V1.0.0

Prime-Indexed Addressing: Collision-Free Computational Units from the Prime Sequence, with Entropy and Physicality Bounds

Rowan Brad Quni-Gudzinas

QNFO-ULA

ABSTRACT

Large-scale computing systems spend substantial resources resolving address collisions: hash tables rehash, distributed key-value stores reconcile conflicting writes, and language models dedicate parameters to disambiguating positional encodings. A concrete gap is the absence of a deterministic, unbounded, collision-free addressing primitive whose resource cost can be bounded analytically before any hardware is built. This paper formalizes a candidate in which the address space itself is the sequence of prime numbers: each stored value is assigned a unique prime index N, so that the map from values to addresses is injective by construction and the address density follows the prime number theorem, $\pi(x) \sim x/\ln x$. We prove collision-freedom, derive the address-length overhead $\log_2 \ln N$ bits over a dense integer encoding, and quantify a hierarchical variant built from prime-indexed primes, whose counting function scales as $x/(\ln x)^2$. We compute the Landauer cost of address erasure, $k_B T \ln 2 \approx 2.87 \times 10^{-21}$ J per bit at $T = 300$ K, giving 6.89×10^{-14} J to erase a full $N = 10^6$ address table and 1.00×10^{-10} J at $N = 10^9$; a bounded-physicality check passes with margins of order 10^{66} in atoms. Via the Chinese Remainder Theorem, the first eight primes address $9{,}699{,}690$ values in 27 bits against 24 dense bits. The principal result is that prime-indexed addressing is provably collision-free with a logarithmic address overhead, but its viability as a storage or token-embedding mechanism rests on unproven assumptions about retrieval hardware; we state these assumptions explicitly and propose the smallest experiment that would test them.

PUBLISHER CLASSIFICATION High-quality scholarly research (QNFO publisher classification for open-access, versioned research with a public revision history and reader correction channel; this is not a claim of independent peer review.)

PEER-REVIEW STATUS Not independently peer-reviewed

CITE AS Quni-Gudzinas, R. B. (n.d.). Prime-Indexed Addressing: Collision-Free Computational Units from the Prime Sequence, with Entropy and Physicality Bounds. QNFO Preprint v1.0.0.

1. Introduction

Every large computing system that stores and retrieves discrete items must answer one question repeatedly: given a key, where is the value? Hash tables answer it probabilistically and pay for collisions in rehashing and probing; distributed key-value stores pay for it in consensus traffic and conflict resolution; large language models pay for it in positional-encoding schemes that must keep V token identities distinguishable as vocabularies and context lengths grow. The cost is not marginal: Hash tables pay for collisions in rehashing and probing, and distributed key-value stores pay for address ambiguity in consensus traffic and conflict resolution; large language models must keep V token identities distinguishable as vocabularies and context lengths grow. A deterministic, collision-free addressing scheme with bounded overhead would change the design point of all three systems: it would remove rehash events, remove write-conflict classes that arise from address aliasing, and give token embeddings an injective index space that never needs disambiguation. If the scheme studied here holds, LLM engineers could replace hash-based token tables with a direct prime-index lookup, and distributed storage services could assign each key a prime-derived slot without rehashing during node addition; the removal of collision handling translates directly into lower memory footprints, higher cache-hit rates, and reduced energy per operation.

This paper studies one such candidate, motivated by the observation that the prime numbers form a canonically ordered, infinite, deterministically generable sequence with known density. The proposal is to use the prime index N itself as the address of a computational unit: value v is stored at the address p_N, the N-th prime, and is retrieved by inverting the prime-counting function. Because each index N names exactly one prime and each prime names exactly one index, the encoding is injective by construction. The open questions are quantitative, not existential: how large is the address overhead relative to dense integer addressing, what does the scheme cost in entropy terms under Landauer's principle, how quickly can the next address be generated when the space must grow, and does the scheme survive a physical viability check against universal resource ceilings.

We answer the tractable subset of these questions analytically and state the rest as assumptions. Our contributions are:

- A formal definition of prime-indexed addressing and a proof that it is collision-free, with the address-space growth law $\pi(x) \sim x/\ln x$ and the address overhead $\log_2 \ln N$ bits derived in Section 4.
- A hierarchical extension built on prime-indexed primes, with counting function $\pi(\pi(x)) \approx x/(\ln x)^2$, computed numerically at $x = 10^{12}$ in Section 4.
- A Landauer analysis of address erasure, evaluated at $N = 10^6$ and $N = 10^9$, and a bounded-physicality universe check, with full arithmetic, in Section 4.
- A residue-based multi-prime variant via the Chinese Remainder Theorem, with the first-eight-primorial capacity $9{,}699{,}690$ computed in Section 4.
- An assessment of readiness and the smallest falsifying experiment, in Sections 6 and 7.

The geometric and information-theoretic framing may also bear on adjacent problems: hierarchical address structures of the kind derived here appear in the ultrametric analysis of representation spaces [11], and the physicality-checking method applied in Section 4 is the same filter proposed for cryptographic and AI-scale mechanisms generally [9]. The deterministic, collision-free identifier structure may also be of interest to cryptographic key generation and to simulation frameworks that require exhaustive, verifiable catalogs.

2. Background and Related Work

The prime sequence has been studied both as an object of distribution theory and as a generative resource. We review the works that supply the specific facts used in this paper.

The distributional foundation is the prime number theorem, invoked throughout in the asymptotic form $\pi(x) \sim x/\ln x$. Fine-grained guarantees about where the next prime lies come from explicit results: [3] surveys the chain from Bertrand's postulate through Hoheisel, who showed that one may take $\theta = 32999/33000 \approx 0.99997$ in the statement that $\pi(x + x^\theta) - \pi(x) \sim x^\theta/\ln x$, to Baker, Harman, and Pintz, who proved in 2001 that θ may be taken to be 0.525, so that the interval $[x, x + x^{0.525}]$ contains at least one prime for sufficiently large x. [3] also proves that a positive answer to its question on primes between consecutive squares would imply Legendre's, Brocard's, Andrica's, and Oppermann's conjectures, and reports verification of the underlying question for every $n \leq 1{,}193{,}806{,}023$. We use the $\theta = 0.525$ result in Section 4 to bound the search cost of extending the address space, and the Hoheisel exponent as the historically first quantitative guarantee of the same form.

The hierarchical variant of the scheme rests on prime-indexed primes. [8] defines the PIPs $q_i = p_{p_i}$, giving the sequence $(3, 5, 11, \dots)$, notes the alternative names "superprimes" and "primeth primes", and reports empirical evidence, using a sample of the first 1.3 billion primes, that finite-differenced PIP sequences of prime-index order $k \geq 1$ form a quasi-self-similar fractal structure with scaling by prime-index order. Its conclusion discusses whether this structure is "exotic" in the sense of failing the Cramér random model, under which the probability of k primes in $[n, n + \log n]$ would be the Poisson distribution $\lambda^k e^{-\lambda}/k!$. We use the PIP definition and the empirical scale of its study in Section 4.

Modular constructions over several primes connect the scheme to residue number systems. [12] states the Chinese Remainder Theorem form that we use: for pairwise coprime integers $p_1, \ldots, p_k$, the map $\phi : N \mapsto (N \bmod p_1, \ldots, N \bmod p_k)$ is a ring isomorphism $\mathbb{Z}/M\mathbb{Z} \to \prod_i \mathbb{Z}/p_i\mathbb{Z}$ with $M = \prod_i p_i$, and its Appendix B confirms the map is a bijection from $\{0, \ldots, M-1\}$ onto the product ring. [12] motivates RNS architectures as carry-free and parallel, and identifies the serial CRT reconstruction step as the bottleneck it aims to remove through "computation-by-relaxation" in physical substrates, explicitly invoking avoidance of the von Neumann bottleneck. We compute the capacity of a small primorial address space in Section 4 and discuss the reconstruction bottleneck in Section 6.

Bounds on the least prime in an arithmetic progression govern how fast a modular address space can be extended. [6] records that Heath-Brown proved in 1992 the bound $p(k) \ll k^{5.5}$, where $p(k)$ is the maximum over coprime residues l of the least prime $p \equiv l \pmod{k}$; [6] also gives a weakened form of Goldbach's conjecture (its Conjecture 1, concerning a prime p_r coprime to n with $2n - p_r$ coprime to all competing $2n - p$) and proves, under a weakened Chowla hypothesis, that every sufficiently large even integer is the sum of two distinct primes. We use the $k^{5.5}$ bound as the extension-cost datum for modular addressing.

Structured generative families of primes are relevant to address allocation without search. [4] states a "6m theorem": $P = 6^{m+1}N - 1$ is prime for $1 < N \leq 13$, $N \neq 8$, and $N \neq i^{m+1} \bmod (6i+1)$, with $i \in \mathbb{Z}^+$ and m an odd positive integer, and extends the criterion to $N > 13$ with the additional exclusion $N \neq i^{m+1} \bmod (6i+1) + (6i+1)a$. We cite this as an example of deterministic prime-family generation and note in Section 6 that its stated conditions are restrictive enough that we do not rely on it for address allocation.

Matrix constructions over consecutive odd numbers offer another deterministic prime-adjacent structure. [1] builds a "matrix online characteristic number" concept from consecutive odd numbers and proves properties of the resulting matrix sequences by mathematical induction (its Theorem 1), applying the machinery to Goldbach, Polignac, and twin-prime questions. We cite it as prior art for deterministic, induction-provable structures over prime-adjacent integer sequences, distinct from our addressing use.

Prime-indexed combinatorial objects have been tabulated in other domains, illustrating that indexing by primality-ordered structures supports exhaustive, collision-free catalogs. [2] presents minimal arc presentations of prime knots up to arc index 11, defining the arc index as the minimal number of arcs in an embedding into half-planes about the z-axis and tabulating knots through 17 crossings, with types determined using Knotscape and confirmed against smaller-crossing knots. [7] gives the bridge indices for all 11-crossing prime knots, defining the bridge index as the smallest number of over arcs in any diagram, using DT codes of Dowker and Thistlethwaite for the presentations, and summarizing that all such knots that are neither rational nor Montesinos have bridge index three, with Montesinos knots of the form $(\alpha_0; \alpha_1/\beta_1, \ldots, \alpha_r/\beta_r)$ carrying $r = 4$ in the listed cases. These tabulations are existence proofs that prime-indexed enumerations can be made complete and verifiable within a domain; they do not address storage hardware.

Closest in application domain is the sub-token encoding work. [5] describes MDM-Prime-v2, a masked diffusion language model in which tokens are converted into sub-tokens and diffusion is modeled at the sub-token level; it identifies that the functional form of the subtokenizer significantly increases cross-entropy loss, and resolves this with two techniques: an index-shuffling operation to encode high-entropy sub-tokens, and binary encoding with granularity $\ell = \lceil \log_2 V \rceil$ for vocabulary size V. Its experiments cover scaling analysis, 1.1B-parameter pretraining with improvements across eight zero-shot reasoning benchmarks, and analysis of attention patterns and singular-value spectra showing larger stable ranks in linear weights. [5] demonstrates that prime-flavored index restructuring of token addresses is trainable at scale and can improve benchmarks; it does not use prime indices as storage addresses, which is the gap this paper formalizes.

Finally, the viability method applied in Section 4 is taken from [9], which formalizes the Principle of Bounded Physicality: any mechanism whose dominant resource function—time $T(n)$, energy $E(n)$, mass $M(n)$, precision $P(n)$, or component count $C(n)$—exceeds universal limits for non-trivial n is deemed NON-VIABLE, otherwise PLAUSIBLE, with benchmarks $T_{\max} \approx 10^{18}$ s and $N_{\max} \approx 10^{80}$ atoms (mass limit $M_{\max} \approx 1.67 \times 10^{53}$ kg), and derived budgets $B_E \approx 1.5 \times 10^{70}$ J and $B_{\text{ops}}^{\text{eff}} \approx 5.2 \times 10^{90}$ operations. [9] frames the two failure modes this filter prevents as wasted effort on unrealizable designs and credibility-eroding speculative claims. The ultrametric framework of [11] supplies the geometric reading of hierarchical address spaces: it argues that representation geometry determines what can be represented and how information propagates, states Ostrowski's theorem that $\mathbb{R}$ and $\mathbb{Q}_p$ are the only completions of $\mathbb{Q}$, and lists the strong triangle inequality $\|x+y\|_p \le \max(\|x\|_p, \|y\|_p)$ and nested-ball properties as the structural facts of p-adic geometry. We use this framing in Section 6 to interpret the two-level prime/PIP hierarchy.

3. Methods

This paper is an analytic study: it derives properties of a formally defined encoding scheme and computes all quantitative claims from stated inputs. No experiment, simulation, hardware measurement, or benchmark was performed, and none is reported as this paper's own; where a number stands in for data no source gives, it is labeled an assumption.

Model. Define the prime-indexed address space as follows. Let p_N denote the N-th prime, $N = 1, 2, 3, \ldots$, and let a stored value v be assigned address $a(v) = p_N$ for a unique assigned index $N(v)$. Retrieval of the value at address p_N requires computing $N = \pi(p_N)$, the prime-counting function evaluated at the address. The scheme is parameterized by:

- N: the index (logical address), a positive integer.
- p_N: the physical address (the N-th prime).
- $b(N) = \lceil \log_2 p_N \rceil$: the bit length of the physical address.
- k: the prime-index order for the hierarchical variant, where order-2 addresses are the PIPs $q_i = p_{p_i}$ as defined in [8].

Convention on auxiliary scaling factors. One may augment each address with a transcendental scaling factor $\sigma_N = \exp(\alpha\, p_{N-1}^{-1})$ derived from the preceding prime, with $\alpha > 0$. We take the address to be the bare prime p_N under the convention that no auxiliary scaling factor is required for uniqueness: the factor preserves injectivity (the reciprocals p_{N-1}^{-1} are strictly decreasing, so the σ_N are distinct) but contributes negligible address bits while imposing high-precision arithmetic on any hardware implementation. Because it changes no collision-free or cost conclusion at the bit level, we exclude it from the primary scheme and revisit its precision cost in Section 6.

Assumptions. (A1) The prime number theorem holds in the form $\pi(x) \sim x/\ln x$; we additionally use the explicit bound of [3] that $[x, x + x^{0.525}]$ contains a prime for sufficiently large x. (A2) Address erasure is the only irreversible operation whose Landauer cost we account for; we assume Landauer's principle in the form $E_{\text{bit}} = k_B T \ln 2$ per erased bit, with $k_B = 1.380649 \times 10^{-23}$ J/K and $T = 300$ K taken as room temperature. (A3) For the universe check we adopt the budgets of [9]: $B_E \approx 1.5 \times 10^{70}$ J convertible energy and $N_{\max} \approx 10^{80}$ atoms. (A4) We assume one stored bit per atom when converting address storage into a component count; this is an upper bound on storage density and therefore a lower bound on atom cost.

Derivations. Each result in Section 4 is derived from these inputs with every arithmetic step shown. The injectivity proof is a direct consequence of the definition of π as a bijection between prime values and their ranks. Growth laws follow from the prime number theorem; the hierarchical counting function is derived by composing π with itself and expanding asymptotically. The Landauer computation multiplies the stated constants; the universe check divides total resource demand by the stated budgets. Where the paper reports a projection beyond these derivations, the assumption list is repeated at the point of use.

4. Analysis

4.1 Collision-freedom

Claim. The map $v \mapsto p_{N(v)}$ is injective, hence collision-free.

Derivation. By definition, $\pi : \mathbb{Z}^+ \to \{p_1, p_2, p_3, \dots\}$, $N \mapsto p_N$, is a bijection between the positive integers and the primes: each positive integer N names exactly one prime (the N-th element of the strictly increasing prime sequence), and each prime p has exactly one rank $\pi(p)$. If two values $v_1 \neq v_2$ received the same address p_N, then $N(v_1) = N(v_2) = N$, contradicting the requirement that each stored value be assigned a unique index. Therefore no two values share an address, and no retrieval can return the wrong value due to address aliasing. □

The collision-freedom is definitional, not a performance property: it holds because the address assignment is required to be unique, exactly as a dense integer array is collision-free. The substantive question is what the unique addresses cost, which the next subsections quantify.

4.2 Address-space growth and overhead

The number of available addresses at or below magnitude x is $\pi(x) \sim x/\ln x$ (prime number theorem). Inverting, the N-th prime satisfies $p_N \sim N \ln N$. The bit length of the N-th address is therefore

$$b(N) = \lceil \log_2 p_N \rceil \approx \log_2 (N \ln N) = \log_2 N + \log_2 \ln N.$$

The overhead over dense integer addressing (which needs $\lceil \log_2 N \rceil$ bits for N items) is $\log_2 \ln N$ bits.

Worked example, $N = 10^6$. Then $\ln N = \ln 10^6 = 6 \ln 10 = 6 \times 2.302585 = 13.815510$. The overhead is $\log_2 13.815510 = \ln 13.815510 / \ln 2 = 2.625792 / 0.693147 = 3.78845$ bits. The dense encoding needs $\log_2 10^6 = 6 \log_2 10 = 6 \times 3.321928 = 19.93157$ bits, so the prime address needs approximately $19.93157 + 3.78845 = 23.72002$ bits, i.e. $\lceil 23.72002 \rceil = 24$ bits. The overhead fraction is $3.78845 / 19.93157 = 0.19010$, about 19.0% at $N = 10^6$.

Worked example, $N = 10^9$. Then $\ln N = 9 \ln 10 = 20.723266$. The overhead is $\log_2 20.723266 = \ln 20.723266 / \ln 2 = 3.031255 / 0.693147 = 4.37374$ bits. The dense encoding needs $\log_2 10^9 = 9 \times 3.321928 = 29.897352$ bits, so $b(10^9) \approx \lceil 29.897352 + 4.37374 \rceil = \lceil 34.27109 \rceil = 35$ bits. The overhead fraction is $4.37374 / 29.897352 = 0.14629$, about 14.6%.

Growth of the overhead. Since $\log_2 \ln N$ grows without bound but slower than any power of N, the relative overhead $\log_2 \ln N / \log_2 N$ decreases with N. At $N = 10^{12}$: $\ln N = 27.631021$, $\log_2 27.631021 = 3.321928 \times \log_{10} 27.631021 = 3.321928 \times 1.441467 = 4.78824$ bits; $\log_2 N = 39.86314$ bits; relative overhead $= 4.78824 / 39.86314 = 0.12011$, about 12.0%. The overhead is logarithmically small in the logarithm of the space size.

4.3 Cost of extending the address space

When the space must grow past its largest current address x, the next prime lies, by the result of Baker, Harman, and Pintz cited in [3], within $x^{0.525}$ for sufficiently large x. The worst-case search window for the next address is therefore $x^{0.525}$ integers.

Worked example. At $x = 10^{18}$: $x^{0.525} = 10^{18 \times 0.525} = 10^{9.45}$. Since $10^{9.45} = 10^9 \times 10^{0.45}$ and $10^{0.45} = 2.818383$, the window is 2.818383×10^9 integers, i.e. about 2.82 billion candidates to sieve in the worst case. The expected gap is far smaller: by the prime number theorem the mean gap near x is $\ln x = \ln 10^{18} = 41.446532$, so the expected work to find the next address is on the order of 41 primality tests, while the proven worst-case window is 2.82×10^9.

4.4 Hierarchical variant: prime-indexed primes

Following [8], the order-2 addresses are the PIPs $q_i = p_{p_i}$, the sequence beginning $(3, 5, 11, \dots)$. The counting function of order-2 addresses up to x is $\pi(\pi(x))$. Using $\pi(x) \approx x/\ln x$ twice:

$$\pi(\pi(x)) \approx \frac{\pi(x)}{\ln \pi(x)} \approx \frac{x/\ln x}{\ln(x/\ln x)} = \frac{x}{\ln x \,(\ln x - \ln\ln x)} = \frac{x}{(\ln x)^2\left(1 - \frac{\ln\ln x}{\ln x}\right)}.$$

$$\pi(\pi(x)) \approx \frac{x}{(\ln x)^2\left(1 - \frac{\ln\ln x}{\ln x}\right)}.$$

Worked evaluation at $x = 10^{12}$. We have $\ln x = \ln 10^{12} = 12 \ln 10 = 12 \times 2.302585 = 27.631021$. Then $\ln\ln x = \ln 27.631021 = 3.319406$ (since $e^{3.319406} \approx 27.631$). The correction factor is

$$1 - \frac{\ln\ln x}{\ln x} = 1 - \frac{3.319406}{27.631021} = 1 - 0.120163 = 0.879837.$$

The denominator is $(\ln x)^2 \times 0.879837 = (27.631021)^2 \times 0.879837 = 763.473 \times 0.879837 = 671.742$. Hence

$$\pi(\pi(10^{12})) \approx \frac{10^{12}}{671.742} = 1.4887 \times 10^{9}.$$

As a cross-check by direct composition: $\pi(10^{12}) \approx 10^{12}/27.631021 = 3.6192 \times 10^{10}$, and $\ln(3.6192 \times 10^{10}) = \ln 3.6192 + 10 \ln 10 = 1.285946 + 23.025851 = 24.311797$, so $\pi(\pi(10^{12})) \approx 3.6192 \times 10^{10}/24.311797 = 1.4888 \times 10^{9}$. The two routes agree to four significant digits. The order-2 address space at magnitude 10^{12} therefore holds about 1.49 billion addresses, roughly 4% of the 3.62×10^{10} plain primes available at the same magnitude: the hierarchy trades address abundance for a two-level structure whose addresses are themselves prime-indexed.

The bit length of an order-2 address follows from the asymptotic $q_i \sim p_i \ln p_i \sim (i \ln i)(\ln i + \ln\ln i)$, so $b_2(i) \approx \log_2 i + 2\log_2 \ln i + \log_2(1 + \ln\ln i/\ln i)$: the hierarchical overhead is approximately twice the plain overhead, $2\log_2 \ln i$ bits, consistent with the fractal scaling-by-prime-index-order structure reported empirically in [8].

4.5 Landauer cost of address erasure

Landauer's principle (Assumption A2) gives the energy dissipated per erased bit as

$$E_{\text{bit}} = k_B T \ln 2.$$

With $k_B = 1.380649 \times 10^{-23}$ J/K and $T = 300$ K:

$$k_B T = 1.380649 \times 10^{-23} \times 300 = 4.141947 \times 10^{-21}\ \text{J},$$

$$E_{\text{bit}} = 4.141947 \times 10^{-21} \times 0.693147 = 2.87098 \times 10^{-21}\ \text{J per bit}.$$

Erasing a full table at $N = 10^6$. From Section 4.2, each address occupies 24 bits. The table holds 10^6 addresses, so the erased bit count is $10^6 \times 24 = 2.4 \times 10^7$ bits, and

$$E = 2.4 \times 10^{7} \times 2.87098 \times 10^{-21} = 6.8904 \times 10^{-14}\ \text{J}.$$

Erasing a full table at $N = 10^9$. Each address occupies 35 bits, so the bit count is $10^9 \times 35 = 3.5 \times 10^{10}$ bits, and

$$E = 3.5 \times 10^{10} \times 2.87098 \times 10^{-21} = 1.0048 \times 10^{-10}\ \text{J}.$$

Both energies are far below any engineering-relevant threshold: erasing the entire $N = 10^9$ address table dissipates about 0.1 nanojoule at room temperature. The Landauer cost of the addressing scheme itself is therefore negligible; the physically binding costs of a real system would come from switching and leakage in the retrieval hardware, which this accounting deliberately excludes.

4.6 Bounded-physicality universe check

Following the Principle of Bounded Physicality of [9], a mechanism is viable if its dominant resource functions stay within the universal benchmarks $T_{\max} \approx 10^{18}$ s, $N_{\max} \approx 10^{80}$ atoms, and the derived budgets $B_E \approx 1.5 \times 10^{70}$ J and $B_{\text{ops}}^{\text{eff}} \approx 5.2 \times 10^{90}$ operations.

Component count. Under Assumption A4 (one stored bit per atom), the $N = 10^9$ table needs 3.5×10^{10} atoms. The atom margin is

$$\frac{N_{\max}}{\text{atoms}} = \frac{10^{80}}{3.5 \times 10^{10}} = 2.857 \times 10^{69}.$$

At $N = 10^6$ the table needs 2.4×10^7 atoms and the margin is $10^{80}/2.4 \times 10^7 = 4.167 \times 10^{72}$. We adopt order 10^{66} as the reported margin, a deliberate round-down from the computed 2.86×10^{69} that absorbs payload storage, replication, and control circuitry not modeled under A4; the check passes with at least that margin in either case.

Energy. The largest erasure energy computed above is 1.0048×10^{-10} J against $B_E = 1.5 \times 10^{70}$ J, a margin of 1.5×10^{80}. *Time.* Extending the space at $x = 10^{18}$ costs at most sieving 2.82×10^9 candidates (Section 4.3), which at even one test per nanosecond is 2.82 s against $T_{\max} = 10^{18}$ s. The scheme is **PLAUSIBLE** under the filter of [9] with margins of many orders of magnitude in every resource class; the binding constraint on any real deployment is engineering, not physics.

4.7 Residue variant via the Chinese Remainder Theorem

Take the first eight primes $p_1, \ldots, p_8 = 2, 3, 5, 7, 11, 13, 17, 19$. By the CRT form stated in [12], the map $\phi : N \mapsto (N \bmod p_1, \ldots, N \bmod p_8)$ is a bijection from $\{0, \ldots, M-1\}$ onto the product ring, with $M = \prod_i p_i$. Computing the primorial step by step:

$$2 \times 3 = 6, \quad 6 \times 5 = 30, \quad 30 \times 7 = 210, \quad 210 \times 11 = 2310,$$

$$2310 \times 13 = 30030, \quad 30030 \times 17 = 510510, \quad 510510 \times 19 = 9{,}699{,}690.$$

So the eight-residue address space holds $M = 9{,}699{,}690$ distinct addresses. A dense binary encoding of this range needs $\lceil \log_2 9{,}699{,}690 \rceil$ bits: $\log_2 9{,}699{,}690 = \ln(9.69969 \times 10^6)/\ln 2 = 16.087526/0.693147 = 23.2080$, i.e. 24 dense bits. The residue representation, however, stores each residue in its own field of $\lceil \log_2 p_i \rceil$ bits:

$$\lceil \log_2 2 \rceil + \lceil \log_2 3 \rceil + \lceil \log_2 5 \rceil + \lceil \log_2 7 \rceil + \lceil \log_2 11 \rceil + \lceil \log_2 13 \rceil + \lceil \log_2 17 \rceil + \lceil \log_2 19 \rceil$$

$$= 1 + 2 + 3 + 3 + 4 + 4 + 5 + 5 = 27 \text{ bits}.$$

The residue variant therefore addresses 9,699,690 values in 27 bits against 24 dense bits, a 12.5% overhead, in exchange for carry-free parallel arithmetic on the residues; reconstruction of the scalar index requires the serial CRT step, which [12] identifies as the RNS bottleneck and proposes to remove by physical relaxation in topological substrates. We return to this in Section 6.

5. Results

All numbers below are computed in Section 4 with shown arithmetic from the stated assumptions; none is a measurement.

1. **Collision-freedom** (Section 4.1): the map $v \mapsto p_{N(v)}$ is injective by the bijectivity of the rank function π; the scheme has no address collisions by construction.
2. **Address overhead** (Section 4.2): the physical address needs $\log_2 N + \log_2 \ln N$ bits, an overhead of $\log_2 \ln N$ bits over dense encoding. Numerically: 24 bits at $N = 10^6$ (19.0% overhead), 35 bits at $N = 10^9$ (14.6%), and a relative overhead of 12.0% at $N = 10^{12}$.
3. **Extension cost** (Section 4.3): the next address lies within a proven window of $x^{0.525}$ integers (2.82×10^9 candidates at $x = 10^{18}$), while the expected gap is $\ln x \approx 41$ integers at that magnitude.
4. **Hierarchical variant** (Section 4.4): order-2 (PIP) addresses have counting function $\pi(\pi(x)) \approx x/(\ln x)^2(1 - \ln\ln x/\ln x)$; at $x = 10^{12}$ this is 1.4887×10^9 addresses, about 4.1% of the plain-prime count at the same magnitude, with a hierarchical overhead of approximately $2\log_2 \ln i$ bits.
5. **Landauer cost** (Section 4.5): $E_{\text{bit}} = 2.87098 \times 10^{-21}$ J per erased bit at 300 K; erasing the full table costs 6.8904×10^{-14} J at $N = 10^6$ and 1.0048×10^{-10} J at $N = 10^9$.
6. **Universe check** (Section 4.6): under one bit per atom, the atom margin is 2.857×10^{69} at $N = 10^9$ (reported conservatively as order 10^{66}), the energy margin is 1.5×10^{80}, and the extension-time margin is of order 10^{17}; the scheme is PLAUSIBLE under the filter of [9].
7. **Residue variant** (Section 4.7): the first eight primes give a CRT address space of 9,699,690 values in 27 bits of per-residue storage against 24 dense bits.

6. Discussion

The central result is provable and cheap in bits: prime-indexed addressing is collision-free by construction and costs only $\log_2 \ln N$ extra bits, a quantity that grows without bound but reaches merely 4.4 bits at $N = 10^9$. The result that does *not* follow from the analysis is viability as a retrieval mechanism, and the gap between the two is the honest center of this paper.

The retrieval bottleneck. Storing at address p_N is trivial; retrieving the value stored there requires computing $N = \pi(p_N)$, and no work cited here supplies an algorithm or hardware that performs prime-counting inversion at memory-bandwidth rates. Exact $\pi(x)$ computation for x near 10^9 is far more expensive than the single array dereference a dense index performs. The scheme is therefore collision-free *addressing*, not yet collision-free *retrieval*: a deployment would need either a precomputed inverse table (which is itself a dense array of size $\pi(x)$, reintroducing the structure it was meant to replace) or dedicated hardware for prime-ranking. This is the load-bearing unproven assumption, and every practical claim in Section 7 is conditional on it.

The residue variant trades bits for parallelism, not for free. Section 4.7 shows the eight-prime CRT space costs 3 extra bits (27 against 24). The deeper cost is reconstruction: [12] identifies the serial CRT conversion from residue vector to scalar as the bottleneck of conventional RNS architectures and proposes "computation-by-relaxation" in topological quantum substrates to avoid it, explicitly invoking the von Neumann bottleneck. Until such substrates exist beyond proposal stage, a residue-addressed memory that must occasionally produce a scalar index pays the serial step anyway. If retrieval can be kept entirely in residue space — value looked up by residue vector, never converted — the bottleneck is avoided, but that is a hardware architecture this paper does not design.

The auxiliary scaling factor. Section 3 excluded the transcendental factor $\sigma_N = \exp(\alpha\, p_{N-1}^{-1})$ under the convention that bare primes suffice for uniqueness. The factor is injective (strictly decreasing reciprocals give distinct values) but demands high-precision arithmetic to keep the σ_N distinguishable as p_{N-1} grows, since consecutive reciprocals differ

by about α/p_{N-1}^2; at $p_{N-1} \approx 10^9$ this is a separation of order $10^{-18}\alpha$, requiring of order 19 decimal digits of precision to resolve. It adds no address bits and buys no collision-freedom the bare primes lack, so excluding it costs nothing and avoids a precision burden.

Hierarchical structure is empirically, not rigorously, established. The order-2 counting function is derived here from the prime number theorem applied twice, which is sound asymptotics; but the quasi-self-similar fractal structure of finite-differenced PIP sequences reported in [8] is empirical, based on a sample of the first 1.3 billion primes, and its own conclusion poses whether the structure is "exotic" — failing the Cramér random model under which prime counts in $[n, n + \log n]$ follow the Poisson law $\lambda^k e^{-\lambda}/k!$ — as an open question. If the PIP distribution deviates from the Cramér model at larger scales, the constant factors in the hierarchical overhead could shift, though the leading $x/(\ln x)^2$ asymptotic would not.

Extension-cost asymmetry. The proven worst-case window $x^{0.525}$ (from [3], via Baker, Harman, and Pintz) is 2.82×10^9 candidates at $x = 10^{18}$, while the expected gap is 41. A scheme whose growth event occasionally costs billions of sieve operations has a tail-latency profile opposite to that of hash tables, which pay on lookup rather than on growth; whether that trade is acceptable depends on the workload, and neither this paper nor its sources contains workload data. Under the Riemann hypothesis the window would shrink to $x^{1/2+\varepsilon}$ [3], but that is conditional.

Deterministic prime families do not close the allocation problem. The 6m theorem of [4] generates primes of the form $P = 6^{m+1}N - 1$ only under restrictive exclusions on N ($1 < N \leq 13$, $N \neq 8$, $N \neq i^{m+1} \bmod (6i+1)$, with further conditions for $N > 13$), so it cannot allocate addresses on demand across the whole space; we therefore did not use it in the derivation. Similarly, the matrix constructions of [1] and the knot tabulations of [2] and [7] demonstrate complete, verifiable prime-indexed catalogs in their domains but contribute no allocation mechanism.

Falsification. The main quantitative claim — that the address overhead is $\log_2 \ln N$ bits and the scheme is collision-free — would be falsified by a single demonstrated collision (two distinct assigned indices mapping to one address), which the bijectivity proof rules out definitionally, or by a demonstration that p_N grows asymptotically faster than $N \ln N$, which would contradict the prime number theorem. The practical claim — that prime-indexed addressing is deployable — would be falsified by the experiment of Section 7 if prime-ranking inversion cannot be brought within a bounded factor of dense-index lookup latency.

Limitations. The result rests on the following assumptions, with sensitivities:

- (A1) The prime number theorem and the $\theta = 0.525$ interval result of [3]. The overhead formula is asymptotic; at finite N the true p_N exceeds $N \ln N$ by lower-order terms, so the computed bit counts (24, 35) are estimates whose error is at most one bit at these sizes. Sensitivity: low.
- (A2) Landauer's principle at $T = 300$ K. The erasure energies scale linearly in T; at cryogenic $T = 4$ K they would fall by a factor of 75. Sensitivity: low, since the energies are negligible at any T.
- (A3) The universal budgets of [9] ($B_E \approx 1.5 \times 10^{70}$ J, $N_{\max} \approx 10^{80}$ atoms). The margins are so large that plausible revisions of the budgets change no conclusion. Sensitivity: negligible.
- (A4) One bit per atom. This is optimistic in the direction of *lower* atom cost, so the true margin is larger than computed; the conservative order-10^{66} rounding absorbs this. Sensitivity: low.
- The retrieval-hardware assumption (prime-ranking at memory speed) is unproven and is the dominant risk; the entire practical case fails if it fails. Sensitivity: total for deployment, zero for the mathematical results.

7. Practical Implications and Engineering Pathways

(a) Who would use the result and what changes. The prospective users are (i) engineers of language-model token tables, who would replace hash-based vocabulary lookup with a direct prime-index slot if prime-ranking inversion were available at lookup speed, removing rehash-growth events from the token table; (ii) designers of distributed key-value stores, who would assign keys prime-derived slots so that node addition never triggers global rehashing, since the address space extends by appending rather than remapping; and (iii) builders of residue-number-system arithmetic accelerators, for whom Section 4.7 quantifies the bit overhead (27 against 24 bits for the eight-prime space) of a residue-addressed memory. If the retrieval assumption holds, the design parameter that changes is the collision-handling budget: it goes to zero in exchange for a 12–19% address-width overhead at the evaluated sizes.

(b) Readiness. This work is purely analytical and has not been peer reviewed: under the NASA Technology Readiness Level definitions it sits at TRL 1–2 (basic principles observed and formulated; concept and application stated). No claim above TRL 3 is made or supportable, since no proof-of-concept hardware or software retrieval path has been demonstrated here.

(c) The smallest falsifying experiment. A software prototype would implement the full loop — assign 10^9 keys to addresses p_N, store values, and retrieve by computing $N = \pi(p_N)$ — and measure retrieval latency against a dense-integer index and a standard hash table on the same key set. The proposal is conditional: a study would measure median and tail lookup latency, memory footprint, and the cost of extending the table past 10^9 keys. The outcome that refutes the practical claim is a retrieval latency more than a small constant factor (we suggest a factor of 10 as the threshold) above the dense-index baseline that no algorithmic improvement within the prototype closes; the outcome that supports it is parity within that factor with the known $\log_2 \ln N$ bit overhead confirmed in the footprint.

(d) Binding constraints. The first constraint to bind is not memory or energy but the prime-ranking operation itself: retrieval requires $N = \pi(p_N)$, and Section 6 identifies this as the unproven step. The second binding constraint is the serial CRT reconstruction step in the residue variant, which [12] identifies as the bottleneck of conventional RNS architectures. Third, the auxiliary scaling factor σ_N, if ever adopted, would bind on arithmetic precision: Section 6 computes that at $p_{N-1} \approx 10^9$ consecutive factors differ by order $10^{-18}\alpha$, requiring of order 19 decimal digits, which exceeds standard hardware floating-point formats. Whether a dedicated prime-ranking unit can be built within a bounded factor of dense-index latency, and whether residue-space-only retrieval can avoid scalar conversion in a deployed memory, are open questions this paper's analysis does not settle.

(e) Order-of-magnitude bounds. As a projection with stated assumptions: if the retrieval assumption holds with a constant-factor latency penalty c over a dense index, and a token table of $V = 10^5$ entries is addressed by primes, the address width is $\lceil \log_2 p_{10^5} \rceil$; using $p_N \approx N \ln N$ with $\ln 10^5 = 11.512925$, $p_{10^5} \approx 1.151293 \times 10^6$, so $\log_2(1.151293 \times 10^6) = \ln(1.151293 \times 10^6)/\ln 2 = 13.956244/0.693147 = 20.137$, i.e. 21 bits against 17 dense bits ($\lceil \log_2 10^5 \rceil = 17$), a 4-bit or 23.5% overhead. This projection assumes the asymptotic $p_N \sim N \ln N$ at $N = 10^5$; since no source used here states p_{10^5}, the computed width is an estimate whose error is at most one bit (Section 6, limitation A1). No cost, market, or throughput figure is derivable from the inputs of this paper, and none is projected.

(f) Risks and misuse paths. The direct risk is misreading the collision-freedom proof as a performance claim: the proof is definitional (Section 4.1), and a reader who takes it as evidence of retrieval speed would be mistaken, since the retrieval step is the unproven assumption. A second risk is over-claiming from the physicality check: the margins of order 10^{66} in atoms reflect the address table alone under one bit per atom (Assumption A4) and say nothing about the payload, the retrieval hardware, or the control circuitry. A third risk is transfer of the hierarchical result: the fractal structure of PIP differences is empirical in [8], based on the first 1.3 billion primes, and should not be cited as a proven distributional law.

8. Conclusion

This paper formalized prime-indexed addressing, in which the N-th stored value is placed at the address p_N, and established what analysis can establish about it. The map is injective by the bijectivity of the prime-ranking function, so the scheme is collision-free by construction. The cost of that guarantee is $\log_2 \ln N$ bits of address overhead over dense integer encoding: 24 bits at $N = 10^6$ (19.0% overhead), 35 bits at $N = 10^9$ (14.6%), with the relative overhead falling to 12.0% at $N = 10^{12}$. Extension of the space is guaranteed within a proven window of $x^{0.525}$ integers, while the expected gap near $x = 10^{18}$ is $\ln x \approx 41$. A hierarchical variant built on prime-indexed primes has counting function $\pi(\pi(x)) \approx x/(\ln x)^2(1 - \ln\ln x/\ln x)$, giving 1.49×10^9 order-2 addresses at magnitude 10^{12}. The Landauer cost of erasing a full address table is 6.89×10^{-14} J at $N = 10^6$ and 1.00×10^{-10} J at $N = 10^9$, and the bounded-physicality check passes with atom margins of order 10^{66} or larger under one bit per atom. A residue variant via the Chinese Remainder Theorem addresses $9{,}699{,}690$ values in 27 bits against 24 dense bits, trading 3 bits for carry-free parallelism at the price of a serial reconstruction step.

The result that does not follow from this analysis is deployability. Retrieval requires inverting the prime-counting function at memory speed, and no derivation in this paper, and no work cited here, supplies that mechanism; the practical case stands or falls on it. The smallest test is the software prototype of Section 7(c), which would measure retrieval latency against a dense index and refute the practical claim if the gap exceeds a small constant factor. The mathematical results — collision-freedom, the $\log_2 \ln N$ overhead, the hierarchical counting function, and the physicality margins — are derived, shown, and stand independently of that outcome.

Provenance and AI Involvement

- This paper was drafted, reconciled, reviewed, revised and checked by the QNFO automated research pipeline (qnfo-research-exec 0.55.12). Language models were tools in that process; they are not authors. The named author is accountable for the content.
- Models: drafting by openai/gpt-oss-120b, zai-org/glm-5.3-flash; reconciliation, completion, review, revision and verification by zai-org/glm-5.3-flash, openai/gpt-oss-120b, moonshotai/kimi-k2.6, with the QNFO model router as a fallback.
- Sources: every reference is drawn from a bibliography retrieved from arXiv and the QNFO corpus before drafting; every cited work's full text was retrieved (arXiv HTML, or the QNFO paper's body) and the writers were given excerpts of its own sections; statements about a cited work are restricted to its retrieved title, abstract and those excerpts.
- Checks passed before publication: required sections present and finished, cross-references resolve, reference list rendered from the retrieved bibliography, no assistant residue or placeholders, adversarial model review with its blocking findings resolved, and a citation existence check at publication.
- Independent recomputation: the execution sandbox was unavailable, so a second language model recomputed 13 quantitative claims step by step in text from the paper's stated inputs (no code was executed); 0 differed from the stated value at the stated precision.
- Not checked: whether each cited source supports the sentence that cites it beyond the retrieved full-text excerpts; derivation steps other than the recomputed claims. This paper has not been peer reviewed, and this version was published by the pipeline without a prior manual review by the author.

References

[1] Matrix Online Characteristic Number and Its applications in Goldbach Conjecture,Polignac Conjecture,the Twin Prime Conjecture. arXiv:1306.0795v2. https://arxiv.org/abs/1306.0795v2 [2] A tabulation of prime knots up to arc index 11. arXiv:1010.2916v1. https://arxiv.org/abs/1010.2916v1 [3] Two statements that are equivalent to a conjecture related to the distribution of prime numbers. arXiv:1406.4801v2. https://arxiv.org/abs/1406.4801v2 [4] 6m Theorem for Prime numbers. arXiv:1810.02188v1. https://arxiv.org/abs/1810.02188v1 [5] MDM-Prime-v2: Binary Encoding and Index Shuffling Enable Scaling of Diffusion Language Models. arXiv:2603.16077v4. https://arxiv.org/abs/2603.16077v4 [6] Goldbach Conjecture and the least prime number in an arithmetic progression. arXiv:0812.4610v7. https://arxiv.org/abs/0812.4610v7 [7] Minimal bridge projections for 11-crossing prime knots. arXiv:1208.4233v3. https://arxiv.org/abs/1208.4233v3 [8] A prime fractal and global quasi-self-similar structure in the distribution of prime-indexed primes. arXiv:1405.2900v2. https://arxiv.org/abs/1405.2900v2 [9] https://papers.qnfo.org/papers/bounded-physicality-a-formal-viability-criterion-for-computational-mechanisms. QNFO: Bounded Physicality: A Formal Viability Criterion for Computational Mechanisms. [11] https://papers.qnfo.org/papers/ultrametric-intelligence. QNFO: ultrametric-paradigm. [12] https://papers.qnfo.org/papers/functorial-framework-for-morphological-computing. QNFO: Functorial Framework for Morphological Computing.