# Quantum Walks and Monte Carlo

WISER Quantum Projects 2025

Team Quantotto: Yevgeny Menaker

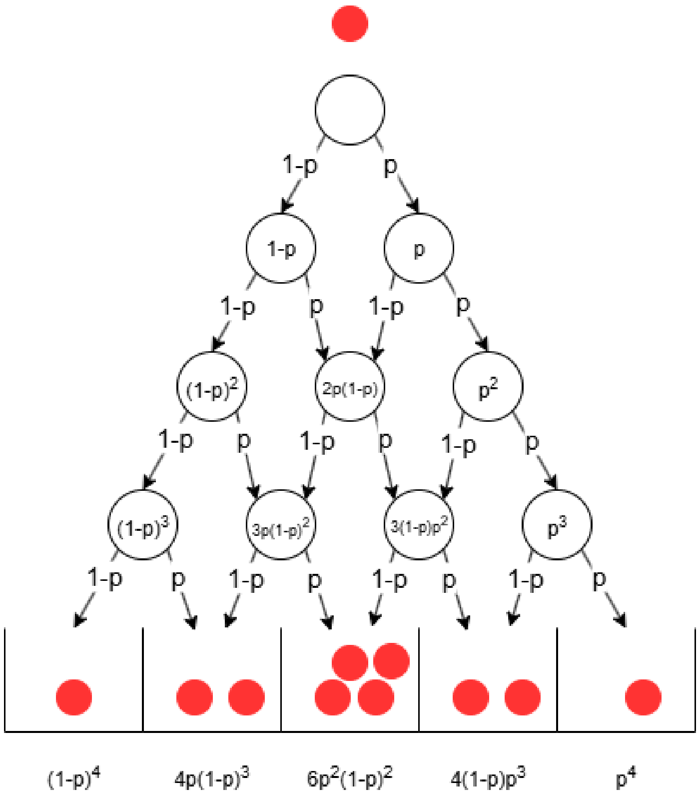
09 August 2025

# Project Goals

- Quantum Galton Board Implementation: Construct a quantum analogue of the n-level Galton Board to illustrate symmetric Gaussian distributions
- State Preparation: Apply the Universal Statistical Simulator method (from research paper) to generate a range of distributions as a superposition of quantum states
- Distribution Adjustment: Alter quantum pegs and coin parameters to produce various distributions, including Exponential and Hadamard Random Walk (Bi-Modal)
- Noise-Resilient Optimization: Design a Galton Board-inspired method to load arbitrary distributions and demonstrate outcomes on noisy simulators using the IBM ibm_torino noise model
- Performance and Depth Analysis: Evaluate circuit accuracy under noisy conditions and examine the relationship between circuit depth and the number of Galton Board levels
- Real Backend Runs: Execute Resulting Circuits on physical QPUs and observe the results

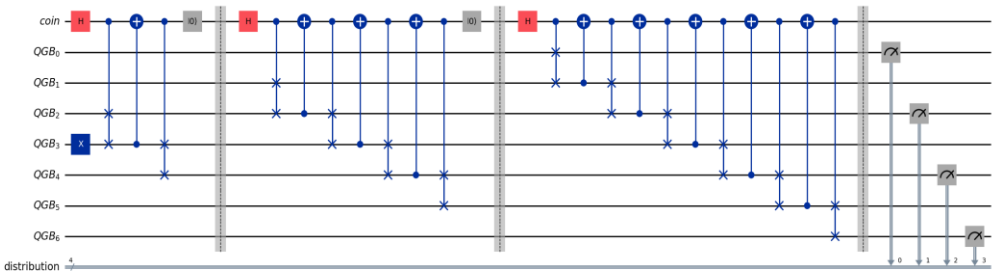
# Galton Board as Decision DAG

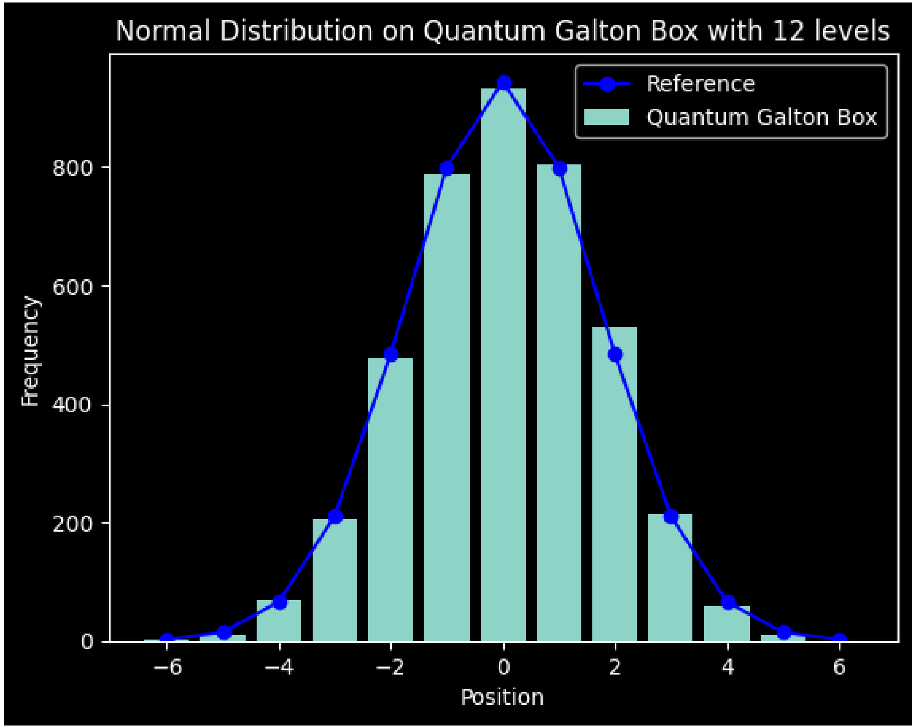
- Galton Board is a decision DAG (Directed Acyclic Graph)
- If p=1/2, a symmetric normal distribution is achieved in the tally bins
- Modifying the probability ( p ) will skew the distribution either to the left or right



# n-levels Galton Board: Quantum Circuit

Diagram for n=3:



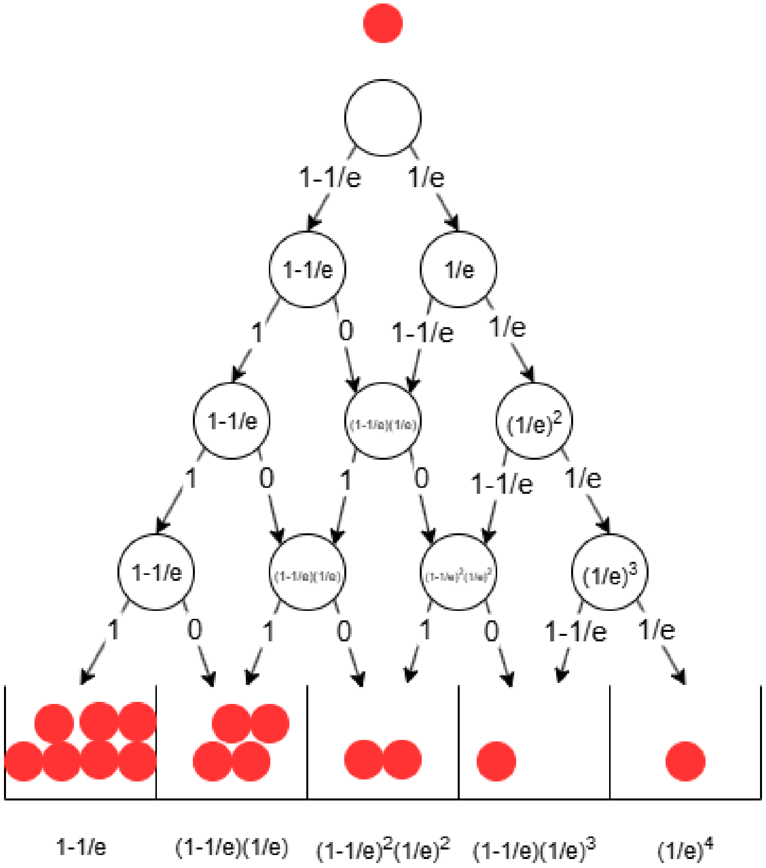




**<u>For n levels</u>**

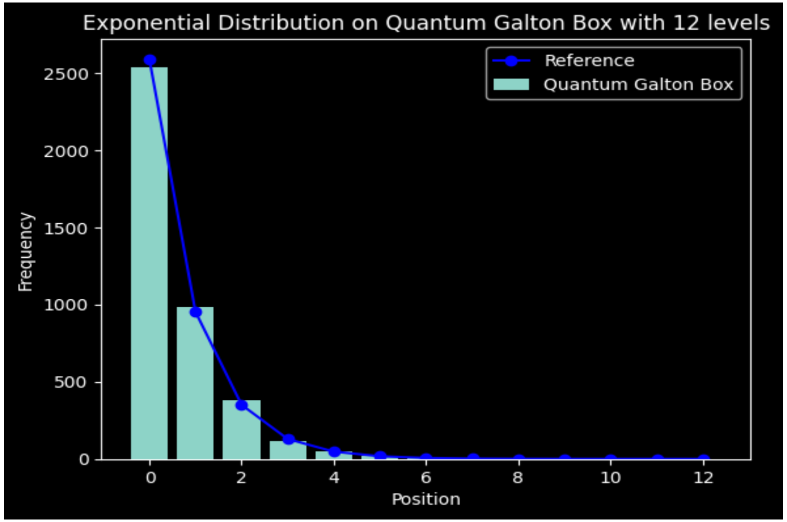
Qubits: 2n + 2

Depth: $4 + 8 + 12 + \ldots + 3n = 2(n^2 + n) = O(n^2)$

# Exponential Distribution

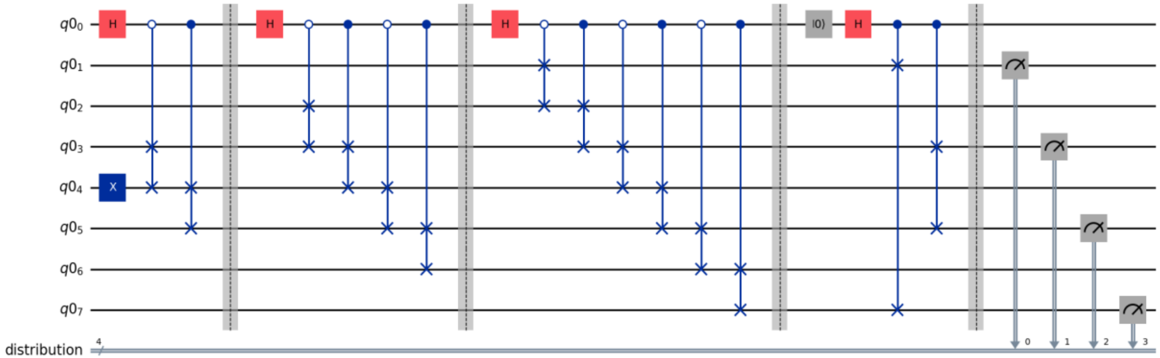


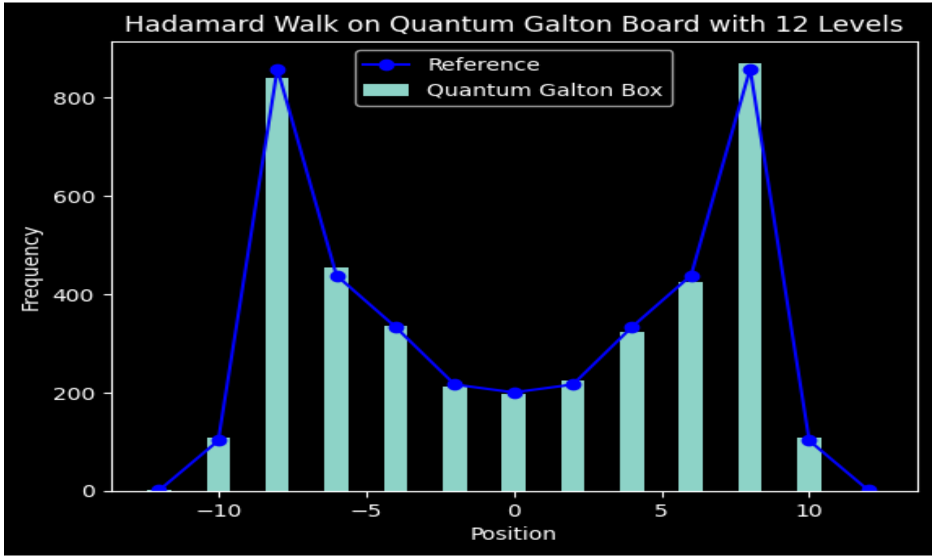

- To achieve a different distribution, probabilities must be adjusted at each level
- The figure illustrates assigning probabilities only to the rightmost quantum peg, requiring either a rotation of the "coin" qubit by a specific angle or flipping it
- In the quantum circuit, this involves modifying the quantum coin function using controlled RY rotations to bias the "coin" accordingly
- The angle of rotation is given by $2arccos(\sqrt{1-\frac{1}{e}})$



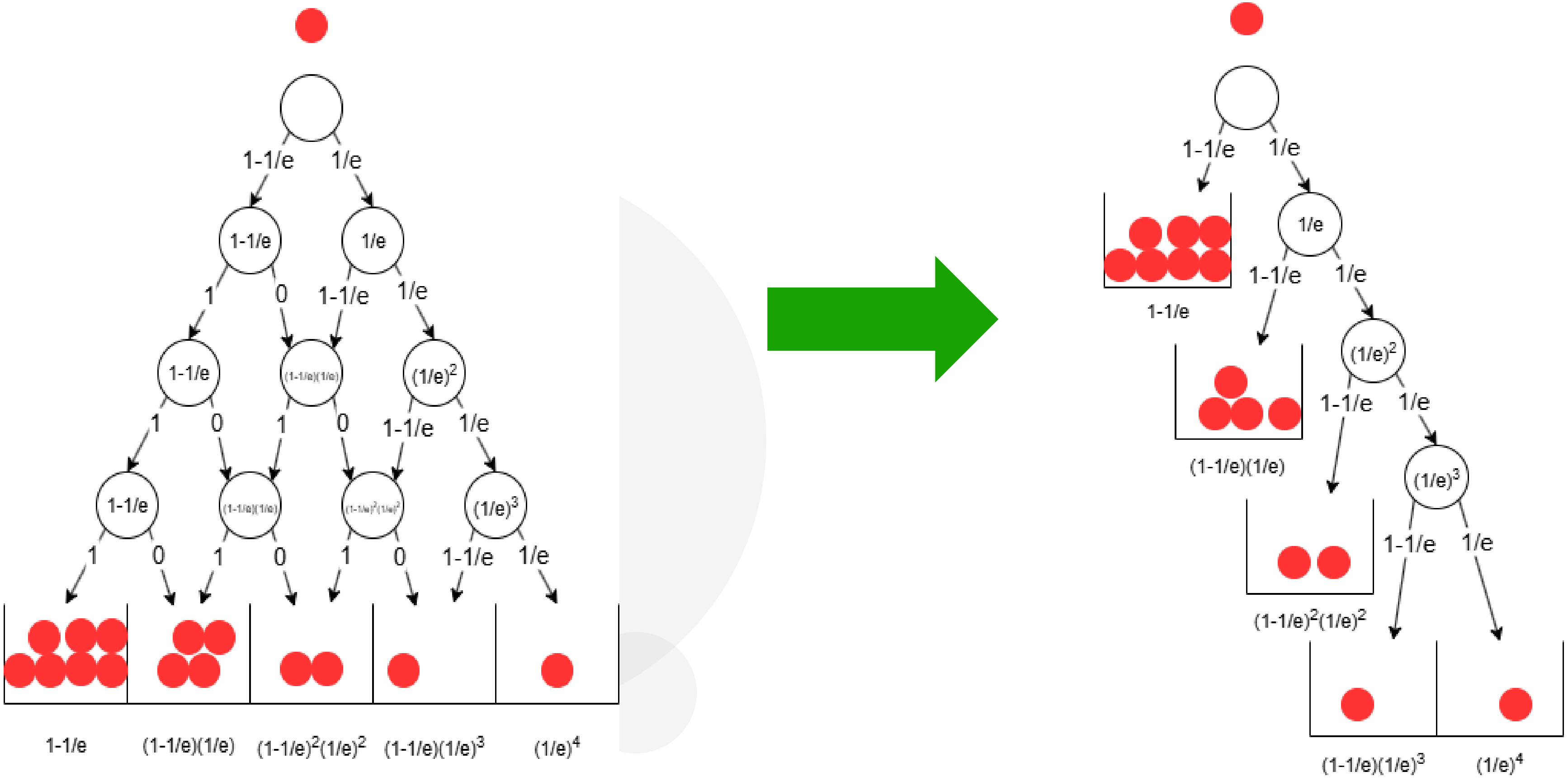
# Hadamard Random Walk (bi-modal) Distribution

- The walk must be conducted in superposition to maintain the interference effect
- Alternate the control value for each CSWAP gate to simulate stepping in superposition
- Refrain from resetting the coin state to preserve interference
- Execute a final swap after all levels have been applied to ensure a symmetric distribution



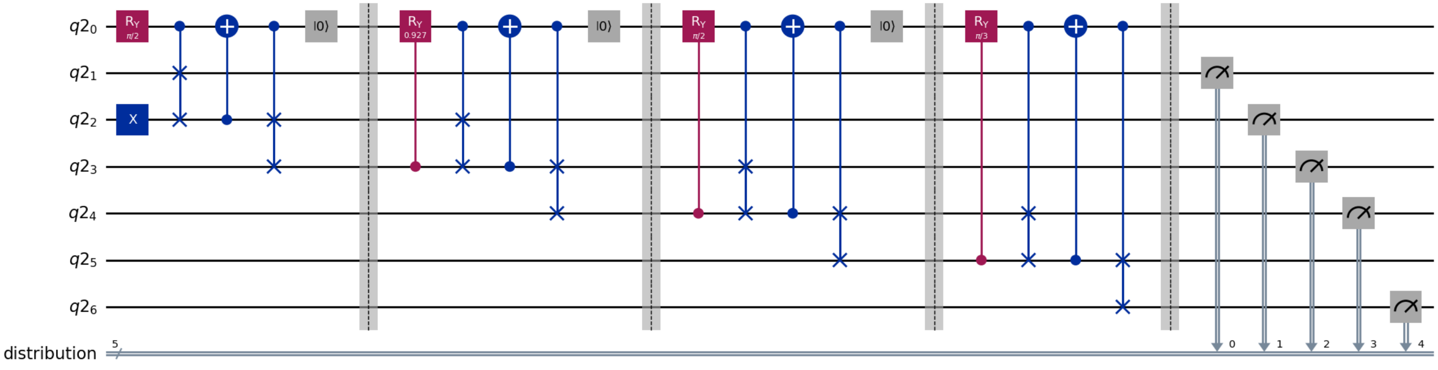
# Symmetric DAG => Optimal Decision Tree



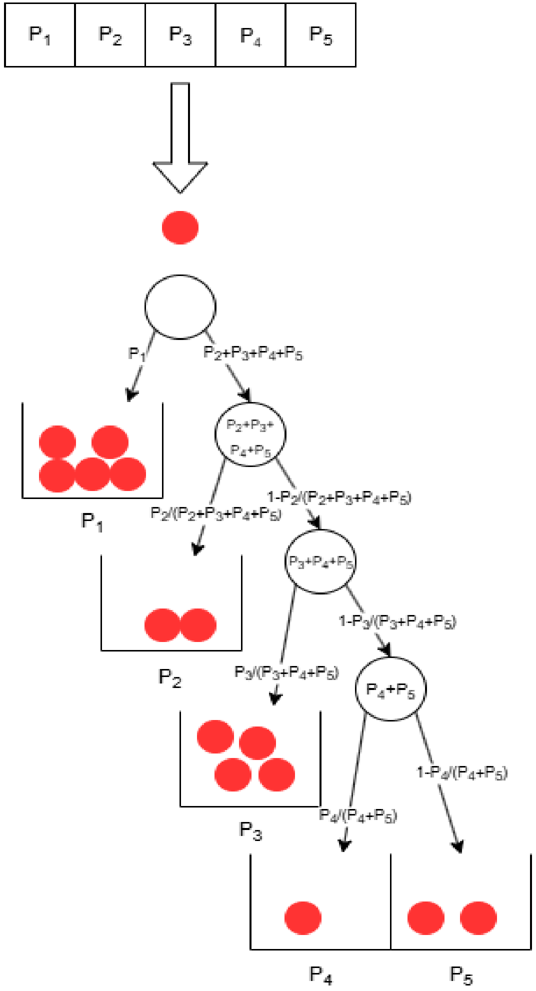

The thinking was inspired by observing the Exponential Distribution decision DAG

# Optimal Decision Tree for an Arbitrary Distribution

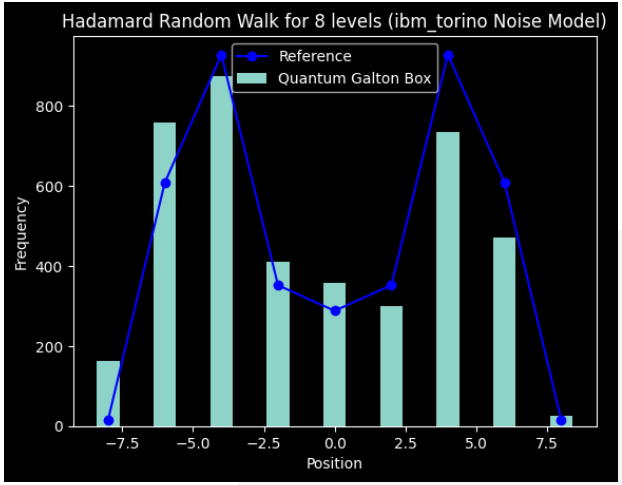
- Given a vector of n+1 probabilities, we can build an optimized coin function
- Quantum circuit will assure the ball reaches each tally bin with the desired probability
- The circuit resembles the one-sided look of the decision tree
- Rotation angle for level `i` is given by: $\theta_i = 2arccos(\sqrt{1 - \frac{p_i}{\sum_{j=i}^{n+1} p_j}})$

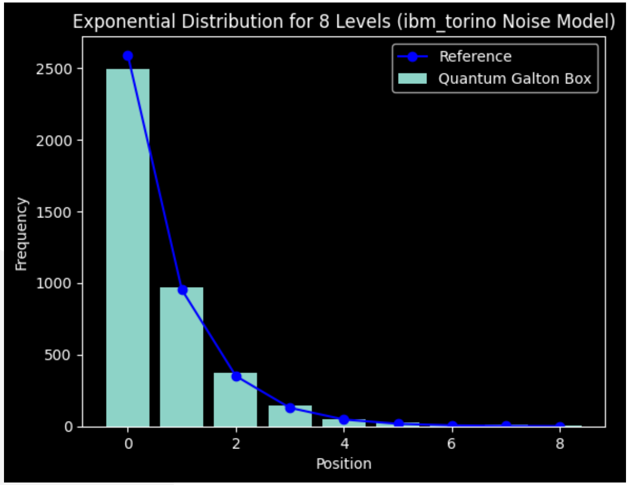


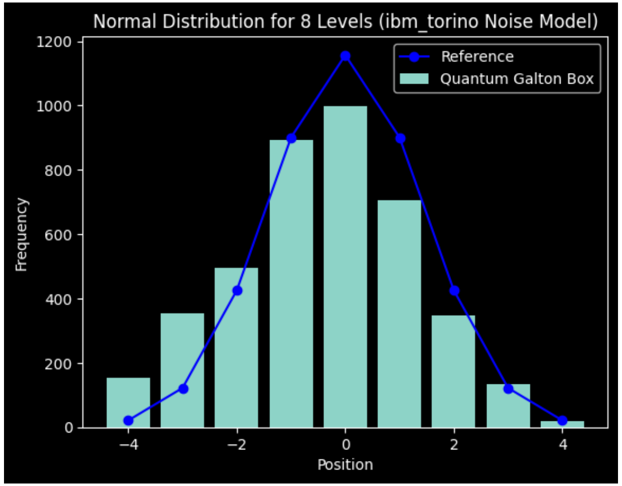

Qubits: n + 3
Depth: 4n = O(n)

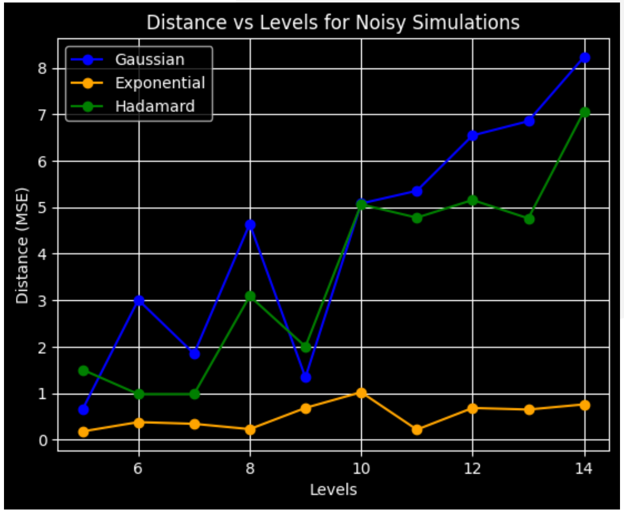


# Noisy Simulations



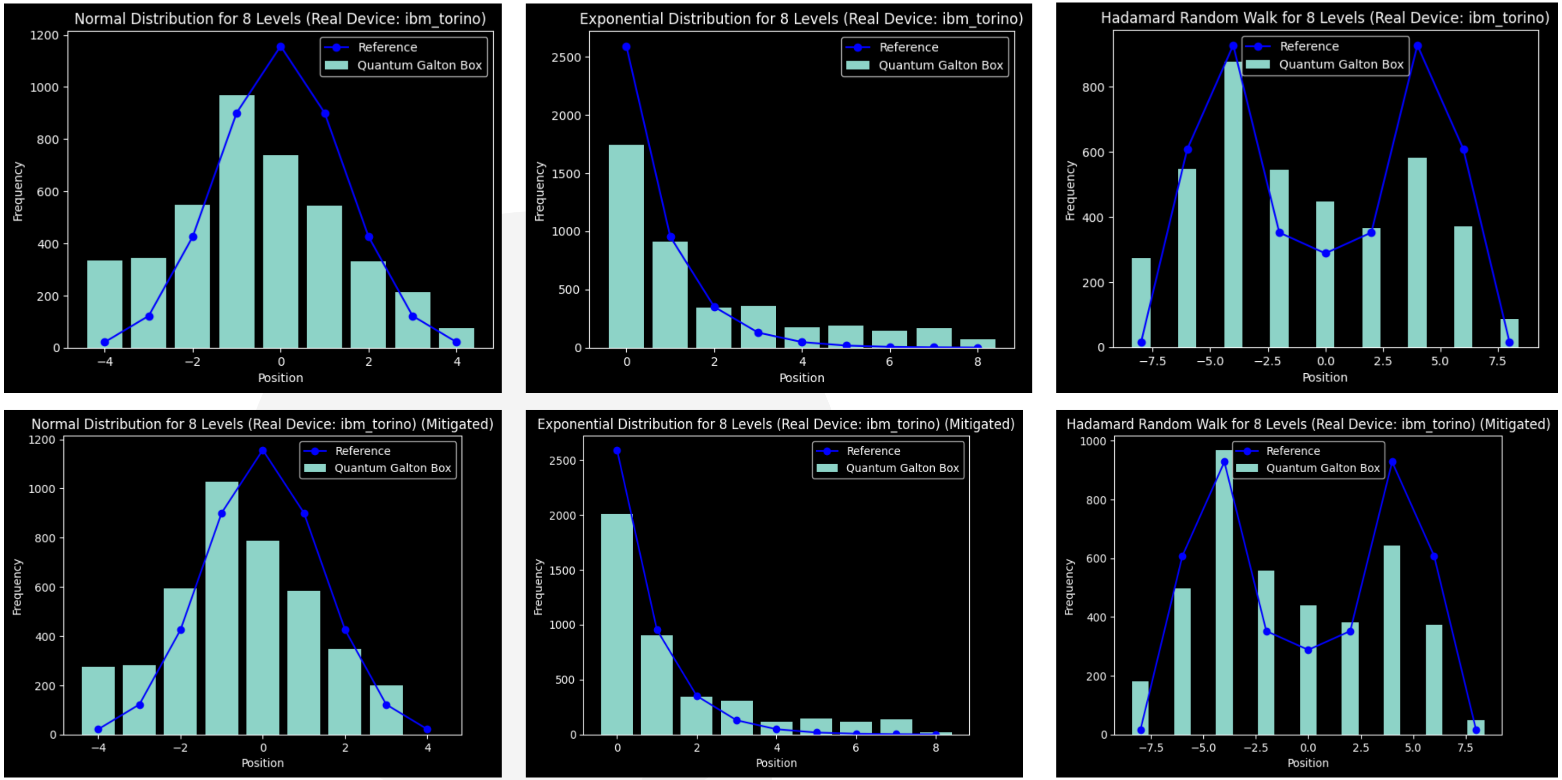



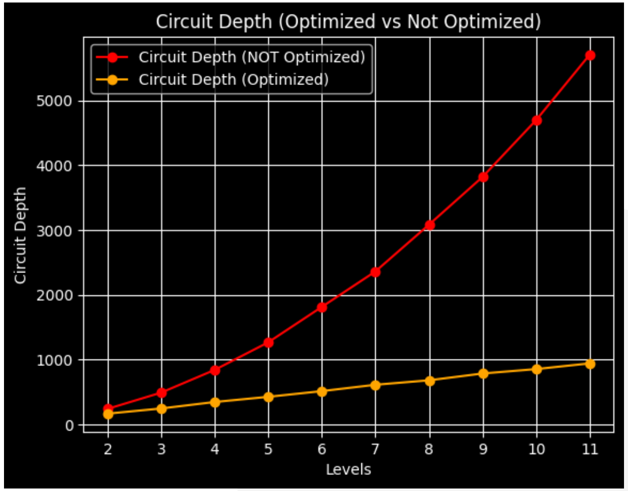

- The exponential distribution demonstrated the greatest improvement in accuracy following the Galton Board optimizations
- By decreasing the number of required qubits from 2n+2 to n+3 and markedly reducing the circuit depth, the results remained sufficiently close to the reference values for all distributions despite the presence of noise
- The visual similarity of the distributions is preserved up to 14 levels with presence of noise

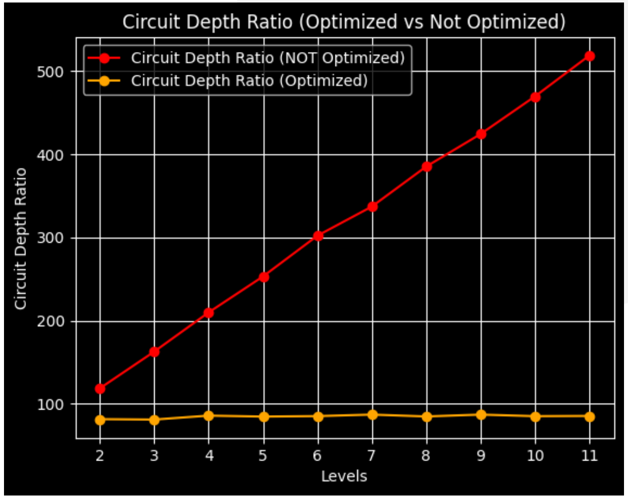
# Real Backend Execution




- Despite large distances from the reference, the distribution shapes are identifiable
- Noise reduction using the M3 package had minimal impact
- Overall, the IBM Torino backend performed impressively!!! 🎉 🎉 🎉

# Circuit Depth Study




**Optimized circuits**

Maintaining a constant depth-to-levels ratio results in a linear dependency on levels

**Non-optimized circuits**

A linearly increasing ratio signifies a quadratic dependency on levels

# Conclusions

- Demonstrated universal distribution simulator as a superposition of Hamming Weight 1 states
- Reduced qubits needed from 2n+2 to n+3
- Improved circuit depth from $O(n^2)$ to $O(n)$ ensuring resource efficiency and noise resilience
- MSE distance stayed below 5 at most levels
- Optimistic outlook for NISQ devices: execution on real backend showcased impressive results

# References


1. "Universal Statistical Simulator", Mark Carney, Ben Varcoe (arXiv:2202.01735)

2. "Quantum random walks - an introductory overview", J. Kempe (arXiv:quant-ph/0303081)

3. "Scalable Mitigation of Measurement Errors on Quantum Computers", Paul D. Nation, Hwajung Kang, Neereja Sundaresan, and Jay M. Gambetta, PRX Quantum 2, 040326 (2021).