SN Computer Science (2020) 1:86
https://doi.org/10.1007/s42979-020-0091-z

ORIGINAL RESEARCH

A Generalized Quantum Algorithm for Assuring Fairness in Random Selection Among 2^N Participants

Ravin Kumar[1]

Published online: 14 March 2020

Abstract
Quantum computing promises to provide a tremendous boost to the computational power of our machines by utilizing superposition and entanglement phenomenon of quantum mechanics. Few quantum algorithms are known which are taking advantage of these quantum phenomena and are providing solutions to problems having significant importance. In this paper, we have proposed a generalized method for designing 2^N qubits circuit such that during measurement only one qubit will be in state-1, while remaining other qubits will hold state-0. Apart from adding fair randomness to the selection process in distributed quantum computing, these generalized quantum circuits can be found very useful in commercial domains requiring transparency and trust in systems requiring fair randomness in decision-making such as in a lottery system. The critical advantage of using our proposed method is that it allows individual's results to be teleported to them, hence making an end-to-end system whose fairness is quantum assured.

Keywords Quantum entanglement · Random selection · Quantum computing · Quantum assurance · Distributed quantum computing

Introduction

Superposition [1] is a phenomenon in quantum mechanics where a qubit can be manipulated such that instead of having one definite state, it is present in all possible states with each having some probability of occurring. It is during the measurement phase that a qubit selects a definite state, while quantum entanglement [2, 3] is a phenomenon by which qubits are manipulated such that their outputs became correlated and cannot be represented as an output of individual qubits. Superposition and quantum entanglement are playing key roles in designing quantum algorithms and are the backbone of quantum teleportation [4]- and quantum cryptography [5]-based systems. Even though these phenomena provide significant advantage to quantum computers, in reality, very few problems [6, 7] have been figured out which can be solved effectively with quantum computing.

A computer can solve only those problems whose solution can be designed using their available gates. Similarly, in quantum computer, there are varieties of quantum gates available with each having its own characteristics and importance in designing the quantum algorithms. In this paper, we are proposing a solution to the problem of assuring fairness in random selection tasks like in lucky draw, and gambling by providing a generalized quantum algorithm.

Quantum Gates

In quantum computers, the basic unit of quantum information is called a 'qubit'. General quantum state of a qubits is represented using its two orthonormal basis states $|0\rangle$ and $|1\rangle$.

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

This article is part of the topical collection "Advances in Computational Intelligence, Paradigms and Applications" guest edited by Young Lee and S. Meenakshi Sundaram.

✉ Ravin Kumar
ravin.kumar.cs.2013@miet.ac.in

[1] Department of Computer Science, Meerut Institute of Engineering and Technology, Meerut 250005, Uttar Pradesh, India

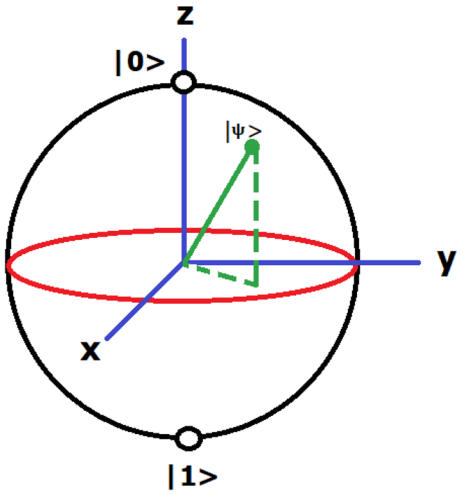

Fig. 1 Bloch sphere representing pure state of a quantum bit

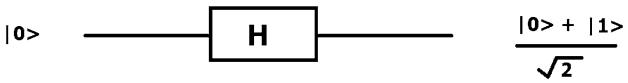

Fig. 2 Hadamard gate applied on $|0\rangle$ input state

$$|1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

A pure qubit is a superposition $|\psi\rangle$ of the basis state, and is described using the linear combination of $|0\rangle$ and $|1\rangle$ states.

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle; \text{such that } |\alpha|^2 + |\beta|^2 = 1$$

Here, $|\alpha|^2$ represents the probability of getting $|0\rangle$ state during measurement of the qubit, and $|\beta|^2$ represents the probability of getting $|1\rangle$ state while measuring the qubit (Fig. 1).

Our proposed algorithm relies on four quantum gates including Hadamard [8], Pauli X [9], controlled Hadamard [10], and controlled NOT [11] gate.

Hadamard Gate

When a qubit present in state $|0\rangle$ or in state $|1\rangle$ is passed through the Hadamard gate, it gets transformed into a superposition of all possible states. In superposition, each state has equal probability of occurring at the time of measurement (Figs. 2, 3). This gate is represented by letter '*H*'.

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

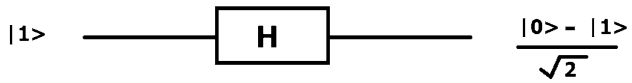

Fig. 3 Hadamard gate applied on $|1\rangle$ input state

Fig. 4 Two Hadamard gates applied on $|0\rangle$ input state

Fig. 5 Two Hadamard gates applied on $|1\rangle$ input state

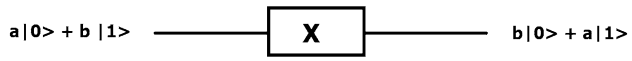

Fig. 6 Pauli *X* gate interchanging the amplitudes of input state

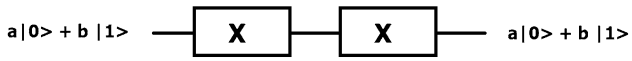

Fig. 7 Two consecutive Pauli *X* gates have no effect on input state

When a qubit with a Hadamard transformation is passed again to the Hadamard gate, it destroys the superposition effect introduced by the first Hadamard gate (Figs. 4, 5).

Pauli *X* Gate

When this quantum gate is applied on a qubit, it interchanges the amplitude of its states (Fig. 6). In other words, it acts similar to the NOT gate of a digital computer. This gate is represented by letter '*X*'.

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Similar to a digital NOT gate, when two consecutive '*X*' gates are applied on input qubit, second '*X*' gate nullifies the effect of first '*X*' gate (Fig. 7).

Controlled Hadamard Gate

This gate consists of two input qubits where the first qubit (i.e. control qubit) decides whether or not to apply Hadamard transformation to the second (i.e. target qubit) qubit. If the control qubit is in state $|1\rangle$, then Hadamard gate is

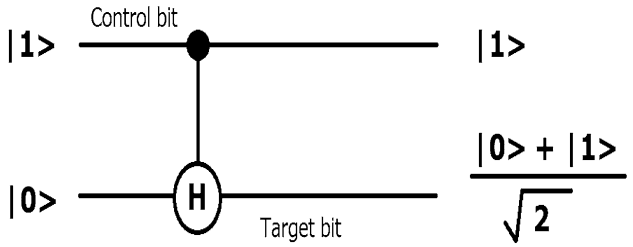

Fig. 8 CH gate with input $|1\rangle$ to control bit, and input $|0\rangle$ to target bit

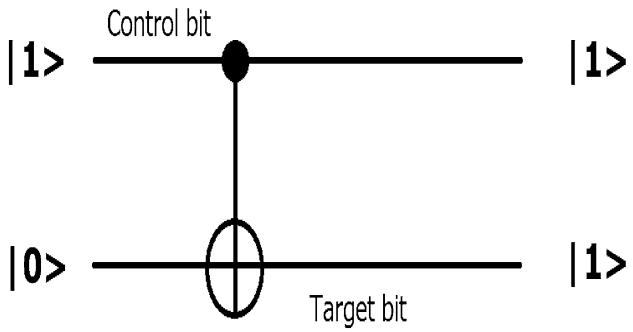

Fig. 9 CX gate with input $|1\rangle$ to control bit, and input $|0\rangle$ to target bit

applied on the second qubit, else no transformation is performed (Fig. 8). This gate is represented by letters 'CH'.

$$\mathrm{CH} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ 0 & 0 & \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix}$$

Controlled NOT Gate

Controlled NOT gate also have a control qubit and a target qubit. When control qubit holds state $|1\rangle$, Pauli X gate is applied to the target qubit (Fig. 9). Controlled NOT gate is represented by letters 'CX' or 'CNOT'.

$$\mathrm{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Quantum Entanglement

Quantum entanglement is a physical phenomenon because of which quantum states of each particle cannot be described independent of each other. This phenomenon still persists even if the entangled particles are separated to a greater distance.

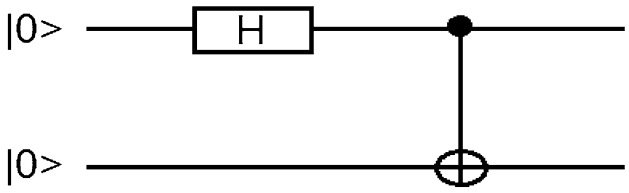

Fig. 10 Positive correlation between two entangled qubits

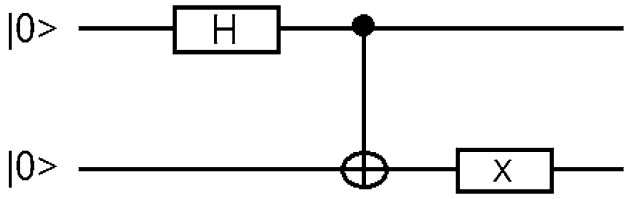

Fig. 11 Negative correlation between two entangled qubits

Two qubits initialized to $|0\rangle$ state can be entangled with positive correlation using CNOT and Hadamard quantum gates. In positive correlated entanglement, output of both the qubits during measurement remains same, i.e. either both are in $|0\rangle$ state or both are in $|1\rangle$ state (Fig. 10).

While designing two entangled qubits (both initialized to $|0\rangle$) with negative correlation, an additional Pauli X gate is applied on the target qubit of CNOT gate. In negative correlated entanglement, output of both the qubits during measurement remains opposite to each other. If one is measured as $|0\rangle$ state, then the other will always be in $|1\rangle$ state (Fig. 11).

Trust Issues with Lucky Draw-Based Systems

Lucky draw is supposed to be an activity of randomly selecting a winner from the available pool of candidates. Ideally, it is supposed that such type of systems will not have any preferential bias towards any participant, but depending on the intensity of preferential bias, all available lucky draw systems can be manipulated at some stage to profit an individual or group of individuals.

Digital systems can be easily manipulated by the authorities to provide desired output in lucky draw, and other randomness-based gambling systems. Depending on the influence one group have on the digital system random function's source code, outcome of the digital system can be controlled or altered (Fig. 12).

Although quantum superposition can be used for having fair randomness for selecting a winner from the list of participants, to perform this operation, Hadamard gates are

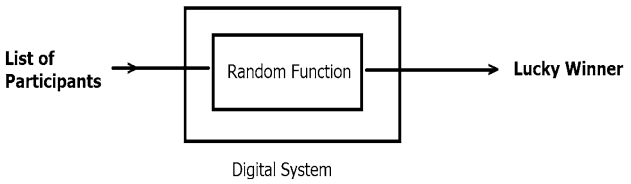

Fig. 12 Winner can be selected by manipulating the source code of random function

applied on N qubits (all initialized to $|0\rangle$ state) to provide equal chance of selecting one candidate out of 2^N total candidates. Although it requires a smaller number of qubits, this approach is not completely quantum assured because no-cloning theorem [12] does not allow the obtained result to be directly teleported to multiple candidates. Thus, the results have to be sent to a classical computer to broadcast it to all the participants. Involvement of digital computer makes this approach vulnerable to alteration (Fig. 13).

Instead of providing a single result, our proposed approach provides results for the entire participants about whether they are the winner or not. This makes it impossible to alter the obtained results by individual(s), and also because we are not sending multiple copies of same result, this enables us to teleport the individual's result to them and making the entire process completely based on quantum computers (Fig. 14).

Proposed Algorithm

Ensuring fairness in random decision-making helps in establishing trust among its users, especially in events like lottery and lucky draw, where the system can be easily altered to directly or indirectly profit an individual or group of individuals. Our proposed quantum circuit designing algorithm randomly provides one qubit with state-1 and the remaining with state-0 while ensuring fairness at each level of random decision-making. Its architecture allows the results to be teleported to other quantum systems for performing further tasks, or even teleporting results directly to the clients which make the system end-to-end quantum secured (Table 1).

Function Quantum_Random_Selection(total_number)

```
Connect H gate in qubit[0]
Connect CX gate to quantum circuit with control bit in qubit[0] and target bit in qubit[1]
Connect X gate in qubit[1]
IF total_number >2
  For index_limit in range 1 to log2(total_number) - 1
    index= -1
    while index <( 2**index_limit )-1
      index = index+1
      index1 = index
      index2 = index + 2**index_limit
      Connect CH gate to quantum circuit with control bit in qubit[index1] and target bit in qubit[index2]
      Connect CX gate to quantum circuit with control bit in qubit[index2] and target bit in qubit[index1]
```

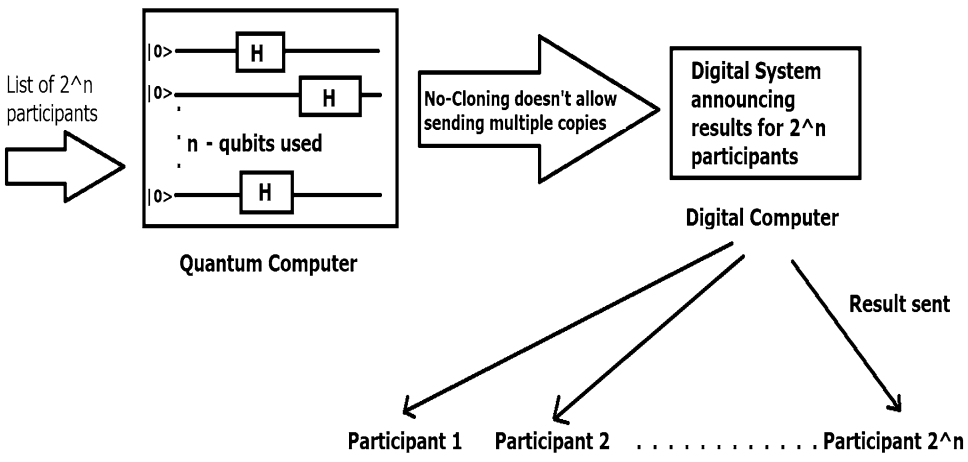

Fig. 13 No-cloning theorem does not allow sending multiple copies of the result

Output state produced by our proposed algorithm with 2^N quantum bits presents equal chances of one qubit holding $|1\rangle$ state, while remaining $2^N - 1$ qubits hold $|0\rangle$. For example in the system of two participants (output state is represented using $|\boldsymbol{\varphi}\rangle$), one qubit always holds $|1\rangle$ state and remaining holds $|0\rangle$ state.

$$|\boldsymbol{\varphi}\rangle = \frac{1}{\sqrt{2^1}}(|10\rangle + |01\rangle)$$

Working Demonstration

Proposed algorithm is implemented using QISKit Library [13] of IBM, and its source code is available in our GitHub repository [14]. For better understanding of the proposed algorithm, two working demonstrations of our algorithm are also provided.

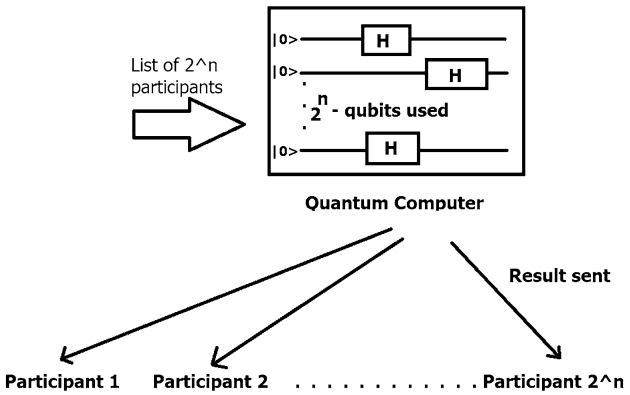

Fig. 14 Our approach provides end-to-end quantum computing-based solution

Table 1 Parameters related detail

S. no	Symbol/variable name	Description
1	total_number	Total number of participants
2	qubit	Array containing quantum bits with each initialized to $\|0\rangle$ state
3	*H*	Represents Hadamard gate
4	CH	Represents controlled Hadamard gate
5	*X*	Represents Pauli *X* gate
6	CX	Represents controlled NOT gate
7	2**A	Represents 2^A

Working Demonstration with Four Participants

To perform random selection among four participants, our proposed algorithm requires four quantum bits (i.e. qubit[] array indexing from 0 to 3) (Fig. 15, Table 2).

Output state $|\boldsymbol{\varphi}\rangle$ has one qubit in $|1\rangle$ state, while remaining holds $|0\rangle$ state.

$$|\boldsymbol{\varphi}\rangle = \frac{1}{\sqrt{2^2}}(|1000\rangle + |0100\rangle + |0010\rangle + |0001\rangle)$$

Working Demonstration with Eight Participants

Our proposed algorithm requires eight quantum bits to perform fair random selection among eight participants (i.e. qubit[] array is indexed from 0 to 7) (Table 3; Fig. 16).

Output state $|\boldsymbol{\varphi}\rangle$ has one qubit in $|1\rangle$ state, while remaining holds $|0\rangle$ state.

$$|\boldsymbol{\varphi}\rangle = \frac{1}{\sqrt{2^3}}(|10000000\rangle + |01000000\rangle + |00100000\rangle + |00010000\rangle + |00001000\rangle + |00000100\rangle + |00000010\rangle + |00000001\rangle)$$

These manipulated qubits can further be teleported (or sent) to the individual participants making our end-to-end system relying only on quantum phenomenon which in turn ensures fairness in each step of our proposed algorithm.

Conclusion

Our proposed quantum algorithm provides a solution to the problem of fair random selection among small number of participants, and has the potential to be used in commercial domains requiring quantum assurance for the random selection tasks. Due to the complete random nature of quantum, it can be used in activities involving lucky draw, and any other similar form of activity dependent on random selection task.

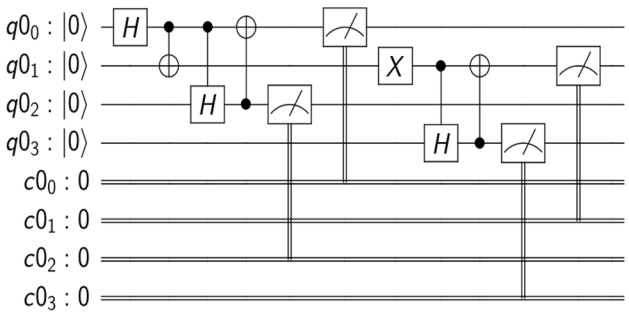

Fig. 15 Visualization of quantum circuit with four participants generated using QISKit library

Table 2 Steps followed by our proposed algorithm for four participants

Step	Operation performed
1	Connect H gate in qubit[0]
2	Connect CX gate to quantum circuit with control bit in qubit[0] and target bit in qubit[1]
3	Connect *X* gate in qubit[1]
4	Connect CH gate to quantum circuit with control bit in qubit[0] and target bit in qubit[2]
5	Connect CX gate to quantum circuit with control bit in qubit[2] and target bit in qubit[0]
6	Connect CH gate to quantum circuit with control bit in qubit[1] and target bit in qubit[3]
7	Connect CX gate to quantum circuit with control bit in qubit[3] and target bit in qubit[1]

Table 3 Steps followed by our proposed algorithm for eight participants

Step	Operation performed
1	Connect *H* gate in qubit[0]
2	Connect CX gate to quantum circuit with control bit in qubit[0] and target bit in qubit[1]
3	Connect *X* gate in qubit[1]
4	Connect CH gate to quantum circuit with control bit in qubit[0] and target bit in qubit[2]
5	Connect CX gate to quantum circuit with control bit in qubit[2] and target bit in qubit[0]
6	Connect CH gate to quantum circuit with control bit in qubit[1] and target bit in qubit[3]
7	Connect CX gate to quantum circuit with control bit in qubit[3] and target bit in qubit[1]
8	Connect CH gate to quantum circuit with control bit in qubit[0] and target bit in qubit[4]
9	Connect CX gate to quantum circuit with control bit in qubit[4] and target bit in qubit[0]
10	Connect CH gate to quantum circuit with control bit in qubit[1] and target bit in qubit[5]
11	Connect CX gate to quantum circuit with control bit in qubit[5] and target bit in qubit[1]
12	Connect CH gate to quantum circuit with control bit in qubit[2] and target bit in qubit[6]
13	Connect CX gate to quantum circuit with control bit in qubit[6] and target bit in qubit[2]
14	Connect CH gate to quantum circuit with control bit in qubit[3] and target bit in qubit[7]
15	Connect CX gate to quantum circuit with control bit in qubit[7] and target bit in qubit[3]

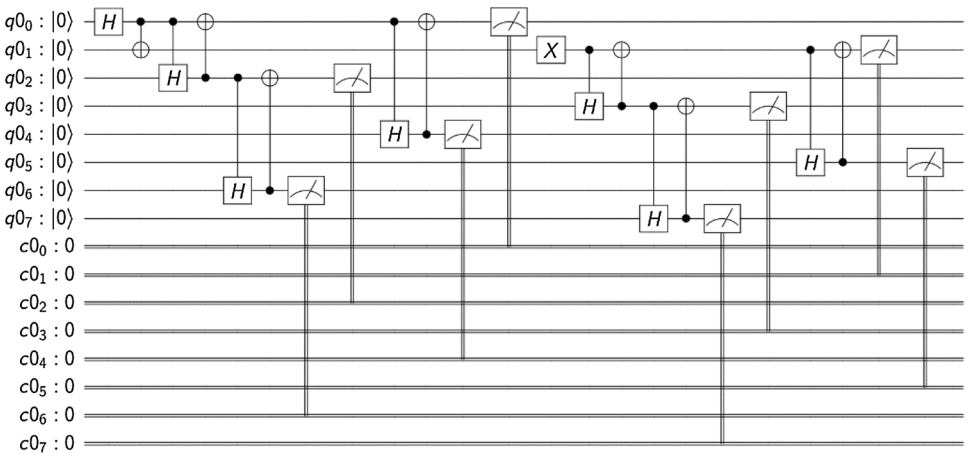

Fig. 16 Visualization of quantum circuit representation of eight participants with QISKit library

References

1. De Ronde C. Quantum superpositions and the representation of physical reality beyond measurement outcomes and mathematical structures. Found Sci. 2016;19:1–28.
2. Witten E. Notes on some entanglement properties of quantum field theory; 2018. arXiv preprint arXiv:1803.04993.
3. Hadjiivanov L, Todorov I. Quantum entanglement. 2015. arXiv preprint arXiv:1506.04262.
4. Luo H, Yuan J, Dai W. A new universal quantum gates and its simulation on GPGPU. In: International conference on cloud computing and security. Cham: Springer; 2017. p. 16–27.

5. Wu C, Yang L. Qubit-wise teleportation and its application in public-key secret communication. Sci China Inf Sci. 2017;60(3):032501.
6. Grover LK. A fast quantum mechanical algorithm for database search. In: Proceedings of the 28th annual ACM symposium on theory of computing. ACM; 1996. p. 212–9.
7. Shor PW. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM Rev. 1999;41(2):303–32.
8. Tipsmark A, Dong R, Laghaout A, Marek P, Ježek M, Andersen UL. Experimental demonstration of a Hadamard gate for coherent state qubits. Phys Rev A. 2011;84(5):050301.
9. Williams CP. Quantum gates. In: Explorations in quantum computing. London: Springer; 2011. p. 51–122. https://www.springer.com/gp/book/9781846288869#aboutBook.
10. Vishnu PK, Joy D, Behera BK, Panigrahi PK. Experimental demonstration of non-local controlled-unitary quantum gates using a five-qubit quantum computer. Quantum Inf Process. 2018;17(10):274.
11. Saeedi M, Zamani MS, Sedighi M. Algebraic characterization of CNOT-based quantum circuits with its applications on logic synthesis. 2007. arXiv preprint arXiv:0712.2963.
12. Wootters WK, Zurek WH. The no-cloning theorem. Phys Today. 2009;62(2):76–7.
13. Github repository. https://github.com/qiskit. Accessed 8 Feb 2019.
14. Github repository. https://github.com/mr-ravin/QrandomSelection. Accessed 8 Feb 2019.