# Project 1 - Quantum Walks and Monte Carlo

Gaël-Pacôme Nguimeya Tematio

August 10, 2025

# *Problem statement*



- Aims of the project
  - ☞ Building quantum circuits that realize a Galton Box-style Monte Carlo process, following the Universal Statistical Simulator.
  - ☞ Implementing a generalized L-layer quantum Galton board to produce samples and study quantum-walk behavior.
- Problem and its Significance
  - ☞ Mapping Monte Carlo sampling to quantum circuits for problems representative of high-dimensional systems.
  - ☞ The Galton Box is a standard Monte Carlo model used for PDE solution methods in settings with complex interactions (e.g., particle transport and quantum systems).
  - ☞ Establishing a correct circuit construction matters because it clarifies how quantum hardware can simulate and potentially accelerate such sampling tasks.

# My solution



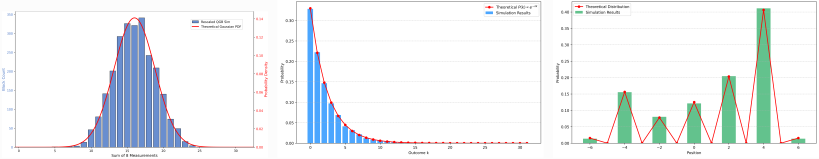
## Objectives

- Implement each task efficiently
- Obtain the best possible accuracy

## Approach

- Generalized $L$-layer QGB: reusable coin qubit, CSWAP sweep, mid-circuit reset.
- Rescaling by block-sum (size 8) to compare with Gaussian prediction.
- Additional samplers: exponential target ($\lambda = 0.4$) and 1D Hadamard walk (6 steps).
- Tools: Python, Qiskit Aer, NumPy, Matplotlib etc...

## Case L = 4




(a) (b) (c)

Figure 1: a) Rescaled 4-layer QGB simulated output, (b) Exponential distribution ($n = 5$, $\lambda = 0.4$), (c) Hadamard quantum walk (6 steps)

| Dist | L | λ/steps | Noise | Shots | Backend | Qubits | Depth | Gates | Fidelity | Score |
|---|---|---|---|---|---|---|---|---|---|---|
| QGB | 4 | n/a | high | 8192 | ibm_toronto | 27 | 630 | 977 | 0.483 | 0.54629 |
| EXP | 4 | 0.10 | medium | 8192 | ibm_toronto | 27 | 5 | 26 | 0.999 | 0.95936 |
| WALK | 4 | steps=5 | low | 8192 | ibm_toronto | 27 | 1972 | 2641 | 0.548 | 0.59842 |

Figure 2: Optimized metrics

# Impact



- Provides a clear, reproducible template for QGB-based Monte Carlo and quantum-walk studies.
- Baseline for evaluating sampling accuracy and circuit design choices in NISQ-era workflows.
- Goals met partially due to limited computational resources.

**Next steps:**

- Scale $L$ and shots; batch and vectorize simulations for speed.
- Add backend-specific noise models and transpiler tuning for CSWAP; test unitary clean-up in place of reset.
- Introduce per-row bias schedules for shaped targets and automate parameter search.
- Run small-$L$ hardware experiments with error mitigation and report distances with confidence intervals.

**Limitations and needs**

- Main constraint: computational resources to simulate larger $L$ and more shots.
- Needed: more CPU/GPU time, memory, and access to quantum hardware for validation.

*Thank you !!!*