

An Adaptive Multilevel Algorithm for Constructing Goldbach Prime Pairs in Very Large Even Integers

Bouchaib Bahbouhi

Independent Researcher, Nantes, France.

***Corresponding Author:** Bouchaib Bahbouhi, Independent Researcher, Nantes, France.

ABSTRACT

The computational construction of Goldbach prime pairs becomes increasingly demanding when the prescribed even integer reaches several thousand decimal digits, because repeated primality testing of unsuccessful candidates can rapidly dominate the computational cost. This work introduces Bahbouhi.b.goldbach.2026, an adaptive multilevel algorithm designed to construct prime pairs p and q satisfying $p + q = E$ for a prescribed very large even integer E . Instead of relying on a direct sequential search, the method organizes candidate selection into a hierarchy of increasingly expensive operations. It combines arithmetic classification modulo 6 and 12, admissible residue rails, destructive congruences, an $r - 1$ divisibility representation, deep modular sieving, survivor-resistance ranking, peripheral and central search branches, local candidate probes, anchor propagation, and an adaptive controller that redirects the search when a branch becomes unproductive. Probable-prime testing is therefore concentrated on candidates that survive the cheaper arithmetic layers. The algorithm was experimentally applied to prescribed even integers ranging from moderate sizes to more than 10,000 decimal digits. The largest externally verified representation retained in the reported experiments is $E = 10^{10068} + 28878$, with $p = 16481$ and $q = 10^{10068} + 12397$. The experiments illustrate the practical scalability of the multilevel approach while identifying the principal computational bottlenecks that emerge at extreme digit lengths. This work is computational and algorithmic and makes no claim of proving Goldbach's conjecture.

Keywords: Goldbach conjecture; strong Goldbach conjecture; additive number theory; computational number theory; Goldbach prime pairs; prime-pair construction; adaptive algorithm; multilevel algorithm; modular arithmetic; modular sieve; destructive congruences; residue classes; residue rails; $r - 1$ representation; primality testing; probable primes; large integers; computational arithmetic; large-scale computation

ARTICLE INFORMATION

Received: 04 October 2026

Accepted: 20 October 2026

Published: 22 October 2026

Cite this article as:

Bouchaib Bahbouhi. An Adaptive Multilevel Algorithm for Constructing Goldbach Prime Pairs in Very Large Even Integers. Open Access Journal of Computer Science and Engineering, 2026; 3(1): 01-09.

<https://doi.org/10.71123/3071-3951.030101>

Copyright:©2026. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.



Introduction

Goldbach's conjecture is one of the oldest unresolved problems in number theory. In its strong form, it states that every even integer greater than 2 can be represented as the sum of two prime numbers. Classical work on additive prime representations includes the Hardy-Littlewood framework [5], Chen's theorem [6], studies of the

exceptional set [7], and the proof of the ternary Goldbach conjecture by Helfgott [8]. Extensive computational work has also verified the binary conjecture over very large finite intervals [1–4]. A proof valid for all even integers nevertheless remains unknown.

The computational problem considered in this article is different from exhaustive verification over a consecutive

interval. A prescribed even integer E is selected first, potentially containing several thousand decimal digits, and the objective is to construct primes p and q such that $p + q = E$. At such sizes, exhaustive enumeration is neither necessary nor computationally attractive. The central problem becomes one of candidate selection: arithmetic information should reject as many unsuitable candidates as possible before expensive probable-prime or primality tests are performed.

A straightforward peripheral search selects successive primes p and tests $q = E - p$. A central search instead examines candidates around $E/2$. Both approaches become inefficient when a long sequence of candidates fails, particularly when E contains thousands of digits. Testing the probable primality of a very large integer is substantially more expensive than checking divisibility by small primes or evaluating modular congruences. Established primality and probable-prime methods provide powerful terminal tests [9–14], while sieve and algorithmic-number-theory methods provide the natural framework for inexpensive preliminary filtering [15–17].

This principle motivated the development of Bahbouhi.b.goldbach.2026. The method is not a single sieve or a fixed scanning rule. It is an adaptive multilevel search architecture in which candidate pairs pass through successive layers of arithmetic selection. The lowest levels use parity, residue classes modulo 6 and 12, admissible Goldbach rails, and destructive congruences. These are followed by an $r - 1$ representation of divisibility, progressively deeper modular sieving, and classification of candidates according to their resistance to elimination. Only the strongest survivors are promoted to probable-prime testing.

The algorithm also treats the geometry of the search as a variable rather than a fixed choice. Peripheral candidates, central candidates, and alternative search regions can be compared. When one branch becomes persistently unproductive, an adaptive relay can redirect computational effort toward another branch. Conversely, a region containing unusually resistant survivors can be investigated more intensively instead of being abandoned prematurely. The procedures called CHIRURGIE and ACHARNEMENT are computational modes, not mathematical theorems.

An important component of the method is the explicit recording of failure. If $q = E - p$ is divisible by a small prime r , then p occupies a residue class determined by E modulo r . Such destructive residue classes can be identified before a large-integer primality test is attempted. Repeated failures therefore provide information that can be reused

in subsequent candidate selection. The $r - 1$ representation gives an equivalent way of encoding divisibility and is used to study sequences in which one or both members of successive candidate pairs remain composite.

The experiments reported here were performed by fixing E before searching for its representation. Thus, the large examples presented later were not generated by first selecting two previously known primes and subsequently defining their sum. Candidate values of p were generated or selected by the algorithm, q was determined by $q = E - p$, and surviving candidates were then subjected to the stated computational verification procedure.

The experimental campaign progressively considered prescribed even integers containing hundreds, thousands, and eventually more than ten thousand decimal digits. Database labels are not treated as primality proofs; probable-prime status, external verification, and proven-prime status are distinguished throughout.

The largest Goldbach representation retained as the record of the present study is $E = 10^{10068} + 28878$, $p = 16481$, and $q = 10^{10068} + 12397$. The identity $p + q = E$ is exact. The computational history and verification associated with this representation are described in the experimental sections.

The objectives of this article are to give a reproducible description of the complete Bahbouhi.b.goldbach.2026 architecture; explain the arithmetic mechanisms used to reduce and rank candidate pairs; report representative large-scale constructions culminating in the retained 10^{10068} -scale record; and examine computational range, bottlenecks, and limitations. The present work is deliberately computational and algorithmic. No proof of Goldbach's conjecture is claimed or implied.

Mathematical and Computational Framework

Throughout this work, E denotes a prescribed even integer. The computational objective is to identify two primes p and q satisfying $p + q = E$. Once p is selected, q is determined uniquely by $q = E - p$. In a central search, with $m = E/2$, the pair is represented as $p = m - d$ and $q = m + d$, where d is the central deviation.

A central design principle is that arithmetic operations have very different computational costs for integers containing thousands of digits. Tests involving parity, small moduli, residue classes, or

divisibility by small primes are inexpensive compared with repeated probable-prime testing. The algorithm therefore orders operations from inexpensive to expensive.

Every prime greater than 3 is congruent to 1 or -1 modulo 6. For E divisible by 6, the two primes greater than 3 must occupy opposite modulo 6 rails. If E is congruent to 2 modulo 6, both primes must be congruent to 1 modulo 6. If E is congruent to 4 modulo 6, both must be congruent to -1 modulo 6. The exceptional possibility involving the prime 3 is checked separately.

Let r be a prime used by the modular sieve. Since $q = E - p$, if p is congruent to E modulo r , then q is divisible by r . Except for $q = r$, such a candidate cannot provide a Goldbach pair. This is called a destructive congruence. Writing $E = a r + s$ and $p = b r + s$ makes the divisibility immediate.

A second representation encodes divisibility through $x - 1 = a r + s$, with $0 \leq s < r$. The terminal condition $s = r - 1$ is equivalent to r dividing x . This $r - 1$ representation is not a new primality criterion; it is an alternative encoding used to analyse chains of candidate failures. After elementary treatment of 2, 3, and 5, diagnostic analysis can focus primarily on primes $r \geq 7$.

Candidate pairs can be classified operationally as C/C, C/P, P/C, or P/P, with P interpreted according to the verification level being discussed. These states permit bilateral analysis of which side of a candidate pair is repeatedly destroyed.

A candidate surviving to greater sieve depth has accumulated more negative divisibility information than one eliminated immediately. Resistance is therefore used as a ranking criterion, never as a proof or strict probability of primality. Deeply resistant composites observed in the experiments are important precisely because they demonstrate this distinction.

The algorithm distinguishes peripheral and central regions and can activate alternative regions through the adaptive controller. Movement is governed by survivor density, destructive-factor depth, recent strong-test yield, and the presence or absence of exceptionally resistant candidates.

The computational pipeline separates filtering from primality determination. Passing a modular sieve is not primality. Passing a probable-prime test is stronger but is not automatically a proof. External database status is also kept distinct from mathematical certification.

The immediate objective is not to enumerate every Goldbach representation of E , but to locate a reliable representation while minimizing expensive large-integer tests. For a peripheral search, the smallest successful p reached under the stated ordering is a useful experimental measure of difficulty; for a central search, the corresponding quantity is the successful deviation d .

The Bahbouhi.b.goldbach.2026 Algorithm

Bahbouhi.b.goldbach.2026 is organized as a hierarchical and adaptive search procedure. The root is the prescribed even integer E . The first branches classify E arithmetically and determine admissible residue rails. Candidate values of p are generated within a selected region, with q determined exactly by $E - p$. Destructive congruences and modular sieves remove candidates containing detectable small factors. Remaining candidates are ranked by resistance, and only a restricted population reaches probable-prime testing.

Initial Audit

Every search begins with an exact audit of E : parity, magnitude, $m = E/2$ when relevant, residues modulo 6 and 12, and useful small factors. This establishes the arithmetic environment before expensive candidate generation.

Admissible 6x Rails

Candidate generation is immediately restricted to the modulo 6 rails compatible with E . For E divisible by 6, the rails are opposite; for E congruent to 2 modulo 6 they are both +1; for E congruent to 4 modulo 6 they are both -1. The prime 3 is handled separately.

Candidate Generation

In the peripheral branch, prime p values are generated in a selected interval and $q = E - p$. In the central branch, admissible deviations d generate $p = m - d$ and $q = m + d$. Alternative regions may be activated when the controller detects persistent poor yield.

Destructive Congruence Layer

For each selected small prime r , E modulo r is reused across candidates. If p is congruent to E modulo r , q is divisible by r and the pair is rejected. In the central branch, both $m - d$ and $m + d$ are checked. The first destructive prime can be recorded for diagnostics.

The $r - 1$ Branch

The $r - 1$ signatures of p and q are followed simultaneously, especially during long unsuccessful chains. This branch provides a common representation for the divisors responsible for C/C, C/P, and P/C states.

MAX-SIEVE FIRST

Candidates surviving elementary filters enter deeper sieving. The principle is to exploit the deepest computationally reasonable modular filtering before launching repeated expensive tests. The optimal depth is dynamic: deeper sieving is useful only while its marginal cost is justified by the strong tests it avoids.

Survivor Resistance and Ranking

Surviving candidates are ranked using depth of survival and available local information. Exceptional survivors receive higher priority. Resistance remains a scheduling signal only; extreme survivors can still be composite.

Local Probes

Around a selected survivor p , the algorithm can inspect $p - 2$ and $p + 2$ diagnostically, respecting admissible rails. Translations by 6 preserve modulo 6 rails, allowing $p + 6n$ and $q - 6n$, or the reverse direction, to be explored locally while preserving $p + q = E$.

q-Anchor Propagation

In families of related even integers, a useful q candidate can be reused as an anchor. For a nearby E' , $p' = E' - q$ is inexpensive to test. Anchor reuse never removes the requirement to audit the exact sum and verification status.

CHIRURGIE

When a branch becomes unusually sparse or repeatedly fails, CHIRURGIE measures survivor density, destructive-factor depths, rail distribution, bilateral states, spacing, asymmetry, and recent strong-test yield. The resulting diagnosis is fed back into lower algorithmic levels.

ACHARNEMENT

When a branch contains exceptionally resistant candidates, the algorithm can persist selectively rather than switching immediately. These candidates receive priority for deeper sieving and strong testing. This mode is restricted to computationally informative survivor populations.

Random Peripheral Sampling

Peripheral subintervals may be sampled to estimate survivor density at selected depths. Regions with unusually strong resistance can receive a priority bonus, while very cold regions can be deprioritized. This changes search order, not validity conditions.

Adaptive Relay

The controller combines sieve information, ranking, local probes, CHIRURGIE, ACHARNEMENT, and recent probable-prime outcomes. A cold branch without extreme survivors can be replaced by another peripheral interval, the center, or another admissible region. A branch with extreme survivors is allowed to persist longer.

Probable-Prime Testing

Only high-priority survivors reach strong testing. Established probable-prime procedures, including those available through GMP and PARI/GP [18,19], are used as terminal rejection and screening tools. If one member is a

verified small prime, computational effort is concentrated on the giant complementary member.

Independent Verification and Exact Audit

A probable-prime result is not automatically promoted to proven-prime status. Important candidates can be checked independently. Before retention, $p + q = E$ is verified exactly and the strongest recorded verification status of each member is stated.

Complete Decision Logic

Starting from E , the algorithm audits residues and rails, generates admissible candidates, applies destructive congruences and the $r - 1$ representation, deepens the sieve according to cost, ranks survivors, applies local probes and anchors when useful, diagnoses cold branches, persists on exceptional survivors, switches branches adaptively, and reserves strong primality testing for the highest-priority survivors. If no pair is found within the computational budget, the search may enlarge or change region; failure of a finite search is never interpreted as evidence against Goldbach's conjecture.

The novelty investigated experimentally lies not in an individual congruence, sieve, or primality test, but in the integration of these established components into an adaptive Goldbach-specific hierarchy in which failure information and search-region performance affect subsequent decisions.

Computational Protocol and Reproducibility

A prescribed E is fixed before each search. Candidate p values are generated according to the active branch and q is calculated exactly as $E - p$. Candidates pass through inexpensive arithmetic layers before strong testing. Sieve depth varies between experiments because the optimal balance changes with digit length and survivor density.

Status terminology is strict. Composite means that compositeness has been established by a divisor or reliable test. Sieve survivor means only that no divisor was detected within the stated range. PRP means probable prime under the specified test. Externally verified prime denotes a candidate independently reported as prime. Proven prime is reserved for a genuine rigorous proof or certificate when the nature of that proof is established.

GMP procedures were used for strong probable-prime testing [18]. A positive probable-prime result is recorded as PRP unless stronger verification is obtained. PARI/GP provided an additional independent environment [19]; its probable-prime and rigorous-prime functions must likewise be distinguished.

FactorDB and dCode were used as external checking services during parts of the campaign. Database classifications document computational history but are not automatically treated as mathematical certificates. Interface markers such as an asterisk are not interpreted as proof that a prime was previously unknown.

For every retained pair, p status, q status, and the exact identity $p + q = E$ are separately audited. Power-of-ten expressions with small offsets are preferred for giant integers because they are exact and reduce transcription risk.

The reported experiments are constructive in the operational sense that E was prescribed before the successful pair was known. The algorithm generated or ranked p candidates, determined $q = E - p$, and then tested the resulting candidate. Database novelty and primality are treated as logically separate properties.

A statement that a candidate survived a sieve to B means only that no tested prime divisor not exceeding B was found. Deep false survivors are retained as diagnostic information because they demonstrate the limits of sieving as a predictor.

The verification hierarchy is: arithmetic admissibility and exact identity; modular-sieve survival; strong probable-prime testing; independent probable-prime confirmation; external prime verification; and, where available, rigorous primality proof or certification. Not every candidate reaches every level.

Reproducibility is required both arithmetically and algorithmically. Another investigator should be able to reconstruct each E , p , and q and reproduce the residue restrictions, sieve stages, ranking, and branch logic. Exact execution times are hardware-dependent and are not treated as universal properties.

Experimental Results

The experimental development proceeded through increasingly large prescribed even integers. Both central and peripheral searches were used. The peripheral branch became particularly useful at extreme digit lengths because the small p can be verified cheaply while only q requires giant-integer testing.

Central experiments included $E = 10^{1000} + 2$, with $d = 2,304,238$ and a recorded probable-prime pair; $E = 10^{1500} + 2$, with $d = 5,749,478$; and $E = 10^{2000} + 2$, with $d = 6,313,132$. A major central milestone was $E = 10^{3000}$, $d = 805,797$, giving $p = 5 \times 10^{2999} - 805,797$ and $q = 5 \times 10^{2999} + 805,797$; both members were externally verified as prime during the campaign.

Several prescribed integers produced long sequences of admissible candidates without simultaneous success. These experiments motivated the operational concept of a Goldbach desert: a finite region that remains unsuccessful under a specified search procedure. The term does not imply absence of a representation outside the tested region.

Peripheral successes included $E = 10^{2500} + 6$ with $p = 9293$ and $q = 10^{2500} - 9287$, and $E = 10^{2500} + 8$ with $p = 25391$ and $q = 10^{2500} - 25383$. Both large q candidates were externally verified as prime during the campaign.

Near 3000 digits, $E = 10^{3000} + 20$ produced $p = 26711$ and $q = 10^{3000} - 26691$; $E = 10^{3000} + 768$ produced $p = 131447$ and $q = 10^{3000} - 130679$; and $E = 10^{3000} + 938$ produced $p = 4967$ and $q = 10^{3000} - 4029$. These cases show substantial variation in the successful small p even for the same exponent scale.

Beyond 3000 digits, verified representations were obtained for $E = 10^{3500} + 96$ and $E = 10^{4000} + 44$. For the difficult case $E = 10^{3900} + 556$, the search eventually reached $p = 124183$ and $q = 10^{3900} - 123627$, externally verified as prime.

Further milestones included $E = 10^{4147} + 6852$ with $p = 78893$ and $q = 10^{4147} - 72041$; $E = 10^{4775} + 91768$ with $p = 9127$ and $q = 10^{4775} + 82641$; $E = 10^{5188} + 564286$ with $p = 64279$ and $q = 10^{5188} + 500007$; and $E = 10^{6000}$ with $p = 11213$ and $q = 10^{6000} - 11213$.

At $E = 10^{6298} + 8798$, the algorithm produced $p = 78079$ and $q = 10^{6298} - 69281$. At $E = 10^{6550} + 6682$, it produced $p = 32371$ and $q = 10^{6550} - 25689$. Both q candidates were externally reported as prime.

The 7000-digit experiments refined survivor resistance. For $E = 10^{7000} + 6434$, $p = 1453$ gave $q = 10^{7000} + 4981$, externally verified as prime. For $E = 10^{7000} + 898$, a deep survivor $p = 5569$ gave $q = 10^{7000} - 4671$ but was subsequently found composite; continuing the search produced $p = 38791$ and $q = 10^{7000} - 37893$, externally verified as prime.

At $E = 10^{7575} + 789434$, several deeply resistant candidates were composite; a later candidate $p = 823261$ gave $q = 10^{7575} - 33827$, retained as PRP rather than promoted beyond the recorded evidence. For $E = 10^{7579} + 67788$, $p = 220529$ and $q = 10^{7579} - 152741$ were externally verified as a prime pair.

At $E = 10^{8000} + 357956$, $p = 359143$ gave the deep survivor $q = 10^{8000} - 1187$, later classified composite. Continuing produced $p = 414017$ and $q = 10^{8000} - 56061$, externally verified as prime.

At $E = 10^{8777} + 633266$, the algorithm reached $p = 82393$ and $q = 10^{8777} + 550873$ after a pronounced sequence of resistant failures. The q candidate passed independent probable-prime testing and was externally reported as prime.

At $E = 10^{9100} + 5643288$, $p = 32609$ gave $q = 10^{9100} + 5610679$. The q candidate passed GMP, an independent PARI/GP probable-prime test, and external prime verification. A second case, $E = 10^{9100} + 42248$, produced $p = 947$ and $q = 10^{9100} + 41301$ with corresponding independent checks.

The next milestone was $E = 10^{9768} + 776896$, with $p = 54517$ and $q = 10^{9768} + 722379$. The large q candidate passed GMP and PARI/GP probable-prime testing and was subsequently externally reported as prime.

The largest retained representation is $E = 10^{10068} + 28878$, $p = 16481$, $q = 10^{10068} + 12397$. Since $16481 + 12397 = 28878$, the Goldbach identity is exact. The small p is prime. The large q passed GMP probable-prime testing, was recorded in FactorDB with probable-prime information, and was subsequently externally reported as prime by dCode. It is therefore reported according to the conservative verification terminology of Section 4.

Experiments continued beyond the record. At $E = 10^{15576} + 4432$, $p = 7591$ gave $q = 10^{15576} - 3159$, which survived destructive sieving to approximately 10^{10} but was subsequently classified composite by PARI/GP. This is a particularly informative false survivor and demonstrates that extending the sieve indefinitely is not necessarily efficient.

Computational Scaling and Practical Limits

The 10,068-digit record is not the maximum scale at which the algorithm can operate. Candidate generation, modular sieving, ranking, and probable-prime testing were experimentally performed at substantially larger sizes, including more than 12,000 and 15,000 digits.

Four limits should be separated: sieve depth, probable-prime testing, search distance, and final certification. Deeper sieving reduces strong-test population but also costs more; testing one giant candidate can remain feasible while testing many becomes expensive; long deserts can dominate total work; and rigorous certification can be much harder than PRP detection.

For $E = 10^{12550} + 6884$, the peripheral branch was restricted initially to $p < 100000$. Among 9,592 primes below 100000, sieving complementary q values by primes up to 10^6 left exactly 1,324 survivors, approximately

13.8%. The first survivors included 37, 41, 67, 83, 151, 163, 181, 277, 353, 397, 421, 463, 521, 541, and 547. Strong testing rejected the recorded candidates through $p = 541$ before the experiment was deliberately stopped.

The same E was examined centrally. Exact audit gives E divisible by 6 and 12 and $m = 5 \times 10^{12549} + 3442$. Admissible central deviations were tested beyond 4,000, but no pair was retained before that branch was stopped. Both geometries remained operational; the obstacle was cumulative unfavorable candidate behavior rather than the magnitude of E itself.

For $E = 10^{15576} + 4432$, $q = 10^{15576} - 3159$ survived sieving to approximately 10^{10} before failing a stronger test. This demonstrates the sieve-versus-test trade-off directly. MAX-SIEVE FIRST is beneficial only while additional sieving is economically justified.

A false survivor is useful computational information. Resistance is correctly interpreted as lack of detected small factors, not as primality. The algorithm therefore uses resistance as a scheduling variable.

The most difficult cases are better characterized by local search-desert length than by digit count alone. A favorable enormous E can require only a few q tests, whereas a smaller unfavorable E can require many. This motivates adaptive ranking, sampling, CHIRURGIE, ACHARNEMENT, and branch switching.

For a prescribed E , the experimental peripheral Goldbach distance is the smallest successful p reached under the stated ordering. A heuristic reference scale investigated during development is $H_G(E) = \log(E) \log \log(E)$, but no uniform logarithmic or polylogarithmic bound is claimed.

A two-dimensional extension considers $E_k = E_0 + 2k$ with a fixed p window. It measures both distance in p for a fixed E and distance in E before a hit occurs within that fixed window. This remains an experimental calibration tool, not a theorem.

The current frontier is practical rather than arithmetic. It depends on implementation efficiency, sieve organization, survivor density, desert length, strong-test cost, and certification resources. More aggressive residue caching, direct admissible-class generation, parallel processing, and dynamic cost-based sieve depth are natural next improvements.

Algorithmic Novelty and Comparison with Existing Approaches

Computational investigations of Goldbach's conjecture have traditionally pursued exhaustive verification below a

bound, with major implementations extending this range through 4×10^{18} [1–4]. Bahbouhi.b.goldbach.2026 addresses a different problem: given one prescribed E that may contain thousands of digits, construct one pair $p + q = E$. It is therefore complementary to exhaustive interval verification.

The theoretical background includes Hardy-Littlewood additive-prime heuristics [5], Chen's theorem [6], exceptional-set work [7], and Helfgott's ternary Goldbach result [8]. The present work does not replace or prove these results.

Several individual components are classical: residue arithmetic, sieving, probabilistic primality testing [9,10], Lucas and pseudoprime methods [11,12], and rigorous primality proving [13,14]. The proposed originality is their organization into a Goldbach-specific adaptive hierarchy.

Compared with fixed sequential peripheral scanning, the architecture precomputes destructive residue information, ranks survivors, moves away from cold regions, and can persist selectively on exceptional survivors. Compared with pure deep sieving, it treats sieve depth as a dynamic resource rather than an end in itself.

Bahbouhi.b.goldbach.2026 is not a new primality test. Its central question is which $q = E - p$ should be sent to established primality procedures. This separation between candidate selection and primality testing is fundamental.

The originality claim is deliberately restricted: the work proposes and experimentally evaluates an adaptive multilevel architecture integrating established tools with Goldbach-specific destructive filtering, survivor resistance, bilateral failure-state analysis, local rail-preserving probes, q -anchor reuse, diagnostic and persistence modes, and dynamic branch selection. Universal optimality is not claimed.

The present architecture also extends the author's earlier central/logarithmic-window work [20] by allowing multiple search geometries, deep destructive sieving, ranking, branch switching, local probes, and anchor propagation.

Discussion

The principal computational result is the construction and evaluation of an adaptive Goldbach search architecture across a wide range of integer sizes. The retained largest externally verified representation is $E = 10^{10068} + 28878$, $p = 16481$, $q = 10^{10068} + 12397$. Its significance is computational: E was fixed first and the pair was subsequently located through the stated pipeline.

A striking empirical feature of many large successes is the small size of p . This explains why peripheral search

can remain practical for enormous E , but the experiments do not establish a uniformly small or polylogarithmically bounded p for every even integer.

The smallest successful peripheral p is a useful experimental Goldbach-distance measure. Comparisons with logarithmic scales inspired by prime-density heuristics [5] may help calibrate initial windows, but the proposed $\log(E) \log \log(E)$ scale remains heuristic.

Difficult cases reveal a distinction between modular admissibility and simultaneous primality. A region can contain many sieve survivors while producing a long sequence of composite strong-test outcomes. This motivates the operational notion of a finite Goldbach desert.

Negative results are informative: early small divisors validate destructive filtering, while deep survivors that fail strong tests expose the point at which sieving loses predictive value. These distinctions motivated the feedback mechanisms.

Probable-prime and proven-prime statuses remain strictly distinct. GMP [18] and PARI/GP [19] provide powerful computational facilities, but the strength of the reported result is tied to the actual test or certificate used. Database status is likewise separated from mathematical status.

Limitations and Future Improvements

The algorithm does not prove that it terminates with a Goldbach pair for every even integer. No finite collection of successful experiments, regardless of digit length, proves the strong Goldbach conjecture.

The current implementation has no proven uniform complexity bound for the number of candidates required before success. Running time depends on both E size and local candidate behavior.

Survivor ranking, random peripheral sampling, ACHARNEMENT, q -anchor exploration, and adaptive branch selection contain heuristic elements. Future controlled benchmarking should compare them against sequential scanning, fixed-depth sieving, central search, and other baselines.

A future implementation should record wall-clock time, processor time, memory, modular operations, survivors at each depth, and strong tests per successful representation. Parallelization is natural because many subintervals and survivor tests are independent.

Residues of E should be cached aggressively and candidates generated directly from compatible residue classes. Dynamic sieve depth should use measured marginal cost rather than a fixed global threshold.

The experimental Goldbach distance should be measured over much larger controlled families. Detection time and rigorous certification time should also be reported separately, especially at extreme digit lengths where certificate construction can dominate.

The long-term objective is a more deterministic controller in which residue structure, survivor density, and recent branch performance predict where expensive computation should be allocated.

Conclusion

Bahbouhi.b.goldbach.2026 is an adaptive multilevel algorithm for constructing Goldbach prime pairs for prescribed very large even integers. Its design follows a cost hierarchy: inexpensive arithmetic operations are applied first and expensive probable-prime or primality procedures are reserved for resistant candidates.

The architecture combines modulo 6 and modulo 12 classification, admissible rails, destructive congruences, an $r - 1$ divisibility representation, deep modular sieving, survivor ranking, peripheral

and central geometries, local probes, q-anchor propagation, diagnostic analysis, targeted persistence, and adaptive branch switching.

The experimental campaign produced representations at progressively larger scales. The largest externally verified representation retained is $E = 10^{10068} + 28878$, $p = 16481$, $q = 10^{10068} + 12397$. Experiments continued beyond this record to more than 12,000 and 15,000 digits, where the architecture remained operational but cumulative costs increased substantially in long deserts.

The present practical frontier is governed not by decimal length alone but by integer size, survivor density, desert length, sieve cost, strong-test cost, and certification cost. Deeply resistant composites demonstrate that sieve survival must remain strictly separated from primality.

The principal contribution is the integration of established arithmetic tools into an adaptive Goldbach-specific search architecture in which candidate failures, survivor resistance, and branch performance influence subsequent decisions. No proof of Goldbach's conjecture is claimed. Future work should focus on controlled benchmarking, dynamic sieve-depth optimization, parallel implementation, systematic calibration of peripheral distance, and rigorous certification of the strongest candidates.

Author Contributions: Conceptualization, B.B.; methodology, B.B.; algorithm development, B.B.; investigation, B.B.; validation, B.B.; formal analysis, B.B.;

data curation, B.B.; writing-original draft preparation, B.B.; writing-review and editing, B.B. The author has read and agreed to the published version of the manuscript.

Funding: This research received no external funding.
Institutional Review Board Statement: Not applicable.
Informed Consent Statement: Not applicable.

Data Availability Statement: All numerical examples required to reconstruct the principal reported Goldbach representations are provided in the article through exact power-of-ten expressions and integer offsets. The computational procedures necessary to reproduce the candidate-generation and filtering strategy are described in the manuscript. Additional computational records may be made available by the author upon reasonable request.

Acknowledgments: The author acknowledges the use of computational tools and independent number-theoretic services for large-integer testing and verification. Generative artificial intelligence tools were used to assist with language editing, manuscript organization, and presentation. All mathematical claims, numerical results, interpretations, and final manuscript content were reviewed and remain the responsibility of the author.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Deshouillers, J.-M.; te Riele, H.J.J.; Saouter, Y. New experimental results concerning the Goldbach conjecture. In *Algorithmic Number Theory*; Buhler, J.P., Ed.; *Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 1998; Volume 1423, pp. 204-215. <https://doi.org/10.1007/BFb0054863>.
2. Richstein, J. Verifying the Goldbach conjecture up to 4×10^{14} . *Math. Comput.* 2001, 70, 1745-1749. <https://doi.org/10.1090/S0025-5718-00-01290-4>.
3. Saouter, Y. Checking the odd Goldbach conjecture up to 10^{20} . *Math. Comput.* 1998, 67, 863-866. <https://doi.org/10.1090/S0025-5718-98-00928-4>.
4. Oliveira e Silva, T.; Herzog, S.; Pardi, S. Empirical verification of the even Goldbach conjecture and computation of prime gaps up to 4×10^{18} . *Math. Comput.* 2014, 83, 2033-2060. <https://doi.org/10.1090/S0025-5718-2013-02787-1>.
5. Hardy, G.H.; Littlewood, J.E. Some problems of *Partitio Numerorum*; III: On the expression of a number as a sum of primes. *Acta Math.* 1923, 44, 1-70. <https://doi.org/10.1007/BF02403921>.
6. Chen, J.-R. On the representation of a larger even integer as the sum of a prime and the product of at most two primes. *Sci. Sin.* 1973, 16, 157-176.

7. Montgomery, H.L.; Vaughan, R.C. The exceptional set of Goldbach's problem. *Acta Arith.* 1975, 27, 353-370.
8. Helfgott, H.A. The ternary Goldbach conjecture is true. *arXiv* 2013, arXiv:1312.7748.
9. Miller, G.L. Riemann's hypothesis and tests for primality. *J. Comput. Syst. Sci.* 1976, 13, 300-317. [https://doi.org/10.1016/S0022-0000\(76\)80043-8](https://doi.org/10.1016/S0022-0000(76)80043-8).
10. Rabin, M.O. Probabilistic algorithm for testing primality. *J. Number Theory* 1980, 12, 128-138. [https://doi.org/10.1016/0022-314X\(80\)90084-0](https://doi.org/10.1016/0022-314X(80)90084-0).
11. Pomerance, C.; Selfridge, J.L.; Wagstaff, S.S., Jr. The pseudoprimes to 25×10^9 . *Math. Comput.* 1980, 35, 1003-1026. <https://doi.org/10.1090/S0025-5718-1980-0572872-7>.
12. Baillie, R.; Wagstaff, S.S., Jr. Lucas pseudoprimes. *Math. Comput.* 1980, 35, 1391-1417. <https://doi.org/10.1090/S0025-5718-1980-0583518-6>.
13. Atkin, A.O.L.; Morain, F. Elliptic curves and primality proving. *Math. Comput.* 1993, 61, 29-68. <https://doi.org/10.1090/S0025-5718-1993-1199989-X>.
14. Agrawal, M.; Kayal, N.; Saxena, N. PRIMES is in P. *Ann. Math.* 2004, 160, 781-793. <https://doi.org/10.4007/annals.2004.160.781>.
15. Sorenson, J.P. Trading time for space in prime number sieves. In *Algorithmic Number Theory*; Buhler, J.P., Ed.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 1998; Volume 1423, pp. 179-195.
16. Bach, E.; Shallit, J. *Algorithmic Number Theory, Volume 1: Efficient Algorithms*; MIT Press: Cambridge, MA, USA, 1996.
17. Crandall, R.; Pomerance, C. *Prime Numbers: A Computational Perspective*, 2nd ed.; Springer: New York, NY, USA, 2005.
18. GNU Project. GNU MP: Number Theoretic Functions; GNU Multiple Precision Arithmetic Library Documentation. Available online: <https://gmplib.org/manual/Number-Theoretic-Functions> (accessed on 15 September 2026).
19. The PARI Group. PARI/GP User's Guide and Reference Manual. Available online: <https://pari.math.u-bordeaux.fr/doc.html> (accessed on 15 September 2026).
20. Bahbouhi, B. A logarithmic window algorithm for constructive Goldbach representations near the centre of large even integers. *Ann. Math. Phys.* 2026.