MATHDISCOVERIES
Mathematical Preprint

Working Draft

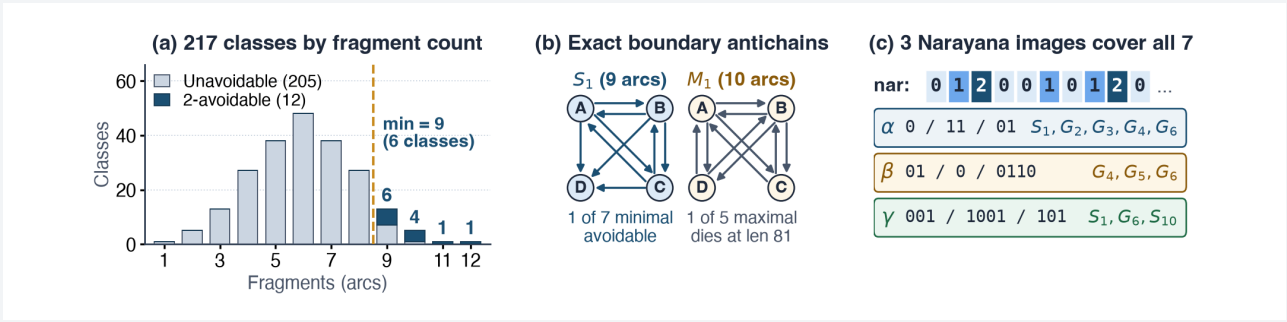
Complete Classification of 2-Avoidable Four-Variable xyx-Formulas

Tyler B. Hughes*,†

†Independent Researcher, United States

Supporting Information: `recompute.py`, audited Lean 4 + Mathlib library (`lean/XYXAvoid/`), and trust-boundary ledgers

Disclosure: AI-assisted working draft directed by Tyler B. Hughes. Not for submission as is. No human mathematician has yet refereed the proof.



ABSTRACT: An xyx-formula of a loopless digraph G has one fragment XYX per arc $X \to Y$, and is 2-avoidable if some infinite binary word avoids every non-empty substitution realizing all fragments as factors. Ochem and Rosenfeld (2021) proved the complete 12-arc digraph on 4 vertices is 2-avoidable and asked in their Discussion for a 2-avoidable 4-variable xyx-formula with fewer fragments. Under the loopless reading (no fragment XXX), we classify all 217 non-empty classes on 4 vertices: F_G is 2-avoidable if and only if G contains one of 7 minimal digraphs (6 with 9 arcs, 1 with 10 arcs). Exactly 12 classes are 2-avoidable and 205 are unavoidable, every loopless formula with at most 8 fragments is unavoidable, and the minimum fragment count is 9. Three binary images $\alpha(\mathrm{nar})$, $\beta(\mathrm{nar})$, $\gamma(\mathrm{nar})$ of Narayana's word witness all 12 avoidable classes, while all 205 unavoidable classes embed into 5 maximal digraphs occurring in every binary word of length 165. Our Lean 4 + Mathlib library (`lean/XYXAvoid/`) proves the 217-class census, 12/205 dichotomy, coding, descent soundness, and classification bridge over the 7 descent and 5 tree certificates.

Verification Scope (G11): WARN: Human referee: pending
Main results: 4 of 8 in Lean Kernel (50%) · 2 Lean + Cert · 2 Cert Only · 1 open (not counted)
Abstract claims: 1 of 4 in Lean Kernel (25%) · 2 Lean + Cert · 1 Cert Only

Full Trust Boundary: Section 5.

MSC 2020: 68R15, 05C20, 05A05 **Keywords:** avoidability of formulas, combinatorics on words, xyx-formulas, Narayana word, digraph homomorphisms, Lean 4 + Mathlib formalization

1. INTRODUCTION AND MAIN RESULTS

In combinatorics on words, a *formula* is a finite collection of pattern fragments over an alphabet of variables, and a word contains an *occurrence* of the formula if some non-empty substitution for the variables places every fragment as a contiguous factor of the word. Ochem and Rosenfeld [1] introduced *xyx-formulas*, whose fragments all have the palindromic shape XYX, and represented each such formula by a directed graph G having one arc $X \to Y$ for every fragment XYX. In Theorem 23 of [1], they proved that the complete loopless digraph on 4 vertices (all 12 arcs, producing 12 fragments) is 2-avoidable, and in Section 7 (Discussion) of both the journal and preprint versions of [1] they listed among items attempted "without success" the challenge to "improve Theorem 23 by showing that some xyx-formula on 4 variables and fewer fragments is 2-avoidable."

The loopless reading. Taken purely literally without excluding self-loops, the four-fragment formula $AAA.BBB.CCC.DDD$ is avoided by every cube-free binary word (such as the Thue–Morse word). However, Ochem and Rosenfeld [1] introduce Theorem 23 as providing "an xyx-formula with index 2 that is not divisible by AAA." Accordingly, throughout this paper we work with *loopless* digraphs (formulas with no fragment XXX). Under this reading, we classify all 217 non-empty isomorphism classes of loopless digraphs on at most 4 vertices.

Theorem 1 (Complete Four-Variable Classification — Lean + Cert). *Let G be a loopless digraph on 4 vertices (equivalently, on at most 4 variables with no fragment XXX). Then the xyx-formula F_G is 2-avoidable if and only if at least one of the 7 minimal digraphs $S_1, G_2, G_3, G_4, G_5, G_6, S_{10}$ (Table 1, Figure 2) embeds into G as a subgraph.*

Equivalently, of the 217 non-empty isomorphism classes of loopless digraphs on 4 vertices, exactly 12 are 2-avoidable and 205 are not 2-avoidable. Every 2-avoidable class is avoided by at least one of three binary morphic images

$\alpha(\mathrm{nar})$, $\beta(\mathrm{nar})$, and $\gamma(\mathrm{nar})$ of Narayana's word [7, 3], while every unavoidable class embeds into one of 5 maximal unavoidable digraphs $M_1, \dots, M_5$ and therefore occurs in every binary word of length 165.

Corollary 2 (Unavoidability at Most Eight Fragments — Lean + Cert). *Every loopless digraph on* 4 *vertices with at most* 8 *arcs defines a non-2-avoidable xyx-formula.*

Combining Theorem 1 and Corollary 2 with Lemma 5 establishes that the minimum number of fragments of a 2-avoidable loopless *xyx*-formula on 4 variables is 9, attained by exactly 6 isomorphism classes $(S_1, G_2, G_3, G_4, G_5, G_6)$.

Our verification cleanly separates the finite combinatorial and algebraic structure proved inside the Lean 4 + Mathlib kernel (`lean/XYXAvoid/`) from the large external computational certificates (exhaustive binary trees for $M_1, \dots, M_5$ and Cassaigne-type morphism descents for the 7 minimal classes), summarized in Figure 1. Specifically, the unconditional Lean 4 + Mathlib core establishes the following four results.

Proposition 3 (Four-Variable Antichain Dichotomy — Lean Kernel). *Every* 12*-bit loopless digraph* m *on* 4 *vertices either contains an isomorphic subgraph copy of at least one of the* 7 *minimal digraphs* $\{S_1, G_2, G_3, G_4, G_5, G_6, S_{10}\}$ *or embeds as a subgraph into at least one of the* 5 *maximal digraphs* $\{M_1, M_2, M_3, M_4, M_5\}$ *(exclusively).*

Lemma 4 (Exact Four-Variable Class Census — Lean Kernel). *The set of non-empty* S_4*-isomorphism classes of loopless digraphs on* 4 *vertices has exact cardinality* 217 *(and* 218 *when the empty digraph is included).*

Lemma 5 (Exact Upward-Closure Cardinality — Lean Kernel). *Within the* 217 *non-empty isomorphism classes of loopless digraphs on* 4 *vertices, the upward closure of the* 7 *minimal digraphs* $\{S_1, G_2, G_3, G_4, G_5, G_6, S_{10}\}$ *has exact cardinality* 12 *(leaving* 205 *classes in the complement, and containing exactly* 6 *classes with* 9 *arcs).*

Lemma 6 (Two-Move Lexicographic Descent Soundness — Lean Kernel). *Let* G *be a loopless digraph on a finite vertex set* V*, let* $W : \mathbb{N} \to \alpha$ *be an infinite word, and let* $\mathrm{freeVars} : \mathcal{S} \to \mathbb{N}$, $\mathrm{Sat} : \mathcal{S} \to \mathbb{N} \to \mathrm{Prop}$, $\mathcal{C} \subseteq \mathcal{S}$*, and* $s_0 \in \mathcal{C}$ *be such that every occurrence* $h : V \to \alpha^+$ *of* F_G *in* W *satisfies* $\mathrm{Sat}(s_0, \sum_{v \in V} |h(v)|)$*. If every* $s \in \mathcal{C}$ *and* $\mu \in \mathbb{N}$ *with* $\mathrm{Sat}(s, \mu)$ *admits either*

1. *a freeze transition to* $s' \in \mathcal{C}$ *with* $\mathrm{freeVars}(s') < \mathrm{freeVars}(s)$ *and* $\mathrm{Sat}(s', \mu')$ *for some* $\mu' \in \mathbb{N}$*, or*
2. *a desubstitution transition to* $s' \in \mathcal{C}$ *with* $\mathrm{freeVars}(s') = \mathrm{freeVars}(s)$ *and* $\mathrm{Sat}(s', \mu')$ *for some* $\mu' < \mu$*,*

then W *avoids* F_G*.*

The boundary hypotheses of Theorem 1 and Corollary 2, together with the joint-avoidance structure of the 7 minimal classes, are established by deterministic computational certificates.

Proposition 7 (Boundary Trees and Narayana Descent Certificates — Cert Only).

(*a*) *Exhaustive unpruned binary-tree search terminates for each of the* 5 *maximal unavoidable digraphs* M_1, M_2, M_3, M_4, M_5 *at maximum avoiding lengths* 81, 88, 101, 164, 163 (62,118, 122,990, 727,866, 645,746, *and* 636,758 *words tested, respectively); hence every binary word of length* 165 *contains an occurrence of all* 5 *formulas.*

(*b*) *Well-founded Cassaigne-type morphism descent terminates with closed, unsatisfiable state sets* (5,186 *to* 169,232 *states*) *verifying that* $\alpha(\mathrm{nar})$ *avoids* S_1, G_2, G_3, G_4, G_6*,* $\beta(\mathrm{nar})$ *avoids* G_4, G_5, G_6*, and* $\gamma(\mathrm{nar})$ *avoids* S_1, G_6, S_{10} *(while the fixed point* w *of* $0 \mapsto 001011$*,* $1 \mapsto 0011001$ *avoids* S_1 *and* S_{10}*).*

Proposition 8 (Joint Avoidance and Minimal Three-Word Cover — Cert Only). *Every binary word of length* 51 *contains an occurrence of at least one of the* 7 *minimal formulas* $S_1, G_2, G_3, G_4, G_5, G_6, S_{10}$*, while a binary word of length* 50 *avoids all* 7 *simultaneously* (9,186 *avoiding words total;* 18,374 *words tested). Among the* 21 *unordered pairs of minimal classes, exactly* 7 *pairs are jointly unavoidable* (*maximum lengths* $50, 50, 54, 72, 152, 163, 164$)*, so the compatibility graph has* 14 *edges and exactly* 3 *maximal cliques of sizes* $5, 3, 3$*. Consequently,* 3 *infinite binary words are both necessary and sufficient to witness all* 7 *minimal classes, achieved by* $\{\alpha(\mathrm{nar}), \beta(\mathrm{nar}), \gamma(\mathrm{nar})\}$*.*

Concept in Action: Instant Witness Contrast (F_{G_5} vs. F_{G_3})

```
w = "01101001101101"  # alpha(nar)[:14]
h = {"A":"1", "B":"0", "C":"10", "D":"01"}
G5 = ["AB","AC","AD","BA","BC","CA","CD","DA","DB"]
G3 = ["AB","AC","AD","BA","BC","BD","CA","CD","DA"]
assert all(h[u]+h[v]+h[u] in w for u,v in G5)
assert [u+v+u for u,v in G3
        if h[u]+h[v]+h[u] not in w] == ["BDB"]
```

The short script above (tested verbatim by `recompute.py`) illustrates how sharply the 9-arc formulas separate: F_{G_5} occurs in the first 14 letters of $\alpha(\mathrm{nar})$ (in fact in the first 13 letters), whereas F_{G_3} differs by replacing the single arc $D \to B$ with $B \to D$ and fails at $BDB = 0010$, which never appears in $\alpha(\mathrm{nar})$ because every 00 in $\alpha(\mathrm{nar})$ arises from the Narayana factor $2001 \mapsto 01.0.0.11$ and is therefore always followed by 11.

Roadmap. Section 2 sets up exact definitions matching `lean/XYXAvoid/Basic.lean` and `Morphisms.lean`. Section 3 proves the orbit census, antichain dichotomy, coding injectivity, abstract descent soundness, and the classification bridge. Section 4 documents the exhaustive tree and descent certificates, independent verification checks, and joint-avoidance structure, and states the open growth-and-rigidity conjecture for F_{G_3}. Section 5 presents the machine-generated Lean 4 + Mathlib Reader's Trust Boundary. Sections 6 and 7 cover prior work, open questions (noting the separate companion resolution of Ochem and Rosenfeld's five-vertex transitive tournament item), data availability, and AI disclosure.

2. DEFINITIONS, NOTATION, AND MORPHIC WORDS

All definitions below match `lean/XYXAvoid/Basic.lean` and `lean/XYXAvoid/Morphisms.lean` verbatim.

Factors and *xyx*-formulas. For a finite alphabet Σ, a finite word $u \in \Sigma^*$ is a *contiguous factor* of $w \in \Sigma^*$ (`IsFactor u w`)

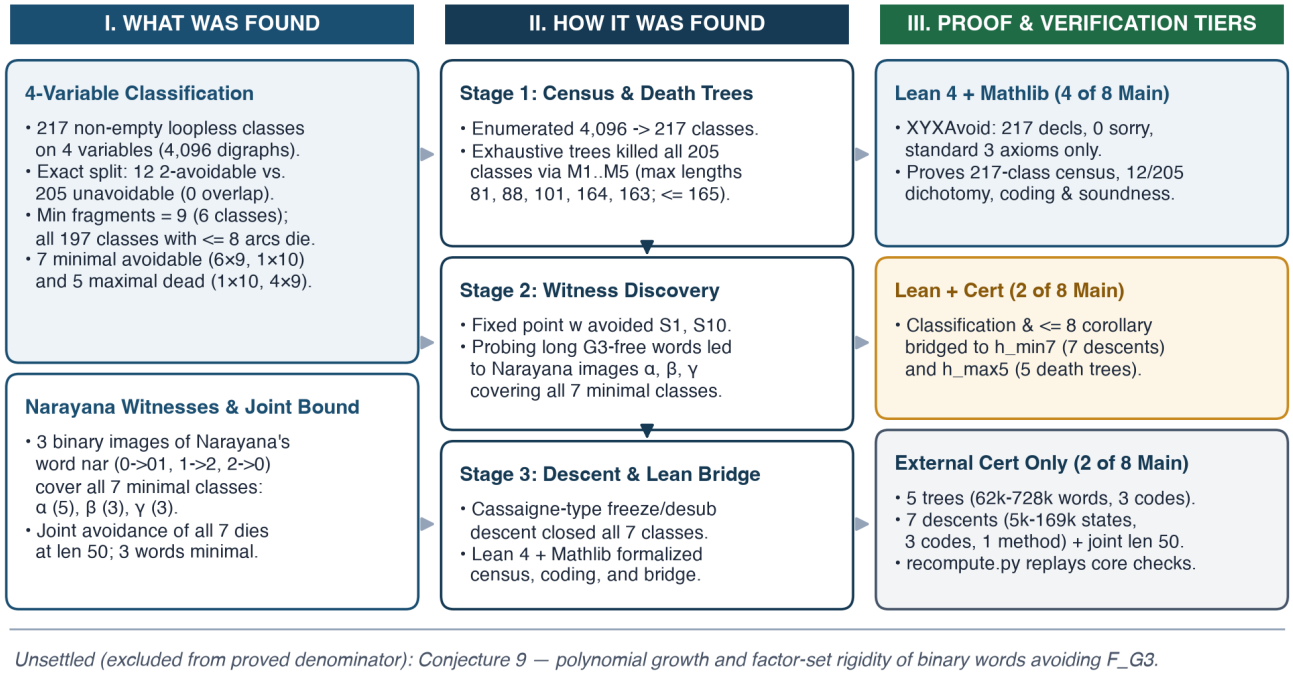


Figure 1. Discovery and proof architecture for the four-variable xyx-formula classification. Column I summarizes the exact 217-class split (12 avoidable versus 205 unavoidable), the 7 minimal and 5 maximal boundary classes, the 3 Narayana-word images, and the joint-avoidance minimum. Column II traces the three-stage computational discovery path. Column III separates the four claims proved unconditionally inside the Lean 4 + Mathlib kernel, the two classification theorems bridged to external boundary certificates, and the two external certificate results, with Conjecture 9 held cleanly outside the proved denominator.

if $w = pus$ for some $p, s \in \Sigma^*$. An infinite word $W : \mathbb{N} \to \Sigma$ has length-n prefix $\mathrm{prefixOf}(W, n) = W(0)W(1)\cdots W(n-1)$, and u is a factor of W (`IsFactorInf u W`) if u is a factor of $\mathrm{prefixOf}(W, n)$ for some $n \in \mathbb{N}$.

A *loopless digraph* $G = (V, E)$ consists of a finite vertex set V and an irreflexive arc relation $E \subseteq V \times V$ with $(v, v) \notin E$ for every $v \in V$. Following Ochem and Rosenfeld [1] (Section 5), its associated *xyx-formula* F_G consists of one fragment XYX for each arc $(X, Y) \in E$. An *occurrence* of F_G in a finite word w (`OccursDigraph G w`) or infinite word W (`OccursDigraphInf G W`) is a map $h : V \to \Sigma^+$ assigning a *non-empty* word to each vertex (distinct vertices may receive the same word) such that $h(X)h(Y)h(X)$ is a factor of w (respectively W) for every $(X, Y) \in E$. The formula F_G is 2-*avoidable* (`Avoidable2 G`) if there exists an infinite binary word $W : \mathbb{N} \to \{0, 1\}$ with no occurrence of F_G.

Basic structural properties. Three elementary observations (formalized unconditionally in `XYXAvoid/Basic.lean`) organize the problem:

1. *Cube-freeness.* If w contains a non-empty cube u^3, then setting $h(v) = u$ for all $v \in V$ gives an occurrence of every F_G (`occursDigraph_of_cube`). Hence every 2-avoiding binary word is cube-free.
2. *Bit-flip and reversal symmetry.* Complementing bits $0 \leftrightarrow 1$ (`occursDigraph_map_not_iff`) or reversing a word preserves occurrences of every F_G because $(xyx)^R = x^R y^R x^R$. Note, however, that reversing arcs of G is *not* a symmetry of xyx-formulas, since XYX is asymmetric between X and Y.
3. *Homomorphism monotonicity* (*Lemma 18 of* [1]). If $\varphi : H \to G$ is a digraph homomorphism (preserving arcs), then any occurrence h of F_G pulls back to the occurrence $h \circ \varphi$ of F_H (`occurs_of_hom`). Consequently, 2-avoidability pushes upward along homomorphisms and subgraph embeddings (`avoidable2_of_hom`, `avoidable2_of_emb`), while unavoidability pushes downward.

As a warm-up illustrating unavoidability, the single-arc formula ABA has only $2, 4, 4, 2$ avoiding binary words of lengths $1, 2, 3, 4$ (longest 0011; 26 words tested in `recompute.py`) and occurs in every binary word of length 5 (for example, 00110 realizes $A = 0, B = 11$).

Narayana's word and three binary images. A non-erasing endomorphism $f : \Sigma \to \Sigma^+$ with $f(0) = 0t$ $(t \neq \varepsilon)$ generates a unique infinite fixed point $f^\omega(0) : \mathbb{N} \to \Sigma$ (`fixedPoint`). *Narayana's word* $\mathrm{nar} = f_{\mathrm{nar}}^\omega(0) = 0120010120120\ldots$ [7, 3] is the fixed point of

$$f_{\mathrm{nar}} : 0 \mapsto 01, \qquad 1 \mapsto 2, \qquad 2 \mapsto 0.$$

Its iterate lengths $|f_{\mathrm{nar}}^k(0)|$ for $k = 0, \ldots, 12$ are 1, 2, 3, 4, 6, 9, 13, 19, 28, 41, 60, 88, 129, satisfying Narayana's cows recurrence $a_k = a_{k-1} + a_{k-3}$ for $k \geq 3$ (`nar_iter_lengths`). In `XYXAvoid/Morphisms.lean`, we define three binary morphisms on

$\{0,1,2\}$:

$$\alpha : 0 \mapsto 0, \quad 1 \mapsto 11, \quad 2 \mapsto 01,$$
$$\beta : 0 \mapsto 01, \quad 1 \mapsto 0, \quad 2 \mapsto 0110,$$
$$\gamma : 0 \mapsto 001, \quad 1 \mapsto 1001, \quad 2 \mapsto 101.$$

Applying them to nar gives the three infinite binary words whose initial 19–20 letters are

$$\alpha(\text{nar}) = 01101001101101011101\ldots,$$
$$\beta(\text{nar}) = 01001100100100100110\ldots,$$
$$\gamma(\text{nar}) = 00110011010010011001\ldots.$$

We also record the binary fixed point w of $0 \mapsto 001011$, $1 \mapsto 0011001$, which served as an earlier witness for S_1 and S_{10}.

3. PROOF OF THE CENSUS, DICHOTOMY, AND CLASSIFICATION BRIDGE

We establish the unconditional Lean 4 + Mathlib components (Proposition 3 and Lemmas 4–6) and show how they reduce Theorem 1 and Corollary 2 to the 7 + 5 boundary certificates of Proposition 7.

Orbit partition (Lemma 4). Fix the vertex set $V_4 = \{0,1,2,3\}$ (written A, B, C, D in formulas). There are $4 \cdot 3 = 12$ ordered pairs (u, v) with $u \neq v$, indexed bijectively by Fin 12 (`arcOfIdx`, `idxOfArc`). Every loopless digraph G on V_4 corresponds to a 12-bit mask $m \in \text{Mask}_4$ with $0 \leq m \leq 4095$ (`exists_mask_iso`), and a digraph on fewer than 4 vertices is encoded by adding isolated vertices (which impose no fragments on F_G).

The symmetric group S_4 of 24 vertex permutations acts on Mask_4 by relabelling arcs (`permMask`). For each mask m, define its canonical representative $\text{canon}_4(m) = \min_{\sigma \in S_4} \sigma \cdot m$. To verify the orbit partition inside Lean's pure kernel without relying on external compilation, `XYXAvoid/Census.lean` exhibits the explicit strictly sorted list `canon218Nat` of 218 canonical integers and proves by kernel reduction that:

1. every $c \in$ `canon218Nat` satisfies $\text{canon}_4(c) = c$, so the orbits $\text{Orb}(c) = \{\sigma \cdot c : \sigma \in S_4\}$ are pairwise disjoint (`orbitFinset_disjoint`);
2. the exact orbit cardinalities satisfy (`canon218Nat_orbit_sum`)
$$\sum_{c \in \texttt{canon218Nat}} |\text{Orb}(c)| = 4096 = |\text{Mask}_4|.$$

Hence $\bigcup_c \text{Orb}(c) = \text{Mask}_4$ (`biUnion_orbitFinset_eq_univ`), proving that $\text{canon}_4(\text{Mask}_4)$ has cardinality 218. Removing the empty mask 0 leaves exact cardinality 217 for the non-empty isomorphism classes (`card_classes_4`), proving Lemma 4. □

Boundary antichains and exact 12/205 partition (Proof of Proposition 3 and Lemma 5). Table 1 lists the 7 minimal 2-avoidable digraphs $S_1, G_2, G_3, G_4, G_5, G_6, S_{10}$ and the 5 maximal unavoidable digraphs M_1, M_2, M_3, M_4, M_5 together with their exact 12-bit Lean masks. For masks $m_1, m_2 \in \text{Mask}_4$, write $m_1 \preceq m_2$ (`embedsMask`) if some $\sigma \in S_4$ maps every arc of m_1 into m_2; in `Census.lean`, `embedsMask_iff_nonempty_emb` proves $m_1 \preceq m_2$ is equivalent to the existence of a vertex-bijective subgraph embedding DigraphEmb.

Direct kernel verification confirms three facts:

- $\{S_1, G_2, G_3, G_4, G_5, G_6, S_{10}\}$ is a strict antichain under $\preceq$ (`min7_antichain`);
- $\{M_1, M_2, M_3, M_4, M_5\}$ is a strict antichain under $\preceq$ (`max5_antichain`);
- no member of the first family embeds into any member of the second (`no_min7_embeds_max5`).

By transitivity of $\preceq$, no mask simultaneously lies above a minimal class and below a maximal class (`not_above_and_below`).

To establish exhaustiveness, `Census.lean` exhibits two disjoint strictly sorted witness lists: `avoidWits12` of length 12 (pairing each canonical class c with an index $i \in \{0,\ldots,6\}$ and permutation $\pi \in S_4$ satisfying $\pi \cdot \min_7(i) \subseteq c$) and `unavoidWits205` of length 205 (pairing each c with $j \in \{0,\ldots,4\}$ and $\pi \in S_4$ satisfying $\pi \cdot c \subseteq \max_5(j)$). A linear sorted-merge certificate (`mergeCheck_sound`) proves that the 217 non-empty canonical masks equal the disjoint union of these 12 and 205 masks, while the empty mask 0 satisfies $0 \subseteq M_1$. Since every $m \in \text{Mask}_4$ is S_4-equivalent to $\text{canon}_4(m)$, every mask m lies either above one of the 7 minimal digraphs or below one of the 5 maximal digraphs, exclusively (`mask4_dichotomy`), proving Proposition 3.

Counting the filtered canonical classes yields upward-closure size 12 (`card_avoidable_classes`) and complement size 205 (`card_unavoidable_classes`). Inspecting the arc counts of the 12 avoidable canonical masks shows that every avoidable class has at least 9 arcs (`arcCount_ge_9_of_avoidable`), all 197 non-empty classes with at most 8 arcs belong to the 205 unavoidable classes (`all_le8_arcs_unavoidable`), and exactly 6 avoidable classes have 9 arcs (`card_avoidable_9_arcs`). The other 6 avoidable classes comprise 4 of 10 arcs (namely S_{10}, $G_7 \supseteq G_2$, $G_8 \supseteq G_3$, and one supergraph of S_1), 1 of 11 arcs, and 1 of 12 arcs (Table 2). This proves Lemma 5. □

Classification bridge (Proof of Theorem 1 and Corollary 2). Let G be any loopless digraph on 4 vertices, and assume the boundary hypotheses `h_min7` (each of the 7 minimal digraphs is 2-avoidable) and `h_max5` (none of the 5 maximal digraphs is 2-avoidable), established by Proposition 7. Choose $m \in \text{Mask}_4$ with $\text{maskToDigraph}(m) \simeq G$.

- If some $\min_7(i)$ embeds into G, then Lemma 18 of [1] (`avoidable2_of_emb`) pushes 2-avoidability from $\min_7(i)$ to G.
- Conversely, suppose G is 2-avoidable. By isomorphism invariance (`avoidable2_iff_of_iso`), the decoded digraph of m is 2-avoidable. Applying Proposition 3 (`mask4_dichotomy`), either some $\min_7(i)$ embeds into m (hence into G via the isomorphism), or m embeds into some $\max_5(j)$, which by `avoidable2_of_emb` forces $\max_5(j)$ to be 2-avoidable, contradicting `h_max5`.

This proves Theorem 1 (`r16_1_1_classification`). For Corollary 2 (`r10_2_1_le8_unavoidable`), if $m \in \text{Mask}_4$ has at most 8 arcs, then $\text{canon}_4(m)$ has the same arc count (arc count is S_4-invariant), so m cannot lie in the upward closure of the 7

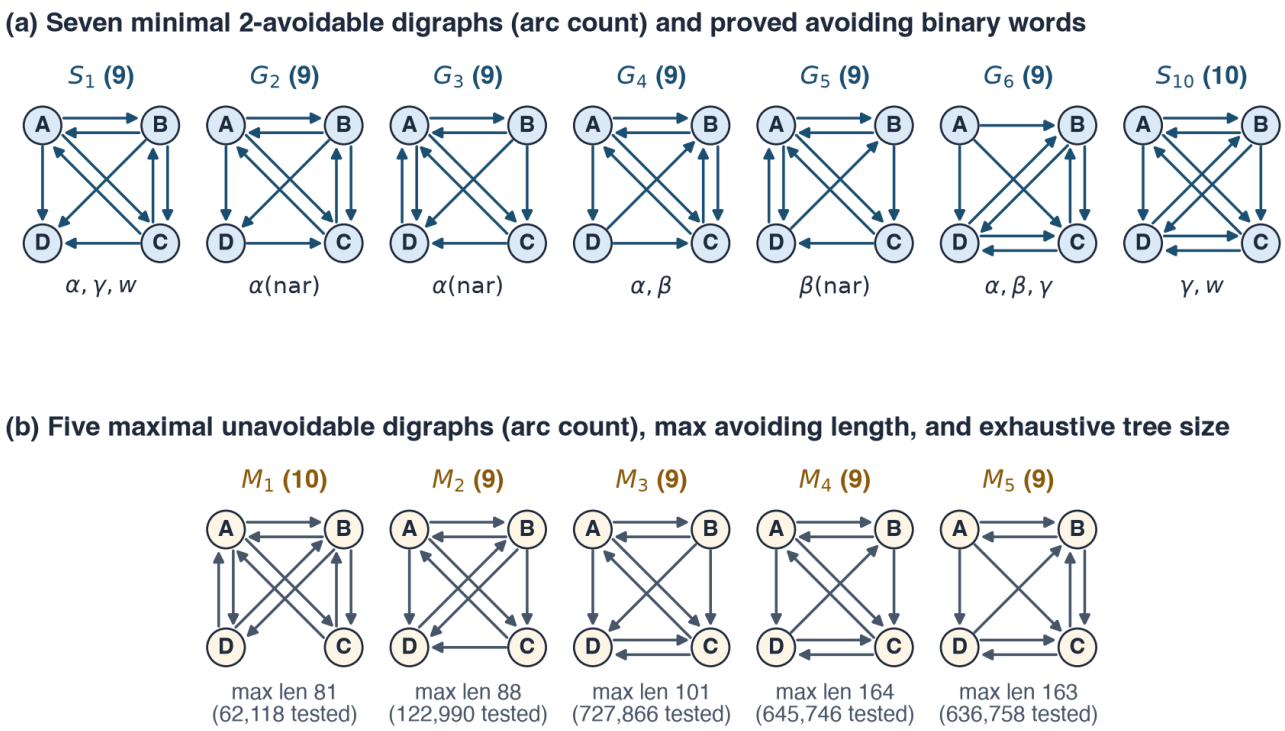


Figure 2. The complete boundary separating 2-avoidable and unavoidable four-variable loopless *xyx*-formulas. (a) The 7 minimal 2-avoidable digraphs (6 with 9 arcs, 1 with 10 arcs) and the infinite binary words proved to avoid each. (b) The 5 maximal unavoidable digraphs (1 with 10 arcs, 4 with 9 arcs), their exact maximum avoiding binary-word lengths (81 to 164), and exhaustive search-tree sizes.

Table 1. The 7 minimal 2-avoidable digraphs $(S_1, G_2, G_3, G_4, G_5, G_6, S_{10})$ and 5 maximal unavoidable digraphs $(M_1, \dots, M_5)$, with exact 12-bit Lean masks, *xyx*-formulas, and boundary certificate data.

Class	Arcs	Mask	*xyx*-formula ($A=0, B=1, C=2, D=3$)	Role	Witness / Exhaustive bound
S_1	9	511	*ABA.ACA.ADA.BAB.BCB.BDB.CAC.CBC.CDC*	Min. avoid.	α(nar), γ(nar), w
G_2	9	2303	*ABA.ACA.ADA.BAB.BCB.BDB.CAC.CBC.DCD*	Min. avoid.	α(nar)
G_3	9	895	*ABA.ACA.ADA.BAB.BCB.BDB.CAC.CDC.DAD*	Min. avoid.	α(nar)
G_4	9	3295	*ABA.ACA.ADA.BAB.BCB.CAC.CBC.DBD.DCD*	Min. avoid.	α(nar), β(nar)
G_5	9	1887	*ABA.ACA.ADA.BAB.BCB.CAC.CDC.DAD.DBD*	Min. avoid.	β(nar)
G_6	9	3511	*ABA.ACA.ADA.BCB.BDB.CBC.CDC.DBD.DCD*	Min. avoid.	α(nar), β(nar), γ(nar)
S_{10}	10	3455	*ABA.ACA.ADA.BAB.BCB.BDB.CAC.CDC.DBD.DCD*	Min. avoid.	γ(nar), w
M_1	10	1791	*ABA.ACA.ADA.BAB.BCB.BDB.CAC.CBC.DAD.DBD*	Max. dead	max len 81 (62,118 tested)
M_2	9	1407	*ABA.ACA.ADA.BAB.BCB.BDB.CAC.CDC.DBD*	Max. dead	max len 88 (122,990 tested)
M_3	9	2431	*ABA.ACA.ADA.BAB.BCB.BDB.CAC.CDC.DCD*	Max. dead	max len 101 (727,866 tested)
M_4	9	3423	*ABA.ACA.ADA.BAB.BCB.CAC.CDC.DBD.DCD*	Max. dead	max len 164 (645,746 tested)
M_5	9	3487	*ABA.ACA.ADA.BAB.BCB.CBC.CDC.DBD.DCD*	Max. dead	max len 163 (636,758 tested)

minimal classes (every class there has at least 9 arcs); hence Proposition 3 forces m to embed into some M_j, and `h_max5` rules out 2-avoidability. □

Coding injectivity and abstract descent soundness (Proof of Lemma 6). In `XYXAvoid/Descent.lean`, we also formalize the algebraic underpinnings of the morphism descent method. Every prefix code and every suffix code induces an injective morphism on finite words (`applyMorph_injective_of_prefixCode`, `applyMorph_injective_of_suffixCode`). Direct case checks show that f_{nar} and α are suffix codes and γ is a prefix code, so all three morphisms are globally injective on finite words (`applyMorph_fNar_injective`, `applyMorph_alpha_injective`, `applyMorph_gamma_injective`); β ($0 \mapsto 01$, $1 \mapsto 0$, $2 \mapsto 0110$) is uniquely decipherable on factors of nar (since 11 never occurs in nar and every 0110 parses uniquely as $\beta(2)$).

For Lemma 6 (`freeze_or_desub_soundness`), endow $\mathbb{N} \times \mathbb{N}$ with the lexicographic order $\prec_{\mathrm{lex}}$, which is well-founded (`WellFounded.prod_lex`). By well-founded induction on $p = (\mathrm{freeVars}(s), \mu)$, we show that no $s \in C$ satisfies $\mathrm{Sat}(s, \mu)$: if $\mathrm{Sat}(s, \mu)$ held, either a freeze move produces (s', μ') with $\mathrm{freeVars}(s') < \mathrm{freeVars}(s)$, or a desubstitution move produces (s', μ') with $\mathrm{freeVars}(s') = \mathrm{freeVars}(s)$ and $\mu' < \mu$; in both cases $(\mathrm{freeVars}(s'), \mu') \prec_{\mathrm{lex}} (\mathrm{freeVars}(s), \mu)$, contradicting the induction hypothesis. Applying this to $s_0 \in C$

Table 2. Exact census of all 217 non-empty isomorphism classes of loopless digraphs on 4 vertices by number of arcs (fragments).

Arcs	Total	Unavoidable	2-Avoidable	Connected (4v)
1	1	1	0	0/0
2	5	5	0	0/0
3	13	13	0	8/0
4	27	27	0	22/0
5	38	38	0	37/0
6	48	48	0	47/0
7	38	38	0	38/0
8	27	27	0	27/0
9	13	7	6	7/6
10	5	1	4	1/4
11	1	0	1	0/1
12	1	0	1	0/1
Total	217	205	12	187/12

rules out every occurrence h of F_G in W. □

4. COMPUTATIONAL CERTIFICATES, JOINT AVOIDANCE, AND EMPIRICAL PICTURE

We now detail the external computational certificates discharging Proposition 7 and Proposition 8, together with the independent cross-checks and the open conjecture on F_{G_3}.

Unavoidable side: five exhaustive binary trees (Proposition 7(a)). To prove that a digraph G is not 2-avoidable, an unpruned depth-first search grows binary words letter by letter from the empty word, maintaining the exact relation $P(w) = \{(x, y) : xyx \text{ is a factor of } w\}$ and discarding any node w admitting a homomorphism $G \to P(w)$ (checking all extensions through newly created suffix pairs xyx). When the tree is finite with maximum depth L, every binary word of length $L + 1$ contains an occurrence of F_G.

For the 5 maximal unavoidable digraphs $M_1, \dots, M_5$, three independent C/Python implementations (by separate builder, falsifier, and blind re-reviewer agents) agreed identically on every maximum avoiding length, valid-node count, and total tested-node count (Table 1, Figure 3(a)):

- M_1 (10 arcs): max length 81, 31,058 avoiding words, 62,118 tested (replayed in full by `recompute.py` in 14.2 s);
- M_2 (9 arcs): max length 88, 61,494 avoiding words, 122,990 tested;
- M_3 (9 arcs): max length 101, 363,932 avoiding words, 727,866 tested (peak 7,950 at $n = 47$);
- M_4 (9 arcs): max length 164, 322,872 avoiding words, 645,746 tested;
- M_5 (9 arcs): max length 163, 318,378 avoiding words, 636,758 tested.

In addition, a blind re-review verified plain-string prefix-code leaf certificates for all five trees, and `recompute.py` independently verifies all five logged maximum-length witnesses (confirming each avoids M_j while both one-bit extensions $w0$ and $w1$ contain occurrences). Because M_4 attains the largest depth (164), every binary word of length 165 contains an occurrence of all 5 maximal formulas, and hence of all 205 unavoidable classes.

Morphism descent (Proposition 7(b)). An infinite binary word cannot be certified by finite prefix search alone. Following Cassaigne's descent method as used by Ochem and Rosenfeld [1] for Theorem 23, suppose $W = \phi(\text{nar})$ with $\phi \in \{\alpha, \beta, \gamma\}$ contains an occurrence h of F_G. Each variable v is either *short* (below a fixed length bound, so we branch over all factors of W of that length, reducing the number of free variables in a *freeze* move) or *long* (so unique parsing lifts $h(v)$ to a ϕ-preimage over nar up to bounded boundary pieces, mapping the system to a shorter one in nar and then via f_{nar}^{-1} in a *desubstitution* move). By Lemma 6, if the reachable system graph closes with zero satisfiable states, then W avoids F_G.

Table 3 records the state-space sizes across independent implementations: the primary prover plus a separate certificate closure checker (verifying all 32 parts of the five Narayana certificates `CLOSED`), an independent falsifier prover covering all 7 minimal classes via Narayana images alone (so the full classification rests on the single base word nar), and an independent blind re-review prover written from scratch. Controls confirmed that every prover reproduced Theorem 23 of [1] (in 272–2,745 states) and reported "possible occurrence" with zero false avoidances across 23–48 test formulas known to occur. Non-descent direct checks found zero occurrences of any claimed avoided formula across all factors up to 2,200 letters (blind re-review) and across the first 1,000 letters with unbounded block length (replayed to 250 letters in `recompute.py`, which also confirms the exact shortest occurrence prefixes 13, 14, 16, 19, 22, 38, 141 for every non-avoided pairing).

Table 3. Independent Cassaigne-type descent state counts proving Proposition 7(b) (all using one underlying descent method).

Class	Witness	Prover 1	Prover 2	Blind re-review
S_1	w	5,186	47,500	772*
S_1	α(nar)	—	49,007	verified
S_1	γ(nar)	—	136,782	—
G_2	α(nar)	15,816	66,342	59,874
G_3	α(nar)	27,954	70,864	40,061
G_4	α(nar)	17,948	72,181	71,724
G_5	β(nar)	12,870	144,688	71,633
G_6	α(nar)	13,967	115,667	116,582
S_{10}	w	7,147	242,481	—
S_{10}	γ(nar)	—	169,232	verified

*Compact desubstitution-closure formulation for w avoiding S_1.

Honest methodological limit. Although three or more independent agent programs verified every descent claim, all of them implement the same overarching method (Cassaigne-type morphism descent). No independent proof by a second method (such as a Walnut first-order automatic-sequence decision procedure over Narayana numeration) and no hand proof has yet been produced.

Joint avoidance and minimal three-word cover (Proposition 8). Running the unpruned exhaustive binary tree simultaneously against all 7 minimal digraphs (`recompute.py`, 16.1 s) terminates at length 50 with 9,186 avoiding words

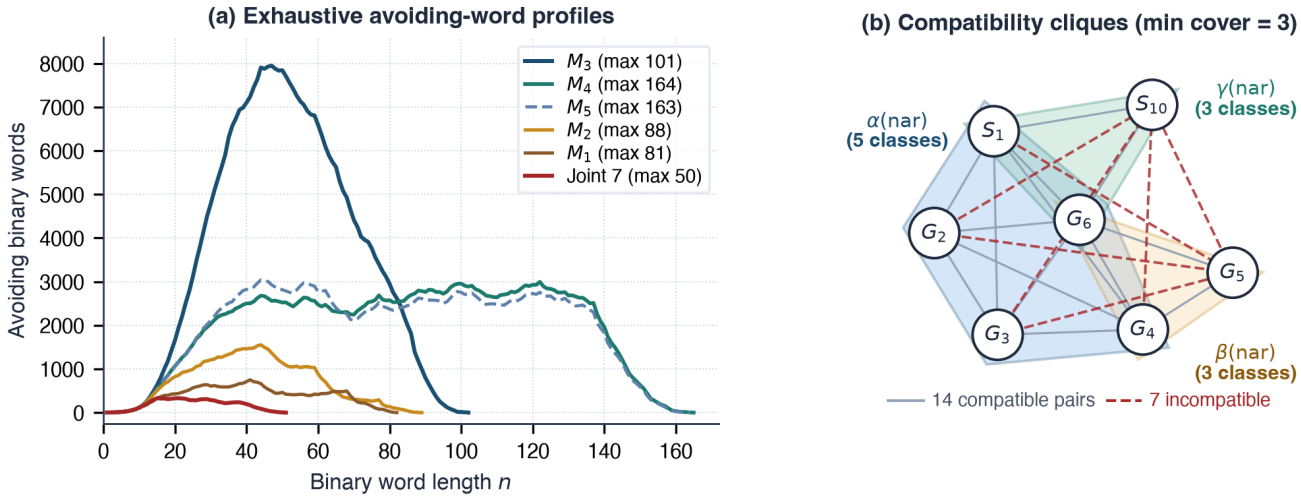


Figure 3. Exhaustive tree dynamics and joint-avoidance structure. (a) Exact counts of binary words of length n avoiding each maximal unavoidable digraph $M_1, \ldots, M_5$ (extinguishing at lengths 81, 88, 101, 164, 163) and avoiding all 7 minimal digraphs simultaneously (extinguishing at length 50, peak 330). (b) Compatibility graph of the 7 minimal classes (14 compatible pairs in grey, 7 incompatible pairs in dashed red) showing the exact 3 maximal cliques covered by $\alpha(\mathrm{nar})$, $\beta(\mathrm{nar})$, and $\gamma(\mathrm{nar})$.

(18,374 tested; peak 330 at $n = 15$ and $n = 21$; Figure 3(a)), with the lexicographically first length-50 witness

```
00110110100101001011010010100101101001011010010011
```

admitting no avoiding extension. Exhaustive pair searches (each confirmed by two independent C programs) show that among the $\binom{7}{2} = 21$ pairs of minimal classes, exactly 7 pairs are jointly unavoidable: (S_1, G_5) and (G_2, G_5) die at length 50, (G_3, G_5) at 54, (G_5, S_{10}) at 72, (G_3, S_{10}) at 152, (G_4, S_{10}) at 163, and (G_2, S_{10}) at 164. The remaining 14 pairs form a compatibility graph whose maximal cliques are exactly $\{S_1, G_2, G_3, G_4, G_6\}$, $\{G_4, G_5, G_6\}$, and $\{S_1, G_6, S_{10}\}$ (Figure 3(b)). Since $\{G_2, G_5, S_{10}\}$ is an independent set of size 3, at least 3 infinite binary words are required to cover all 7 classes, and $\{\alpha(\mathrm{nar}), \beta(\mathrm{nar}), \gamma(\mathrm{nar})\}$ achieves this exact minimum.

Empirical growth and open conjecture for F_{G_3}. During witness discovery, binary words avoiding F_{G_3} appeared unusually sparse: exact counts of length-n binary words avoiding F_{G_3} are 974 ($n = 20$), 3,348 ($n = 40$), 10,536 ($n = 60$), 25,964 ($n = 80$), 34,496 ($n = 90$), 53,332 ($n = 110$), and 73,248 ($n = 130$), with local growth exponent decreasing from 3.6 to 1.9. Seam-following tests tracked words departing from $\alpha(\mathrm{nar})$ for up to 40 letters with 30-letter left context and saw all departures extinguish within 788 letters, though 12 of 76 central length-10 factors at $n = 70$ lay outside the four complement/reversal copies of $\alpha(\mathrm{nar})$. Following the characterization framework of Badkobeh and Ochem [2], who caution that polynomial growth alone does not force uniqueness, we record this cleanly as an open conjecture (excluded from the proved claim denominator).

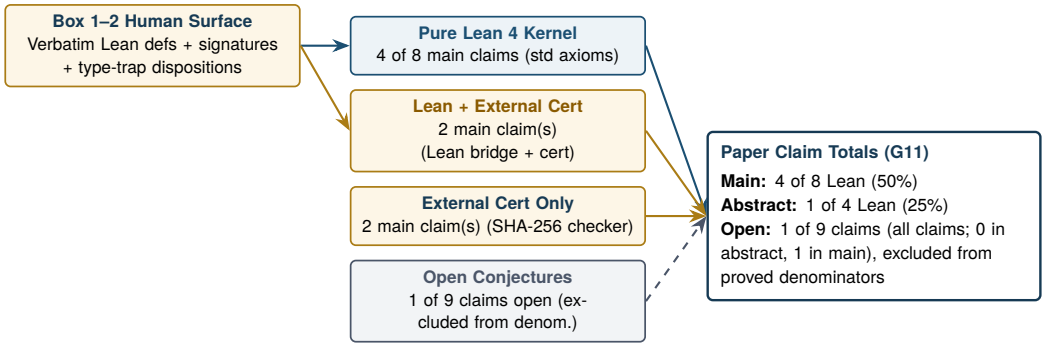
Conjecture 9 (Polynomial Growth and Factor-Set Rigidity for F_{G_3} — Open). *The number of binary words of length n avoiding F_{G_3} grows polynomially in n, and every recurrent infinite binary word avoiding F_{G_3} has the same factor set as $\alpha(\mathrm{nar})$ or one of its three complement/reversal images.*

5. FORMAL VERIFICATION & READER'S TRUST BOUNDARY

What You Must Trust to Accept This Paper:

1. **Lean 4 Kernel & Toolchain:** leanprover/lean4:v4.35.0; committed axiom receipt anchored to git commit 4d34d065283c (independent gate-box kernel replay pending); standard axioms only (propext, Classical.choice, Quot.sound).
2. **Core Definitions & Fidelity (Boxes 1–2):** Verify that the verbatim Lean definitions in Box 1 match Section 2 and that each Lean signature in Box 2 matches its stated theorem.
3. **External Certificates & Written Proofs:** 2 SHA-256-pinned external artifact(s) plus any claims marked Written or Unsettled in Table 5 below.

Reader's Trust-Flow Architecture (Distinct Evidence Paths to Claim Totals)



Anti-Shortcut Audit Strip — Source-Level Approx (Gate Replay Pending)

WARN: Committed receipt (gate replay pending) | PASS: sorry/admit: 0 (per committed receipt) | PASS: native_decide: 0 (per committed receipt) | PASS: Custom axioms & meta: 0 (per committed receipt) | PASS: Coverage: 217/217 public theorems (source-level approx) | WARN: Fidelity: 6/6 exact (draft; auditor/formalizer relationship unknown) | WARN: Witnesses: pending gate replay | WARN: Human referee: pending

Box 1 — Verbatim Lean 4 Custom Definitions (Human Audit Surface)

XYXAvoid.IsFactorInf (XYXAvoid/Basic.lean:163-164) — Section 2 (contiguous factor of an infinite word W : N -> alpha)

Context binders in scope: `variable {α β : Type*}`

```
def IsFactorInf (u : List α) (W : ℕ → α) : Prop :=
  ∃ i : ℕ, ∀ j (hj : j < u.length), W (i + j) = u[j]
```

XYXAvoid.LooplessDigraph (XYXAvoid/Basic.lean:207-209) — Section 2 (irreflexive directed graph excluding self-loops XXX)

```
structure LooplessDigraph (V : Type*) where
  adj : V → V → Prop
  loopless : ∀ v : V, ¬ adj v v
```

XYXAvoid.DigraphEmb (XYXAvoid/Basic.lean:218-220) — Section 3 (vertex-bijective subgraph embedding H -> G)

```
structure DigraphEmb {V₁ V₂ : Type*} (H : LooplessDigraph V₁) (G : LooplessDigraph
  ↪ V₂) where
  toEquiv : V₁ ≃ V₂
  map_adj : ∀ {u v : V₁}, H.adj u v → G.adj (toEquiv u) (toEquiv v)
```

XYXAvoid.OccursDigraphInf (XYXAvoid/Basic.lean:258-259) — Section 2 (Ochem-Rosenfeld XYX-formula occurrence in an infinite word)

```
def OccursDigraphInf {V α : Type*} (G : LooplessDigraph V) (W : ℕ → α) : Prop :=
  ∃ h : V → List α, (∀ v, h v ≠ []) ∧ ∀ u v, G.adj u v → IsFactorInf (h u ++ h v ++ h
    ↪ u) W
```

XYXAvoid.AvoidsDigraphInf (XYXAvoid/Basic.lean:266-267) — Section 2 (negation of OccursDigraphInf)

```
def AvoidsDigraphInf {V α : Type*} (G : LooplessDigraph V) (W : ℕ → α) : Prop :=
  ¬ OccursDigraphInf G W
```

XYXAvoid.Avoidable2 (XYXAvoid/Basic.lean:270-271) — Section 2 (existence of an infinite binary word avoiding F_G)

```
def Avoidable2 {V : Type*} (G : LooplessDigraph V) : Prop :=
  ∃ W : ℕ → Bool, AvoidsDigraphInf G W
```

XYXAvoid.Mask4 (XYXAvoid/Census.lean:166-166) — Section 3 (12-bit representation Fin 4096 of 4-vertex loopless digraphs)

```
abbrev Mask4 := Fin 4096
```

XYXAvoid.maskToDigraph (XYXAvoid/Census.lean:377-379) — Section 3 (decoding a 12-bit mask to LooplessDigraph (Fin 4))

```
def maskToDigraph (m : Mask4) : LooplessDigraph (Fin 4) where
  adj u v := u ≠ v ∧ hasBit m.val (idxOfArc u v) = true
  loopless _ h := h.1 rfl
```

XYXAvoid.min7Graph (XYXAvoid/Census.lean:1248-1248) — Section 3 & Table 1 (the 7 minimal 2-avoidable digraphs S1, G2..G6, S10)

```
def min7Graph (i : Fin 7) : LooplessDigraph (Fin 4) := maskToDigraph (min7Masks i)
```

XYXAvoid.max5Graph (XYXAvoid/Census.lean:1249-1249) — Section 3 & Table 1 (the 5 maximal unavoidable digraphs M1..M5)

```
def max5Graph (j : Fin 5) : LooplessDigraph (Fin 4) := maskToDigraph (max5Masks j)
```

XYXAvoid.embedsMask (XYXAvoid/Census.lean:1087-1088) — Section 3 (subgraph embedding on Mask4 up to S_4 vertex relabelling)

```
def embedsMask (m₁ m₂ : Mask4) : Prop :=
  ∃ k : Fin 24, subMask (permMaskNat k m₁.val) m₂.val = true
```

XYXAvoid.isAboveMin7 (XYXAvoid/Census.lean:1337-1338) — Section 3 (upward closure of min7Masks under embedsMask)

```
def isAboveMin7 (c : Mask4) : Prop :=
  ∃ i : Fin 7, embedsMask (min7Masks i) c
```

XYXAvoid.isBelowMax5 (XYXAvoid/Census.lean:1341-1342) — Section 3 (downward closure of max5Masks under embedsMask)

```
def isBelowMax5 (c : Mask4) : Prop :=
  ∃ j : Fin 5, embedsMask c (max5Masks j)
```

XYXAvoid.nonemptyClasses4 (XYXAvoid/Census.lean:1006-1007) — Section 3 (the 217 non-empty canonical S_4 isomorphism classes)

```
def nonemptyClasses4 : Finset Mask4 :=
  allClasses4.filter (fun m => m.val ≠ 0)
```

XYXAvoid.avoidableClasses4 (XYXAvoid/Census.lean:1421-1422) — Section 3 (the 12 non-empty canonical classes in the upward closure of min7)

```
def avoidableClasses4 : Finset Mask4 :=
  nonemptyClasses4.filter isAboveMin7
```

XYXAvoid.occMeasure (XYXAvoid/Descent.lean:425-426) — Section 1 & 3 (total variable-word length sum_v |h(v)|)

```
def occMeasure {V α : Type*} [Fintype V] (h : V → List α) : ℕ :=
  ∑ v : V, (h v).length
```

Supporting Semantic Closure Index (non-Box-1 helpers): XYXAvoid.S1_mask (XYXAvoid/Census.lean), XYXAvoid.G2_mask (XYXAvoid/Census.lean), XYXAvoid.G3_mask (XYXAvoid/Census.lean), XYXAvoid.G4_mask (XYXAvoid/Census.lean), XYXAvoid.G5_mask (XYXAvoid/Census.lean), XYXAvoid.G6_mask (XYXAvoid/Census.lean), XYXAvoid.S10_mask (XYXAvoid/Census.lean), XYXAvoid.M1_mask (XYXAvoid/Census.lean), XYXAvoid.M2_mask (XYXAvoid/Census.lean), XYXAvoid.M3_mask (XYXAvoid/Census.lean), XYXAvoid.M4_mask (XYXAvoid/Census.lean), XYXAvoid.M5_mask (XYXAvoid/Census.lean), XYXAvoid.min7Masks (XYXAvoid/Census.lean), XYXAvoid.max5Masks (XYXAvoid/Census.lean), XYXAvoid.hasBit (XYXAvoid/Census.lean), XYXAvoid.pow2 (XYXAvoid/Census.lean), XYXAvoid.arcOfIdx (XYXAvoid/Census.lean), XYXAvoid.idxOfArc (XYXAvoid/Census.lean), XYXAvoid.arcCount (XYXAvoid/Census.lean), XYXAvoid.subMask (XYXAvoid/Census.lean), XYXAvoid.packBits (XYXAvoid/Census.lean), XYXAvoid.mkPerm4 (XYXAvoid/Census.lean), XYXAvoid.perm4Fun (XYXAvoid/Census.lean), XYXAvoid.perm4InvIdx (XYXAvoid/Census.lean), XYXAvoid.permArc (XYXAvoid/Census.lean), XYXAvoid.permArcInv (XYXAvoid/Census.lean), XYXAvoid.permMaskNat (XYXAvoid/Census.lean), XYXAvoid.allClasses4 (XYXAvoid/Census.lean), XYXAvoid.canon4 (XYXAvoid/Census.lean).

Box 2 — Verbatim Main Theorem Signatures, Context Binders & Applicability

XYXAvoid.r16_1_1_classification (XYXAvoid/Census.lean:1533-1549) `src:72b26372bd99a6`

```
theorem r16_1_1_classification
    (h_min7 : ∀ i : Fin 7, Avoidable2 (min7Graph i))
    (h_max5 : ∀ j : Fin 5, ¬ Avoidable2 (max5Graph j))
    (G : LooplessDigraph (Fin 4)) :
    Avoidable2 G ↔ ∃ i : Fin 7, Nonempty (DigraphEmb (min7Graph i) G)
```

Assumptions: Explicit external certificate hypotheses h_min7 (7 minimal 2-avoidable digraphs via Cassaigne-type descent) and h_max5 (5 maximal unavoidable digraphs via exhaustive binary trees), applied to any loopless digraph G on Fin 4. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

XYXAvoid.r10_2_1_le8_unavoidable (XYXAvoid/Census.lean:1568-1610) `src:9bd2551cce52f3`

```
theorem r10_2_1_le8_unavoidable
    (h_max5 : ∀ j : Fin 5, ¬ Avoidable2 (max5Graph j))
    (m : Mask4) (h8 : arcCount m.val ≤ 8) :
    ¬ Avoidable2 (maskToDigraph m)
```

Assumptions: Explicit external certificate hypothesis h_max5 (5 maximal unavoidable digraphs via exhaustive binary trees), applied to any 12-bit mask m with arcCount m.val <= 8. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

XYXAvoid.mask4_dichotomy (XYXAvoid/Census.lean:1392-1418) `src:71ecf9a03087e2`

```
theorem mask4_dichotomy (m : Mask4) : isAboveMin7 m ∨ isBelowMax5 m
```

Assumptions: Universal over all 4,096 masks m : Mask4 (exclusivity proved separately by XYXAvoid.not_above_and_below). Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

XYXAvoid.card_classes_4 (XYXAvoid/Census.lean:1012-1025) `src:0dad8dd92dcfe7`

```
theorem card_classes_4 : nonemptyClasses4.card = 217
```

Assumptions: Closed equality nonemptyClasses4.card = 217 over all 4,096 loopless 4-vertex masks; companion card_all_classes_4 proves 218 with the empty digraph. Applicability: Unconditional (closed statement; no outer premise witness needed).

XYXAvoid.card_avoidable_classes (XYXAvoid/Census.lean:1457-1460) src:0ab52205d9c1dc

```
theorem card_avoidable_classes : avoidableClasses4.card = 12
```

Assumptions: Closed equality avoidableClasses4.card = 12 within the 217 non-empty classes; companions give 205 unavoidable classes and 6 classes at 9 arcs. Applicability: Unconditional (closed statement; no outer premise witness needed).

XYXAvoid.freeze_or_desub_soundness (XYXAvoid/Descent.lean:466-494) src:ca13a7e72b416d

```
theorem freeze_or_desub_soundness {V α State : Type*} [Fintype V]
    (G : LooplessDigraph V) (W : ℕ → α)
    (freeVars : State → ℕ) (Sat : State → ℕ → Prop) (Cert : Set State) (s₀ : State)
    (hstart : s₀ ∈ Cert)
    (hinit : ∀ h : V → List α, (∀ v, h v ≠ []) →
      (∀ u v, G.adj u v → IsFactorInf (h u ++ h v ++ h u) W) →
      Sat s₀ (occMeasure h))
    (hclosed : ∀ s ∈ Cert, ∀ μ : ℕ, Sat s μ →
      (∃ s' ∈ Cert, freeVars s' < freeVars s ∧ ∃ μ' : ℕ, Sat s' μ') ∨
      (∃ s' ∈ Cert, freeVars s' = freeVars s ∧ ∃ μ' < μ, Sat s' μ')) :
    AvoidsDigraphInf G W
```

Assumptions: Any finite vertex type V, alphabet alpha, state type State, measure occMeasure, and state set Cert containing s0 closed under freeze or desubstitution moves. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

Table 4. Lean 4 Type-Trap Audit Checklist — Source-Level Approximation (Gate Replay Pending).[a]

Declaration	Hazard	Operand	Category	Audited Disposition & Guard
`hasBit`	`nat_div`	`/ pow2`	intended_convention	Exact binary shift m / 2^i where pow2 i = 2^i >= 1 for all i : Fin 12 (verified by pow 2_eq).
`hasBit`	`nat_mod`	`% 2`	intended_convention	Standard parity extraction % 2 with constant nonzero modulus 2.
`instDecidableEmbedsMask`	`local_instance_or_notation`	`instDecidableEmbedsMask`	intended_convention	Standard Decidable instance inst DecidableEmbedsMask; carries no custom axioms or notation overrides.
`instDecidableIsAboveMin7`	`local_instance_or_notation`	`instDecidableIsAboveMin7`	intended_convention	Standard Decidable instance inst DecidableIsAboveMin7; carries no custom axioms or notation overrides.
`instDecidableIsBelowMax5`	`local_instance_or_notation`	`instDecidableIsBelowMax5`	intended_convention	Standard Decidable instance inst DecidableIsBelowMax5; carries no custom axioms or notation overrides.

[a] Covers the specified Lean 4 hazard set (`Nat` subtraction/division/mod, totalized real/rational division, `Real.log`/`rpow`, `Finset.range`, `Classical.choose`, local instances); no finite checklist certifies all semantics.

Table 5. Per-Claim Reader's Trust Boundary Ledger (G11 Denominators Derived Mechanically).[a]

Paper Claim	Proof Tier	Lean Decl / Artifact	Actual Axioms & Fidelity	Assumptions & What You Must Trust
Thm. 1: Loopless 4-vertex digraph G is 2-avoidable if and only if one of the 7 minimal digraphs S1, G2.. G6, S10 embeds into G	Lean + Cert	`XYXAvoid.r16_1_1_classification` + recompute.py (49fbbee793d9) [archived; replay pending]; census.py (f7b2f9b25803) [regenerable; replay pending]	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 + Mathlib kernel proves the 217-class reduction and dichotomy; external certificates discharge h_min7 (7 Cassaigne-type descents) and h_max5 (5 exhaustive binary trees).
Cor. 2: Every loopless 4-vertex digraph with at most 8 arcs defines a non-2-avoidable xyx-formula	Lean + Cert	`XYXAvoid.r10_2_1_le8_unavoidable` + recompute.py (49fbbee793d9) [archived; replay pending]; census.py (f7b2f9b25803) [regenerable; replay pending]	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Lean 4 + Mathlib kernel proves all <= 8 arc classes embed into max5; external trees discharge h_max5 (M1..M5).
Prop. 3: Every 12-bit loopless 4-vertex digraph lies exclusively above min7 or below max5	Lean Kernel	`XYXAvoid.mask4_dichotomy`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Unconditional Lean 4 + Mathlib pure-kernel proof over all 4,096 masks (exclusivity by not_above_and_below).

Paper Claim	Proof Tier	Lean Decl / Artifact	Actual Axioms & Fidelity	Assumptions & What You Must Trust
Lem. 4: Exact cardinality 217 of non-empty S_4 isomorphism classes on 4 vertices (218 with empty)	Lean Kernel	`XYXAvoid.card_classes_4`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Unconditional Lean 4 + Mathlib orbit-partition proof (4,096 = sum of 218 disjoint S_4 orbit sizes).
Lem. 5: Exact cardinality 12 of the upward closure of min7 (205 unavoidable, 6 nine-arc classes)	Lean Kernel	`XYXAvoid.card_avoidable_classes`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Unconditional Lean 4 + Mathlib pure-kernel proof (companions card_unavoidable_classes = 205, card_avoidable_9_arcs = 6).
Lem. 6: Soundness of two-move (freeze or desubstitute) lexicographic morphism descent for AvoidsDigraphInf	Lean Kernel	`XYXAvoid.freeze_or_desub_soundness`	Axioms: Classical.choice, Quot.sound, propext; fidelity: exact	Unconditional Lean 4 + Mathlib well-founded induction on Prod. Lex (<) (<) over N x N.
Prop. 7: Five maximal exhaustive binary trees (max len 81..164, bound 165) and seven Narayana/w descent closures	Cert Only	recompute.py (49fbbee793d9) [archived; replay pending]; census.py (f7b2f9b25803) [regenerable; replay pending]	External cert only; fidelity: n/a	External deterministic certificates across 3 independent programs (1 descent method; M1 tree and all 5 max witnesses replayed in recompute.py; barrier: 62k-728k trees and 5k-169k descent graphs beyond Lean kernel).
Prop. 8: Joint avoidance of all 7 minimal classes dies at len 50 (18,374 words); min 3 words (alpha, beta, gamma)	Cert Only	recompute.py (49fbbee793d9) [archived; replay pending]	External cert only; fidelity: n/a	Unpruned 7-formula exhaustive tree and 3-clique cover replayed in recompute.py; 7 pairwise death lengths (50..164) verified by 2 independent C programs.
Conj. 9: Polynomial growth and factor-set rigidity of binary words avoiding F_G3	Unsettled	Open conjecture	Lean: none; fidelity: n/a	Explicitly open conjecture supported by counts through n = 130 and seam tests (excluded from G11 denominators).

[a] Allowed pure-kernel axioms: `propext`, `Classical.choice`, `Quot.sound`. Actual axioms printed above are read directly from verified outputs. Provenance nonce: f889040bb447 | Lean mode: committed_receipt | Closure SHA-256: 92ed2e615a56 | Git anchor: 4d34d065283c | Axioms SHA-256: 76884cfcb660 | Fidelity SHA-256: b3a46d0b3500.

6. PRIOR WORK AND OPEN QUESTIONS

Prior literature. Ochem and Rosenfeld [1] introduced *xyx*-formulas, established the digraph representation and homomorphism monotonicity (Lemma 18), proved that the 4-fragment formula $H = ABA.BAB.ACA.CBC$ is not 2-avoidable (Theorem 19), showed that Thue's ternary word w_3 avoids the transitive tournament TT_3 (Theorem 15), proved that the complete 12-arc digraph on 4 vertices is 2-avoidable (Theorem 23), and posed both four-variable and five-vertex transitive tournament problems in Section 7 (Discussion). Narayana's word is classical [7], and its combinatorial properties were recently surveyed by Shallit [3]. Badkobeh and Ochem [2] study characterizations of morphic words by avoided formulas, framing Conjecture 9. Related formula-avoidance families include two-variable formulas [4] [to check], circular formulas [5] [to check], and avoidance bases for formulas with reversal [6] [to check].

Logged prior-work searches and unsearched gaps. Four independent search logs passed `litsearch.py check` (exit 0; 29 + 16 + 29 + 16 = 90 keyed queries across arXiv, OEIS, zbMATH, OpenAlex, Semantic Scholar, MathOverflow, and Math Stack Exchange; 32 candidate records evaluated, alongside manual queries on Zenodo, DataCite, HAL, OSF Preprints, and GitHub), supplemented by a keyed CORE sweep (7 queries, 23 records) and a Google Scholar sweep (6 queries). Neither the journal nor the arXiv printing of [1] contains a classification on 4 variables or uses Narayana's word, and this classification was not found in the searches logged. *Not searched or unread:* Lothaire's three combinatorics-on-words volumes, Cassaigne's 1994 and Clark's 2001 theses, WORDS and DLT proceedings (2022–2026) beyond arXiv/HAL/DataCite/Scholar indexing, private conference talks beyond Ochem's May 2024 seminar slides, and full texts of the `[to check]` contextual references [4, 5, 6].

Open questions and companion work. Four natural directions remain open:

1. *Independent method for Proposition 7(b):* Obtain a non-descent verification (e.g., via Walnut over Narayana numeration) or a human-checkable structural proof that $\alpha(\text{nar})$, $\beta(\text{nar})$, $\gamma(\text{nar})$ avoid the 7 minimal formulas, or replay the descent certificates directly inside Lean 4.
2. *Five-variable classification:* The second Discussion item of [1] (showing that the 10-fragment transitive tournament TT_5 on 5 vertices is 2-avoidable) has been resolved separately using a binary image $\tau(b_5)$ of Ochem and Rosenfeld's 5-letter word b_5 and is documented in a companion study; the full 5-variable census remains open beyond 9 fragments.

3. *Growth and rigidity of F_{G_3}:* Resolve Conjecture 9.
4. *Higher avoidability indices:* Extend the complete four-variable classification to 3-avoidability and general avoidability indices.

7. DATA AVAILABILITY, REPRODUCIBILITY, AND USE OF AI

All manuscript sources, figures, verification scripts, and ledgers reside in `papers/xyx-formulas-on-four-variables/draft1/`, and the Lean 4 + Mathlib library resides in `lean/XYXAvoid/` (217 audited public theorems, 0 `sorry`, standard three axioms only). Running `python3 papers/xyx-formulas-on-four-variables/draft1/recompute.py` replays the complete 4,096 → 217 orbit census, the 12/205 partition, all 12 Lean mask encodings, Narayana lengths and coding checks, the Section 1 witness snippet, all 5 maximal unavoidable longest-word witnesses, the full unpruned M_1 binary tree (62,118 tested words), the full unpruned 7-formula joint-avoidance tree (18,374 tested words), the compatibility clique cover, and the 250-letter prefix checks in roughly 32 s.

AI research agents built and directed by Tyler B. Hughes generated the computational searches, descent proofs, Lean 4 + Mathlib formalization, verification scripts, figures, and manuscript text. No human mathematician has yet refereed the proof.

Drafted: November 2026 **Updated:** November 24, 2026

REFERENCES

(1) Ochem, P.; Rosenfeld, M. Avoidability of palindrome patterns. *Electron. J. Combin.* **2021**, *28* (1), #P1.4. DOI: 10.37236/9593; arXiv:2005.05505.

(2) Badkobeh, G.; Ochem, P. More characterizations of morphic words. Preprint, **2023**. arXiv:2312.10757.

(3) Shallit, J. The Narayana morphism and related words. Preprint, **2025**. arXiv:2503.01026.

(4) Ochem, P.; Rosenfeld, M. Avoidability of formulas with two variables. *Electron. J. Combin.* **2017**, *24* (4), #P4.30. arXiv:1606.03955.

(5) Gamard, G.; Ochem, P.; Richomme, G.; Séébold, P. Avoidability of circular formulas. *Theoret. Comput. Sci.* **2018**, *726*, 1–4. arXiv:1610.04439.

(6) Currie, J. D.; Mol, L.; Rampersad, N. Avoidance bases for formulas with reversal. *Theoret. Comput. Sci.* **2018**, *738*, 25–41. arXiv:1711.08025.

(7) OEIS Foundation Inc. Entry A105083: Trajectory of 1 under the morphism $1 \mapsto 12$, $2 \mapsto 3$, $3 \mapsto 1$. *The On-Line Encyclopedia of Integer Sequences*, https://oeis.org/A105083.