# An Efficient Quantum Binary-Neuron Algorithm for Accurate Multi-Story Floor Localization

Yousef Zook Computer and Systems Engineering Alexandria University Alexandria, Egypt es-yousif.mohamed@alexu.edu.eg® Ahmed Shokry Computer Science and Engineering Pennsylvania State University PA, USA ahmed.shokry@psu.edu© Moustafa Youssef Computer Science and Engineering University of New South Wales Sydney, Australia m.yousief@unsw.edu.au®

Abstract—Accurate floor localization in a multi-story environment is an important but challenging task. Among the current floor localization techniques, fingerprinting is the mainstream technology due to its accuracy in noisy environments. To achieve accurate floor localization in a building with many floors, we have to collect sufficient data on each floor, which needs significant storage and running time; preventing fingerprinting techniques from scaling to support large multi-story buildings, especially on a worldwide scale.

In this paper, we propose a quantum algorithm for accurate multi-story localization. The proposed algorithm leverages quantum computing concepts to provide an exponential enhancement both space and running time compared to the classical in counterparts. In addition, it builds on an efficient binary-neuron implementation that can be implemented using fewer qubits compared to the typical non-binary neurons, allowing for easier deployment with near-term quantum devices. We implement the proposed algorithm on a real IBM quantum machine and evaluate it on three real indoor testbeds. Results confirm the exponential saving in both time and space for the proposed quantum algorithm, while keeping the same localization accuracy compared to the traditional classical techniques, and using half the number of qubits required for other quantum localization algorithms.

*Index Terms*—floor localization, quantum applications, quantum neuron, quantum computing.

#### I. INTRODUCTION

Indoor localization has become an important requirement in smart cities and Internet of Things (IoT)-based applications as people spend more time indoors [1]. Accurate 3D localization is crucial in many commercial and emergency situations, such as indoor multi-floor navigation and emergency response services, where getting an accurate location in a large multi-story building can make the difference between life and death [2].

Different techniques have been proposed to solve the floor estimation problem [3, 4, 5, 6]. Among the current floor localization techniques, fingerprinting is the mainstream technology due to its accuracy in noisy environments [3]. These techniques have two phases: the offline calibration phase and the online phase. In the offline phase, the data coming from the different signal sources (e.g. WiFi access points, APs) is collected at the different floors in the building. Then, in the online phase, the system matches the heard signals from a user's device at an unknown floor with the fingerprint data to estimate her floor. The closest floor in the fingerprint is reported as the estimated floor. The number of signal sources used in the matching process affects the final floor estimation accuracy; i.e., the higher the number of signal sources, the more precise the accuracy will be[7, 8]. However, this increases the matching time and the space needed to store a large fingerprint, especially for large multi-story buildings. This drawback poses a major difficulty for fingerprinting techniques in real-world floor localization systems.

Recently, quantum fingerprint-based localization algorithms started to appear to enable a large-scale worldwide localization [9, 10, 11, 12, 13, 14, 15], mainly focusing on **2D** *localization in a single floor*. However, these algorithms require a relatively large number of qubits, making them less suitable for near-term quantum devices. This is even *worse* when we apply them to the multi-floor localization problem, where the *fingerprint is much larger*.

In this paper, we propose an efficient quantum fingerprintbased floor localization algorithm for near-term quantum devices. The proposed algorithm uses the quantum binary neuron to provide an exponential saving in time and space compared to its classical counterparts [16]. Furthermore, this implementation requires nearly *half* the number of qubits compared to current quantum localization algorithms [9, 10, 11, 12]. This allows our algorithm to use double the number of signal sources, leading to better localization accuracy and/or more efficient implementation.

We implement our quantum algorithm on an IBM Quantum Experience machine and deploy it in three real multi-story buildings. The results confirm the ability of our quantum algorithm to achieve the same accuracy as classical floor localization but with the potential exponential saving in both space and running time.

The rest of the paper is organized as follows: Section II gives a background on the floor localization problem and the binary neuron. We discuss the details of the proposed quantum floor localization algorithm in Section III. Then, we evaluate the performance of our system in Section IV. Finally, we conclude our work in Section V.

# II. BACKGROUND

# A. The Floor Estimation Problem

Fingerprint-based floor localization algorithms [3] depend on the pre-installed signal sources (e.g., WiFi or cellular infras-



**Fig. 1:** A general artificial neuron model diagram. The neuron input  $\vec{\Phi}$  is processed with the neuron weight vector  $\vec{\Psi}$  using dot product to produce the output of the neuron (the activation value). A binary neuron has binary inputs and weight vectors.

tructure, also called reference points) to estimate users' floor. Generally, they work in two phases: the offline calibration phase, where the signal information, such as the Received Signal Strength (RSS) and the signal source ID, is collected at different locations on each floor to construct the fingerprint database. The fingerprint is stored as vectors, where each entry in the vector represents the RSS from one reference point, alongside the floor number where these vectors are collected. In the online tracking phase, the online signal sources information is matched to the fingerprint vectors, and the closest floor in the signal space becomes the estimated floor. *Classically*, matching the online information to the fingerprint takes o(MN) *time and space complexity*, where *M* is the number of vectors in the fingerprint and *N* is the number of signal sources.

As the floor localization accuracy depends on the density of the signal sources, the matching time and space requirements limit the accuracy of the fingerprint-based techniques, especially for large buildings.

#### B. Binary Neuron

An artificial neuron mimics what is done in a real biological neuron. It takes signals as inputs, processes these signals, and activates an output signal based on this process. An artificial neuron is basically a mathematical representation of the biological neuron (Figure 1). Specifically, the output activation of a neuron is calculated as the weighted sum of multiplying each input  $\phi_i$  with a weight  $\psi_i$  as<sup>1</sup>:

$$\langle \Psi | \Phi \rangle = \sum_{i=0}^{N-1} \Psi_i \Phi_i \tag{1}$$

A binary neuron is a special version of the general neuron, where the neuron takes a *binary-valued input* vector  $\vec{\Phi}$ , then it is combined with a *binary-valued weight vector*  $\vec{\Psi}$ . The output is a response function evaluated from the inner product of the two vectors, representing the neuron *activation* value.

In this paper, we use a quantum implementation of this binary neuron to estimate the user floor number, where the



**Fig. 2:** System architecture for the quantum fingerprint-based floor localization using quantum neuron.

fingerprint samples are the inputs to the neuron and the user's sample represents the neuron weight. The output activation value of this neuron is going to be used as an indicator of the similarity between the user sample and each fingerprint sample.

# III. THE PROPOSED QUANTUM FLOOR ESTIMATION ALGORITHM

To reduce the matching time and space requirements, we use the quantum binary-neuron circuit [16]. The basic idea is to use an instance of the quantum binary neuron to calculate the similarity between the online RSS vector (i.e., user sample) and each one of the fingerprint training vectors (i.e., fingerprint sample), one at a time. The floor that gives the highest activation is reported as the estimated location for the user sample.

### A. Floor Estimation Algorithm

Figure 2 shows the architecture of the proposed quantum floor localization system. In the offline phase, the fingerprint data is collected across the building floors by collecting the signal sources IDs along with the floor number as the ground truth in a vector. Each entry in this vector represents a specific signal source. Then, in the online phase, we collect the user signal sources IDs heard by the user's device at an unknown floor. We compare the online RSS vector to vector at each fingerprint location using the quantum binary neuron. The floor in the fingerprint sample with the highest activation value is returned as the estimated floor number.

The RSS vectors are binary vectors: each entry is either -1 or +1, where -1 at a vector index *i* means that the signal source number *i* is not heard, and +1 represents that the signal source is heard.

Assuming that the user online sample is  $\Psi = \begin{bmatrix} \psi_0 & \dots & \psi_{N-1} \end{bmatrix}$  and the fingerprint sample number l is  $\Phi^l = \begin{bmatrix} \phi_0^l & \dots & \phi_{N-1}^l \end{bmatrix}$ , where N is the number of the signal sources and  $\phi_i^l, \psi_i \in \{-1, 1\}$ , we can encode the vectors value using amplitude encoding [17] to the following quantum states,

<sup>&</sup>lt;sup>1</sup>The Dirac notation is usually used to mathematically describe quantum systems. The  $|.\rangle$  notation is called ket and represents a column vector, the  $\langle .|$  notation is called bra and represents a row vector, and the  $\langle .|.\rangle$  notation is called braket and represents the dot product of the two vectors.



Fig. 3: Binary neuron quantum circuit to calculate neuron activation between a fingerprint sample  $\Phi^l$ , and a user sample  $\Psi$ .  $|\gamma_i\rangle$  represents the joint system state at different positions in the circuit.

$$|\Psi\rangle = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} \psi_i |i\rangle \tag{2}$$

$$|\Phi^l\rangle = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} \phi_i^l \left|i\right\rangle \tag{3}$$

Where  $|i\rangle$  is the binary encoding for *i*. Note the exponential saving in the space where a vector of *N* entries is encoded in  $\log(N)$  qubits.

To find the fingerprint sample  $\Phi^l$  that is most similar to the online sample  $\Psi$ , we can use a binary neuron, where each fingerprint sample  $\Phi^l$  at location l acts as an input to the neuron and the online sample  $\Psi$  acts as the neuron weight vector. Hence, we can find the neuron activation value using the dot product  $\langle \Psi | \Phi^l \rangle$ .

Figure 3 shows the quantum circuit for implementing the quantum binary similarity algorithm between the two vectors  $\Phi^l$  and  $\Psi$  (i.e.  $\langle \Psi | \Phi^l \rangle$ ). The input to the circuit is a single ancilla qubit at state  $|0\rangle$  and *n*-qubit register initially at state  $|0\rangle^{\otimes n}$ , where *N* is the vectors length representing the *N* signal sources, and  $n = \log(N)$ .

The circuit consists of three stages:  $\Phi^l$  Encoding,  $\Psi$  Entangling, and Measurement stages.

1) The  $\Phi^l$  Encoding Stage: Initially, the quantum system is in state:

$$|\gamma_0\rangle = |0\rangle |0\rangle^{\otimes n} \tag{4}$$

The first step is to construct the fingerprint binary state vector  $\Phi^l$ . This is achieved by applying a unitary transformation  $U_{\Phi^l}$  on the data qubits such that:

$$U_{\Phi^l} \left| 0 \right\rangle^{\otimes n} = \left| \Phi^l \right\rangle \tag{5}$$

where  $U_{\Phi^l}$  is a gate that obtains  $|\Phi^l\rangle$  from  $|0\rangle^{\otimes n}$  using the Hypergraph States Generation Subroutine [16]. This moves the quantum state to:

$$|\gamma_1\rangle = |0\rangle |\Phi^l\rangle \tag{6}$$

2) The  $\Psi$  Entangling Stage: The second step is to apply another unitary transformation  $U_{\Psi}$  to the data register of the circuit in quantum state  $|\gamma_1\rangle$ . The quantum state moves to:

$$|\gamma_2\rangle = |0\rangle U_{\Psi} |\Phi^l\rangle = |0\rangle |\eta\rangle \tag{7}$$

where  $|\eta\rangle = U_{\Psi} |\Phi^l\rangle$ .

The  $U_{\Psi}$  is a gate that moves a quantum state  $|\Psi\rangle$  to state  $|1\rangle^{\otimes n}$ , i.e.:

$$U_{\Psi} |\Psi\rangle = |1\rangle^{\otimes n} = |N-1\rangle \tag{8}$$

Similar to  $U_{\Phi}$ , the  $U_{\Psi}$  gate can be realized using the Hypergraph States Generation Subroutine [16].

3) The Measurement Stage: The last step is to measure the neuron activation value, i.e. the dot product  $\langle \Psi | \Phi^l \rangle$ . Using equations 7 and 8, the required similarity can be written as:

$$\langle \Psi | \Phi^l \rangle = \langle \Psi | U_{\Psi}^{\dagger} U_{\Psi} | \Phi^l \rangle = \langle N - 1 | \eta \rangle \tag{9}$$

Since  $|\eta\rangle = \eta_0 |0\rangle + \eta_1 |1\rangle + ... + \eta_{N-1} |N-1\rangle$ , Equation 9 reduces to

$$\langle \Psi | \Phi^l \rangle = \eta_{N-1} \tag{10}$$

Hence, the squared neuron activation value is the probability of measuring the data register in state  $|N - 1\rangle$ . To measure this probability, we add the final CNOT gate that flips the ancilla qubit if the data register is in state  $|N - 1\rangle = |1\rangle^{\otimes n} = [0, 0, ..., 1]$ ,

$$|\langle \Psi | \Phi^l \rangle|^2 = P(a=1) = \eta_{N-1}^2$$
(11)

#### B. Example

In this section, we give a simple example of the quantum binary neuron circuit mentioned in the previous section using two fingerprint vectors collected at two locations:  $l_0$  and  $l_1$ , and a user sample vector, each with four signal sources:  $S_i$  for  $i \in \{0..3\}$ .

At location  $l_0$ , all signal sources are heard except  $S_3$ , i.e. the fingerprint sample at location  $l_0$  is:  $\Phi^{l_0} = [1, 1, 1, -1]$ , and at location  $l_1$ , signal sources  $S_2$  and  $S_3$  are not heard, i.e. fingerprint sample at location  $l_1$  is  $\Phi^{l_1} = [1, 1, -1, -1]$ , where 1 means that the signal source is being heard at the location and -1 means that it is not heard. Let the user sample be  $\Psi = [1, 1, 1, 1]$ , i.e. all signal sources are heard at the current online user location.

To find the neuron activation value for the fingerprint samples and the user sample vectors, we construct the two quantum circuits shown in Figures 4. The first subfigure (a) shows the simple implementation of the  $U_{\Phi^{l_0}}$  and  $U_{\Psi}$  to find the neuron activation value (dot product) of the first fingerprint sample  $\Phi^{l_0}$  with the user sample  $\Psi$ . The  $U_{\Phi^{l_0}}$ gate is implemented by applying the Hadamard gates (H) on the data register, which produces an equal superposition state of  $\frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$ . Then the Controlled Z (CZ) gate is added to flip the sign of the basis state  $|11\rangle$  by rotating the qubits around the Z-axis of the Bloch sphere [18]. The  $U_{\Psi}$  gate moves state  $|\Psi\rangle$  to state  $|1\rangle^{\otimes n}$  according to Equation 8. This is done by applying H gates to move the  $|\Psi\rangle = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$  state to state  $|0\rangle^{\otimes n}$  then adding NOT gates (X) to move it to state  $|1\rangle^{\otimes n}$ .

The second subfigure (b) shows the quantum circuit implementation for calculating the similarity to the second fingerprint  $\Phi^{l_1}$  sample. Similar to the first circuit, the  $U_{\Phi^{l_1}}$  is



(a) Quantum circuit for calculating activation value for the first fingerprint sample  $\Phi^{l_0} = [1, 1, 1, -1]$  and the user sample  $\Psi = [1, 1, 1, 1]$ 



(b) Quantum circuit for calculating activation value for the second fingerprint sample  $\Phi^{l_1} = [1, 1, -1, -1]$  and the user sample  $\Psi = [1, 1, 1, 1]$ 

Fig. 4: Binary neuron quantum circuits to calculate neuron activation values between the fingerprint samples  $\Phi^{l_i}$  and the user sample  $\Psi = [1, 1, 1, 1]$ .

implemented by applying H gates on the data register qubits then a Z gate on the second data qubit to move the data register to state  $\frac{1}{2}(|00\rangle + |01\rangle - |10\rangle - |11\rangle)$ . To measure the probability  $P(a_0 = 1)$ , we run the first

To measure the probability  $P(a_0 = 1)$ , we run the first quantum circuit for k times (number of shots) and count the number of times where the ancilla qubit output is 1. Then using Equation 12, we can find the neuron activation value.

$$\langle \Psi | \Phi^l \rangle |^2 = P(a=1) = \frac{count(a=1)}{k} \simeq 0.5067$$
 (12)

Similarly, we run the second circuit (b) for the same number of shots (k) to get the activation value for the location  $l_1$  as  $\langle \Psi | \Phi^{l_1} \rangle = 0$ . Since the activation value of the first fingerprint  $\langle \Psi | \Phi^{l_0} \rangle$  is higher than the activation value of the second fingerprint  $\langle \Psi | \Phi^{l_1} \rangle$ , we estimate the user's floor as the floor number of the first fingerprint sample's location  $l_0$ .

### C. Complexity Analysis

In the offline phase, to store a sample vector of size N, we need N memory units in a classical computer. This makes the total space complexity needed to store the fingerprint data with M samples o(MN) in the classical algorithm. In contrast, the quantum floor localization algorithm stores a vector sample of size N in  $q = 1 + \log(N)$ , where one qubit is used as an ancilla and  $\log(N)$  qubits register is used as the data register, i.e. the quantum algorithm has a space complexity of order  $o(M \log(N))$ .

In the online phase, the fingerprint and user vectors are encoded and entangled in the quantum circuit using  $U_{\Phi t}$  and  $U_{\Psi}$  unitary transformations, which can be achieved efficiently using the Hypergraph States Generation Subroutine [16]. Hence, using efficient unitary transformations, the proposed quantum algorithm achieves  $o(M \log(N))$  time bound.

Furthermore, the fingerprint data vector  $\Phi^l$  can be loaded directly from quantum sensors [19] in the future or from a quantum random access memory (QRAM) [20], where the binary representation of  $\Phi_i^l$  is loaded in parallel into the quantum data register, which allows us to use a single quantum circuit for matching all fingerprint samples with a user's sample.

# IV. EVALUATION

In this section, we evaluate the quantum binary neuron algorithm in three real testbeds and implement it on a real IBM quantum machine. We start by describing the testbeds. Then, we show the results of comparing the proposed quantum algorithm with its classical counterpart as well as the state-of-the-art quantum fingerprint localization systems [9, 10, 11]. Finally, we discuss the different advantages of the proposed quantum algorithm.

#### A. Environment Setup

The collected data cover three different buildings of Jaume I University [21]. We deploy the proposed quantum algorithm on the first building (four floors), the second building (four floors), and the third building (five floors). We used the already deployed 16 WiFi access points (APs) as signal sources (N = 16). The floors area is around  $110m^2$  [21].

We deployed the proposed quantum algorithm on each building and compared the results to the classical counterpart. Furthermore, we compared the proposed quantum circuit to the state-of-the-art quantum fingerprint localization circuit [10, 11]. We used the real 5-qubits *ibmq\_manila* quantum computer [22].

# B. Performance Evaluation

1) Comparison to classical binary neuron: Figure 5 shows the floor localization accuracy CDF for the three buildings using the Quantum Binary Neuron (QBN) algorithm, the classical counterpart, and the random floor classifier as a worstcase bound for localization. The figure shows that both the quantum and classical algorithms achieve the same accuracy. This comes with an exponential gain in both the running time and the space.

Figure 6, shows the space and running time complexity between the classical and the proposed quantum algorithm. The figure shows that the proposed quantum algorithm achieves  $o(M \log(N))$  compared to o(MN). This highlights the exponential gain of the proposed quantum algorithm over its classical counterpart due to the fact that the quantum algorithm grows in a logarithmic manner with N.

2) Comparison to state-of-the-art quantum localization: To validate that the quantum binary neuron gives similar accuracy to the state-of-the-art quantum localization [9, 10, 11] while using a lower number of qubits, we run the quantum binary neuron circuit and the circuit used in the state-of-the-art quantum algorithm [9, 10, 11] on the same testbeds. However, we use only **four APs** in this experiment, as the previous



Fig. 5: Floor localization error CDF comparing the proposed quantum algorithm, the classical counterpart, and a random classifier (as a baseline) for the three testbeds.



Fig. 6: The complexity of the classical algorithm (o(MN)) and the proposed quantum algorithm (o(MLog(N))).

quantum algorithms would require more than the 5-qubits available on the used *ibmq\_manila* quantum machine. This highlights the advantage of our proposed algorithm that can use up to 16 APs on this machine leading to better localization accuracy and making it more suitable for near-term quantum devices.

Figure 7 validates that the quantum binary neuron circuit can achieve a similar accuracy to the quantum circuit used in the state-of-the-art quantum localization algorithms [9, 10, 11]. The proposed quantum binary neuron requires almost half the number of qubits  $(1 + \log(N))$  compared to the previous quantum algorithms that require  $(1 + 2\log(N))$  qubits.

Moreover, Figure 8, confirms that the higher the number of signal sources, the more precise the accuracy will be. This highlights the importance of using more signal sources, which can be achieved in low number of qubits  $(1 + \log(N))$  using the proposed quantum binary neuron algorithm. This makes it more feasible to be implemented on near-term quantum devices.

#### C. Discussion

The results of running our quantum algorithm on three real testbeds show that the proposed quantum floor localization algorithm has a similar accuracy as its classical counterpart but with the exponential gain in time and space, and has a similar accuracy to the state-of-the-art quantum algorithms [9, 10, 11] while using a lower number of qubits.

The classical floor estimation algorithm requires o(MN) space and running time. On the other side, the proposed quantum algorithm requires  $o(M \log(N))$  space and time. This highlights the quantum computing gain and enables fingerprint-based floor localization systems to be deployed in large-scale multi-story buildings while maintaining a similar localization accuracy as the classical algorithm [7, 8].

Furthermore, the proposed quantum algorithm is useful for the offline calibration phase as it reduces the fingerprint vector size from o(N) to  $o(\log(N))$ , which mitigates the data storage overhead and saves the required bandwidth for applications where the data is downloaded on the users' devices when the matching process is done on users' local devices (e.g., for privacy-sensitive applications).

Comparing the proposed quantum algorithm to the stateof-the-art quantum localization algorithms that require  $1 + 2\log(N)$  number of qubits, results show that the proposed quantum binary neuron algorithm achieves similar accuracy while using  $1 + \log(N)$  qubits only compared to these algorithms. This makes it able to achieve higher accuracy on near-term quantum machines, where the number of qubits is limited.

Finally, a single quantum circuit can be used for different fingerprint samples by loading the fingerprint data from a quantum random access memory (QRAM) or quantum sensors instead of the unitary transformation  $U_{\Phi^l}$  in the first stage, and both space and time complexity can be further enhanced to  $o(\log(MN))$  by encoding all the fingerprint locations using quantum states. And state preparation routines could be used to reduce the number and depth of the circuit gates.

#### V. CONCLUSION

In this paper, we introduced a fingerprint-based quantum algorithm for multi-story large-scale floor localization. The proposed quantum algorithm requires  $o(M \log(N))$  time and space, unlike its classical counterpart, which needs o(MN) time and space for a fingerprint with M locations and N signal sources. We implemented the proposed quantum algorithm on a real IBM quantum machine and deployed it in three real testbeds. Results show that our quantum algorithm can



Fig. 7: Floor localization error CDF comparing using the quantum binary neuron circuit, and the quantum circuit mentioned in [9, 10, 11] which requires nearly double the number of qubits compared to the proposed quantum binary neuron circuit.



Fig. 8: Floor localization accuracy using different number of signal sources (N).

achieve the same accuracy as the classical algorithm but with the exponential gain in time and space. This confirms the gain of using quantum algorithms to solve large-scale floor localization problems, where more data can be used with lower number of computing resources. Furthermore, we show the advantage of using the proposed quantum binary neuron circuit for near-term quantum machines where the number of qubits is limited. The proposed algorithm can use nearly half the number of qubits compared to other state-of-the-art quantum localization algorithms while keeping the localization accuracy. Currently, we are expanding our research in multiple directions including exploring different quantum similarity metrics implementations, extending our exponential saving to the fingerprint samples (M), handling the intrinsic practical considerations of quantum processors in the algorithm development, obtaining theoretical quantum performance bounds, among others.

## REFERENCES

- Qiao Wan, Xiaoqi Duan, Yue Yu, Ruizhi Chen, and Liang Chen. Self-Calibrated Multi-Floor Localization Based on Wi-Fi Ranging/Crowdsourced Fingerprinting and Low-Cost Sensors. *Remote Sensing*, 14(21):5376, 2022.
- [2] Asmaa Saeed, Ahmed Wasfey, Hamada Rizk, and Hirozumi Yamaguchi. CellStory: Extendable cellular signals-based floor estimator using deep learning. In 2022 18th International Conference on Intelligent Environments (IE), pages 1–4. IEEE, 2022.

- [3] Litao Han, Li Jiang, Qiaoli Kong, Ji Wang, Aiguo Zhang, and Shiming Song. Indoor Localization within Multi-Story Buildings Using MAC and RSSI Fingerprint Vectors. *Sensors*, 19(11):2433, May 2019.
- [4] K. Kim, S. Lee, and K. Huang. A scalable deep neural network architecture for multi-building and multi-floor indoor localization based on Wi-Fi fingerprinting. *Big Data Analytics*, 3(4), 2018.
- [5] Rizanne Elbakly, Moustafa Elhamshary, and Moustafa Youssef. HyRise: A Robust and Ubiquitous Multi-Sensor Fusion-Based Floor Localization System. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 2(3), sep 2018.
- [6] S. Mostafa, K. A. Harras, and M. Youssef. A survey of indoor localization systems in multi-floor environments. 2022.
- [7] Paramvir Bahl and Venkata N Padmanabhan. RADAR: An in-building RF-based user Location and Tracking System. In *IEEE INFOCOM 2000.*, volume 2, pages 775–784. IEEE, 2000.
- [8] Moustafa Youssef and Ashok Agrawala. The Horus WLAN Location Determination System. In Proceedings of the 3rd international conference on Mobile systems, applications, and services, pages 205–218, 2005.
- [9] Ahmed Shokry and Moustafa Youssef. Challenge: Quantum Computing for Location Determination. arXiv eprints, pages arXiv–2106, 2021.
- [10] Ahmed Shokry and Moustafa Youssef. QLoc: A Realistic Quantum Fingerprint-based Algorithm for Large Scale Localization. *IEEE International Conference on Quantum Computing and Engineering (QCE)*, 2022.
- [11] Ahmed Shokry and Moustafa Youssef. A Quantum Algorithm for RF-based Fingerprinting Localization Systems. *IEEE Conference on Local Computer Networks (LCN)*, 2022.
- [12] Ahmed Shokry and Moustafa Youssef. Towards Quantum Computing for Location Tracking and Spatial Systems. In Proceedings of the 29th International Conference on Advances in Geographic Information Systems, pages 278–281, 2021.
- [13] Yousef Zook, Ahmed Shokry, and Moustafa Youssef. A quantum fingerprinting algorithm for next generation

cellular positioning. *IEEE Journal on Selected Areas in Communications*, 42(1):93–102, 2024.

- [14] Ahmed Shokry and Moustafa Youssef. An efficient quantum euclidean similarity algorithm for worldwide localization, 2024.
- [15] Ahmed Shokry and Moustafa Youssef. A deployable quantum access points selection algorithm for large-scale localization, 2024.
- [16] F. Tacchino, C. Macchiavello, D. Gerace, and D. Bajoni. An artificial neuron implemented on an actual quantum processor. *npj quantum information*, 2019.
- [17] Vivek V Shende, Stephen S Bullock, and Igor L Markov. Synthesis of Quantum Logic Circuits. *IEEE Transactions* on Computer-Aided Design of Integrated Circuits and Systems, 25(6):1000–1010, 2006.
- [18] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge

University Press, USA, 10th edition, 2011.

- [19] Mario E Rivero-Angeles. Quantum-based wireless sensor networks: A review and open questions. *International Journal of Distributed Sensor Networks*, 17(10):15501477211052210, 2021.
- [20] Vittorio Giovannetti, Seth Lloyd, and Lorenzo Maccone. Quantum Random Access Memory. *Phys. Rev. Lett.*, 100:160501, Apr 2008.
- [21] Joaquín Torres-Sospedra, Raúl Montoliu, Adolfo Martínez-Usó, Joan P. Avariento, Tomás J. Arnau, Mauri Benedito-Bordonau, and Joaquín Huerta. UJIIndoorLoc: A new multi-building and multi-floor database for WLAN fingerprint-based indoor localization problems. In 2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN), pages 261–270, 2014.
- [22] IBM Quantum. https://quantum-computing.ibm.com/. 19, 2021.