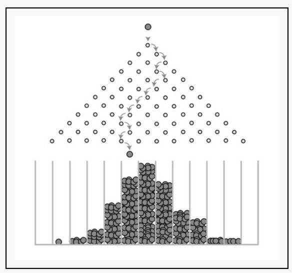
Quantum Galton Board

Monte Carlo Simulation via Quantum Circuits

Raouf Ould Ali & Rassim Cherir
WISER Quantum Project

Introduction

This work implements the quantum Galton board approach described in the Universal Statistical Simulator (Carney & Varcoe, 2022). We aim to generate a Quantum Galton Board (QGB), which is essentially a quantum circuit that imitates the behavior of a physical Galton board and allows us to generate statistical distributions.

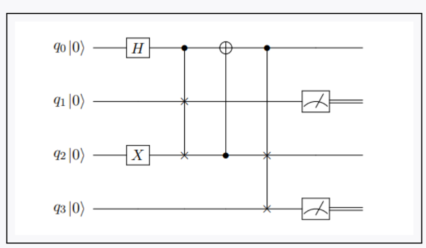


Constructing the Standard Board

In the standard construction of a Galton board, when a ball hits a peg, it has an equal probability of going either right or left, ultimately generating a normal distribution of outcomes at the last level, as illustrated. The probability of the ball falling into column k is:

$$\frac{1}{2^n}\binom{n}{k}$$

The proof is straightforward by induction on the board's levels.
To build the circuit for the QGB, we design a circuit for a single peg and reuse this block to construct the entire board.



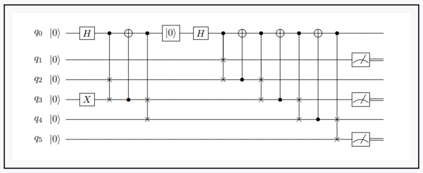
We mimic the ball by applying an X gate on qubit q_2, giving the state $|0100\rangle$ (where the '1' in position 2 represents the ball's presence). Applying a Hadamard gate then gives the superposition:

$$\frac{1}{\sqrt{2}}(|0100\rangle + |0101\rangle)$$

The state of the "control" qubit q_0 determines whether the ball goes right or left. After applying the $SWAP$ and $CNOT$ gates, we obtain the final state:

$$\frac{1}{\sqrt{2}}(|1001\rangle + |0011\rangle) = \frac{1}{\sqrt{2}}(|100\rangle + |001\rangle)\,|1\rangle$$

simulating the ball falling into qubit q_1 or q_3.
To generalize to the 3-peg construction (2 levels), we replicate the peg block three times: once on the first level and twice on the second. A $CNOT$ gate is added between the pegs of each level to put the control qubit into a superposition again, allowing us to simulate all outcomes simultaneously. Resetting the control qubit is needed once per level since each peg must act independently based on the previous result. However, within the same level, resetting is unnecessary as the ball cannot pass through two pegs on the same level.



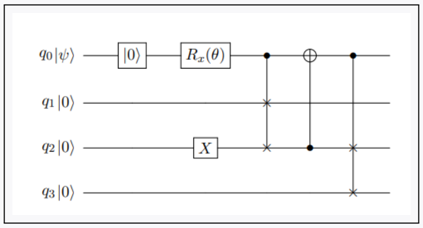
This approach requires $2n + 2$ qubits and $O(n^2)$ gates for an n-level QGB.

Experimental Results

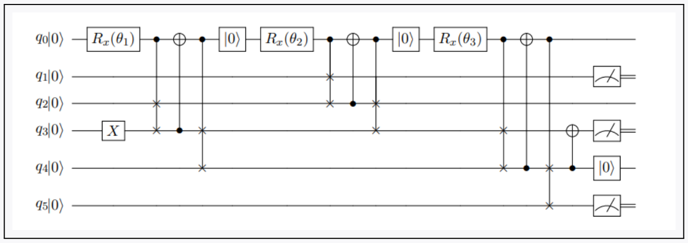
Due to the number of qubits and gates required, real QGB simulations are challenging. They have been performed on a single peg, producing noisy results but still showing peaks at the expected states. For larger QGBs, classical simulation techniques have been used, yielding results that closely match theoretical expectations.

Biased QGB

To obtain a biased peg, we simply replace the Hadamard gate with an $RX(\theta)$ gate, allowing us to choose the desired probability distribution. The $RX(\theta)$ gate provides probabilities $\cos^2(\theta/2)$ for the $|0\rangle$ state and $\sin^2(\theta/2)$ for the $|1\rangle$ state, enabling precise control over the bias.



Note that to control the bias of **every** peg, we must reset the control qubit once per peg, not just once per level.



The number of gates remains $O(n^2)$ in this case.