Addendum: Analysis of Random Number Generation Performance Discrepancies

Abstract

This addendum examines the initial assumptions about the performance characteristics of Pseudo-Random Number Generators (PRNG), true Quantum Random Number Generators (QRNG) on hardware, and simulated QRNG, compared to the actual findings from our analysis. We assumed PRNG would exhibit higher throughput, lower latency, and lower quality of randomness, while true Quantum hardware QRNG would show lower throughput, higher latency, and higher quality of randomness, with simulated QRNG falling in between. However, the data revealed that both simulated and hardware-based QRNG performed worse in randomness quality than a well-optimized C++ PRNG, contradicting theoretical expectations. This discrepancy is attributed to current technical limitations, which are explored in this report.

1 Introduction

Our project aimed to compare the performance of PRNG, true Quantum hardware RNG, and simulated QRNG in terms of throughput, latency (execution time), and randomness quality. Initial assumptions were grounded in theoretical expectations: PRNG, being a classical algorithm, should prioritize speed and efficiency (higher throughput, lower latency) at the cost of randomness quality, while true Quantum hardware RNG should offer superior randomness due to inherent quantum uncertainty, albeit with lower throughput and higher latency due to hardware constraints. Simulated QRNG was expected to fall between these, balancing randomness and performance. However, the empirical results from our analysis in Google Colab revealed unexpected outcomes, prompting an investigation into current technical limitations.

2 Initial Assumptions

We hypothesized the following performance characteristics based on theoretical principles:

- **PRNG**:
  - Higher throughput (e.g., millions of bits/second) due to efficient classical algorithms.
  - Lower latency (e.g., milliseconds for large samples) due to computational simplicity.

  - Lower quality of randomness, as PRNGs are deterministic and periodic, potentially detectable in statistical tests (e.g., lower entropy, non-uniform distributions).

- **True Quantum Hardware QRNG**:
  - Lower throughput (e.g., tens to hundreds of bits/second) due to quantum hardware constraints, queue times, and shot limitations on platforms like IBM Quantum.
  - Higher latency (e.g., minutes to hours for large samples) due to hardware processing and queue delays.
  - Higher quality of randomness, leveraging quantum superposition and measurement for true unpredictability (e.g., high entropy, uniform distributions).

- **Simulated QRNG**:
  - Moderate throughput (e.g., thousands to hundreds of thousands of bits/second) on simulators like Qiskit's `AerSimulator`, balancing quantum simulation overhead and classical computation.
  - Moderate latency (e.g., seconds for large samples) in simulated environments.
  - Moderate to high quality of randomness, depending on simulation parameters (e.g., qubits, shots), but potentially less than true hardware due to idealized simulation.

These assumptions were based on the theoretical advantages of quantum randomness (intrinsic unpredictability) versus classical PRNGs (deterministic efficiency), with simulations expected to bridge the gap.

3 Actual Findings from the Analysis

Our empirical analysis in Google Colab, using data generated with 500–10,000 samples, yielded results that contradicted these assumptions:

- **PRNG**:
  - Achieved high throughput (e.g., 1.2–2.5 million bits/second with 10,000 samples in 0.27 seconds) and low latency (e.g., 0.01–0.27 seconds).
  - Demonstrated consistently high quality of randomness (entropy 7.9–8.0 bits/byte, K-S p-value ¿ 0.05, autocorrelation 0, quality score 1.0000), far exceeding expectations of lower randomness quality. This was attributed to a well-optimized C++ and Python implementation using `numpy.random`, producing nearly uniform 32-bit integers.

- **Simulated QRNG**:
  - Exhibited lower throughput (e.g., 134K–165K bits/second with 10,000 samples in 1.94 seconds) and moderate latency (e.g., 0.24–1.94 seconds), aligning with expectations of simulation overhead.

  - Surprisingly showed lower quality of randomness (entropy 4.1–4.9 bits/byte, K-S p-value = 0.0000, high autocorrelation -0.5, quality score 0.0000–0.0686) than PRNG, contradicting the assumption of moderate to high randomness. This was linked to simulation parameters (30 qubits, 200 shots), introducing bias and non-uniformity.

- **True Quantum Hardware QRNG (Limited Testing)**:
  - Initial tests (e.g., 500 samples, 280 seconds execution time) suggested lower throughput (57 bits/second) and higher latency, but randomness quality was also lower (entropy 1.37–2.4, p-value = 0.0000, autocorrelation 0.61) than expected, due to hardware noise, limited shots (250), and circuit design on IBM Quantum's free-tier.

The data revealed that both simulated and hardware-based QRNG performed worse in randomness quality than the well-optimized PRNG, despite theoretical expectations of superior quantum randomness.

4 Discrepancy and Technical Limitations

The unexpected findings can be attributed to current technical limitations, which undermine the theoretical advantages of quantum randomness:

- **Quantum Simulation Limitations**:
  - **Bias in Parameters**: Using 30 qubits with 200 shots in `AerSimulator` introduced bias, as the Hadamard gate on many qubits and limited shots failed to produce uniform 32-bit integers. Reducing qubits (e.g., to 6) and increasing shots (e.g., to 1000) could improve randomness, but simulation overhead limits scalability.
  - **Idealized Simulation**: `AerSimulator` assumes noiseless quantum behavior, but real quantum systems have noise, which isn't fully captured, leading to non-random patterns (e.g., high autocorrelation, non-uniform histograms).

- **Quantum Hardware Limitations**:
  - **Hardware Noise and Bias**: IBM Quantum's free-tier backends (e.g., 250 shots per circuit, 64 circuits) introduce noise, measurement errors, and bias (e.g., Hadamard gate imperfections), reducing randomness quality (e.g., entropy 1.37, p-value = 0.0000).
  - **Low Throughput and High Latency**: Queue times, limited shots, and hardware constraints result in low throughput (57 bits/second) and high latency (280 seconds for 500 samples), far below theoretical expectations for quantum advantage.
  - **Quota Constraints**: Free-tier limits (e.g., 5–10 jobs/day) restrict sample size and optimization, preventing full exploitation of quantum randomness.

- **PRNG Optimization**: The C++ and Python PRNG implementations (`std::mt19937`, `numpy.random`) are highly optimized, producing near-ideal uniform random numbers with minimal latency and high throughput, exceeding theoretical expectations of lower randomness quality due to deterministic nature.

These limitations explain why PRNG outperformed both simulated and hardware-based QRNG in randomness quality, despite theoretical predictions. Current quantum technology and simulation tools are not yet mature enough to fully realize quantum randomness advantages, particularly under free-tier constraints and idealized simulation assumptions.

5 Recommendations for Future Work

To align empirical results with theoretical expectations, consider the following:

- **Improve Simulated QRNG**:
  - Increase shots (e.g., 1000) and reduce qubits (e.g., 6) in `AerSimulator` to enhance uniformity and reduce bias.
  - Incorporate noise models or error mitigation to better mimic real quantum hardware, improving randomness quality.
- **Optimize Quantum Hardware QRNG**:
  - Use higher shots (e.g., 1000–2000) and error mitigation on IBM Quantum, leveraging paid plans or less busy backends to reduce noise and bias.
  - Scale to larger sample sizes (e.g., 10,000–100,000) by managing quotas through staggered runs or paid access, balancing latency and throughput.
- **Compare with Advanced PRNGs**: Test state-of-the-art PRNGs (e.g., xorshift, PCG) to verify if PRNG quality remains superior, ensuring fair comparison with quantum methods.
- **Statistical Validation**: Use additional tests (e.g., Anderson-Darling, Dieharder) to confirm randomness, and increase sample sizes (e.g., 100,000) to detect subtle differences, accounting for technical limitations.

6 Conclusion

The discrepancy between our assumptions and findings highlights current technical limitations in quantum simulation and hardware, particularly noise, shot limitations, and resource constraints. While PRNG exceeds expectations with high randomness quality, low latency, and high throughput, both simulated and hardware-based QRNG fall short due to immature quantum technology. Future improvements in simulation parameters, hardware optimization, and sample scaling are essential to realize quantum randomness advantages, aligning empirical performance with theoretical potential.