Formal Verification of Quantum SWAP Test Using Weakest Precondition Logic and Finite-Sample Performance Bounds for Indoor Localization

Marc Jermaine Pontiveros[1*] and Henry N Adorna[1]

Department of Computer Science, University of the Philippines Diliman, Quezon City 1103, Philippines
*Corresponding author: mpontiveros@up.edu.ph
hnadorna@up.edu.ph

Abstract. This paper presents a practical framework for quantum indoor localization by focusing on the implementation details of its core subroutine, the SWAP test. The work integrates three key components: formal methods, statistical analysis, and simulation. First, we demonstrate the utility of modern verification tools by formally proving the correctness of the SWAP test using a Hoare-based quantum program logic, confirming that an acceptance outcome correctly projects the input states onto the symmetric subspace. Second, building on this verified foundation, we derive explicit finite-sample bounds on the number of measurements (N) needed to estimate state overlap within an additive error ϵ with failure probability δ. By applying Hoeffding's inequality, this analysis provides practical guidance on measurement complexity, quantifying a critical parameter for experimental implementations. Finally, to illustrate the end-to-end workflow, we introduce a web-based simulation that encodes Received Signal Strength (RSS) data into quantum states and uses a sequential SWAP test comparison for location identification. Our integrated approach provides a tangible roadmap for implementing quantum fingerprinting subroutines, affirming the feasibility of the core components while transparently outlining the practical challenges and resource requirements that guide future work.

Keywords: Quantum Fingerprinting, SWAP Test, Indoor Localization, Formal Verification, Measurement Bounds, Weakest Precondition Logic

1 Introduction

Quantum fingerprinting leverages quantum states to create compact, efficiently comparable representations of classical data [1] [12]. A cornerstone of this approach is the SWAP test, a quantum subroutine that estimates the similarity—or the squared overlap—between two states, $|\psi_1\rangle$ and $|\psi_2\rangle$. The probability of an *accept* outcome, corresponding to a projection onto the symmetric subspace, is

J. Caro et al. (eds.), *Proceedings of the Workshop on Computation: Theory and Practice (WCTP 2025)*, Atlantis Highlights in Computer Sciences 24,
https://doi.org/10.2991/978-94-6239-638-8_3

given by:

$$p_{\text{acc}} = \frac{1 + |\langle\psi_1|\psi_2\rangle|^2}{2} \tag{1}$$

Due to its fundamental role, the SWAP test is a key component in numerous quantum algorithms, from machine learning to communication protocols [3] [5] [10].

A promising application is quantum indoor localization, where Received Signal Strength (RSS) vectors are encoded as quantum fingerprints [8]. The SWAP test provides an efficient subroutine for comparing a user's current RSS state against a single database entry. While this subroutine can be embedded within larger quantum search algorithms to achieve an exponential speedup over classical linear search, potentially, this paper focuses on a different, more foundational problem: the practical implementation of the comparison step itself. A critical detail often overlooked in theoretical proposals is that the circuit's probabilistic nature necessitates multiple runs to confidently estimate the overlap. Without a rigorous analysis, determining the required number of trials, or *shots*, is left to guesswork, risking either insufficient accuracy or unnecessary computational cost.

This paper provides a practical implementation roadmap by focusing on three key aspects of the SWAP test subroutine. Our contributions are as follows:

1. **Formal Verification:** While the SWAP test is well-understood, we demonstrate the capability of modern formal methods by providing a proof of its correctness using the quantum program verification tool by Feng and Xu [2]. We rigorously verify that an *accept* outcome correctly corresponds to the projector $E_{acc} = \frac{1}{2}(I + SWAP)$, showcasing the application of these tools to foundational quantum circuits.
2. **Finite-Sample Analysis:** We treat each circuit run as a Bernoulli trial and apply concentration inequalities to derive an explicit formula for the minimum number of shots, $N(\varepsilon, \delta)$, required to estimate the overlap within an error ε with failure probability δ. This analysis provides concrete, finite-shot estimates for implementing the quantum localization subroutine, quantifying a critical resource for practical deployment.
3. **Simulation Prototype:** We develop a web-based simulation of the end-to-end localization process. Our prototype encodes 4-dimensional RSS vectors into 2-qubit states. It uses the SWAP test for pairwise comparison, serving as both a proof-of-concept and an educational tool that bridges the gap between the theoretical subroutine and its practical application. The source code is made available in [6].

2 Quantum Fingerprint Matching Protocol

The indoor localization problem can be modeled as a fingerprint matching task. A user's location is found by comparing a newly measured Received Signal Strength (RSS) vector (the *test fingerprint*) against a pre-compiled database

of RSS vectors from known locations. Classically, this requires computing numerous high-dimensional vector distances. The quantum approach provides an efficient subroutine for this comparison by encoding the classical vectors into quantum states and estimating their similarity using the SWAP test.

The protocol consists of three main phases:

1. **State Preparation:** Classical RSS fingerprints are encoded into normalized quantum states using amplitude encoding, a standard technique for representing classical data in quantum systems [9].
2. **State Comparison:** The SWAP test circuit (Fig. 1) is used to estimate the overlap, or squared inner product $|\langle\psi_1|\psi_2\rangle|^2$, between the test state $|\psi_2\rangle$ and a database state $|\psi_1\rangle$.
3. **Decision:** The estimated overlap, which indicates similarity, is used to identify the most likely location. For our prototype, we sequentially test against each candidate location and select the one with the highest overlap.

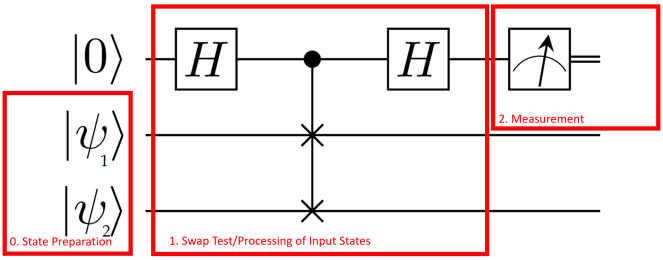


Fig. 1: The quantum fingerprint matching protocol. It involves (1) State preparation to encode classical data into quantum states $|\psi_1\rangle$ and $|\psi_2\rangle$, (2) Processing via the SWAP test, and (3) Measurement of the ancilla qubit to estimate the state overlap.

2.1 The SWAP Test: Circuit and Theory

The core of the quantum comparison is the SWAP test, which uses an ancillary qubit to mediate an interaction between two quantum states $|\psi_1\rangle$ and $|\psi_2\rangle$. The circuit, shown in Fig. 1, evolves the system as follows.

The initial state of the system, comprising an ancilla initialized to $|0\rangle$ and the two fingerprint states, is $|0\rangle|\psi_1\rangle|\psi_2\rangle$.

1. A Hadamard gate is applied to the ancilla, creating a superposition:

$$\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)|\psi_1\rangle|\psi_2\rangle \tag{2}$$

2. A Controlled-SWAP (CSWAP) gate, also known as a Fredkin gate, is applied. This gate swaps $|\psi_1\rangle$ and $|\psi_2\rangle$ if and only if the ancilla is in the state $|1\rangle$. The system state becomes:

$$\frac{1}{\sqrt{2}}(|0\rangle|\psi_1\rangle|\psi_2\rangle + |1\rangle|\psi_2\rangle|\psi_1\rangle) \tag{3}$$

3. A second Hadamard gate is applied to the ancilla, transforming the state into:

$$\frac{1}{2}\big(|0\rangle(|\psi_1\rangle|\psi_2\rangle + |\psi_2\rangle|\psi_1\rangle) + |1\rangle(|\psi_1\rangle|\psi_2\rangle - |\psi_2\rangle|\psi_1\rangle)\big) \tag{4}$$

The final step is to measure the ancilla qubit. The terms can be understood using the SWAP operator, which acts as $SWAP(|i\rangle|j\rangle) = |j\rangle|i\rangle$. The projectors onto the symmetric and anti-symmetric subspaces of the combined state $|\psi_1\rangle|\psi_2\rangle$ are $P_{\text{sym}} = \frac{I+SWAP}{2}$ and $P_{\text{antisym}} = \frac{I-SWAP}{2}$, respectively.

The final state can be rewritten as:

$$|0\rangle P_{\text{sym}}(|\psi_1\rangle|\psi_2\rangle) + |1\rangle P_{\text{antisym}}(|\psi_1\rangle|\psi_2\rangle) \tag{5}$$

Measuring the ancilla in the state $|0\rangle$ (an "accept" outcome) corresponds to projecting the two states onto the symmetric subspace. The probability of this outcome is directly related to the squared inner product of the states:

$$p_{\text{acc}} = P(0) = \langle\psi_1\psi_2|P_{\text{sym}}|\psi_1\psi_2\rangle = \frac{1 + |\langle\psi_1|\psi_2\rangle|^2}{2} \tag{6}$$

2.2 Implementation and Application

State Preparation via Amplitude Encoding To implement the protocol, classical RSS vectors must first be encoded into quantum states. Given a real-valued vector $\mathbf{v} = (x_0, x_1, \ldots, x_{n-1})^\top$, we first normalize it such that $||\mathbf{v}||_2 = 1$. Then, using amplitude encoding, we prepare a state in an n-dimensional Hilbert space (requiring $N_{\text{qubits}} = \lceil \log_2(n) \rceil$ qubits):

$$|\psi_{\mathbf{v}}\rangle = \sum_{i=0}^{n-1} x_i |i\rangle \tag{7}$$

For example, a normalized 2-dimensional vector $\mathbf{v}_1 = (0.39, 0.92)^\top$ is encoded as the single-qubit state $|\psi_1\rangle = 0.39|0\rangle + 0.92|1\rangle$.

Measurement and Overlap Estimation A single run of the SWAP test yields a binary outcome (0 or 1). To estimate p_{acc}, the circuit is executed N_{shots} times. Let N_{accept} be the number of times the ancilla is measured as 0. The overlap can then be estimated by rearranging Eq. (6):

$$|\langle\psi_1|\psi_2\rangle|^2 \approx \frac{2N_{\text{accept}}}{N_{\text{shots}}} - 1 \tag{8}$$

The location corresponding to the database fingerprint $|\psi_1\rangle$ that yields the highest estimated overlap with the test fingerprint $|\psi_2\rangle$ is chosen as the user's location.

Extension to Multi-Qubit States The protocol naturally generalizes to higher-dimensional vectors. A classical vector of dimension n is encoded into a quantum state using $N_{\text{qubits}} = \lceil \log_2(n) \rceil$ qubits, as defined in Eq. (7). For two such states, $|\psi_1\rangle$ and $|\psi_2\rangle$, the SWAP test is implemented by applying a CSWAP gate between each corresponding pair of qubits in their respective registers, all controlled by the same shared ancilla. This procedure is outlined in Algorithm 1.

Algorithm 1 Quantum Fingerprint Matching Subroutine

Input: Two N_{qubits}-qubit registers ψ_1, ψ_2; an ancilla qubit a; number of shots N_{shots}.
Output: Estimated squared overlap $|\langle\psi_1|\psi_2\rangle|^2$.

Initialize $N_{\text{accept}} = 0$.
for $j \leftarrow 1$ to N_{shots} **do**
    Prepare ancilla a in $|0\rangle$.
    Apply Hadamard gate to a.
    for $i \leftarrow 0$ to $N_{\text{qubits}} - 1$ **do**
        Apply CSWAP$(a, \psi_{1i}, \psi_{2i})$.
    end for
    Apply Hadamard gate to a.
    Measure qubit a.
    if outcome is 0 **then**
        $N_{\text{accept}} \leftarrow N_{\text{accept}} + 1$.
    end if
end for
return $(2 \times N_{\text{accept}}/N_{\text{shots}}) - 1$.

3 Formal Verification of the SWAP Test using Weakest Precondition Logic

While the SWAP test's correctness is well-established analytically, this section provides a formal verification to validate its acceptance condition within a rigorous proof framework. The primary goal of this exercise is to showcase the capability of modern formal methods for quantum program verification, rather than to prove a new result. To illustrate, we employ the proof assistant from *Verification of Nondeterministic Quantum Programs* by Feng and Xu [2], which implements a Hoare-style logic for quantum programs. The tool, *NQPV*, allows us to define a quantum program S, specify a desired postcondition Q, and automatically compute the weakest precondition $wp(S, Q)$ that must hold on an initial state for that postcondition to be satisfied.

3.1 Verification Setup

We model the SWAP test as a program acting on an ancilla qubit a and two data registers, x and y. The NQPV pseudocode below shows the setup for the

case where x and y are two-qubit registers (composed of qubits x_0, x_1 and y_0, y_1, respectively).

```
def pf_swap_2qubit := proof [a x0 x1 y0 y1] :
    {{ Zero[a] I[x0] I[x1] I[y0] I[y1] }};

    a *= H;

    // CSWAP(a; x0 <-> y0)
    [a x0 y0] *= CCX;
    [a y0 x0] *= CCX;
    [a x0 y0] *= CCX;

    // CSWAP(a; x1 <-> y1)
    [a x1 y1] *= CCX;
    [a y1 x1] *= CCX;
    [a x1 y1] *= CCX;

    a *= H;

    {{ P0[a] }}
end

show pf_swap_2qubit end
```

The program is written using a Hoare Logic framework, as $\{\{P\}\}S\{\{Q\}\}$, where P is the precondition, S is the quantum program, and Q is the postcondition. NQPV uses weakest precondition calculus to work backward from the postcondition {{P0[a]}} (i.e., the ancilla is measured as 0) to compute the necessary condition on the initial state of the data registers.

3.2 Verification Results

Using the NQPV assistant, we computed the weakest precondition for the program with respect to the postcondition that the ancilla is measured in the $|0\rangle$ state. The resulting operator on the Hilbert space of the data registers (x and y) was:

$$WP_{\text{SWAP-test}}(P_0[a]) = \frac{I_{xy} + SWAP_{xy}}{2} \tag{9}$$

This result precisely matches the theoretical symmetric subspace projector P_{sym}. This formally shows two related things: first, that the measurement operator (POVM element) corresponding to an "accept" outcome is indeed P_{sym}, and second, that the weakest precondition for guaranteeing an "accept" outcome with probability 1 is that the initial joint state must lie entirely within that symmetric subspace. For a general state, the acceptance remains probabilistic, governed by the expectation value $\langle\psi|P_{\text{sym}}|\psi\rangle$.

This verification was successfully performed for both single-qubit data registers (a 2-dimension scenario) and the two-qubit registers (a 4-dimension scenario, shown above). Both cases were handled automatically by the NQPV tool, although the larger state space required tighter numerical tolerances in the semidefinite programming solver, with a maximal deviation on the order of 10^{-16}. This result demonstrates that formal verification tools have matured to the point of handling non-trivial quantum circuits.

However, it is important to consider the scalability of this approach. While NQPV was effective for these small, few-qubit circuits, formally verifying quantum programs with significantly more qubits or greater circuit depth remains a formidable challenge due to the exponential growth of the state space. The techniques used here, based on semidefinite programming, face scaling limitations. Therefore, while formal methods provide the highest level of assurance for foundational subroutines like the SWAP test, developing scalable verification techniques for larger, more complex quantum algorithms is a critical and active area of ongoing research.

4 Finite-Sample Analysis: SWAP Test Shot Complexity

4.1 Bounding the Overlap Error

Each run of the SWAP test is a Bernoulli trial yielding an "accept" with probability p_{acc}. After N shots, we obtain an estimate $\hat{p} = N_{\text{accept}}/N$. To bound the error of this estimate, we can use Hoeffding's inequality:

$$P(|\hat{p} - p_{\text{acc}}| \geq \varepsilon_p) \leq 2e^{-2N\varepsilon_p^2} \tag{10}$$

However, the quantity of practical interest is not p_{acc} but the squared overlap, $F = |\langle\psi_1|\psi_2\rangle|^2$. These are related by $F = 2p_{\text{acc}} - 1$. This linear relationship means that an error ε_p in estimating p_{acc} propagates to an error of $\varepsilon_F = 2\varepsilon_p$ in the estimate of F.

To provide a guarantee on the overlap error ε_F, we must ensure the probability error ε_p is bounded by $\varepsilon_F/2$. Substituting $\varepsilon_p = \varepsilon_F/2$ into Eq. (10) gives us the required number of shots. This leads to our first main result.

Theorem 1 (Shot Bound for Pairwise Overlap Estimation). *To estimate the squared overlap $F = |\langle\psi_1|\psi_2\rangle|^2$ to within an additive error ε_F with confidence $1 - \delta$, the number of shots N must satisfy:*

$$N \geq \frac{2}{\varepsilon_F^2} \ln\left(\frac{2}{\delta}\right) \tag{11}$$

4.2 Practical Application: Distinguishing Two Locations

The bound in Theorem 1 allows us to calculate the resources needed to distinguish two locations. Suppose we are choosing between a correct location L_A and an incorrect one L_B. Their true squared overlaps with the test state are F_A and

F_B, respectively. To reliably distinguish them, our estimation error ε_F must be less than half the gap between them, i.e., $\varepsilon_F < |F_A - F_B|/2$.

Example: A test state is compared to two database fingerprints. The true squared overlap with Location A is $F_A = 0.8$ (corresponding to $p_A = 0.9$) and with Location B is $F_B = 0.4$ (corresponding to $p_B = 0.7$).

- The overlap gap is $\Delta F = |0.8 - 0.4| = 0.4$.
- We require an overlap error $\varepsilon_F < \Delta F/2 = 0.2$. Let's set a target error of $\varepsilon_F = 0.1$ for a safety margin.
- We desire 99% confidence, so the failure probability is $\delta = 0.01$.
- Using Eq. (11), the minimum shots needed for *each* comparison is:

$$N \geq \frac{2}{(0.1)^2} \ln\left(\frac{2}{0.01}\right) = 200 \ln(200) \approx 200 \times 5.298 \approx 1060$$

Table 1: Minimum shots (N) for a given confidence ($1 - \delta$) and error tolerance. The error on the acceptance probability (ε_p) and the corresponding error on the squared overlap ($\varepsilon_F = 2\varepsilon_p$) are shown. All values are rounded up.

Confidence Level $(1-\delta)$	**Failure Prob.** (δ)	**Error on** p_{acc} (ε_p)	**Error on Overlap** (F) (ε_F)	**Required Shots** (N)
90%	0.1	0.10	0.20	150
		0.05	0.10	600
95%	0.05	0.10	0.20	185
		0.05	0.10	738
99%	0.01	0.10	0.20	265
		0.05	0.10	1060

4.3 Extension: Distinguishing Multiple Locations

In practice, we must identify the best match from a database of K candidate locations. This is a multi-hypothesis testing problem. To guarantee that we select the correct location with an overall confidence of $1 - \delta$, we must account for the increased probability of error. Using the union bound (Bonferroni correction), we can bound the total failure probability by ensuring the failure probability of each of the $K - 1$ pairwise comparisons against the true best match is at most $\delta' = \delta/(K - 1)$.

Corollary 1 (Shot Bound for K-Way Selection). *To correctly identify the location with the highest squared overlap from a set of K candidates, where the minimum gap between the best and second-best overlap is ΔF, the number of shots N for each comparison must satisfy:*

$$N \geq \frac{2}{(\Delta F/2)^2} \ln\left(\frac{2(K-1)}{\delta}\right) = \frac{8}{(\Delta F)^2} \ln\left(\frac{2(K-1)}{\delta}\right) \tag{12}$$

This shows that the required number of shots grows logarithmically with the number of locations in the database, a crucial factor for resource planning.

4.4 Discussion: Tighter Bounds and Idealized Assumptions

The analysis so far relies on Hoeffding's inequality, which is general but often conservative because it does not use any information about the sample mean itself. Tighter confidence intervals, such as the Clopper-Pearson or Wilson score interval, can provide the same statistical guarantee with fewer shots, especially when p_{acc} is close to 0 or 1. For a more advanced analysis, concentration inequalities like the empirical Bernstein bound could offer further improvements by incorporating the sample variance.

Beyond the choice of inequality, it is critical to acknowledge that this entire analysis rests on idealized assumptions. The shot counts derived are theoretical lower bounds under noise-free conditions and assume that each shot is independent and identically distributed (i.i.d.). On real hardware, systematic errors from gate imperfections, SPAM, and decoherence will introduce bias, and temporal drift can violate the i.i.d. assumption. A robust implementation would require characterization of these effects and could potentially employ adaptive shot allocation strategies (e.g., successive elimination) to focus resources on distinguishing the most competitive candidate locations.

5 Indoor Localization with Quantum Fingerprinting: A Simulation

5.1 Overview of the RF Fingerprinting Method

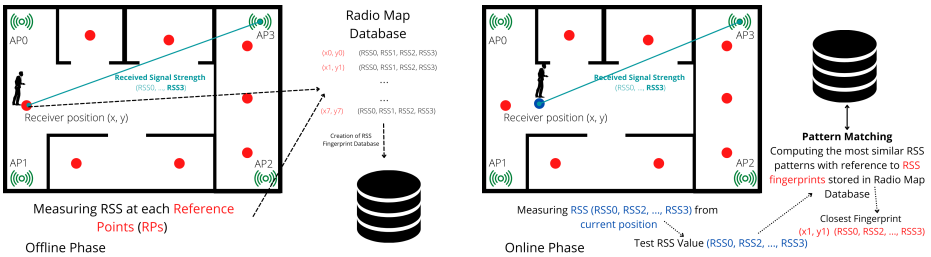
Radio Frequency (RF) fingerprinting for indoor localization operates in two phases: an offline preparation phase and an online matching phase (Fig. 2). The quantum approach adapts this framework by replacing the classical distance calculation with an efficient quantum similarity estimation using the SWAP test subroutine. Our work demonstrates this process through an interactive web-based simulation that allows for dynamic configuration and visualization of the entire workflow.

5.2 Offline Phase: Simulating the Radio Map

The offline phase builds a database of location fingerprints, or a "Radio Map." In our simulation, this database is generated dynamically for a floor plan with a fixed set of 7 RPs and 4 APs. The application calculates the RSS vector for each RP using the Standard Log-Distance Path Model:

$$\text{RSSI}(d) = A_{\text{ref}} - 10 \cdot n \cdot \log_{10}(d/d_0) \tag{13}$$

The model parameters—reference RSSI (A_{ref}), path loss exponent (n), and reference distance (d_0)—are user-configurable. To simulate real-world variability, users can also introduce Gaussian noise and apply signal averaging. We note that this path loss model is a common idealization and does not account for complex real-world effects such as multipath fading or physical obstructions, which are important considerations for a production system [7] [4].



(a) Offline phase: A radio map is built by recording RSS vectors at known reference points (RPs).

(b) Online phase: A user's current RSS vector is compared against the radio map to find the closest match.

Fig. 2: Conceptual overview of RF-Fingerprinting for Indoor Localization.

5.3 Online Phase: Quantum Location Determination

The online phase begins when a user's location is tested. The process is as follows:

1. **Measurement:** The application calculates the "test" RSS vector for the user's current location using the same configurable path loss model.
2. **Quantum State Preparation:** The test vector and each vector from the radio map are transformed into valid quantum states via two automated pre-processing steps:
   - **Padding:** Vectors are padded with zeros to ensure their dimension is a power of 2, as required by amplitude encoding.
   - **Normalization:** Each vector is normalized to have a Euclidean norm of 1, with its elements becoming the amplitudes of the quantum state.

   This process achieves significant data compression, encoding an n-dimensional vector into only $\lceil \log_2(n) \rceil$ qubits, a key feature of many quantum algorithms.
3. **Similarity Comparison:** The prototype performs a **sequential search**, using the SWAP test to compare the user's test state against each reference state from the radio map. The underlying circuits are built and run using Qiskit's high-performance Aer simulator. The number of shots for each comparison is chosen based on the finite-sample analysis in Section 4 to ensure a statistically reliable estimate.
4. **Location Estimation:** The reference point whose fingerprint yields the highest similarity score (i.e., highest "accept" probability) is identified as the user's most probable location.

5.4 The Web-Based Simulation Prototype

To provide a tangible proof-of-concept, we developed an interactive web application using Qiskit [11] and Streamlit (Fig. 3). The application serves as an end-to-end demonstration, designed to bridge the gap between the abstract theory of the SWAP test and its practical application.

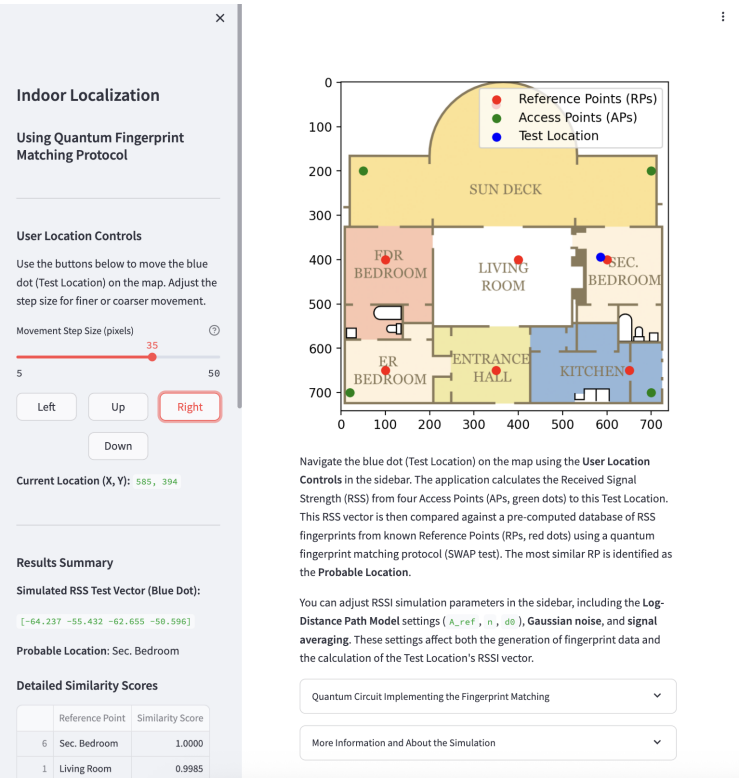
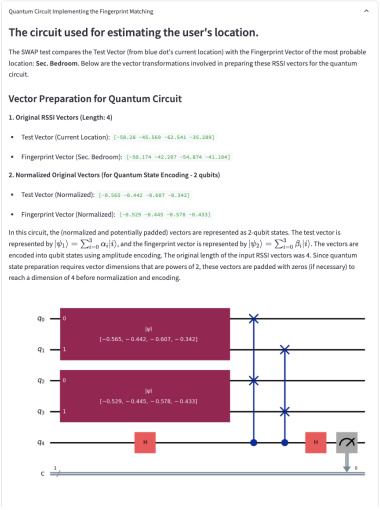
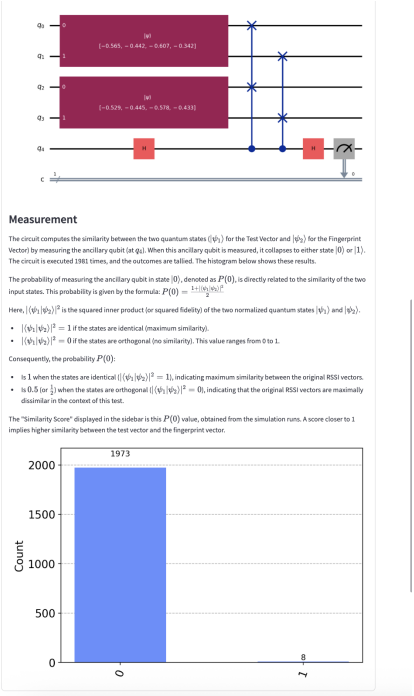


Fig. 3: The main interface of the web-based simulation prototype. The central map displays the APs (green), RPs (red), and the user's test location (blue). The sidebar contains controls for navigation and simulation parameters.



(a) Pre-processing of RSSI vectors before quantum encoding.



(b) The Qiskit quantum circuit and resulting measurement histogram.

Fig. 4: Visualization of the quantum components in the simulation prototype, showing (a) the vector pre-processing steps and (b) the final circuit and measurement results for a comparison.

The prototype's core is its configurable simulation engine. Rather than using static data, RSSI values are calculated dynamically, allowing users to model different environments by adjusting path loss parameters, noise levels, and signal averaging. A central focus is explainability. The sidebar provides a continuously updated table of similarity scores, giving a clear ranking of potential locations. An expandable interface (Fig. 4a) details the full vector pre-processing pipeline from raw RSSI values to final normalized amplitudes. Finally, the application renders the specific Qiskit circuit and resulting measurement histogram for the best-matching comparison (Fig. 4b), linking theory, simulation, and outcome transparently.

6 Numerical Experiments

To validate our theoretical analysis and assess the practical performance of the SWAP test-based classifier, we conducted a series of numerical experiments using our simulation framework. The goal was to measure the empirical top-1 localization accuracy as a function of the number of shots (N) and to compare it against our theoretical bounds and standard classical baselines.

6.1 Experimental Setup

The experiment was run on the simulated environment described in Section 5 ($K = 7$ reference points), with a fixed random seed to ensure reproducibility. For each shot count N on the x-axis, we performed 50 trials for each of the 7 reference points, for a total of 350 independent trials per data point. In each trial, a test RSSI vector was generated by adding Gaussian noise with a standard deviation of $\sigma = 5.0$ dBm to a ground-truth signal. This noisy vector was then classified using three methods:

1. **Quantum SWAP Test Circuit:** The test vector was compared against all K ground-truth reference vectors using the SWAP test for various shot counts (N). The location with the highest estimated squared overlap was chosen.
2. **Classical Baseline 1 (Cosine Similarity):** The test vector was compared against reference vectors using cosine similarity.
3. **Classical Baseline 2 (Euclidean Distance):** The test vector was compared against reference vectors using Euclidean distance on the raw (unnormalized) RSSI vectors.

For a fair comparison, the quantum and cosine similarity methods used the exact same vector pre-processing steps (padding and normalization). The shaded error bands on the subsequent plot represent the 95% confidence interval for the binomial accuracy proportion.

6.2 Results and Discussion

The results of the experiment are shown in Fig. 5. The plot demonstrates a clear relationship between measurement shots and classification accuracy. As predicted, the Quantum SWAP Test Circuit's performance improves with N, successfully overcoming the simulated signal noise. With a sufficient number of shots ($N > 1500$), its accuracy is competitive with the cosine similarity baseline, but remains slightly below the top-performing Euclidean distance baseline.

This result is consistent with our noise-free theoretical framework. For the specific geometry of our radio map, the empirical minimum overlap gap was calculated to be $\Delta F \approx 0.0837$. Using Corollary 1 with $K = 7$ and $\delta = 0.01$, the theoretical shot count required to guarantee 99% classification confidence is $N \approx 8,100$. This large number correctly predicts that this map configuration is a "hard" problem, with at least two reference points that are very difficult to distinguish. Because this value lies far beyond our 2,000-shot sweep, it is omitted from the figure, but its calculation explains why the quantum method's accuracy had not yet surpassed the classical baselines. This demonstrates our formula's utility as a tool for assessing problem difficulty a priori.

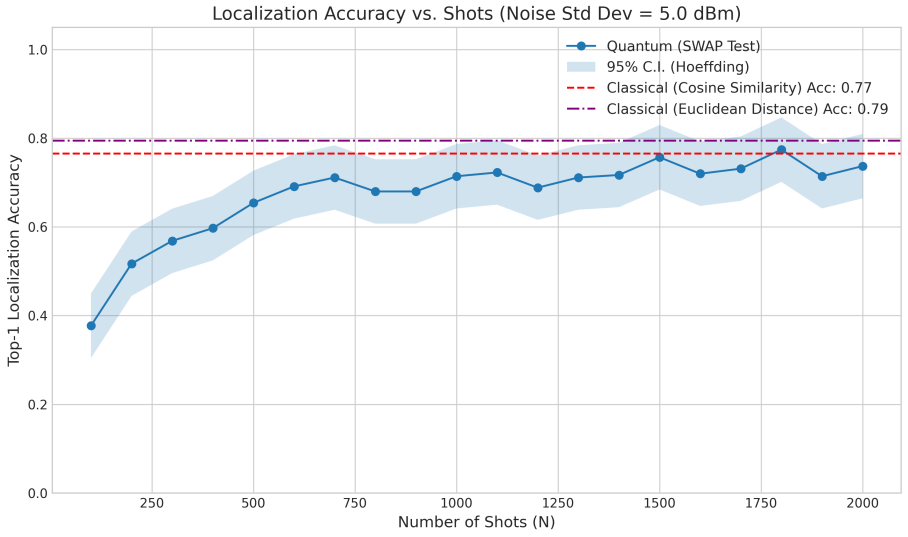


Fig. 5: Empirical localization accuracy vs. number of shots (N). The shaded area shows the 95% confidence interval over 350 trials per data point. The Quantum SWAP Test Circuit's performance is compared against two classical baselines.

7 Conclusion and Future Work

In this work, we presented a practical roadmap for implementing and analyzing the SWAP test subroutine for quantum indoor localization. Our integrated approach combined a formal verification of the circuit's correctness, a statistical analysis to quantify the necessary measurement shots, and a proof-of-concept simulation. By verifying the SWAP test with weakest precondition logic and deriving finite-sample bounds for its execution, we established a rigorous foundation for its practical use. The web-based prototype demonstrated the end-to-end workflow of a sequential-search localization protocol, showcasing the feasibility of its core quantum components in a simulated four-AP environment.

The primary limitations of this study are rooted in its idealized models. The statistical analysis assumes a perfect, noise-free quantum device, providing a theoretical lower bound on measurement costs. Similarly, the localization prototype uses a simplified log-distance model for RSS values, which does not capture the complexities of real-world indoor environments.

Future work should focus on bridging this gap between theory and practice. Key directions include:

1. **Execution on Real Hardware:** A significant next step is to execute the protocol on current quantum computers, such as those available through IBM Quantum Experience. This would involve confronting the challenges of gate errors, SPAM errors, and decoherence, necessitating the use of quantum error mitigation techniques to improve result fidelity.
2. **Achieving Quantum Speedup:** To realize the full potential of quantum fingerprinting, the SWAP test subroutine should be integrated into a larger quantum search algorithm (e.g., based on amplitude amplification). This would be a crucial step toward achieving a true end-to-end algorithmic speedup over classical methods.
3. **Enhancing Environmental Models:** The simulation could be improved by incorporating more sophisticated radio propagation models that account for physical obstacles and multipath fading, leading to more realistic performance estimates.

By addressing these areas, the foundational work presented here can be extended toward a truly practical and advantageous quantum solution for indoor localization.

References

1. Buhrman, H., Cleve, R., Watrous, J., de Wolf, R.: Quantum fingerprinting. Physical Review Letters **87**, 167,902 (2001)
2. Feng, Y., Xu, Y.: Verification of nondeterministic quantum programs. In: Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS 2023), pp. 18:1–18:18 (2023)

3. Harrow, A.W., Hassidim, A., Lloyd, S.: Quantum algorithm for linear systems of equations. Physical Review Letters **103**(15) (2009). [Online]. Available: https://doi.org/10.1103/physrevlett.103.150502
4. Kunhoth, J., Karkar, A., Al-Maadeed, S., Al-Ali, A.: Indoor positioning and wayfinding systems: a survey. Human-centric Computing and Information Sciences **10**(1) (2020). [Online]. Available: https://doi.org/10.1186/s13673-020-00222-0
5. Nishimura, H.: A survey: SWAP test and its applications to quantum complexity theory. In: Algorithmic Foundations for Social Advancement, pp. 243–257. Springer (2025). Forthcoming
6. Pontiveros, M.J.: Indoor localization using fingerprint matching in qiskit. [Online] (2022). Available: https://github.com/marcjermaine-pontiveros/CS297-IQA-Demo
7. Prabakar, A.: RSSI-based accurate indoor localization scheme for wireless sensor networks. [Online] (2015). Available: https://www.youtube.com/watch?v=CWvRJdF7oVE&t=199s
8. Shokry, A., Youssef, M.: Quantum computing for location determination. In: Proceedings of the 29th International Conference on Advances in Geographic Information Systems (ACM SIGSPATIAL 2021) (2021)
9. Weigold, M., Barzen, J., Leymann, F., Salm, M.: Expanding data encoding patterns for quantum algorithms. In: 2021 IEEE 18th International Conference on Software Architecture Companion (ICSA-C). IEEE (2021). [Online]. Available: https://doi.org/10.1109/icsa-c52384.2021.00025
10. Wiebe, N., Kapoor, A., Svore, K.M.: Quantum algorithms for nearest-neighbor methods for supervised and unsupervised learning. Quantum Information and Computation **15**(3&4), 316–356 (2015). [Online]. Available: https://doi.org/10.26421/qic15.3-4-7
11. Wille, R., van Meter, R., Naveh, Y.: IBM's qiskit tool chain: Working with and developing for real quantum computers. In: 2019 Design, Automation & Test in Europe Conference & Exhibition (DATE). IEEE (2019). [Online]. Available: https://doi.org/10.23919/date.2019.8715261
12. de Wolf, R.: Quantum computing: Lecture notes. arXiv preprint (2022). Version from 2022