Quantum Walks and Monte Carlo Simulation

Archana Singh (archanasingh133@gmail.com)

1 Quantum Galton Board (QGB)

The Quantum Galton Board (QGB) simulates quantum scattering over layers of entangled pegs [1]. Each layer uses a control qubit to redistribute amplitude through CSWAP and CNOT operations.

Initialization

The Quantum Galton Board (QGB) circuit initializes a "ball" at the center of a line of qubits and simulates its coherent scattering through a binary tree-like structure using quantum gates.

Let the total number of qubits be n and the number of levels (layers) be d, such that $n = 2d + 2$. The initial state of the system places the ball at the middle qubit (position mid $= \lfloor n/2 \rfloor$), with all other qubits in $|0\rangle$:

$$|\psi_0\rangle = |0\rangle^{\otimes n}, \quad \text{then} \quad X_{\text{mid}} |\psi_0\rangle$$

Additionally, the control qubit (say, qubit 0) is placed in the state $|+\rangle$.

Quantum Peg Transformation

Each level applies a "quantum peg" unit, which redistributes the ball left and right via controlled-SWAP operations. For level i, the peg acts on three working qubits $(q_{l_i}, q_{m_i}, q_{r_i})$, where:

$$l_i = \text{mid} - i, \quad m_i = \text{mid}, \quad r_i = \text{mid} + i$$

The quantum peg applies the following unitary:

$$U_{\text{peg}}^{(i)} = \text{CSWAP}(c, l_i, m_i) \cdot \text{CNOT}(m_i \to c) \cdot \text{CSWAP}(c, m_i, r_i)$$

This routing operation transfers amplitudes symmetrically (or asymmetrically if biased gates like $R_x(\theta)$ are used) to the left and right positions, creating a quantum superposition of paths.

Qubit Selection Strategy for Pegs: In practice, for each level j beyond the first, pegs are applied at multiple positions across the qubit line. If mid is the center qubit, the peg targets are determined as follows:

- Let $\text{mid}_j = \text{mid} - j$
- Iterate i from 0 to j
- At each step, apply the peg on qubit $\text{mid}_j + 2i$ (skip one qubit between pegs to avoid overlap)

This creates $j + 1$ pegs per level, symmetrically spreading the walk.

Coherent Evolution and Measurement

After d such layers, the final quantum state is a superposition over many paths, where each path corresponds to a binary outcome based on the sequence of SWAPs taken: $|\psi_{\text{final}}\rangle = \sum_i \alpha_i |x_i\rangle$,

where $|x_i\rangle$ is a computational basis state with a '1' at the final position of the ball. Each α_i encodes the complex amplitude of the walk ending in position x_i.

The measurement collapses this to a single position, yielding a **discrete probability distribution** over the $n-1$ measured qubits.

Postprocessing: Measurements yield one-hot bitstrings. A block of 8 such values is summed: $X_i = \sum_{j=1}^{8} x_j^{(i)}, \quad x_j \in \{0, 1, \ldots, n\}$. Histogramming X_i produces a discrete approximation to a Gaussian.

2 Hadamard Quantum Walk

A discrete-time quantum walk is built from coin + position qubits [2]. Each step applies:

- A coin flip: $H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$
- A coin-controlled shift: increment if coin = 0, decrement if coin = 1

Controlled logic uses: CINC: Apply $\text{CCX}(\text{coin}, q_i, q_{i+1})$, CDEC: reverse

Initialization: Walker starts in position $x_0 = 2^{n-1}$ (center of range). Position register uses binary encoding.

Output: Final histogram after t steps shows bimodal interference pattern with faster-than-classical spreading.

3 Exponential-Step Quantum Walk

We implement a one-dimensional discrete-time coined walk whose step-length is geometrically distributed, yielding a (truncated) exponential distribution over positions. The circuit uses three registers: (i) a *position* register $\{q_0,\dots,q_{n-1}\}$ storing a one-hot "ball" (initialized at the center q_{mid}); (ii) a single *continue* ancilla $|c\rangle$ initialized to $|1\rangle$; and (iii) a bank of *coin* ancillas $\{|u_j\rangle\}_{j=1}^{L}$, one per walk level.

At level j, we prepare a biased coin via $|u_j\rangle \leftarrow R_y(2\alpha)\,|0\rangle\,, \qquad \Pr[u_j{=}1] = \sin^2\alpha =: r, \quad \Pr[u_j{=}0] = 1-r,$ so r is the probability to *continue*. If the coin yields 0, we flip the flag $|c\rangle$ to $|0\rangle$ (stop condition). Conditioned on $c{=}1$, we perform a right-shift of the one-hot position by exactly one site using a controlled SWAP chain across $(q_{\text{mid}},\dots,q_{n-1})$. Concretely, each controlled SWAP is realized without a native Fredkin using three Toffolis (CCX), ensuring compatibility with standard gate sets.

After L levels, the number of executed shifts K equals the count of consecutive "continue" outcomes before the first "stop", i.e., $\Pr[K=k] = (1-r)\,r^k, \qquad k \in \{0,1,\dots\}$, truncated by register boundaries and the chosen depth L. Since the measured position is $X = x_0 + K$, the marginal over positions follows a discrete exponential (geometric) law with mean $\mathbb{E}[K] = \frac{r}{1-r}$ and decay rate $\lambda \approx -\ln r$ (continuous analogue). Thus, the circuit converts per-level Bernoulli "continue/stop" decisions into an exponentially decaying step-length distribution, which appears as an exponential tail in the histogram of one-hot position measurements. No mid-circuit resets or classical control are required; the stop logic is maintained coherently by the flag qubit c.

4 Circuit Gate Counts, TVD, and Noise Model

Table 1 summarizes the total variation distance (TVD) with respect to the ideal output distribution, along with detailed gate counts, for the Hadamard walk, Gaussian Galton board, and exponential walk circuits. An additional column is provided for the TVD obtained when running the circuits under the noise model described below.

Noise model. To evaluate circuit robustness, we used a simple dephasing noise model implemented in Qiskit Aer. The model applies *phase damping* channels to simulate loss of quantum coherence without energy relaxation:

- **Single-qubit gates**: phase damping probability $p_1 = 0.01$ applied to all H, X, S_x, R_x, R_y, and R_z gates.
- **Two-qubit gates**: phase damping probability $p_2 = 0.02$ applied independently to each qubit of a CX gate.
- **Idle qubits**: optional phase damping probability $p_{\text{idle}} = 0.001$ applied to identity operations to mimic idle-time dephasing between gates.

This noise model was injected into the simulator so that the corresponding dephasing channel is applied immediately after each relevant gate operation, allowing the impact of realistic coherence loss to be quantified via the "TVD (noisy)" metric in Table 1.

Table 1: Gate counts and TVD (vs. ideal) for the three circuits. The "TVD (noisy)" column is to be filled after running the noisy simulations.

Circuit	TVD (ideal)[1]	TVD (noisy)	ccx	cswap	cx	h	x	ry	measure	reset	barrier	Total
Hadamard walk	0	0.5	20	0	10	5	11	0	3	0	0	49
Gaussian Galton board	0.4525	0.6087	0	20	16	4	1	0	9	3	1	54
Exponential walk	0.2245		90	0	6	0	14	6	12	0	6	134

Observation: Gaussian arises from symmetric walk; exponential requires biased dynamics.

References

[1] Mark Carney and Ben Varcoe, *Universal Statistical Simulator*, arXiv:2202.01735 [quant-ph], 2022.

[2] S. E. Venegas-Andraca, *Quantum random walks: an introductory overview*, Quantum Inf. Process. **11**, 1015–1106 (2012).