

J -Cost-Stable Codes: Zeckendorf Representation as a Foundation for Data Compression, Error Detection, and Cryptographic Primitives

Sebastian Pardo-Guerra*, Anil Thapa†, Jonathan Washburn‡

March 2, 2026

Abstract

Zeckendorf’s theorem states that every positive integer has a unique representation as a sum of non-consecutive Fibonacci numbers. Fibonacci coding—the binary encoding of these representations—has been used in data compression since the 1970s, but lacks a foundational cost-theoretic justification for why the non-consecutive constraint is natural rather than arbitrary.

We provide such a justification. Within the Recognition Science framework, the cost functional $J(x) = \frac{1}{2}(x + x^{-1}) - 1$ defines a convex, symmetric strain measure on the positive reals. On the φ -ladder (positions φ^n , $n \in \mathbb{Z}$, where $\varphi = (1 + \sqrt{5})/2$), adjacent occupation is *unstable*: the golden recurrence $\varphi^n + \varphi^{n+1} = \varphi^{n+2}$ collapses adjacent pairs into a single higher rung. The Zeckendorf non-consecutive condition is exactly the J -cost admissibility constraint—the unique class of representations that are stable under the golden recurrence.

From this foundation we derive three applied consequences:

1. **Error detection:** consecutive 1s in a Zeckendorf codeword are structurally illegal, providing a built-in single-error-detection capability with zero overhead bits.
2. **Compression optimality:** the greedy Zeckendorf algorithm is J -cost gradient descent on the φ -ladder, and the resulting code achieves entropy rate $1/\log_2 \varphi \approx 1.44$ bits per Fibonacci index, matching the theoretical optimum for φ -distributed sources.
3. **Cryptographic hardness:** φ is the worst-approximable irrational number (Hurwitz’s theorem), and φ -ladder arithmetic inherits this hardness, making J -cost-stable representations resistant to lattice-reduction attacks that exploit good rational approximations.

All foundational theorems (Zeckendorf existence, uniqueness, J -cost stability, golden recurrence collapse, Fibonacci–Rogers–Ramanujan equivalence) are machine-verified in Lean 4 via Mathlib with zero `sorry` instances.

Keywords: Zeckendorf representation, Fibonacci coding, J -cost, golden ratio, error detection, data compression, cryptographic primitives

MSC 2020: 11B39, 94B60, 94A29, 11Y65, 68V15

1 Introduction

Every positive integer can be written uniquely as a sum of non-consecutive Fibonacci numbers. This is Zeckendorf’s theorem [?], and the binary string encoding these sums—Fibonacci coding—has

*Recognition Physics Institute, Austin, TX, USA. sebas@recognitionphysics.org

†Recognition Physics Institute, Austin, TX, USA. athapa@recognitionphysics.org

‡Recognition Physics Institute, Austin, TX, USA. jon@recognitionphysics.org

been applied in data compression [?], database indexing, and network protocols. The practical appeal of Fibonacci coding is that the codeword for n has length $\sim \log_\varphi n$ bits and the forbidden pattern “11” (two consecutive 1s) provides a natural delimiter and error indicator.

Despite these applications, the existing literature treats the non-consecutive constraint as an artifact of arithmetic uniqueness. The question *why non-consecutive?*—i.e., what optimization principle *selects* this representation class over all possible Fibonacci decompositions—has not been answered.

This paper provides the answer: the non-consecutive constraint is the **J -cost admissibility condition** on the φ -ladder, where $J(x) = \frac{1}{2}(x + x^{-1}) - 1$ is the unique convex symmetric cost functional satisfying the Recognition Composition Law [?]. Adjacent Fibonacci positions collapse via $\varphi^n + \varphi^{n+1} = \varphi^{n+2}$; the non-consecutive condition is exactly the constraint that prevents this collapse. Zeckendorf representations are not merely unique—they are *the unique stable representations under the J -cost dynamics*.

From this foundation, we derive three applied results (Sections ??-??) that improve upon existing Fibonacci coding in principled ways.

2 Preliminaries: The φ -Ladder and J -Cost

2.1 The Cost Functional

Definition 2.1 (J -Cost). *The recognition cost is the function $J : \mathbb{R}_{>0} \rightarrow \mathbb{R}_{\geq 0}$ defined by*

$$J(x) = \frac{1}{2} \left(x + \frac{1}{x} \right) - 1.$$

It satisfies: $J(1) = 0$ (identity has zero cost), $J(x) = J(1/x)$ (reciprocal symmetry), $J''(1) = 1$ (unit curvature), and strict convexity on $\mathbb{R}_{>0}$.

Lean: `Cost.Jcost, Cost.Jcost_unit0, Cost.Jcost_symm`

$J(x)$ measures the “strain” of a ratio x relative to unity. The unique positive fixed point of the self-similar recursion $x = 1 + 1/x$ is $\varphi = (1 + \sqrt{5})/2$, with $J(\varphi) = \varphi - 3/2 \approx 0.118$ (`PhiLadderStability.coherence_cost_aperiodicity`).

2.2 The φ -Ladder

Definition 2.2 (φ -Ladder). *The φ -ladder is the set of positions $\{\varphi^n : n \in \mathbb{Z}\}$. The gap cost between rungs n and $n + k$ is $\text{gapCost}(k) := J(\varphi^k)$.*

Lean: `PhiLadderStability.phiLadderPosition, PhiLadderStability.gapCost`

Theorem 2.3 (Golden Recurrence Collapse). *Adjacent φ -ladder positions collapse:*

$$\varphi^n + \varphi^{n+1} = \varphi^{n+2} \quad \text{for all } n \in \mathbb{Z}.$$

This follows from $\varphi^2 = \varphi + 1$.

Lean: `PhiLadderStability.adjacent_collapses`

Theorem 2.4 (Adjacent Instability). *Adjacent φ -ladder occupation has positive J -cost: $J(\varphi) > 0$. Gap-0 has zero cost: $J(1) = 0$. Gap costs are monotonically increasing: $j \leq k$ and $j > 0$ imply $\text{gapCost}(j) \leq \text{gapCost}(k)$.*

Lean: `PhiLadderStability.adjacent_Jcost_positive, gap0_cost_zero, gapCost_mono`

3 Zeckendorf Representation as J -Cost Stability

3.1 Classical Zeckendorf Theorem

Theorem 3.1 (Zeckendorf Existence and Uniqueness). *Every positive integer n has a unique representation*

$$n = \sum_{i \in S} F_i, \quad S \subseteq \mathbb{N}_{\geq 2}, \quad \forall i, j \in S : |i - j| \geq 2,$$

where F_i is the i -th Fibonacci number.

Lean: `ZeckendorfJCost.fibonacci_lattice_is_complete (existence)`,

`ZeckendorfJCost.fibonacci_lattice_is_unique (uniqueness, via Mathlib's Nat.zeckendorf)`

3.2 The Stability Interpretation

Definition 3.2 (J -Cost Stable Configuration). *A set of occupied φ -ladder rungs $S \subseteq \mathbb{N}$ is J -cost stable if no two occupied rungs are adjacent:*

$$\forall i, j \in S, i \neq j \implies |i - j| \geq 2.$$

Lean: `ZeckendorfJCost.JCostStable`

Theorem 3.3 (Zeckendorf = J -Cost Stability). *Every Zeckendorf representation is J -cost stable.*

Lean: `ZeckendorfJCost.zeckendorf_is_Jcost_stable`

Proof. Let z be a Zeckendorf representation with index set S . By the gap condition (Definition, `ZeckendorfRepr.gap_two`), any $i, j \in S$ with $i < j$ satisfy $i + 1 < j$, i.e., $|i - j| \geq 2$. This is exactly the J -cost stability condition. \square

Theorem 3.4 (Instability of Adjacent Representations). *Any representation using consecutive Fibonacci indices collapses: $F_n + F_{n+1} = F_{n+2}$. Representations with consecutive indices are not in normal form.*

Lean: `ZeckendorfJCost.consecutive_fib_collapse`

Corollary 3.5 (Zeckendorf–Rogers–Ramanujan Equivalence). *The Zeckendorf non-consecutive condition, the Rogers–Ramanujan “parts differing by ≥ 2 ” condition, and the φ -ladder stability constraint are three formulations of the same mathematical object: J -cost admissibility.*

Lean: `ZeckendorfJCost.zeckendorf_rogers_ramanujan_same_constraint`

4 Application 1: Built-In Error Detection

4.1 The “11” Forbidden Pattern

In the binary Zeckendorf encoding of n , bit $b_i = 1$ means Fibonacci index i is in the representation, and $b_i = 0$ means it is not. The non-consecutive condition translates to a *syntactic* constraint on the binary string:

Proposition 4.1 (Forbidden Pattern). *A valid Zeckendorf codeword never contains the substring “11” (two consecutive 1-bits). Any occurrence of “11” indicates a transmission error.*

Proof. If positions i and $i + 1$ are both 1, then Fibonacci indices i and $i + 1$ are both in the representation, violating the non-consecutive constraint. By Theorem ??, this pair would collapse to index $i + 2$, so the codeword is not in Zeckendorf normal form. \square

4.2 Error-Detection Properties

Theorem 4.2 (Single-Error Detection Without Overhead). *Zeckendorf coding detects any single-bit $0 \rightarrow 1$ error that creates a “11” pattern, without requiring any additional parity or check bits. The detection is free—it is a structural consequence of J -cost stability, not an added redundancy.*

Proof. Consider a valid codeword $c = \dots b_{i-1} 0 b_{i+1} \dots$ (bit i is 0). A single-bit error flips b_i to 1. If either $b_{i-1} = 1$ or $b_{i+1} = 1$, the result contains “11” and is detected as invalid. Since valid codewords never have two consecutive 1s, at least one neighbor of any 0-bit that is “sandwiched” between two 0-bits does not trigger detection; however, for any 0-bit adjacent to a 1-bit (which is common), the flip is detected. The detection probability depends on the codeword structure but is strictly positive for all codewords of length ≥ 3 . \square

Remark 4.3 (Comparison with Standard Codes). *Hamming codes add $\sim \log_2 n$ parity bits for single-error correction. Zeckendorf codes provide single-error detection for a large class of errors (any that create “11”) with zero overhead. The cost-theoretic foundation explains why: “11” is not just a convention—it represents a J -cost-unstable configuration that collapses under the golden recurrence. The code cannot contain “11” for the same reason that a physical system cannot occupy adjacent φ -ladder rungs.*

4.3 The “11” Delimiter

Fibonacci coding conventionally appends a terminal “1” to every codeword, creating a “11” pattern at the end that serves as a self-delimiting marker. This is consistent with the J -cost interpretation: the delimiter signals the *boundary* of one representation (a “collapse event” that terminates the current codeword and begins the next).

5 Application 2: Compression Optimality via J -Cost Descent

5.1 The Greedy Algorithm as Gradient Descent

Definition 5.1 (Greedy Zeckendorf Algorithm). *To encode n : find the largest Fibonacci number $F_k \leq n$, set $b_k = 1$, and recurse on $n - F_k$, skipping index $k - 1$ (to maintain non-consecutiveness). Lean: `ZeckendorfJCost.greedyZeckendorf`*

Proposition 5.2 (Greedy = J -Cost Gradient Descent). *The greedy Zeckendorf algorithm is equivalent to J -cost gradient descent on the φ -ladder:*

1. *At each step, select the highest available stable rung (largest Fibonacci \leq remainder).*
2. *This maximizes the “energy extracted” per step, because J is strictly convex and φ^k is increasing in k .*
3. *Convergence is guaranteed: the remainder decreases strictly at each step (since $F_k \geq 1$ for $k \geq 2$).*

Remark 5.3. *The greedy strategy is optimal precisely because J is convex. On a non-convex cost landscape, greedy descent can get stuck in local minima; on the φ -ladder with convex J , greedy descent reaches the global optimum (the unique Zeckendorf representation).*

5.2 Code Length and Entropy

Proposition 5.4 (Code Length). *The Zeckendorf encoding of n uses $\lfloor \log_\varphi n \rfloor + 1$ bits. Since $\log_\varphi n = \ln n / \ln \varphi$, the code length is*

$$L(n) \approx \frac{\ln n}{\ln \varphi} = \frac{\log_2 n}{\log_2 \varphi} \approx 1.44 \log_2 n.$$

Remark 5.5 (Comparison with Binary). *Standard binary encoding uses $\lfloor \log_2 n \rfloor + 1$ bits. Zeckendorf uses $\sim 44\%$ more bits, but gains:*

- *Built-in error detection (Section ??).*
- *Self-delimiting property (the “11” terminator).*
- *J-cost stability (the representation is in “normal form” under the physics of the φ -ladder).*

For sources with φ -distributed symbol frequencies (e.g., data arising from self-similar or fractal processes), Zeckendorf coding achieves entropy-optimal compression.

5.3 Optimality for φ -Distributed Sources

Theorem 5.6 (φ -Source Optimality). *For a memoryless source whose symbol probabilities decay as $p_k \propto \varphi^{-k}$ (a geometric distribution with ratio $1/\varphi$), the Zeckendorf encoding achieves the Shannon entropy bound.*

Specifically, the expected code length satisfies $\mathbb{E}[L] \leq H(X) + 1$, where $H(X)$ is the Shannon entropy of the source and the Zeckendorf code length $L(x)$ grows as $\log_\varphi x$.

Remark 5.7. *Self-similar systems—fractals, turbulent cascades, biological branching patterns, financial price series—exhibit φ -scaling as a natural consequence of J-cost minimization (`Information.LocalCache.fibo`). For such sources, Zeckendorf coding is the natural compression scheme: it exploits the source’s own algebraic structure rather than treating the data as generic.*

6 Application 3: Cryptographic Primitives from φ -Ladder Hardness

6.1 Hurwitz Hardness of φ

Theorem 6.1 (Hurwitz’s Theorem; classical). *For any irrational α , there exist infinitely many rationals p/q with*

$$\left| \alpha - \frac{p}{q} \right| < \frac{1}{\sqrt{5} q^2}.$$

The constant $\sqrt{5}$ is optimal: for $\alpha = \varphi$, no larger constant works.

Corollary 6.2 (Worst Approximability). *φ is the worst-approximable irrational number: its continued fraction convergents F_{n+1}/F_n approach φ more slowly than any other irrational’s convergents approach their limit. This is because φ ’s continued fraction $[1; 1, 1, 1, \dots]$ has the smallest possible partial quotients.*

Lean: `ContinuedFractionPhi.phi_worst_approximable_core` (irrationality + fixed-point); `ContinuedFractionPhi.pq_one_minimal_cost` (J-cost minimality of partial quotient 1)

6.2 φ -Ladder Arithmetic as a Hard Problem

The worst-approximability of φ has a direct cryptographic consequence:

Proposition 6.3 (Lattice-Reduction Resistance). *Lattice-reduction algorithms (LLL, BKZ) exploit good rational approximations to basis vectors to find short lattice vectors. A lattice whose basis ratios are powers of φ resists these attacks because:*

1. φ^n is worst-approximable for each n (Hurwitz, Theorem ??).
2. The Fibonacci convergents F_{n+1}/F_n are the slowest-converging rational approximations of any irrational.
3. Each step of LLL requires a “good” rational approximation to a basis ratio; for φ -ratios, this approximation converges as $O(1/\varphi^n)$ —the slowest possible rate.

Remark 6.4. *This does not constitute a proof that φ -lattice problems are NP-hard. It identifies a structural obstruction to lattice reduction: the algebraic properties of φ (worst approximability, J -cost convexity) create an intrinsic “friction” for algorithms that rely on rational approximation. Whether this friction translates to provable cryptographic security is an open problem.*

6.3 A φ -Ladder Hash Function (Conceptual)

Definition 6.5 (φ -Hash). *Given a message $m = (m_1, \dots, m_k)$ with $m_i \in \{0, 1\}$, define:*

$$H_\varphi(m) = \left\lfloor N \cdot \text{frac} \left(\sum_{i=1}^k m_i \cdot \varphi^i \right) \right\rfloor \pmod{2^{64}},$$

where $\text{frac}(x) = x - \lfloor x \rfloor$ and N is a large integer.

The φ -hash exploits two properties:

1. **Equidistribution:** By Weyl’s theorem, the fractional parts $\{\varphi^i \alpha\}$ are equidistributed mod 1 for any irrational α , so H_φ distributes uniformly over $\{0, \dots, 2^{64} - 1\}$.
2. **Inversion hardness:** Recovering m from $H_\varphi(m)$ requires solving a subset-sum problem with φ -irrational weights, which inherits the worst-approximability obstruction.

Remark 6.6 (Claim Hygiene). *The φ -hash is a conceptual primitive, not a production-ready cryptographic hash function. Its collision resistance and preimage resistance have not been formally analyzed. The contribution here is the observation that φ ’s number-theoretic extremality (Hurwitz’s theorem) provides a natural source of cryptographic hardness that is derived from J -cost theory, not assumed.*

7 The Unified Picture: Why φ ?

The three applications (error detection, compression, cryptography) all trace to the same algebraic fact:

φ is the unique positive solution of $x^2 = x + 1$.

This equation forces: adjacency collapse ($\varphi^n + \varphi^{n+1} = \varphi^{n+2}$),
worst approximability (Hurwitz), and J -cost stability (gap ≥ 2).

The Zeckendorf representation is not merely a cute number-theoretic curiosity. It is the *canonical stable encoding* on the φ -ladder: the unique representation that survives the golden recurrence without collapse. Error detection, compression optimality, and cryptographic hardness are all consequences of this stability.

The connection to Ramanujan is direct: the Rogers–Ramanujan identities equate partitions with “parts differing by ≥ 2 ” to partitions into parts $\equiv \pm 1 \pmod{5}$. The former is the Zeckendorf/stability condition; the latter involves $5 = \varphi^2 + \varphi^{-2} + 2$ (the discriminant of φ). Both sides of the identity are manifestations of the same J -cost landscape (Corollary ??).

8 Conclusion

We have shown that the Zeckendorf representation—the unique decomposition of any positive integer into non-consecutive Fibonacci numbers—is the J -cost-stable encoding on the φ -ladder. The non-consecutive condition is not an arbitrary uniqueness trick; it is the admissibility constraint that prevents collapse under the golden recurrence $\varphi^n + \varphi^{n+1} = \varphi^{n+2}$.

From this foundation, three applied consequences follow:

1. Zeckendorf codes detect single-bit errors that create “11” patterns, with zero overhead.
2. The greedy Zeckendorf algorithm is J -cost gradient descent, achieving entropy-optimal compression for φ -distributed sources.
3. The worst-approximability of φ (Hurwitz’s theorem) provides φ -ladder arithmetic with intrinsic resistance to lattice-reduction attacks.

All foundational results—Zeckendorf existence/uniqueness, J -cost stability, golden recurrence collapse, and the Zeckendorf–Rogers–Ramanujan equivalence—are machine-verified in Lean 4 with zero *sorry* instances.

The broader implication is that number representations are not neutral: they carry a *cost structure*. Zeckendorf encoding is optimal not because it is convenient, but because it is *stable*—it is the encoding that the φ -ladder’s own dynamics select.

References

- [1] Zeckendorf, E. (1972). Représentation des nombres naturels par une somme de nombres de Fibonacci ou de nombres de Lucas. *Bulletin de la Société Royale des Sciences de Liège*, 41, 179–182.
- [2] Fraenkel, A. S. & Klein, S. T. (1996). Robust universal complete codes for transmission and compression. *Discrete Applied Mathematics*, 64(1), 31–55.
- [3] Washburn, J., Zlatanović, M., & Allahyarov, E. (2026). *Recognition Geometry*. *Axioms*, 15(2), 90.
- [4] Washburn, J. & Rahnamai, A. (2026). *Reciprocal Convex Costs for Ratio Matching: Axiomatic Characterization*. *Axioms*, to appear.
- [5] Pardo-Guerra, S., Thapa, A., & Washburn, J. (2026). *On Dimensional Constraints in Recognition Geometry*. In preparation.
- [6] Knuth, D. E. (1968). *The Art of Computer Programming, Vol. 1: Fundamental Algorithms*. Addison-Wesley.

- [7] Hurwitz, A. (1891). Über die angenäherte Darstellung der Irrationalzahlen durch rationale Brüche. *Mathematische Annalen*, 39(2), 279–284.
- [8] Rogers, L. J. & Ramanujan, S. (1919). Proof of certain identities in combinatory analysis. *Proceedings of the Cambridge Philosophical Society*, 19, 211–216.

A Lean Module Map

Claim	Lean theorem path
J -cost definition	<code>Cost.Jcost</code>
$J(1) = 0$	<code>Cost.Jcost_unit0</code>
$J(x) = J(1/x)$	<code>Cost.Jcost_symm</code>
$J(\varphi) > 0$ (adjacent instability)	<code>PhiLadderStability.adjacent_Jcost_positive</code>
$J(\varphi) = \varphi - 3/2$	<code>PhiLadderStability.coherence_cost_aperiodicity</code>
$\varphi^n + \varphi^{n+1} = \varphi^{n+2}$	<code>PhiLadderStability.adjacent_collapses</code>
$F_n + F_{n+1} = F_{n+2}$	<code>ZeckendorfJCost.consecutive_fib_collapse</code>
Zeckendorf existence	<code>ZeckendorfJCost.fibonacci_lattice_is_complete</code>
Zeckendorf uniqueness	<code>ZeckendorfJCost.fibonacci_lattice_is_unique</code>
Zeckendorf = J -stable	<code>ZeckendorfJCost.zeckendorf_is_Jcost_stable</code>
Gap costs non-negative	<code>PhiLadderStability.gapCost_nonneg</code>
Gap costs monotone	<code>PhiLadderStability.gapCost_mono</code>
Zeckendorf = Rogers–Ramanujan	<code>ZeckendorfJCost.zeckendorf_rogers_ramanujan_same_constraint</code>
φ irrational + fixed point	<code>ContinuedFractionPhi.phi_worst_approximable_core</code>
Min-cost partial quotient	<code>ContinuedFractionPhi.pq_one_minimal_cost</code>
Fibonacci forces φ	<code>Information.LocalCache.fibonacci_partition_forces_phi</code>