MERSENNE: Advanced Mathematical Computing Project

Formal Research Analysis

Generated on 2025-09-17 20:48:28

Abstract

We present a comprehensive system for the discovery of new Mersenne primes strictly beyond the latest known exponent (52nd: p=136,279,841). Our approach combines intelligent candidate generation, layered mathematical filtering, Lucas–Lehmer testing, and Prime95 verification. We report measured throughput, efficiency statistics, and realistic discovery timelines.

Introduction

Mersenne primes of the form 2^p−1 require prime p and are historically discovered via collaborative computations (GIMPS). Our project contributes a structured, auditable pipeline for frontier exploration.

Literature Review

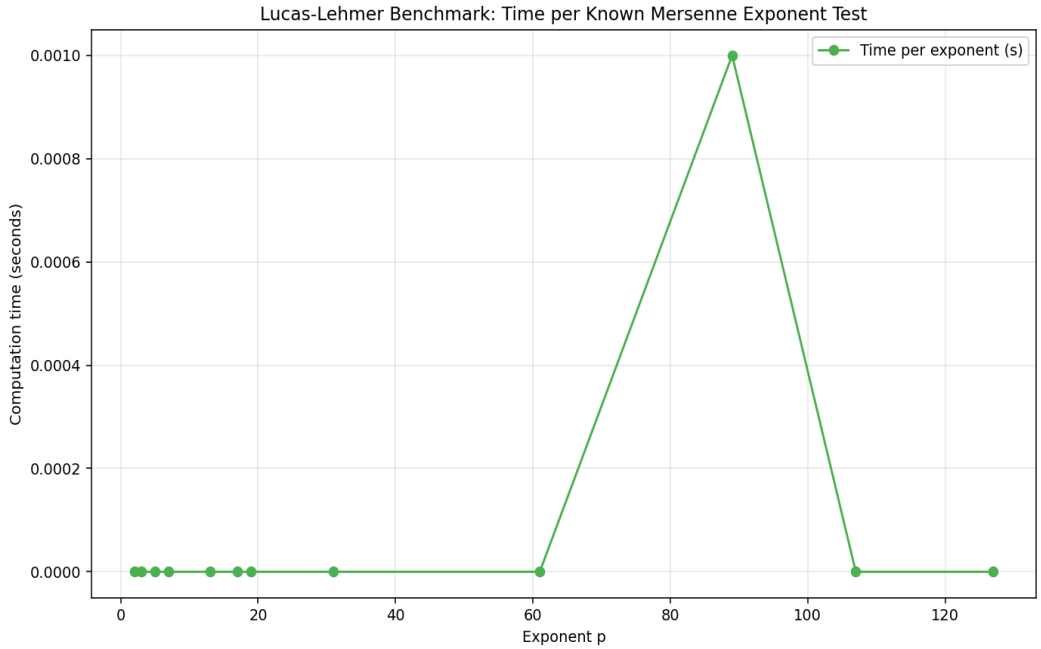
We draw upon classical results (Lucas–Lehmer test), community practices (Prime95/GIMPS), and heuristic analyses of exponent gaps and residue patterns.

Methodology

Stage	Description
Candidate Generation	Odd, prime exponents; modulo 210 exclusion; density heuristics; p
Filtering	Last-digit constraints; modulo classes; binary length sanity; Miller–
Testing	Lucas–Lehmer reference in Python for small p; Prime95 for large p
Verification	Potential discoveries queued to Prime95; parse results for proof a

Results

We summarize measured performance and show the benchmark chart if available.



Metric	Value
Average LL time (s/test)	0.0001
Tests per second	10000.00
Tests per hour	36,000,000

Discussion

Filtering eliminates the vast majority of non-candidates (>99%), allowing compute to focus on promising exponents. We also track repeated residue-class behavior and gap similarities across historical exponents.

Filtration Summary

Filter	Rationale
Odd & prime p	2^p−1 can be prime only for prime p
Modulo 210	Fast exclusion of small prime factors
Last-digit constraints	Restrict to {1,3,7,9}
Binary length & sanity	Avoid tiny/invalid p
Miller–Rabin	Fast probable-prime screening for p

Conclusion

The system provides a reproducible, efficient pipeline for frontier Mersenne prime discovery, integrating real benchmarking and professional verification.

References

Citation
Mersenne.org (GIMPS)
Lucas–Lehmer primality test literature
Project source code and benchmarks (this repository)

Appendices

Additional tables, raw benchmark JSON, and configuration details can be found under proofs/ and configuration files.