not accelerate any numerical computations. We think Deutsch-Jozsa algorithm needs more specifications so as to facilitate the construction of the wanted quantum oracle and check its correctness.

6 Acknowledgements

We thank the National Natural Science Foundation of China (61303200, 61411146001).

References

[1] Z.J. Cao and Z.F. Cao: On Shor's Factoring Algorithm with More Registers and the Problem to Certify Quantum Computers. IACR Cryptology ePrint Archive 2014: 721 (2014)

[2] Z.J. Cao, Z.F. Cao and L.H. Liu: Remarks on Quantum Modular Exponentiation and Some Experimental Demonstrations of Shor's Algorithm. IACR Cryptology ePrint Archive 2014: 828 (2014)

[3] Z.J. Cao, Z.F. Cao and L.H. Liu: Comment on Demonstrations of Shor's Algorithm in the Past Decades. IACR Cryptology ePrint Archive 2015: 1207 (2015)

[4] A. Dang, et al.: Optimising Matrix Product State Simulations of Shor's Algorithm, arXiv:1712.07311v2 (2017)

[5] D. Deutsch and R. Jozsa: Rapid solutions of problems by quantum computation. Proceedings of the Royal Society of London A, 439, 553 (1992)

[6] D-Wave Systems, PDF, 01-2017, `http://www.dwavesys.com/sites/default/files/D-Wave%202000Q%20Tech%20Collateral_0117F.pdf`

[7] L. K. Grover: A fast quantum mechanical algorithm for database search. Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing. pp. 212C219 (1996)

[8] E. Lucero, et al.: Computing prime factors with a Josephson phase qubit quantum processor. Nature Physics 8, 719-723, 2012. arXiv:1202.5707 (2012)

[9] `http://www.research.ibm.com/ibm-q/`

[10] C.Y. Lu, et al.: Demonstration of a Compiled Version of Shor's Quantum Factoring Algorithm Using Photonic Qubits, Physical Review Letters 99 (25): 250504, arXiv:0705.1684 (2007)

[11] B. Lanyon, et al.: Experimental Demonstration of a Compiled Version of Shor's Algorithm with Quantum Entanglement", Physical Review Letters 99 (25): 250505. arXiv:0705.1398 (2007)

[12] E. Martín-López, et al.: Experimental realization of Shor's quantum factoring algorithm using qubit recycling. Nature Photonics. doi:10.1038/nphoton.2012.259 (2012)

[13] P. Shor: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM J. Comput. 26 (5): 1484-1509 (1997)

[14] L. Vandersypen, et al.: Experimental realization of Shor's quantum factoring algorithm using nuclear magnetic resonance, Nature 414 (6866): 883-887, arXiv:quant-ph/0112176 (2001)