Thermodynamic Bounding for Cryptographic Prime Generation

Drift Systems Inc.

December 2025

Abstract

Cryptographic prime generation for 4096-bit keys is dominated by the computational cost of probabilistic primality tests (Miller-Rabin). We present the **Adaptive Sigma Sieve**, a pre-computation filter that rejects candidate integers based on their "Arithmetic Friction" (Hamming weight distribution). By dynamically scaling acceptance bounds ($\mu \pm k\sigma$) based on the bit-length L, we filter the statistical tails of the binomial distribution. Empirical results demonstrate a **29.6% to 31.6% reduction** in modular exponentiation cycles, effectively increasing the yield of high-entropy primes without compromising security.

1 Introduction

The generation of RSA and ECC primitives relies on "Blind Search": generating random odd integers n and testing for primality. As key lengths L increase, the density of primes decreases ($1/\ln n$), while the cost of testing increases ($O(k \cdot n^3)$).

Existing acceleration methods focus on modular sieving (Wheel Factorization). However, these methods ignore the internal additive structure of the candidates. Our research indicates that primes exhibit specific "Laminar" signatures in their binary representation[cite: 503].

2 The Scaling Problem

For a random integer of length L, the Hamming weight $H(n)$ follows a binomial distribution approaching a normal distribution:

$$\mu = \frac{L}{2}, \quad \sigma = \frac{\sqrt{L}}{2} \tag{1}$$

Static percentage filters (e.g., accepting 45%-55% weight) fail at high bit depths ($L = 4096$) because the relative standard deviation shrinks, causing the filter to either reject valid candidates or accept total noise[cite: 424].

3 Methodology: The Adaptive Sigma Sieve

We define a dynamic acceptance interval A_L derived from the thermodynamic entropy bounds of the system:

$$A_L = \left[\frac{L}{2} - k\frac{\sqrt{L}}{2}, \quad \frac{L}{2} + k\frac{\sqrt{L}}{2} \right] \tag{2}$$

Where k is a tuning coefficient ($0.8 \leq k \leq 1.2$)[cite: 431].

3.1 Hardware Implementation

To implement this at line-rate for 4096-bit keys, we utilize a **Split-Loop Adder Tree** on a Gowin GW2A-18 FPGA. This architecture bypasses synthesis iteration limits by calculating partial sums $S_{0..7}$ in parallel and aggregating them in a single clock cycle.

4 Results

In a sample of $N = 100,000$ candidates, the filter rejected candidates residing in the turbulent tails ($\sigma > 0.4$).

- **512-bit:** 29.6% Computational Savings
- **4096-bit:** 31.6% Computational Savings [cite: 405]

This confirms that the "Goldilocks Zone" of arithmetic friction scales linearly with key size, providing a consistent efficiency gain for hyperscale HSMs.