MATHDISCOVERIES
Mathematical Preprint

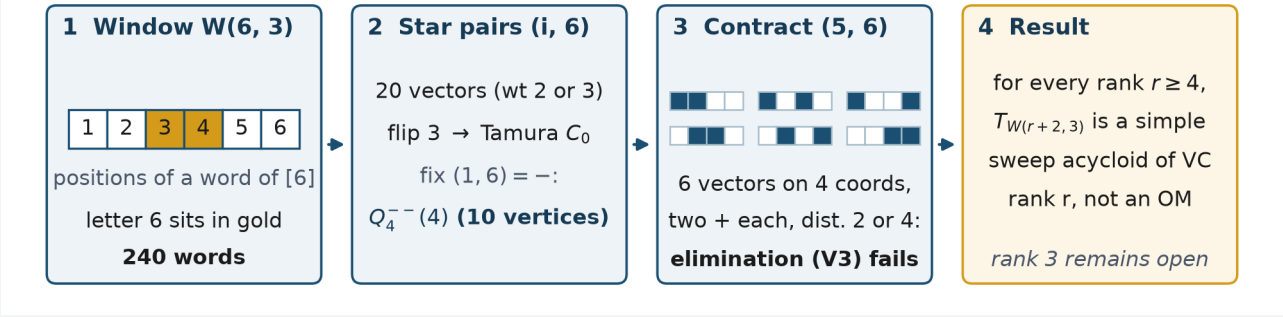
Working Draft

# Padrol–Philippe (Combinatorica 2024) Question 7.21 Answered No in Every Rank $r \geq 4$: Window Sweep Acycloids $W(n,m)$

Tyler B. Hughes
Independent Researcher, United States

*Supporting Information:* `recompute.py`, *Lean 4 library* `lean/SweepAcycloid`, *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:** Padrol and Philippe's Question 7.21 (whether every sweep acycloid is an oriented matroid) has answer no in every rank $r \geq 4$ (VC dimension); rank 3 remains open. For $m \geq 1$ and $2m \leq n$, let $W(n,m)$ be the words of $[n]$ with at least $m-1$ letters on each side of $n$: it is reversal-closed, bijects with $\{0,\dots,n-2m+1\} \times S_{n-1}$ of size $(n-2m+2)(n-1)!$, and for $n \geq 4$ geodesic adjacent-swap walks exist and every pair is flipped by a single swap, yielding a simple allowable graph and simple sweep acycloid $T_{W(n,m)}$. A pair family is shattered if and only if it is a forest whose $n$-component has size at most $n-m+1$, giving VC dimension $n-1$ ($m=1, n \geq 2$) and $n-2$ ($m \geq 2, n \geq 4$) and breaking matroid augmentation for $m \geq 3$. For $m \geq 3$, $T_{W(n,m)}$ is no oriented-matroid tope set, witnessed by a non-sweep potential sweep, a Knauer–Marc minor $Q^{--}_{n-2m+4}(n-2m+4)$, and Tamura's $C_0$ at $(6,3)$. For $n \geq 4$ and $m \leq 2$, $W(n,m)$ is realizable over every linearly ordered commutative ring. All 14 of 14 main results (16 of 16 numbered claims, 100%) are proved in pure-kernel Lean 4.

**Verification Scope (G11):** WARN: Human referee: pending
Main results: 14 of 14 in Lean Kernel (100%)
Abstract claims: 14 of 14 in Lean Kernel (100%)

**Full Trust Boundary:** Section 5.

**MSC 2020:** 52C40, 05B35, 05A05 **Keywords:** sweep acycloid, oriented matroid, allowable graph of permutations, VC dimension, excluded minor

## 1. INTRODUCTION AND MAIN RESULTS

Sweeping a hyperplane across a finite set of points in a generic direction lists the points in a total order, and the orders obtained over all directions encode the oriented geometry of the configuration. We show that the natural combinatorial axioms for such sets of orders admit explicit families that no oriented matroid, and hence no point configuration, can produce.

Padrol and Philippe [1] abstracted sets of sweep orders to *allowable graphs of permutations* and the corresponding sets of sign vectors to *sweep acycloids* [1, Definitions 7.5, 7.10, and 7.13; Theorem 7.15], and proved that the tope set of every sweep oriented matroid is a sweep acycloid [1, Corollary 7.18]. They asked whether the converse holds:

> *Question 7.21* [1]. Is every sweep acycloid an oriented matroid?

They had not been able to construct a counterexample [1, Section 7.4], and the question reappears in Philippe's thesis [2, Section 3.7.4, Question 3.7.21, pp. 125–127].

We answer the question in the negative with an explicit two-parameter family. For integers $m \geq 1$ and $2m \leq n$, let $W(n,m)$ be the set of permutations of $[n]$, written as words, in which the distinguished letter $n$ has at least $m-1$ letters before it and at least $m-1$ letters after it. For example, $W(6,3)$ consists of the 240 words of $[6]$ in which the letter 6 stands in position 3 or 4. Write $T_{W(n,m)}$ for the set of sign vectors of these words (Section 2). In Theorem 1, *rank* means VC dimension (Definition 5), because [1, Section 3.1] defines rank only for oriented matroids; the two notions agree on every finite loopless oriented matroid (Remark 23).

**The proof arc and the companion results.** Theorem 1 below is the case $m = 3$ of a uniform analysis over all admissible $(n,m)$, which runs in three steps. First, geodesic adjacent-swap walks and single-pair edges (Theorem 8) make $W(n,m)$ a simple allowable graph and $T_{W(n,m)}$ a sim-

ple sweep acycloid (Theorem 9). Second, a pair family is shattered if and only if it is a forest whose $n$-component has at most $n-m+1$ vertices (Theorem 10), which fixes the VC dimension (Theorem 11 and Proposition 12). Third, restriction to the star pairs $(i,n)$ and a repaired contraction rule (Lemma 15) turn any oriented matroid with topes $T_{W(n,m)}$, $m \geq 3$, into one with a constant-weight tope set, which violates covector elimination (Theorem 17). Companion results count $W(n,m)$ (Corollary 7), give three independent witnesses of non-matroidality (Theorems 13, 18 and 19) and Tamura's $C_0$ inside $T_{W(6,3)}$ (Proposition 21), and show that the threshold $m=3$ is sharp (Theorem 22). Figure 1 maps every numbered result to its proof in the pinned Lean 4 library of Section 5: all 14 of 14 main results (16 of 16 numbered claims, 100%) are proved inside the Lean 4 kernel without Mathlib or external axioms.

**Theorem 1** (Negative answer to Question 7.21 in every rank $r \geq 4$). *Let $r \geq 4$ be an integer and put $n = r+2$. Then:*

(a) *$W(n,3)$ is a simple allowable graph of permutations (Definition 2, formalizing [1, Definition 7.5]);*

(b) *$T_{W(n,3)}$ is a simple acycloid and a sweep acycloid on the pairs of $[n]$ (Definitions 3 and 4, formalizing [1, Definition 7.10 and the tope reading of Definition 7.13]; extension to all covectors follows by [1, Section 7.4 and Lemma 7.17]);*

(c) *some injective family of $r$ pairs is shattered by $T_{W(n,3)}$, and no injective family of $r+1$ pairs is, so $T_{W(n,3)}$ has VC dimension $r$;*

(d) *no oriented matroid on the pairs of $[n]$ has $T_{W(n,3)}$ as its set of topes.*

*Consequently, Question 7.21 of [1] has the answer no in every rank $r \geq 4$ (rank meaning VC dimension).*

**Discovery and proof architecture.** The counterexample was found by an automated SAT-solver search (CaDiCaL, with DRAT proofs checked by `drat-trim`) and a C census over simple sweep acycloids on $n=6$ letters, whose least non-matroidal simple example was a relabelled copy of $W(6,3)$. It was generalized analytically and put through independent falsification and blind review, which repaired a misworded contraction rule (Lemma 15(b)–(c)) and contributed the da Silva witness (Theorem 18) and the identification of Tamura's $C_0$ (Proposition 21). The upstream SAT, DRAT, and C search files are not included in this artifact bundle, and no theorem or minimality claim in this paper relies on them.

The self-contained Python snippet below checks the smallest case $W(6,3)$ directly from permutations of 6 letters: its 240 words, its 20 star vectors on $(i,6)$, its 10-vertex Knauer–Marc minor $Q_4^{--}(4)$ obtained by fixing $(1,6)$ to $-$, and its 6 terminal vectors after contracting $(5,6)$, whose pairwise Hamming distances are only 2 or 4.

**Instant Witness Check (run verbatim by `recompute.py`)**

```
from itertools import combinations, permutations
W = [p for p in permutations(range(6)) if p.index(5) in (2, 3)]
star = {tuple(i in p[:p.index(5)] for i in range(5)) for p in W}
qmm = {v[1:] for v in star if not v[0]}
cut = {v[:4] for v in star if v[:4] + (not v[4],) in star}
dist = {sum(a != b for a, b in zip(u, v))
        for u, v in combinations(cut, 2)}
assert len(W) == 240 and len(star) == 20 and len(qmm) == 10
assert len(cut) == 6 and dist == {2, 4}
```

The box uses 0-based letters, so letter 5 of the code is letter 6 of the text. Section 2 fixes the definitions, Section 3 gives the proof arc (the proofs of the supporting lemmas and of four companion results are in Appendix A), Section 4 records the exact computations, Section 5 presents the Lean trust boundary, and Section 6 discusses prior work and open problems.

## 2. DEFINITIONS AND CONVENTIONS

Throughout, $[n]=\{1,\dots,n\}$ and $E_n=\{(i,j):1\leq i<j\leq n\}$. The Lean 4 library indexes letters by $0,\dots,n-1$ through $a \mapsto a-1$; the distinguished letter $x=n$ corresponds to $n-1$ in `Fin n`. A permutation $\sigma \in S_n$ is written as the word $\sigma(1)\sigma(2)\cdots\sigma(n)$, where $\sigma^{-1}(a)$ is the 1-based position of letter $a$. Its reversal is the word $\overline{\sigma}$ given by $\overline{\sigma}(t)=\sigma(n+1-t)$. Two permutations differ by an *adjacent transposition* if one swaps $\sigma(k)$ and $\sigma(k+1)$ for some $1\leq k<n$.

**Window sets and sign vectors.** For integers $m \geq 1$ and $2m \leq n$, define the *window set*

$$W(n,m)=\{\sigma\in S_n : m\leq \sigma^{-1}(n)\leq n+1-m\}.$$

Following [1, Section 3.2, map (5)], each permutation $\sigma \in S_n$ defines a sign vector $X^\sigma \in \{+,0,-\}^{E_n}$ with $X^\sigma_{(i,j)}=+$ if $\sigma^{-1}(i)<\sigma^{-1}(j)$ and $-$ otherwise. Write $T_S=\{X^\sigma:\sigma\in S\}$ for any $S\subseteq S_n$. The *inversion distance* $d(\sigma,\tau)$ is the number of pairs $e\in E_n$ with $X^\sigma_e \neq X^\tau_e$.

**Allowable graphs and acycloids.** We restate Padrol and Philippe's Definitions 7.1, 7.5, 7.10, and 7.13 in the simple-move form formalized in `Draft2Upgrades.lean`. For any $T\subseteq\{+,0,-\}^E$, elements $e,f\in E$ are *parallel in $T$* if either $X_e=X_f$ for all $X\in T$ or $X_e=-X_f$ for all $X\in T$ ([1, Section 3.1, Footnote 4]).

**Definition 2** (Simple allowable graph of permutations). A *simple move* $\sigma \to \mu$ flipping $e\in E_n$ is an adjacent transposition whose sign vectors differ on $e$ and agree on every $f\neq e$. A *simple allowable sequence* inside $S\subseteq S_n$ from $\sigma$ to $\tau$ is a sequence of simple moves staying in $S$ in which no pair $e\in E_n$ is flipped twice [1, Definition 7.1]. A subset $S\subseteq S_n$ is a *simple allowable graph of permutations* ([1, Definition 7.5]) if $S$ is nonempty, closed under word reversal (P1), and every $\sigma,\tau\in S$ are joined by a simple allowable sequence inside $S$ (P2) (for single-pair moves, the intersection axiom (P3) holds automatically).

**Definition 3** (Acycloid and simple acycloid). A nonempty subset $T\subseteq\{+,0,-\}^E$ is an *acycloid* ([1, Definition 7.10]; [3, Section 3]) if all elements of $T$ share the same support, $T=-T$, and for all distinct $X,Y\in T$ there exists $f\in E$ with $X_f=$

| (i) What was found | (ii) Origin and verification | (iii) How it was proved |
|---|---|---|
| **Q7.21 answered no in every rank $r \geq 4$** | Discovery: $n = 6$ SAT + C → $W(n, m)$ | **Lean kernel** Thm 1 |
| $W(n, m)$ reversal-closed, explicit bijection | Analytic $W(n, m)$; check: $n = 4$ to 8 | **Lean kernel** Prop 6 |
| exact cardinality $(n - 2m + 2)(n - 1)!$ | Analytic recursion; check: $n = 4$ to 8 | **Lean kernel** Cor 7 |
| geodesic swap walks, single-pair edges | Analytic swaps; check: BFS $n = 6$ | **Lean kernel** Thm 8 |
| $W(n, m)$ allowable graph, $T_{W(n,m)}$ sweep acycloid | Analytic Defs. 7.5, 7.10, 7.13 (tope) | **Lean kernel** Thm 9 |
| shattered sets = forests, $\lvert C_F(n)\rvert \leq n - m + 1$ | Analytic forests; check: Cayley $n = 6, 7$ | **Lean kernel** Thm 10 |
| VC rank exactly $n - 2$ for $n \geq 4, m \geq 2$ | Analytic VC bound; check: $n = 6, 7$ | **Lean kernel** Thm 11 |
| VC rank of $W(n, 1) = S_n$ is $n - 1$ $(n \geq 2)$ | Analytic specialization at $m = 1$ | **Lean kernel** Prop 12 |
| shattered complex fails matroid (I3) $(m \geq 3)$ | Analytic star $A$ vs. disjoint tree $B$ | **Lean kernel** Thm 13 |
| restriction and repaired contraction | Blind review repaired contraction rule | **Lean kernel** Lem 15 |
| every finite oriented matroid has a tope | Analytic max-support on Fin($k$) | **Lean kernel** Lem 16 |
| not an oriented matroid for $m \geq 3$ | Written from $n = 6$; check: (6, 3), (7, 3) | **Lean kernel** Thm 17 |
| da Silva: potential sweep, not a sweep | Blind review witness; check: $n = 6$ sweeps | **Lean kernel** Thm 18 |
| explicit Knauer-Marc minor $Q_N^{--}(N)$ $(m \geq 3)$ | Analytic halfspaces; check: $n = 6, 7$ | **Lean kernel** Thm 19 |
| Tamura's $C_0$ in $W(6, 3)$ (reorient coord 3) | Blind review + Handa 1993 Prop. 7.7 | **Lean kernel** Prop 21 |
| realizable for $n \geq 4, m \leq 2$ over ordered rings | Analytic simplex; check: $n = 4$ to 7 | **Lean kernel** Thm 22 |
| rank 3: is every sweep acycloid an OM? | not addressed by this work | **Unsettled** |

**Figure 1.** Discovery and proof architecture: (i) what was found, (ii) how each claim originated and was checked (upstream $n = 6$ SAT and C search files are not included in this bundle), and (iii) how each numbered claim was proved (proof badges are read directly from `trust-boundary.json`). All 14 main results and 2 supporting lemmas (16 of 16 numbered claims) are proved in pure-kernel Lean 4. The rank-3 case of Question 7.21 is unsettled and is excluded from the denominator.

$-Y_f \neq 0$ and a vector $Z \in T$ such that $Z_e = -X_e$ for every $e$ parallel to $f$ in $T$ and $Z_e = X_e$ otherwise. An acycloid is *simple* if every $X \in T$ has full support (no zero coordinates) and every parallelism class is a singleton.

**Definition 4** (Sweep acycloid). A triple $(a, b, c) \in \{+, 0, -\}^3$ is *orthogonal to* $(+, +, -)$ if either $a = b = c = 0$, or both $\{+, +, -\}$ and $\{-, -, +\}$ share at least one nonzero entry with $(a, b, c)$ (for full-support triples, this excludes only $(+, +, -)$ and $(-, -, +)$; [1, Lemma 3.8]). An acycloid $T$ on $E_n$ is a *sweep acycloid* ([1, Definition 7.13]) if every $X \in T$ satisfies orthogonality to $(+, +, -)$ on every triple $(i, j), (j, k), (i, k)$ with $i < j < k$, and whenever $\{a, b\}$ and $\{b, c\}$ are parallel in $T$ for distinct $a, b, c \in [n]$, $\{a, b\}$ and $\{a, c\}$ are also parallel in $T$.

**Remark on Definition 7.13(i) (tope versus covector wording).** The published clause (i) of [1, Definition 7.13] reads verbatim: "(i) its topes fulfill the transitivity condition from Lemma 3.8; namely for every covector $X$ and every choice of $1 \leq i < j < k \leq n$, the triple $(X_{(i,j)}, X_{(j,k)}, X_{(i,k)})$ is orthogonal to the sign vector $(+, +, -)$." Definition 4 and the Lean 4 predicate `IsSweepAcycloid` impose this orthogonality condition on the topes $X \in T$, which is the reading used in Padrol and Philippe's proof of [1, Theorem 7.15]. The extension from topes and parallelism classes to all covectors (faces) of $T$ is established in [1, Section 7.4 and Lemma 7.17] and enters here as a cited written step rather than a Lean 4 step.

### Shattered families and VC dimension.

**Definition 5** (VC dimension). A family $P = (e_1, \ldots, e_k)$ of pairs in $E_n$ is *shattered* by $T_S$ if for every $s \in \{+, -\}^k$ there exists $\sigma \in S$ with $X^{\sigma}_{e_t} = s_t$ for all $1 \leq t \leq k$. The *VC dimension* of $T_S$ is the largest $k$ for which an injective family of $k$ pairs in $E_n$ is shattered by $T_S$.

Viewing a family $P$ of pairs in $E_n$ as an undirected graph on vertex set $[n]$, we say $P$ is a *forest* if removing any edge index $r$ leaves no path in $P \setminus \{r\}$ between the endpoints of $P(r)$ (so a forest is automatically injective, with no repeated edges), and we write $C_P(x) \subseteq [n]$ for the connected component of vertex $x$ in $P$.

**Oriented matroids, minors, and Knauer–Marc's $Q_N^{--}(N)$.** On $\{+,0,-\}$, define sign negation by $-0=0$, $-(+)=-$, $-(-)=+$, conformal composition by $(a\circ b)=a$ if $a\neq 0$ and $b$ otherwise, and the conformal partial order $X\preceq Y$ if $X_e\in\{0,Y_e\}$ for all $e\in E$. A sign-vector system $\mathcal{L}\subseteq\{+,0,-\}^E$ satisfies the *oriented-matroid covector axioms* (V0)–(V3) ([1, Definition 3.1]) if $0\in\mathcal{L}$ (V0), $-\mathcal{L}=\mathcal{L}$ (V1), $X,Y\in\mathcal{L}\Rightarrow X\circ Y\in\mathcal{L}$ (V2), and for $X,Y\in\mathcal{L}$ and $e\in E$ with $X_e=-Y_e\neq 0$ there exists $Z\in\mathcal{L}$ with $Z_e=0$ and $Z_f=(X\circ Y)_f$ whenever $X_f\neq -Y_f$ or $X_f=Y_f=0$ (V3). The maximal elements of $\mathcal{L}$ under $\preceq$ are its *topes*. For any map $\iota: F\to E$ the *restriction* is $\mathcal{L}|_\iota=\{X\circ\iota: X\in\mathcal{L}\}$, and for $e\in E$ the *contraction* is $\mathcal{L}/e=\{X|_{E\setminus\{e\}}: X\in\mathcal{L},\ X_e=0\}$. For $N\geq 4$, Knauer and Marc [4, Section 5, p. 21] define the partial cube $Q_N^{--}(N)$ by deleting from the $N$-cube $\{+,-\}^N$ the all-minus vertex, all $N$ vertices of weight 1, and the all-plus vertex; its vertex set is

$$Q_N^{--}(N)=\{v\in\{+,-\}^N: 2\leq \#\{i: v_i=+\}\leq N-1\}.$$

**Sweeps, da Silva's partition, Tamura's $C_0$, and realizations.** An *ordered partition* $I=(B_1,\dots,B_k)$ of $[n]$ is refined by $\sigma\in S_n$ if every element of $B_a$ precedes every element of $B_b$ in $\sigma$ whenever $a<b$. The composition $I\circ\sigma\in S_n$ lists the blocks $B_1,\dots,B_k$ in order and breaks ties inside each block according to $\sigma$. Following [1, Definition 7.16], $I$ is a *sweep* of $S$ if $I\circ\sigma\in S$ for every $\sigma\in S$, and a *potential sweep* of $S$ if some $\sigma\in S$ refines $I$ and $I\circ\overline{\sigma}\in S$ for every $\sigma\in S$ refining $I$. For $n\geq 4$, put $I^{\mathrm{dS}}=(\{1\},\{3,\dots,n\},\{2\})$.

In Handa [3, Proposition 7.7, p. 262] (citing A. Tamura, private communication, 1985, reference [48], p. 265), the five circuit representatives of Tamura's non-matroidal acycloid $C_0$ on five elements are $0{+}{-}{+}{+}$, $+0{-}{+}{+}$, $+{+}0{+}{+}$, $+{+}{-}0{+}$, and $+{+}{-}{+}0$; a full-support vector $v\in\{+,-\}^5$ is a tope of $C_0$ if none of these five circuits or their negatives conforms to $v$.

Finally, for a linearly ordered commutative ring $R$ with $0<1$, points $P:[n]\to R^d$, and direction $u\in R^d$, a permutation $\sigma\in S_n$ is the *sweep order* of $P$ along $u$ if $\sigma^{-1}(i)<\sigma^{-1}(j)$ holds if and only if $\langle u,P(i)\rangle<\langle u,P(j)\rangle$ for all $i,j\in[n]$.

## 3. MAIN THEOREMS AND PROOFS

**Proposition 6** (Reversal and bijection). *Let $m\geq 1$ and $2m\leq n$. Then:*

(a) *$W(n,m)$ is closed under word reversal: $\sigma\in W(n,m)$ implies $\overline{\sigma}\in W(n,m)$;*

(b) *there is an explicit bijection between $W(n,m)$ and $\{0,\dots,n-2m+1\}\times S_{n-1}$.*

*Proof.* (a) Reversing a word sends the 1-based position $p=\sigma^{-1}(n)$ to $n+1-p$; the condition $m\leq p\leq n+1-m$ is invariant under $p\mapsto n+1-p$. (b) Send $\sigma\in W(n,m)$ to the pair $(\sigma^{-1}(n)-m,\ \hat{\sigma})$, where $\hat{\sigma}\in S_{n-1}$ deletes $n$ from the word $\sigma$. The inverse inserts $n$ at position $m+q$ in $\hat{\sigma}$ for $0\leq q\leq n-2m+1$. □

**Corollary 7** (Window cardinality). *For all integers $m\geq 1$ and $2m\leq n$, there is an explicit bijection between $W(n,m)$ and $\mathrm{Fin}((n-2m+2)(n-1)!)$, and every bijection of $W(n,m)$ with $\mathrm{Fin}(c)$ satisfies $c=(n-2m+2)(n-1)!$.*

The proof is in Appendix A.

**Theorem 8** (Geodesic walks and single-pair edges). *Let $n\geq 4$, $m\geq 1$, and $2m\leq n$. Then:*

(a) *for all $\sigma,\tau\in W(n,m)$ there is an adjacent-transposition walk inside $W(n,m)$ from $\sigma$ to $\tau$ of length $d(\sigma,\tau)$;*

(b) *every adjacent-transposition walk inside $W(n,m)$ from $\sigma$ to $\tau$ has length at least $d(\sigma,\tau)$;*

(c) *for every pair $e\in E_n$ there exist $\sigma,\tau\in W(n,m)$ differing by one adjacent transposition whose sign vectors differ on $e$ and agree on every other pair;*

(d) *for every $\sigma\in S_n$ and $1\leq i<j<k\leq n$, if $X^\sigma_{(i,j)}=X^\sigma_{(j,k)}$ then $X^\sigma_{(i,k)}=X^\sigma_{(i,j)}$.*

*Proof.* (a) Proceed by induction on $d(\sigma,\tau)$. If $\sigma\neq\tau$, some adjacent pair $(a,b)$ in $\sigma$ is inverted in $\tau$ (otherwise $\sigma$ would be sorted by $\tau^{-1}$, forcing $\sigma=\tau$). Swapping $(a,b)$ reduces the inversion distance by 1 and stays in $W(n,m)$ unless $n\in\{a,b\}$ and the swap pushes $n$ outside $[m,n+1-m]$. Suppose $n$ stands at the left boundary $p=m$ and $(a,b)=(a,n)$ is inverted in $\tau$. Since $a$ precedes $n$ in $\sigma$ and follows $n$ in $\tau$, while $\tau^{-1}(n)\geq m$, there must exist another letter $c$ that follows $n$ in $\sigma$ and precedes $n$ in $\tau$. If some adjacent pair strictly on the left of $n$ or strictly on the right of $n$ is inverted in $\tau$, swapping it reduces $d(\sigma,\tau)$ by 1 without moving $n$. Otherwise, the letters to the right of $n$ in $\sigma$ are already in $\tau$-order; let $d$ be the letter immediately right of $n$ in $\sigma$. Since the right block is in $\tau$-order, $\tau^{-1}(d)\leq\tau^{-1}(c)<\tau^{-1}(n)$, so $(n,d)$ is an adjacent $\tau$-inverted pair, and swapping it moves $n$ rightward to position $m+1\leq n+1-m$, which lies inside the window because $2m\leq n$. The right-boundary case $p=n+1-m$ is symmetric. (b) Each adjacent swap changes exactly one coordinate of $X^\sigma$, hence changes $d(\cdot,\tau)$ by at most 1. (c) For $e=(i,j)$ with $i<j<n$, place $i,j$ in adjacent slots avoiding position $m$ (possible since $n\geq 4$) and put $n$ at position $m$. For $e=(i,n)$, place $i$ at position $m$ and $n$ at $m+1$ (possible since $2m\leq n$ gives $m+1\leq n+1-m$). Swapping them stays in $W(n,m)$ and flips only $e$. (d) Immediate from transitivity of $<$ on $\sigma^{-1}$. □

**Theorem 9** (Simple allowable graph and simple sweep acycloid). *Let $n\geq 4$, $m\geq 1$, and $2m\leq n$. Then $W(n,m)$ is a simple allowable graph of permutations (Definition 2), and $T_{W(n,m)}$ is both a simple acycloid (Definition 3) and a sweep acycloid (Definition 4).*

*Proof.* Nonemptiness of $W(n,m)$ follows by applying Theorem 8(c) to pair $(1,2)$, and reversal closure (P1) is Proposition 6(a). Along any geodesic walk of length $d(\sigma,\tau)$ from Theorem 8(a), each step flips a single pair $e$ and drops inversion distance to $\tau$ by 1, which forces $X^\mu_e=X^\tau_e$ and therefore prevents $e$ from ever being flipped again along the remaining geodesic; this yields a simple allowable sequence inside $W(n,m)$ and verifies Definition 2.

For $T_{W(n,m)}$, every $X^\sigma$ has full support and reversal closure gives $-T_{W(n,m)}=T_{W(n,m)}$. Theorem 8(c) shows that no two distinct pairs $e\neq f$ are parallel in $T_{W(n,m)}$, so every parallelism class is a singleton. For distinct $X^\sigma,X^\tau\in T_{W(n,m)}$, the very first step $\sigma\to\mu$ of a geodesic walk from Theorem 8(a) flips a single pair $f$ on which $X^\sigma_f=-X^\tau_f$ and leaves

all other pairs fixed, proving the acycloid axiom of Definition 3. Finally, Theorem 8(d) forbids $(+,+,-)$ and $(-,-,+)$ on every triple $(i,j),(j,k),(i,k)$, verifying orthogonality to $(+,+,-)$, and singleton parallelism classes make the triangle parallelism axiom vacuous. □

**Theorem 10** (Exact characterization of shattered sets). *Let $m \geq 1$ and $2m \leq n$, and let $P = (e_1, \dots, e_k)$ be any family of pairs in $E_n$. Then $P$ is shattered by $T_{W(n,m)}$ if and only if $P$ is a forest on $[n]$ whose connected component $C_P(n)$ containing letter $n$ satisfies $|C_P(n)| + m \leq n + 1$ (equivalently $|C_P(n)| \leq n - m + 1$).*

*Proof. Necessity.* Suppose $P$ is shattered by $T_{W(n,m)}$. If $P$ were not a forest, some edge $e_r = (u, v)$ would have a shortest path $u = w_0, w_1, \dots, w_L = v$ in $P \setminus \{r\}$ (with $L = 1$ if $e_r$ is repeated). Orienting every path step forward ($w_t$ before $w_{t+1}$) and $e_r$ backward ($v$ before $u$) defines a consistent sign pattern on $P$ that no total order $\sigma \in S_n$ can realize, a contradiction. Hence $P$ is a forest, and in particular 2-colorable. Orient every edge in $C_P(n)$ by increasing graph distance from $n$ (endpoints of every edge in a 2-colorable graph have opposite parity, hence distinct distances from $n$). Any word $\sigma \in W(n,m)$ realizing this pattern places $n$ strictly before every other vertex of $C_P(n)$, so at least $|C_P(n)| - 1$ letters follow $n$ in $\sigma$. Thus $\sigma^{-1}(n) + |C_P(n)| - 1 \leq n$, and combining with $m \leq \sigma^{-1}(n)$ gives $|C_P(n)| + m \leq n + 1$.

*Sufficiency.* Let $P$ be a forest with $c = |C_P(n)| \leq n - m + 1$, and fix any target sign pattern on $P$. Inducting on edges, shift one component at a time to obtain an injective topological rank $\mathrm{rk} : [n] \to \mathbb{N}$ realizing the target orientation on every edge of $P$. Compress rk to ranks $r_{\mathrm{in}} : C_P(n) \to \{0, \dots, c-1\}$ inside $C_P(n)$ and $r_{\mathrm{out}} : ([n] \setminus C_P(n)) \to \{0, \dots, n-c-1\}$ outside $C_P(n)$. Let $a = r_{\mathrm{in}}(n) \in \{0, \dots, c-1\}$, and set $p_{\mathrm{left}} = \max(0,\ m - 1 - a)$. Because $c \leq n - m + 1$, we have $p_{\mathrm{left}} \leq m - 1 \leq n - c$. Place the first $p_{\mathrm{left}}$ vertices of $[n] \setminus C_P(n)$ before all of $C_P(n)$ (preserving $r_{\mathrm{out}}$), then place $C_P(n)$ in order of $r_{\mathrm{in}}$, and finally place the remaining vertices of $[n] \setminus C_P(n)$. Every edge of $P$ connects vertices in the same side of the partition $(C_P(n), [n] \setminus C_P(n))$, so every target sign is preserved, while letter $n$ lands at 1-based position $p_{\mathrm{left}} + a + 1 = \max(a + 1, m)$. Since $m \leq \max(a + 1, m)$ and $a + 1 \leq c \leq n - m + 1$ while $m \leq n - m + 1$, the resulting word lies in $W(n,m)$. □

**Theorem 11** (VC dimension for $m \geq 2$). *Let $n \geq 4$, $m \geq 2$, and $2m \leq n$. Some injective family of $n - 2$ pairs is shattered by $T_{W(n,m)}$, and no injective family of $n - 1$ pairs is. Hence the VC dimension of $T_{W(n,m)}$ is exactly $n - 2$.*

*Proof.* The path $P = \{(1,2), (2,3), \dots, (n-2, n-1)\}$ has $n-2$ edges, is acyclic, and leaves letter $n$ isolated ($|C_P(n)| = 1 \leq n - m + 1$), so it is shattered by Theorem 10 (and directly by inserting $n$ at position $m$ in the inductive path realization). Conversely, any shattered family $P$ of $n - 1$ pairs is a forest on $n$ vertices (Theorem 10), hence a spanning tree with $|C_P(n)| = n$, violating $|C_P(n)| \leq n - m + 1 \leq n - 1$. □

**Proposition 12** (VC dimension for $m = 1$). *Let $n \geq 2$. Some injective family of $n-1$ pairs is shattered by $T_{W(n,1)} = T_{S_n}$, and no injective family of $n$ pairs is. Hence the VC dimension of $T_{W(n,1)}$ is exactly $n - 1$.*

*Proof.* The path $\{(1,2), \dots, (n-1, n)\}$ is a forest of $n-1$ edges and trivially satisfies $|C_P(n)| \leq n = n-1+1$, so it is shattered by Theorem 10. Conversely, adding each edge of a forest on $n$ vertices strictly decreases the number of connected components (counted by component minima), so no forest on $n$ vertices has $n$ edges. □

**Theorem 13** (Failure of matroid augmentation on the shattered complex). *Let $m \geq 3$ and $2m \leq n$, and set $a = n - m$ and $b = a + 1$. Then there exist injective families $A$ of $a$ pairs and $B$ of $b$ pairs, both shattered by $T_{W(n,m)}$, such that for every $e \in B \setminus A$, the family $A \cup \{e\}$ is not shattered by $T_{W(n,m)}$. Thus the shattered complex of $T_{W(n,m)}$ fails the matroid augmentation axiom (I3).*

*Proof.* Since $m \geq 3$ and $2m \leq n$, we have $a = n - m \geq 3$ and $a + 2 \leq n - 1$. Take $A$ to be the star $\{(i, n) : 1 \leq i \leq a\}$ of size $a$, whose $n$-component is $\{1, \dots, a, n\}$ of size $a + 1 = n - m + 1$. Take $B$ to be the tree on $\{1, \dots, a+2\}$ formed by the path $(1,2), \dots, (a-1, a)$ together with the two leaves $(1, a+1)$ and $(1, a+2)$, which has $b = a+1$ edges and leaves $n$ isolated ($|C_B(n)| = 1$). Both $A$ and $B$ are shattered by Theorem 10. Every edge $e \in B$ lies outside $A$: if $e$ is a path edge $(j, j+1)$ $(1 \leq j < a)$, then $A \cup \{e\}$ contains the triangle $j$–$(j+1)$–$n$ and is not a forest; if $e = (1, w)$ with $w \in \{a+1, a+2\}$, then $A \cup \{e\}$ attaches $w$ to $C_A(n)$, raising $|C_{A \cup \{e\}}(n)|$ to $a + 2 = n - m + 2 > n - m + 1$. In every case $A \cup \{e\}$ fails Theorem 10. □

**Remark 14** (Phase transition and analytic Cayley counts). By Theorem 10, for $m = 1$ the shattered sets are the independent sets of the graphic matroid $M(K_n)$; for $m = 2$ the bound $|C_P(n)| \leq n - 1$ rules out only spanning trees, so they form the $(n-2)$-truncation of $M(K_n)$, still a matroid; for $m \geq 3$ the bound $|C_P(n)| \leq n - m + 1 \leq n - 2$ breaks augmentation (Theorem 13). Since the shattered subsets of a finite loopless oriented matroid are the independent sets of its underlying matroid (Remark 23), this is an independent obstruction to oriented matroidality. For $m \geq 2$ the shattered $(n-2)$-sets are the 2-component spanning forests of $K_n$ with $|C_F(n)| = c \leq n - m + 1$; by Cayley's formula their number is

$$\sum_{c=1}^{n-m+1} \binom{n-1}{c-1} c^{c-2} (n-c)^{n-c-2},$$

with $1^{-1} = 1$ (evaluated in Section 4 and Figure 2(a)).

**Lemma 15** (Restriction and repaired contraction). *Let $\mathcal{L}$ be a sign-vector system on an arbitrary ground set $E$ satisfying the oriented-matroid covector axioms (V0)–(V3).*

(a) *For every map $\iota : F \to E$ the restriction $\mathcal{L}|_\iota$ satisfies (V0)–(V3). If every tope of $\mathcal{L}$ has full support and $\mathcal{L}$ has a tope, then the topes of $\mathcal{L}|_\iota$ are exactly the vectors $T \circ \iota$ with $T$ a tope of $\mathcal{L}$.*

(b) *For every $e \in E$ the contraction $\mathcal{L}/e$ satisfies (V0)–(V3). If every tope of $\mathcal{L}$ has full support and some $Y_0$ has both $(Y_0, +)$ and $(Y_0, -)$ as topes, then the topes of $\mathcal{L}/e$ are exactly the vectors $Y$ with both $(Y, +)$ and $(Y, -)$ topes of $\mathcal{L}$.*

(c) *The second hypothesis of (b) cannot be weakened to "$e$ is not a loop": on two elements, $\{00, ++, --\}$ satisfies (V0)–(V3), its topes have full support, the second element is not a loop, no $Y$ has both extensions as topes, and yet the zero vector is a tope of the contraction at the second element.*

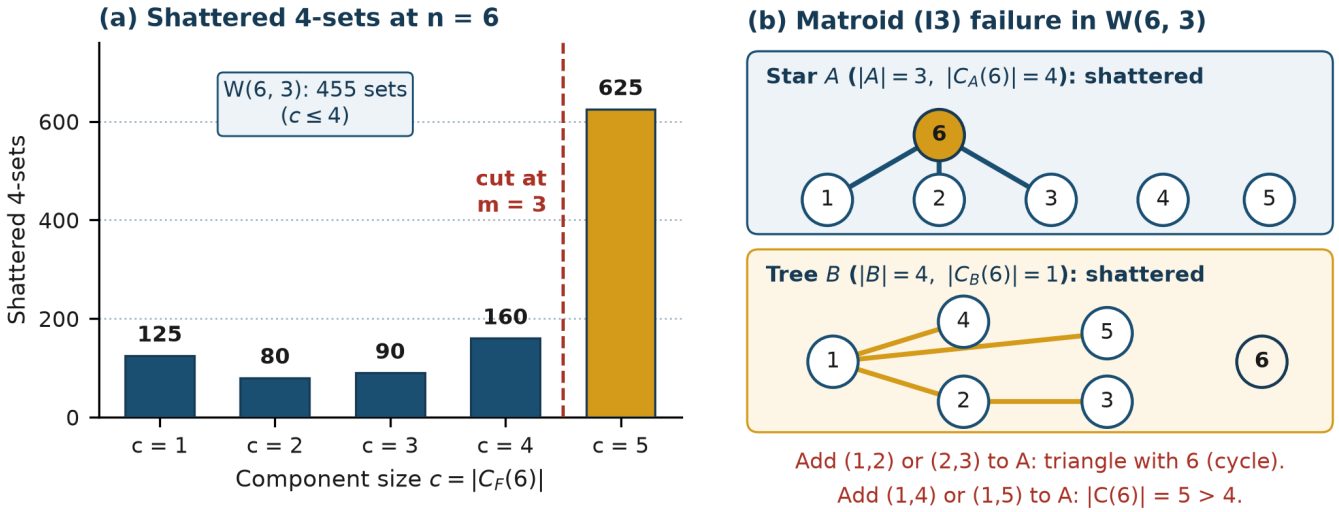



**Figure 2.** The shattered-set phase transition at $n = 6$ (Theorems 10 and 13). (a) Exact Cayley decomposition of the 1,080 two-component spanning forests of $K_6$ by component size $c = |C_F(6)|$: $W(6,2)$ shatters all 1,080 (the 4-truncation of the graphic matroid $M(K_6)$), while $W(6,3)$ enforces $c \le 4$, cutting off the 625 forests with $c = 5$ and leaving $125 + 80 + 90 + 160 = 455$. (b) The star $A$ ($|A| = 3$, $|C_A(6)| = 4$) and disjoint tree $B$ ($|B| = 4$) witnessing the failure of matroid augmentation (I3) in $W(6,3)$.

The proof is in Appendix A.

**Lemma 16** (Finite-ground-set tope existence). *Every oriented matroid $\mathcal{L}$ on a finite ground set $[k]$ possesses at least one tope.*

The proof is in Appendix A.

**Theorem 17** (Not an oriented matroid). *Let $m \ge 3$ and $2m \le n$. There is no oriented matroid on $E_n$ whose set of topes is $T_{W(n,m)}$.*

*Proof.* Suppose $\mathcal{L}$ were such an oriented matroid; its topes have full support and $T_{W(n,m)}$ is nonempty. Restrict along $i \mapsto (i,n)$ for $1 \le i \le n-1$. By Lemma 15(a), the topes of the restriction are the full sign vectors on $n-1$ elements whose number of plus signs lies in $[m-1, n-m]$, because the letters other than $n$ can be ordered arbitrarily before and after $n$. Write $B(k,a,b)$ for the full sign vectors on $k$ elements with between $a$ and $b$ plus signs. For $0 \le a < b \le k$, both $(Y,+)$ and $(Y,-)$ belong to $B(k,a,b)$ if and only if $Y \in B(k-1,a,b-1)$, which is nonempty since $a \le b-1 \le k-1$. Applying Lemma 15(b) repeatedly for $n-2m+1$ contractions of the last coordinate yields an oriented matroid whose topes are $B(2k,k,k)$ with $k = m-1 \ge 2$.

Take $X$ equal to $+$ on $\{1,\dots,k\}$ and $-$ elsewhere, and $Y$ equal to $+$ on $\{2,\dots,k+1\}$ and $-$ elsewhere. Elimination (V3) at coordinate 1 produces a covector $Z$ with $Z_1 = 0$, $Z = +$ on $\{2,\dots,k\}$, and $Z = -$ on $\{k+2,\dots,2k\}$. If $Z_{k+1} = +$, then $Z \circ X$ is a tope with $k+1$ plus signs; if $Z_{k+1} = -$, then $Z \circ Y$ has $k+1$ minus signs; if $Z_{k+1} = 0$, taking $T \in B(2k,k,k)$ to be $+$ on $\{1,\dots,k-1\} \cup \{k+1\}$ gives a tope $Z \circ T$ with $k+1$ plus signs. Every case contradicts constant weight $k$. □

**Theorem 18** (A potential sweep that is not a sweep). *Let $m \ge 3$ and $2m \le n$. The ordered partition $I^{\mathrm{dS}} = (\{1\}, \{3,\dots,n\}, \{2\})$ is a potential sweep of $W(n,m)$ and not a sweep.*

*Proof.* A word placing 1 first, 2 last, and $n$ at position $m$ refines $I^{\mathrm{dS}}$ and lies in $W(n,m)$. For any $\sigma \in W(n,m)$ refining $I^{\mathrm{dS}}$, the composition $I^{\mathrm{dS}} \circ \overline{\sigma}$ keeps 1 first and 2 last while reversing the middle block $\{3,\dots,n\}$, moving $n$ from position $p$ to $n+1-p$, which stays in the window. For the failure, take $\tau \in W(n,m)$ with $1,2$ in positions $1,2$ and $n$ in position $m$ (possible because $m \ge 3$); then $I^{\mathrm{dS}} \circ \tau$ moves 2 to the end and shifts $n$ to position $m-1$, outside the window. □

**Theorem 19** (Explicit Knauer–Marc excluded partial-cube minor). *Let $m \ge 3$ and $2m \le n$, and set $k_+ = m-3$, $k_- = m-2$, and $N = n-2m+4$. Then $N \ge 4$, and a sign vector $v \in \{+,0,-\}^N$ satisfies*

$$v_i = X^{\sigma}_{(k_+ + k_- + i,\, n)} \quad (1 \le i \le N)$$

*for some $\sigma \in W(n,m)$ with $X^{\sigma}_{(j,n)} = +$ for $1 \le j \le k_+$ and $X^{\sigma}_{(k_+ + j,\, n)} = -$ for $1 \le j \le k_-$ if and only if $v \in Q_N^{--}(N) = \{v \in \{+,-\}^N : 2 \le \#\{i : v_i = +\} \le N-1\}$.*

The proof is in Appendix A.

**Remark 20** (Independent bridges to the published criteria). By [1, Corollaries 7.18 and 7.20], a sweep acycloid $T_S$ is an oriented matroid if and only if every potential sweep of $S$ is a sweep (da Silva [5]), and also if and only if the tope graph of $T_S$ avoids the excluded partial-cube minor family $\mathcal{Q}^- = \{Q_r^{-*}, Q_r^{--}(s) : r \ge 4,\ 1 \le s \le r\}$ of Knauer and Marc [4, Theorem 1.1, Section 5 (p. 21), and Corollary 7.2]. Restricting $T_{W(n,m)}$ to the star pairs contracts all non-star edges of the partial cube, and fixing $k_+ + k_- = 2m-5$ star coordinates takes $2m-5$ halfspace restrictions (without hyperplane contractions). So Theorems 18, 19 and 13 give three further routes to Theorem 17 through these cited theorems; Theorem 17 itself is proved in Lean 4 without any external criterion.

**Proposition 21** (Tamura's acycloid inside $W(6,3)$). *For a sign vector $v \in \{+,0,-\}^5$ the following are equivalent: (i) $v_i = X^{\sigma}_{(i,6)}$ for $1 \le i \le 5$ and some $\sigma \in W(6,3)$; (ii) $v$ has full support*

*and exactly two or three plus signs; (iii) $v$ with its third coordinate negated is a tope of Tamura's acycloid $C_0$ [3, Proposition 7.7, p. 262].*

The proof is in Appendix A.

**Theorem 22** (Realizability for $m \le 2$). *Let $n \ge 4$ and let $R$ be a linearly ordered commutative ring with $0 < 1$.*

(a) *There is $P : [n] \to R^{n-1}$ whose sweep orders over all directions are exactly the words of $W(n,1) = S_n$.*

(b) *There is $P : [n] \to R^{n-2}$ whose sweep orders over all directions are exactly the words of $W(n,2)$.*

The proof is in Appendix A.

*Proof of Theorem 1.* Apply Theorems 9, 11, and 17 with $n = r + 2 \ge 6$ and $m = 3$. □

**Remark 23** (Rank and VC dimension). In any finite loopless oriented matroid the subsets shattered by the tope set are precisely the independent sets of its underlying matroid [7]: an independent restriction is free, so every sign pattern lifts to a tope, whereas a dependent subset contains a signed circuit whose sign pattern is excluded by circuit–covector orthogonality (not formalized in Lean here). Hence VC dimension coincides with oriented-matroid rank whenever a finite loopless oriented matroid exists. For example, for $m = 2$ and $n \ge 4$, $W(n,2)$ is the sweep-order set of a configuration spanning $\mathbb{R}^{n-2}$ (Theorem 22(b)), whose sweep oriented matroid has rank $n-2$ in [1, Definition 3.2], matching Theorem 11; similarly $W(n,1)$ spans $\mathbb{R}^{n-1}$ and has VC dimension $n-1$ (Proposition 12).

## 4. COMPUTATIONAL VERIFICATION AND EMPIRICAL PICTURE

Every number in this section is recomputed from the definitions by `recompute.py` in exact integer arithmetic, reading each computed sentence from this source file. The cardinality formula of Corollary 7 was confirmed by enumeration for every $n$ from 4 to 8 and every admissible $m$.

**The contraction chain and Knauer–Marc minors.** In the proof of Theorem 17, at $(n,m) = (6,3)$ the star restriction has $20 = \binom{5}{2} + \binom{5}{3}$ vectors and the terminal contraction has $6 = \binom{4}{2}$. At $(n,m) = (7,3)$ the star restriction has $50 = \binom{6}{2} + \binom{6}{3} + \binom{6}{4}$ vectors and the terminal contraction has $6 = \binom{4}{2}$, with 2 contractions passing through 20 vectors. As the Python box checks at $(6,3)$, the 6 terminal vectors pairwise differ in 2 or 4 coordinates. For Theorem 19, at $(n,m) = (6,3)$ the single halfspace restriction fixing $(1,6)$ to $-$ produces $Q_4^{--}(4)$ with 10 vertices on 4 coordinates, and at $(n,m) = (7,3)$ the same single halfspace restriction produces $Q_5^{--}(5)$ with 25 vertices on 5 coordinates.

**Shattered sets and sweeps.** For Theorems 10 and 11, at $n = 6$ there are 1,365 four-element subsets of $E_6$ and 3,003 five-element subsets: $W(6,2)$ shatters all 1,080 two-component spanning forests $(125+80+90+160+625 = 1{,}080$ by component size $c = 1,\dots,5)$, whereas $W(6,3)$ shatters exactly 455 four-element sets $(125+80+90+160 = 455$, cutting off the 625 forests with $c = 5)$ and 0 five-element sets. At $n = 7$, the Cayley sum of Remark 14 gives 13,377 two-component spanning forests in total $(1{,}296+750+720+960+1{,}875+7{,}776 = 13{,}377)$, of which $W(7,3)$ shatters exactly 5,601 five-element sets (all $c \le 5$), verified by direct enumeration over all 20,349 five-element subsets of $E_7$. For Theorem 13 at $(6,3)$, the star $A$ of size 3 and tree $B$ of size 4 are both shattered, while all 4 single-edge extensions $A \cup \{e\}$ $(e \in B)$ are not shattered. Table 1 reports words, VC dimension, sweeps, and potential sweeps at $n = 6$; the script also checks every sweep count by literal composition and verifies Theorem 8(a) by breadth-first search from every word.

**Table 1.** Window sets at $n = 6$: words, VC dimension, sweeps, potential sweeps, and potential sweeps that are not sweeps.

| | words | VC | sweeps | potential | gap |
|---|---|---|---|---|---|
| $W(6,1)$ | 720 | 5 | 4,683 | 4,683 | 0 |
| $W(6,2)$ | 480 | 4 | 2,521 | 2,521 | 0 |
| $W(6,3)$ | 240 | 4 | 1,051 | 1,071 | 20 |

The realizable windows have gap 0, and all 20 non-sweep potential sweeps of $W(6,3)$ have block sizes $(1,4,1)$, matching $I^{\mathrm{dS}}$.

**Witnesses.** For Theorem 18, the word $\tau = 126345 \in W(6,3)$ gives $I^{\mathrm{dS}} \circ \tau = 163452$, with the letter 6 in position 2. For Proposition 21, the full sign vectors to which none of the circuits of $C_0$ conforms, reoriented at element 3, are exactly the 20 vectors of the star restriction of $T_{W(6,3)}$. As a discriminating check on the conformity direction, $+{+}{+}{+}{+}$ is excluded because the circuit $+{+}0{+}{+}$ conforms to it. For Theorem 22, for every $n$ from 4 to 7, every word of $W(n,1)$ and of $W(n,2)$ is the sweep order of the constructed integer direction, with exact large-term cancellation; the reverse inclusion is proved analytically.

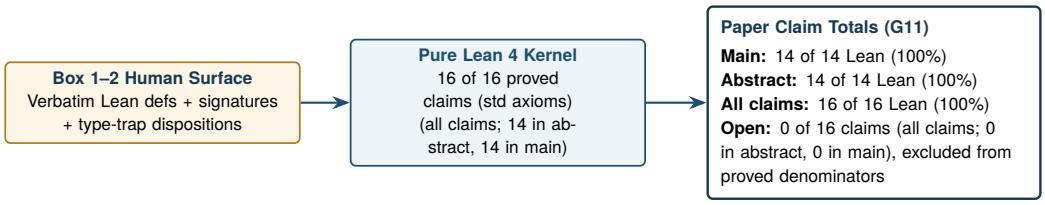
**Lean replay.** The file `Verify.lean` lists 72 declarations; the replayed `axioms.out` has 74 lines (2 multi-line wraps), and 0 of them name an axiom outside `propext`, `Quot.sound`, and `Classical.choice`. The library (toolchain `v4.35.0`) uses no Mathlib or external dependency. Eight core theorems were graded `exact` cross-family by an OpenAI blind Lean auditor (`lean/SweepAcycloid/audit.md`), and the eight theorems of `Draft2Upgrades.lean` by an Anthropic blind referee (for Theorems 1 and 9 under the disclosed tope reading of [1, Definition 7.13(i)]). `Sanity.lean` holds 17 closed non-vacuity witnesses, 16 of them for the 16 numbered claims (plus the superseded predecessor witness). Section 5 is generated mechanically from the manifest and the Lean sources.

## 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 ded7fe7259b4 (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 Artifacts & Remaining Claims:** No external certificate artifacts are required; any claims marked Written or Unsettled rely on standard written proofs or remain explicitly unproved.

**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: 72/72 public theorems (source-level approx) | WARN: Fidelity: 16/16 exact (draft cross-family)
WARN: Witnesses: pending gate replay | WARN: Human referee: pending

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

**SweepAcycloid.AdjSwapFlips (SweepAcycloid/Draft2Upgrades.lean:216-217) — Section 2**

```
def AdjSwapFlips {n : Nat} (σ μ : Perm n) (e : Pair n) : Prop :=
  IsAdjSwap σ μ ∧ signVec σ e ≠ signVec μ e ∧ ∀ f : Pair n, f ≠ e → signVec σ f =
    ↪ signVec μ f
```

**SweepAcycloid.Bijection (SweepAcycloid/Defs.lean:35-39) — Section 2**

```
structure Bijection (α β : Type) where
  toFun : α → β
  inv : β → α
  left_inv : ∀ a : α, inv (toFun a) = a
  right_inv : ∀ b : β, toFun (inv b) = b
```

**SweepAcycloid.Conforms (SweepAcycloid/Defs.lean:113-114) — Section 2**

```
def Conforms {E : Type} (X Y : E → Sign) : Prop :=
  ∀ e : E, X e = Sign.zero ∨ X e = Y e
```

**SweepAcycloid.ContractOM (SweepAcycloid/Defs.lean:139-141) — Section 2**

```
def ContractOM {E : Type} (L : (E → Sign) → Prop) (e : E)
    (Y : {f : E // f ≠ e} → Sign) : Prop :=
  ∃ X : E → Sign, L X ∧ X e = Sign.zero ∧ ∀ f : {f : E // f ≠ e}, Y f = X f.val
```

**SweepAcycloid.GraphPathAvoid (SweepAcycloid/Draft2Upgrades.lean:576-583) — Section 2**

```
inductive GraphPathAvoid {n k : Nat} (P : Fin k → Pair n) (forbid : Fin k) :
    Fin n → Fin n → Prop where
  | refl (u : Fin n) : GraphPathAvoid P forbid u u
  | step {u w v : Fin n} (r : Fin k) :
      r ≠ forbid →
      (((P r).fst = u ∧ (P r).snd = w) ∨ ((P r).fst = w ∧ (P r).snd = u)) →
      GraphPathAvoid P forbid w v →
      GraphPathAvoid P forbid u v
```

**SweepAcycloid.InWindow (SweepAcycloid/Defs.lean:66-67) — Section 2**

```
def InWindow (n m : Nat) (σ : Perm n) : Prop :=
  ∀ x : Fin n, x.val + 1 = n → m ≤ (σ.inv x).val + 1 ∧ (σ.inv x).val + m ≤ n
```

**SweepAcycloid.IsAcycloid (SweepAcycloid/Draft2Upgrades.lean:172-180) — Section 2**

```
def IsAcycloid {E : Type} (T : (E → Sign) → Prop) : Prop :=
  (∃ X : E → Sign, T X) ∧
  (∀ X Y : E → Sign, T X → T Y → ∀ e : E, X e ≠ Sign.zero ↔ Y e ≠ Sign.zero) ∧
  (∀ X : E → Sign, T X → T (fun e => (X e).negate)) ∧
  (∀ X Y : E → Sign, T X → T Y → X ≠ Y →
    ∃ f : E, ((X f = Sign.pos ∧ Y f = Sign.neg) ∨ (X f = Sign.neg ∧ Y f = Sign.pos)) ∧
      ∃ Z : E → Sign, T Z ∧
        (∀ e : E, ParallelIn T f e → Z e = (X e).negate) ∧
        (∀ e : E, ¬ ParallelIn T f e → Z e = X e))
```

**SweepAcycloid.IsAdjSwap (SweepAcycloid/Defs.lean:76-80) — Section 2**

```
def IsAdjSwap {n : Nat} (σ τ : Perm n) : Prop :=
  ∃ k : Nat, ∃ hk : k + 1 < n,
    τ.toFun ⟨k, Nat.lt_trans (Nat.lt_succ_self k) hk⟩ = σ.toFun ⟨k + 1, hk⟩ ∧
    τ.toFun ⟨k + 1, hk⟩ = σ.toFun ⟨k, Nat.lt_trans (Nat.lt_succ_self k) hk⟩ ∧
    ∀ i : Fin n, i.val ≠ k → i.val ≠ k + 1 → τ.toFun i = σ.toFun i
```

**SweepAcycloid.IsForest (SweepAcycloid/Draft2Upgrades.lean:586-587) — Section 2**

```
def IsForest {n k : Nat} (P : Fin k → Pair n) : Prop :=
  ∀ r : Fin k, ¬ GraphPathAvoid P r (P r).fst (P r).snd
```

**SweepAcycloid.IsFullSupport (SweepAcycloid/Defs.lean:117-118) — Section 2**

```
def IsFullSupport {E : Type} (X : E → Sign) : Prop :=
  ∀ e : E, X e ≠ Sign.zero
```

**SweepAcycloid.IsKnauerMarcQmm (SweepAcycloid/Draft2Upgrades.lean:2327-2330) — Section 2**

```
def IsKnauerMarcQmm (N : Nat) (v : Fin N → Sign) : Prop :=
  IsFullSupport v ∧
  2 ≤ sumFin N (fun i => if v i = Sign.pos then 1 else 0) ∧
  sumFin N (fun i => if v i = Sign.pos then 1 else 0) + 1 ≤ N
```

**SweepAcycloid.IsOrientedMatroid (SweepAcycloid/Defs.lean:121-128) — Section 2**

```
def IsOrientedMatroid {E : Type} (L : (E → Sign) → Prop) : Prop :=
  L (fun _ => Sign.zero) ∧
  (∀ X : E → Sign, L X → L (fun e => (X e).negate)) ∧
  (∀ X Y : E → Sign, L X → L Y → L (fun e => (X e).comp (Y e))) ∧
  (∀ X Y : E → Sign, L X → L Y → ∀ e : E, X e = Sign.pos → Y e = Sign.neg →
    ∃ Z : E → Sign, L Z ∧ Z e = Sign.zero ∧
      ∀ f : E, ¬ (X f = Sign.pos ∧ Y f = Sign.neg) →
        ¬ (X f = Sign.neg ∧ Y f = Sign.pos) → Z f = (X f).comp (Y f))
```

**SweepAcycloid.IsPartitionComp (SweepAcycloid/Defs.lean:165-167) — Section 2**

```
def IsPartitionComp {n : Nat} (blk : Fin n → Nat) (σ τ : Perm n) : Prop :=
  ∀ i j : Fin n, τ.inv i < τ.inv j ↔
    (blk i < blk j ∨ (blk i = blk j ∧ σ.inv i < σ.inv j))
```

**SweepAcycloid.IsPotentialSweep (SweepAcycloid/Defs.lean:175-178) — Section 2**

```
def IsPotentialSweep {n : Nat} (S : Perm n → Prop) (blk : Fin n → Nat) : Prop :=
  (∃ σ : Perm n, S σ ∧ RefinesPartition σ blk) ∧
  (∀ σ ρ : Perm n, S σ → RefinesPartition σ blk → IsReverse σ ρ →
    ∃ τ : Perm n, IsPartitionComp blk ρ τ ∧ S τ)
```

**SweepAcycloid.IsReverse (SweepAcycloid/Defs.lean:70-71) — Section 2**

```
def IsReverse {n : Nat} (σ τ : Perm n) : Prop :=
  ∀ i j : Fin n, i.val + j.val + 1 = n → τ.toFun i = σ.toFun j
```

**SweepAcycloid.IsShattered (SweepAcycloid/Defs.lean:106-108) — Section 2**

```
def IsShattered {n k : Nat} (S : Perm n → Prop) (P : Fin k → Pair n) : Prop :=
  ∀ pattern : Fin k → Bool, ∃ σ : Perm n, S σ ∧
    ∀ r : Fin k, (signVec σ (P r) = Sign.pos) ↔ pattern r = true
```

**SweepAcycloid.IsSimpleAcycloid (SweepAcycloid/Draft2Upgrades.lean:183-186) — Section 2**

```
def IsSimpleAcycloid {E : Type} (T : (E → Sign) → Prop) : Prop :=
  IsAcycloid T ∧
  (∀ X : E → Sign, T X → IsFullSupport X) ∧
  (∀ e f : E, ParallelIn T e f → e = f)
```

**SweepAcycloid.IsSimpleAllowableGraph (SweepAcycloid/Draft2Upgrades.lean:232-235) — Section 2**

```
def IsSimpleAllowableGraph {n : Nat} (S : Perm n → Prop) : Prop :=
  (∃ σ : Perm n, S σ) ∧
  (∀ σ : Perm n, S σ → ∃ τ : Perm n, IsReverse σ τ ∧ S τ) ∧
  (∀ σ τ : Perm n, S σ → S τ → ∃ used : Pair n → Prop, SimpleAllowableSeq S used σ τ)
```

**SweepAcycloid.IsSweep (SweepAcycloid/Defs.lean:170-171) — Section 2**

```
def IsSweep {n : Nat} (S : Perm n → Prop) (blk : Fin n → Nat) : Prop :=
  ∀ σ : Perm n, S σ → ∃ τ : Perm n, IsPartitionComp blk σ τ ∧ S τ
```

**SweepAcycloid.IsSweepAcycloid (SweepAcycloid/Draft2Upgrades.lean:206-213) — Section 2**

```
def IsSweepAcycloid {n : Nat} (T : (Pair n → Sign) → Prop) : Prop :=
  IsAcycloid T ∧
  (∀ X : Pair n → Sign, T X →
    ∀ (i j k : Fin n) (hij : i.val < j.val) (hjk : j.val < k.val),
      OrthTriplePPM (X ⟨i, j, hij⟩) (X ⟨j, k, hjk⟩) (X ⟨i, k, Nat.lt_trans hij hjk⟩))
        ↪ ∧
  (∀ (a b c : Fin n) (hab : a ≠ b) (hbc : b ≠ c) (hac : a ≠ c),
    ParallelIn T (unorderedPair a b hab) (unorderedPair b c hbc) →
    ParallelIn T (unorderedPair a b hab) (unorderedPair a c hac))
```

**SweepAcycloid.IsSweepOrder (SweepAcycloid/Defs.lean:242-244) — Section 2**

```
def IsSweepOrder {R : Type} (ord : LinearOrderedCommRing R) {n d : Nat}
    (P : Fin n → Fin d → R) (u : Fin d → R) (σ : Perm n) : Prop :=
  ∀ i j : Fin n, σ.inv i < σ.inv j ↔ ord.lt (dotProd ord u (P i)) (dotProd ord u (P
    ↪ j))
```

**SweepAcycloid.IsTamuraC0Tope (SweepAcycloid/Defs.lean:190-193) — Section 2**

```
def IsTamuraC0Tope (v : Fin 5 → Sign) : Prop :=
  IsFullSupport v ∧
  ∀ c : Fin 5, ¬ Conforms (tamuraC0Circuit c) v ∧
    ¬ Conforms (fun i => (tamuraC0Circuit c i).negate) v
```

**SweepAcycloid.IsTope (SweepAcycloid/Defs.lean:131-132) — Section 2**

```
def IsTope {E : Type} (L : (E → Sign) → Prop) (T : E → Sign) : Prop :=
  L T ∧ ∀ U : E → Sign, L U → Conforms T U → U = T
```

**SweepAcycloid.LinearOrderedCommRing (SweepAcycloid/Defs.lean:206-226) — Section 2**

```
structure LinearOrderedCommRing (R : Type) where
  zero : R
  one : R
  add : R → R → R
  mul : R → R → R
  neg : R → R
  lt : R → R → Prop
  add_assoc : ∀ a b c : R, add (add a b) c = add a (add b c)
  zero_add : ∀ a : R, add zero a = a
  add_comm : ∀ a b : R, add a b = add b a
  add_left_neg : ∀ a : R, add (neg a) a = zero
  mul_assoc : ∀ a b c : R, mul (mul a b) c = mul a (mul b c)
  one_mul : ∀ a : R, mul one a = a
  mul_comm : ∀ a b : R, mul a b = mul b a
  left_distrib : ∀ a b c : R, mul a (add b c) = add (mul a b) (mul a c)
  lt_irrefl : ∀ a : R, ¬ lt a a
  lt_trans : ∀ a b c : R, lt a b → lt b c → lt a c
  lt_trichotomy : ∀ a b : R, lt a b ∨ a = b ∨ lt b a
  zero_lt_one : lt zero one
  add_lt_add_left : ∀ a b c : R, lt a b → lt (add c a) (add c b)
  mul_pos : ∀ a b : R, lt zero a → lt zero b → lt zero (mul a b)
```

**SweepAcycloid.OrthTriplePPM (SweepAcycloid/Draft2Upgrades.lean:189-192) — Section 2**

```
def OrthTriplePPM (a b c : Sign) : Prop :=
  (a = Sign.zero ∧ b = Sign.zero ∧ c = Sign.zero) ∨
  ((a = Sign.pos ∨ b = Sign.pos ∨ c = Sign.neg) ∧
   (a = Sign.neg ∨ b = Sign.neg ∨ c = Sign.pos))
```

**SweepAcycloid.Pair (SweepAcycloid/Defs.lean:29-32) — Section 2**

```
structure Pair (n : Nat) where
  fst : Fin n
  snd : Fin n
  lt : fst.val < snd.val
```

**SweepAcycloid.ParallelIn (SweepAcycloid/Draft2Upgrades.lean:168-169) — Section 2**

```
def ParallelIn {E : Type} (T : (E → Sign) → Prop) (e f : E) : Prop :=
  (∀ X : E → Sign, T X → X e = X f) ∨ (∀ X : E → Sign, T X → X e = (X f).negate)
```

**SweepAcycloid.Perm (SweepAcycloid/Defs.lean:22-26) — Section 2**

```
structure Perm (n : Nat) where
  toFun : Fin n → Fin n
  inv : Fin n → Fin n
  left_inv : ∀ i : Fin n, inv (toFun i) = i
  right_inv : ∀ i : Fin n, toFun (inv i) = i
```

**SweepAcycloid.Perm.comp (SweepAcycloid/PermBase.lean:88-92) — Section 2**

```
def Perm.comp {n : Nat} (σ τ : Perm n) : Perm n where
  toFun := fun i => σ.toFun (τ.toFun i)
  inv := fun i => τ.inv (σ.inv i)
  left_inv := fun i => by simp [σ.left_inv, τ.left_inv]
  right_inv := fun i => by simp [τ.right_inv, σ.right_inv]
```

**SweepAcycloid.RefinesPartition (SweepAcycloid/Defs.lean:161-162) — Section 2**

```
def RefinesPartition {n : Nat} (σ : Perm n) (blk : Fin n → Nat) : Prop :=
  ∀ i j : Fin n, blk i < blk j → σ.inv i < σ.inv j
```

**SweepAcycloid.RestrictOM (SweepAcycloid/Defs.lean:135-136) — Section 2**

```
def RestrictOM {E F : Type} (L : (E → Sign) → Prop) (ι : F → E) (Y : F → Sign) : Prop
  ↪ :=
  ∃ X : E → Sign, L X ∧ ∀ f : F, Y f = X (ι f)
```

**SweepAcycloid.Sign (SweepAcycloid/Defs.lean:42-46) — Section 2**

```
inductive Sign where
  | neg
  | zero
  | pos
  deriving DecidableEq, Repr
```

**SweepAcycloid.Sign.comp (SweepAcycloid/Defs.lean:55-57) — Section 2**

```
def Sign.comp : Sign → Sign → Sign
  | Sign.zero, b => b
  | a, _ => a
```

**SweepAcycloid.Sign.negate (SweepAcycloid/Defs.lean:49-52) — Section 2**

```
def Sign.negate : Sign → Sign
  | Sign.neg => Sign.pos
  | Sign.zero => Sign.zero
  | Sign.pos => Sign.neg
```

**SweepAcycloid.SimpleAllowableSeq (SweepAcycloid/Draft2Upgrades.lean:221-227) — Section 2**

```
inductive SimpleAllowableSeq {n : Nat} (S : Perm n → Prop) :
    (Pair n → Prop) → Perm n → Perm n → Prop where
  | nil {σ : Perm n} : S σ → SimpleAllowableSeq S (fun _ => False) σ σ
  | cons {used : Pair n → Prop} {σ μ τ : Perm n} {e : Pair n} :
      S σ → AdjSwapFlips σ μ e → ¬ used e →
      SimpleAllowableSeq S used μ τ →
      SimpleAllowableSeq S (fun f => f = e ∨ used f) σ τ
```

**SweepAcycloid.TopeSet (SweepAcycloid/Defs.lean:149-150) — Section 2**

```
def TopeSet {n : Nat} (S : Perm n → Prop) (T : Pair n → Sign) : Prop :=
  ∃ σ : Perm n, S σ ∧ T = signVec σ
```

**SweepAcycloid.WalkIn (SweepAcycloid/Defs.lean:83-86) — Section 2**

```
inductive WalkIn {n : Nat} (S : Perm n → Prop) : Nat → Perm n → Perm n → Prop where
  | nil {σ : Perm n} : S σ → WalkIn S 0 σ σ
  | cons {L : Nat} {σ μ τ : Perm n} :
      S σ → IsAdjSwap σ μ → WalkIn S L μ τ → WalkIn S (L + 1) σ τ
```

**SweepAcycloid.appendPair (SweepAcycloid/Draft2Upgrades.lean:590-591) — Section 2**

```
def appendPair {n a : Nat} (A : Fin a → Pair n) (e : Pair n) : Fin (a + 1) → Pair n :=
  fun i => if h : i.val < a then A ⟨i.val, h⟩ else e
```

**SweepAcycloid.compSize (SweepAcycloid/Draft2Upgrades.lean:572-573) — Section 2**

```
def compSize {n k : Nat} (P : Fin k → Pair n) (x : Fin n) : Nat :=
  sumFin n (fun v => if graphReachSteps P n x v = true then 1 else 0)
```

**SweepAcycloid.daSilvaBlock (SweepAcycloid/Defs.lean:181-182) — Section 2**

```
def daSilvaBlock {n : Nat} (i : Fin n) : Nat :=
  if i.val = 0 then 0 else if i.val = 1 then 2 else 1
```

**SweepAcycloid.dotProd (SweepAcycloid/Defs.lean:236-238) — Section 2**

```
def dotProd {R : Type} (ord : LinearOrderedCommRing R) {d : Nat}
    (u v : Fin d → R) : R :=
  sumFinR ord d (fun r => ord.mul (u r) (v r))
```

**SweepAcycloid.existsFinBool (SweepAcycloid/Draft2Upgrades.lean:553-557) — Section 2**

```
def existsFinBool : (k : Nat) → (Fin k → Bool) → Bool
  | 0, _ => false
  | k + 1, f =>
      f ⟨k, Nat.lt_succ_self k⟩ ||
        existsFinBool k (fun i => f ⟨i.val, Nat.lt_trans i.isLt (Nat.lt_succ_self k)⟩)
```

**SweepAcycloid.extendAt (SweepAcycloid/Defs.lean:144-146) — Section 2**

```
def extendAt {E : Type} [DecidableEq E] (e : E) (s : Sign)
    (Y : {f : E // f ≠ e} → Sign) (f : E) : Sign :=
  if h : f = e then s else Y ⟨f, h⟩
```

**SweepAcycloid.fact (SweepAcycloid/Draft2Upgrades.lean:47-49) — Section 2**

```
def fact : Nat → Nat
  | 0 => 1
  | k + 1 => (k + 1) * fact k
```

**SweepAcycloid.graphAdjBool (SweepAcycloid/Draft2Upgrades.lean:560-562) — Section 2**

```
def graphAdjBool {n k : Nat} (P : Fin k → Pair n) (u v : Fin n) : Bool :=
  existsFinBool k (fun r =>
    ((P r).fst == u && (P r).snd == v) || ((P r).fst == v && (P r).snd == u))
```

**SweepAcycloid.graphReachSteps (SweepAcycloid/Draft2Upgrades.lean:565-569) — Section 2**

```
def graphReachSteps {n k : Nat} (P : Fin k → Pair n) : Nat → Fin n → Fin n → Bool
  | 0, u, v => u == v
  | s + 1, u, v =>
      graphReachSteps P s u v ||
        existsFinBool n (fun w => graphReachSteps P s u w && graphAdjBool P w v)
```

**SweepAcycloid.invDist (SweepAcycloid/Defs.lean:97-102) — Section 2**

```
def invDist {n : Nat} (σ τ : Perm n) : Nat :=
  sumFin n (fun i =>
    sumFin n (fun j =>
      if h : i.val < j.val then
        if signVec σ ⟨i, j, h⟩ ≠ signVec τ ⟨i, j, h⟩ then 1 else 0
      else 0))
```

**SweepAcycloid.parallelOM2 (SweepAcycloid/Defs.lean:153-156) — Section 2**

```
def parallelOM2 (X : Fin 2 → Sign) : Prop :=
  (X ⟨0, Nat.zero_lt_succ 1⟩ = Sign.zero ∧ X ⟨1, Nat.lt_succ_self 1⟩ = Sign.zero) ∨
  (X ⟨0, Nat.zero_lt_succ 1⟩ = Sign.pos ∧ X ⟨1, Nat.lt_succ_self 1⟩ = Sign.pos) ∨
  (X ⟨0, Nat.zero_lt_succ 1⟩ = Sign.neg ∧ X ⟨1, Nat.lt_succ_self 1⟩ = Sign.neg)
```

**SweepAcycloid.reorientAt2 (SweepAcycloid/Defs.lean:196-197) — Section 2**

```
def reorientAt2 (v : Fin 5 → Sign) (i : Fin 5) : Sign :=
  if i.val = 2 then (v i).negate else v i
```

**SweepAcycloid.signVec (SweepAcycloid/Defs.lean:60-61) — Section 2**

```
def signVec {n : Nat} (σ : Perm n) (e : Pair n) : Sign :=
  if σ.inv e.fst < σ.inv e.snd then Sign.pos else Sign.neg
```

**SweepAcycloid.starPair6 (SweepAcycloid/Defs.lean:200-201) — Section 2**

```
def starPair6 (i : Fin 5) : Pair 6 :=
  ⟨⟨i.val, Nat.lt_trans i.isLt (Nat.lt_succ_self 5)⟩, ⟨5, Nat.lt_succ_self 5⟩, i.isLt⟩
```

**SweepAcycloid.sumFin (SweepAcycloid/Defs.lean:89-93) — Section 2**

```
def sumFin : (k : Nat) → (Fin k → Nat) → Nat
  | 0, _ => 0
  | k + 1, f =>
      f ⟨k, Nat.lt_succ_self k⟩ +
        sumFin k (fun i => f ⟨i.val, Nat.lt_trans i.isLt (Nat.lt_succ_self k)⟩)
```

**SweepAcycloid.sumFinR (SweepAcycloid/Defs.lean:229-233) — Section 2**

```
def sumFinR {R : Type} (ord : LinearOrderedCommRing R) : (d : Nat) → (Fin d → R) → R
  | 0, _ => ord.zero
  | d + 1, f =>
      ord.add (f ⟨d, Nat.lt_succ_self d⟩)
        (sumFinR ord d (fun i => f ⟨i.val, Nat.lt_trans i.isLt (Nat.lt_succ_self d)⟩))
```

**SweepAcycloid.tamuraC0Circuit (SweepAcycloid/Defs.lean:186-187) — Section 2**

```
def tamuraC0Circuit (c i : Fin 5) : Sign :=
  if i = c then Sign.zero else if i.val = 2 then Sign.neg else Sign.pos
```

**SweepAcycloid.unorderedPair (SweepAcycloid/Draft2Upgrades.lean:195-199) — Section 2**

```
def unorderedPair {n : Nat} (a b : Fin n) (hne : a ≠ b) : Pair n :=
  if h : a.val < b.val then ⟨a, b, h⟩
  else ⟨b, a, by
    have hv : a.val ≠ b.val := fun heq => hne (Fin.ext heq)
    omega⟩
```

**Selected Concrete Sanity Examples (checked in Lean; examples only, not global identity proofs):** SweepAcycloid.example_signVec_id3 (signVec (Perm.id 3) ⟨⟨0, by decide⟩, ⟨1, by decide⟩, by decide⟩ = Sign.pos); SweepAcycloid.example_invDist_id3 (invDist (Perm.id 3) (Perm.id 3) = 0); SweepAcycloid.example_invDist_rev3 (invDist (Perm.id 3) (revPerm (Perm.id 3)) = 3); SweepAcycloid.example_tamuraC0Circuit_0_2 (tamuraC0Circuit ⟨0, by decide⟩ ⟨2, by decide⟩ = Sign.neg); SweepAcycloid.example_reorientAt2_pos (reorientAt2 (fun _ => Sign.pos) ⟨2, by decide⟩ = Sign.neg); SweepAcycloid.example_daSilvaBlock_1 (daSilvaBlock (⟨1, by decide⟩ : Fin 4) = 2); SweepAcycloid.example_fact_5 (fact 5 = 120).

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

**SweepAcycloid.claim_q721_direct_all_ranks_ge_4 (SweepAcycloid/Draft2Upgrades.lean:536-548)** src:b7ba99267b504b

```
theorem claim_q721_direct_all_ranks_ge_4 (r : Nat) (hr : 4 ≤ r) :
    IsSimpleAllowableGraph (InWindow (r + 2) 3) ∧
    IsSimpleAcycloid (TopeSet (InWindow (r + 2) 3)) ∧
    IsSweepAcycloid (TopeSet (InWindow (r + 2) 3)) ∧
    (∃ P : Fin r → Pair (r + 2), Function.Injective P ∧ IsShattered (InWindow (r + 2)
      ↪ 3) P) ∧
    (∀ P : Fin (r + 1) → Pair (r + 2), Function.Injective P →
      ¬ IsShattered (InWindow (r + 2) 3) P) ∧
    (¬ ∃ L : (Pair (r + 2) → Sign) → Prop,
      IsOrientedMatroid L ∧
      ∀ T : Pair (r + 2) → Sign, IsTope L T ↔ TopeSet (InWindow (r + 2) 3) T)
```

**Assumptions:** The hypotheses printed in the signature; nothing beyond the Box 1 definitions. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

**SweepAcycloid.claim1_reversal_and_bijection (SweepAcycloid/PermWalks.lean:67-196)** src:57be6f095a54e6

```
theorem claim1_reversal_and_bijection (n m w k : Nat)
    (hm : 1 ≤ m) (h2m : 2 * m ≤ n) (hw : w + 2 * m = n + 2) (hk : k + 1 = n) :
    (∀ σ : Perm n, InWindow n m σ → ∃ τ : Perm n, IsReverse σ τ ∧ InWindow n m τ) ∧
    Nonempty (Bijection {σ : Perm n // InWindow n m σ} (Fin w × Perm k))
```

**Assumptions:** The hypotheses printed in the signature; nothing beyond the Box 1 definitions. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

**SweepAcycloid.claim_window_cardinality (SweepAcycloid/Draft2Upgrades.lean:151-162)** src:62df652383aaed

```
theorem claim_window_cardinality (n m w k : Nat)
    (hm : 1 ≤ m) (h2m : 2 * m ≤ n) (hw : w + 2 * m = n + 2) (hk : k + 1 = n) :
    Nonempty (Bijection {σ : Perm n // InWindow n m σ} (Fin (w * fact k))) ∧
    ∀ c : Nat, Nonempty (Bijection {σ : Perm n // InWindow n m σ} (Fin c)) → c = w *
      ↪ fact k
```

**Assumptions:** The hypotheses printed in the signature; nothing beyond the Box 1 definitions. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

**SweepAcycloid.claim2_isometric_simple_sweep_acycloid (SweepAcycloid/PermWalks.lean:1011-1042)** src:be55435cde3c55

```
theorem claim2_isometric_simple_sweep_acycloid (n m : Nat)
    (hn : 4 ≤ n) (hm : 1 ≤ m) (h2m : 2 * m ≤ n) :
    (∀ σ τ : Perm n, InWindow n m σ → InWindow n m τ →
      WalkIn (InWindow n m) (invDist σ τ) σ τ) ∧
    (∀ L : Nat, ∀ σ τ : Perm n, WalkIn (InWindow n m) L σ τ → invDist σ τ ≤ L) ∧
    (∀ e : Pair n, ∃ σ τ : Perm n, InWindow n m σ ∧ InWindow n m τ ∧
      IsAdjSwap σ τ ∧ signVec σ e ≠ signVec τ e ∧
      ∀ f : Pair n, f ≠ e → signVec σ f = signVec τ f) ∧
    (∀ σ : Perm n, ∀ i j k : Fin n, ∀ hij : i.val < j.val, ∀ hjk : j.val < k.val,
      signVec σ ⟨i, j, hij⟩ = signVec σ ⟨j, k, hjk⟩ →
      signVec σ ⟨i, k, Nat.lt_trans hij hjk⟩ = signVec σ ⟨i, j, hij⟩)
```

**Assumptions:** The hypotheses printed in the signature; nothing beyond the Box 1 definitions. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

**SweepAcycloid.claim_simple_allowable_graph_and_sweep_acycloid (SweepAcycloid/Draft2Upgrades.lean:402-530)** src:2c0cae899d3bc7

```
theorem claim_simple_allowable_graph_and_sweep_acycloid (n m : Nat)
    (hn : 4 ≤ n) (hm : 1 ≤ m) (h2m : 2 * m ≤ n) :
    IsSimpleAllowableGraph (InWindow n m) ∧
    IsSimpleAcycloid (TopeSet (InWindow n m)) ∧
    IsSweepAcycloid (TopeSet (InWindow n m))
```

**Assumptions:** The hypotheses printed in the signature; nothing beyond the Box 1 definitions. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

**SweepAcycloid.claim_shattered_characterization (SweepAcycloid/Draft2Upgrades.lean:1347-1671)** src:24e60db6d0e55d

```
theorem claim_shattered_characterization (n m k : Nat)
    (hm : 1 ≤ m) (h2m : 2 * m ≤ n) (x : Fin n) (hx : x.val + 1 = n)
    (P : Fin k → Pair n) :
    IsShattered (InWindow n m) P ↔
      IsForest P ∧ compSize P x + m ≤ n + 1
```

**Assumptions:** The hypotheses printed in the signature; nothing beyond the Box 1 definitions. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

**SweepAcycloid.claim3_vc_dimension_eq (SweepAcycloid/VCDim.lean:921-925)** src:b81f2065465f6d

```
theorem claim3_vc_dimension_eq (n m d : Nat)
    (hn : 4 ≤ n) (hm : 2 ≤ m) (h2m : 2 * m ≤ n) (hd : d + 2 = n) :
    (∃ P : Fin d → Pair n, Function.Injective P ∧ IsShattered (InWindow n m) P) ∧
    (∀ P : Fin (d + 1) → Pair n, Function.Injective P → ¬ IsShattered (InWindow n m)
      ↪ P)
```

**Assumptions:** The hypotheses printed in the signature; nothing beyond the Box 1 definitions. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

**SweepAcycloid.claim_vc_m1 (SweepAcycloid/Draft2Upgrades.lean:1675-1964)** src:e6aa1200e3e030

```
theorem claim_vc_m1 (n d : Nat) (hn : 2 ≤ n) (hd : d + 1 = n) :
    (∃ P : Fin d → Pair n, Function.Injective P ∧ IsShattered (InWindow n 1) P) ∧
    (∀ P : Fin (d + 1) → Pair n, Function.Injective P → ¬ IsShattered (InWindow n 1)
      ↪ P)
```

**Assumptions:** The hypotheses printed in the signature; nothing beyond the Box 1 definitions. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

**SweepAcycloid.claim_shattered_not_matroid (SweepAcycloid/Draft2Upgrades.lean:1970-2320)** src:533979a1241988

```
theorem claim_shattered_not_matroid (n m a b : Nat)
    (hm : 3 ≤ m) (h2m : 2 * m ≤ n) (ha : a + m = n) (hb : b = a + 1) :
    ∃ (A : Fin a → Pair n) (B : Fin b → Pair n),
      Function.Injective A ∧ Function.Injective B ∧
      IsShattered (InWindow n m) A ∧ IsShattered (InWindow n m) B ∧
      ∀ j : Fin b, (∀ i : Fin a, B j ≠ A i) →
        ¬ IsShattered (InWindow n m) (appendPair A (B j))
```

**Assumptions:** The hypotheses printed in the signature; nothing beyond the Box 1 definitions. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

**SweepAcycloid.claim4_restriction_and_repaired_contraction (SweepAcycloid/OrientedMatroid.lean:330-357)** src:6a472496bfc32c

```
theorem claim4_restriction_and_repaired_contraction :
    (∀ (E F : Type) (L : (E → Sign) → Prop) (ι : F → E),
      IsOrientedMatroid L →
      IsOrientedMatroid (RestrictOM L ι) ∧
      ((∀ T : E → Sign, IsTope L T → IsFullSupport T) →
       (∃ T : E → Sign, IsTope L T) →
        ∀ Y : F → Sign, IsTope (RestrictOM L ι) Y ↔
          ∃ T : E → Sign, IsTope L T ∧ ∀ f : F, Y f = T (ι f))) ∧
    (∀ (E : Type) [DecidableEq E] (L : (E → Sign) → Prop) (e : E),
      IsOrientedMatroid L →
      IsOrientedMatroid (ContractOM L e) ∧
      ((∀ T : E → Sign, IsTope L T → IsFullSupport T) →
       (∃ Y0 : {f : E // f ≠ e} → Sign,
          IsTope L (extendAt e Sign.pos Y0) ∧ IsTope L (extendAt e Sign.neg Y0)) →
        ∀ Y : {f : E // f ≠ e} → Sign, IsTope (ContractOM L e) Y ↔
          (IsTope L (extendAt e Sign.pos Y) ∧ IsTope L (extendAt e Sign.neg Y)))) ∧
    (IsOrientedMatroid parallelOM2 ∧
      (∀ T : Fin 2 → Sign, IsTope parallelOM2 T → IsFullSupport T) ∧
      (∃ X : Fin 2 → Sign, parallelOM2 X ∧ X ⟨1, Nat.lt_succ_self 1⟩ ≠ Sign.zero) ∧
      (¬ ∃ Y : {f : Fin 2 // f ≠ ⟨1, Nat.lt_succ_self 1⟩} → Sign,
        IsTope parallelOM2 (extendAt ⟨1, Nat.lt_succ_self 1⟩ Sign.pos Y) ∧
        IsTope parallelOM2 (extendAt ⟨1, Nat.lt_succ_self 1⟩ Sign.neg Y)) ∧
      IsTope (ContractOM parallelOM2 ⟨1, Nat.lt_succ_self 1⟩) (fun _ => Sign.zero))
```

**Assumptions:** The hypotheses printed in the signature; nothing beyond the Box 1 definitions. Applicability: Unconditional (closed statement; no outer premise witness needed).

**SweepAcycloid.claim_exists_tope_fin (SweepAcycloid/Draft2Upgrades.lean:2489-2544)** src:0d0ad469573ca6

```
theorem claim_exists_tope_fin (k : Nat) (L : (Fin k → Sign) → Prop)
    (hL : IsOrientedMatroid L) :
    ∃ T : Fin k → Sign, IsTope L T
```

**Assumptions:** The hypotheses printed in the signature; nothing beyond the Box 1 definitions. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

**SweepAcycloid.claim5_not_oriented_matroid (SweepAcycloid/OrientedMatroid.lean:981-1016)** src:c94f9a29b5cb69

```
theorem claim5_not_oriented_matroid (n m : Nat) (hm : 3 ≤ m) (h2m : 2 * m ≤ n) :
    ¬ ∃ L : (Pair n → Sign) → Prop,
      IsOrientedMatroid L ∧ ∀ T : Pair n → Sign, IsTope L T ↔ TopeSet (InWindow n m) T
```

**Assumptions:** The hypotheses printed in the signature; nothing beyond the Box 1 definitions. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

**SweepAcycloid.claim6_dasilva_potential_not_sweep (SweepAcycloid/DaSilva.lean:22-327)** src:ef578180d06bee

```
theorem claim6_dasilva_potential_not_sweep (n m : Nat) (hm : 3 ≤ m) (h2m : 2 * m ≤ n)
  ↪ :
    IsPotentialSweep (InWindow n m) (@daSilvaBlock n) ∧
    ¬ IsSweep (InWindow n m) (@daSilvaBlock n)
```

**Assumptions:** The hypotheses printed in the signature; nothing beyond the Box 1 definitions. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

**SweepAcycloid.claim_knauer_marc_minor (SweepAcycloid/Draft2Upgrades.lean:2337-2483)** src:f75cd7ecb530b1

```
theorem claim_knauer_marc_minor (n m N kp kn : Nat)
    (hm : 3 ≤ m) (h2m : 2 * m ≤ n) (hkp : kp + 3 = m) (hkn : kn + 2 = m)
    (hN : kp + kn + N + 1 = n) :
    4 ≤ N ∧
    ∀ v : Fin N → Sign,
      (∃ σ : Perm n, InWindow n m σ ∧
        (∀ j : Fin kp, signVec σ ⟨⟨j.val, by have := j.isLt; omega⟩, ⟨kp + kn + N, by
          ↪ omega⟩, by have := j.isLt; dsimp; omega⟩ = Sign.pos) ∧
        (∀ j : Fin kn, signVec σ ⟨⟨kp + j.val, by have := j.isLt; omega⟩, ⟨kp + kn +
          ↪ N, by omega⟩, by have := j.isLt; dsimp; omega⟩ = Sign.neg) ∧
        (∀ i : Fin N, v i = signVec σ ⟨⟨kp + kn + i.val, by have := i.isLt; omega⟩,
          ↪ ⟨kp + kn + N, by omega⟩, by have := i.isLt; dsimp; omega⟩)) ↔
      IsKnauerMarcQmm N v
```

**Assumptions:** The hypotheses printed in the signature; nothing beyond the Box 1 definitions. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

**SweepAcycloid.claim7_tamura_c0_band63 (SweepAcycloid/TamuraC0.lean:25-212)** src:3cd8fb7e17c405

```
theorem claim7_tamura_c0_band63 :
    ∀ v : Fin 5 → Sign,
      ((∃ σ : Perm 6, InWindow 6 3 σ ∧ ∀ i : Fin 5, v i = signVec σ (starPair6 i)) ↔
        (IsFullSupport v ∧
          (sumFin 5 (fun i => if v i = Sign.pos then 1 else 0) = 2 ∨
           sumFin 5 (fun i => if v i = Sign.pos then 1 else 0) = 3))) ∧
      ((∃ σ : Perm 6, InWindow 6 3 σ ∧ ∀ i : Fin 5, v i = signVec σ (starPair6 i)) ↔
        IsTamuraC0Tope (reorientAt2 v))
```

**Assumptions:** The hypotheses printed in the signature; nothing beyond the Box 1 definitions. Applicability: Unconditional (closed statement; no outer premise witness needed).

**SweepAcycloid.claim8_realizable_m1_m2 (SweepAcycloid/Realizability.lean:893-900)** src:562a906da3992b

```
theorem claim8_realizable_m1_m2 (n d1 d2 : Nat)
    (hn : 4 ≤ n) (hd1 : d1 + 1 = n) (hd2 : d2 + 2 = n)
    (R : Type) (ord : LinearOrderedCommRing R) :
    (∃ P : Fin n → Fin d1 → R,
      ∀ σ : Perm n, InWindow n 1 σ ↔ ∃ u : Fin d1 → R, IsSweepOrder ord P u σ) ∧
    (∃ P : Fin n → Fin d2 → R,
      ∀ σ : Perm n, InWindow n 2 σ ↔ ∃ u : Fin d2 → R, IsSweepOrder ord P u σ)
```

**Assumptions:** The hypotheses printed in the signature; nothing beyond the Box 1 definitions. Applicability: Instantiable — closed instantiation witness PENDING gate-box replay.

**Table 2.** Lean 4 Type-Trap Audit Checklist — Source-Level Approximation (Gate Replay Pending).[a]

| Declaration | Hazard | Operand | Category | Audited Disposition & Guard |
|---|---|---|---|---|
| `claim1_reversal_and_bijection` | `nat_div` | `/ /` | intended_convention | The token pair is the subtype separator // of the Lean notation {x : T // p x}; no natural-number division occurs. |
| `claim1_reversal_and_bijection` | `nat_div` | `/ InWindow` | intended_convention | The token pair is the subtype separator // of the Lean notation {x : T // p x}; no natural-number division occurs. |
| `claim4_restriction_and_repaired_contraction` | `nat_div` | `/ /` | intended_convention | The token pair is the subtype separator // of the Lean notation {x : T // p x}; no natural-number division occurs. |
| `claim4_restriction_and_repaired_contraction` | `nat_div` | `/ f` | intended_convention | The token pair is the subtype separator // of the Lean notation {x : T // p x}; no natural-number division occurs. |
| `ContractOM` | `nat_div` | `/ /` | intended_convention | The token pair is the subtype separator // of the Lean notation {x : T // p x}; no natural-number division occurs. |
| `ContractOM` | `nat_div` | `/ f` | intended_convention | The token pair is the subtype separator // of the Lean notation {x : T // p x}; no natural-number division occurs. |
| `extendAt` | `nat_div` | `/ /` | intended_convention | The token pair is the subtype separator // of the Lean notation {x : T // p x}; no natural-number division occurs. |
| `extendAt` | `nat_div` | `/ f` | intended_convention | The token pair is the subtype separator // of the Lean notation {x : T // p x}; no natural-number division occurs. |
| `claim_window_cardinality` | `nat_div` | `/ /` | intended_convention | The token pair is the subtype separator // of the Lean notation {x : T // p x}; no natural-number division occurs. |
| `claim_window_cardinality` | `nat_div` | `/ InWindow` | intended_convention | The token pair is the subtype separator // of the Lean notation {x : T // p x}; no natural-number division occurs. |

[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 3.** 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: for every rank r >= 4, W(r+2, 3) is a simple allowable graph of permutations, T_{W(r+2, 3)} is a simple sweep acycloid of VC dimension r, and no oriented matroid has T_{W(r+2, 3)} as its tope set | Lean Kernel | `SweepAcycloid.claim_q721_direct_all_ranks_ge_4` | Axioms: Classical.choice, Quot.sound, propext; fidelity: exact | Pure-core Lean 4 kernel and the Box 1 definitions only (tope reading of Padrol-Philippe Def. 7.13(i); extension to all covectors cited from Sec. 7.4 and Lem. 7. 17); the hypotheses are those of the Box 2 signature. |
| Prop. 6: for m >= 1 and 2m <= n, W(n, m) is closed under word reversal and bijects explicitly with Fin(n-2m+2) x S_{n-1} | Lean Kernel | `SweepAcycloid.claim1_reversal_and_bijection` | Axioms: Classical.choice, Quot.sound, propext; fidelity: exact | Pure-core Lean 4 kernel and the Box 1 definitions only; the hypotheses are those of the Box 2 signature. |
| Cor. 7: for m >= 1 and 2m <= n, W(n, m) bijects explicitly with Fin((n-2m+2)(n-1)!) and every bijection with Fin(c) satisfies c = (n-2m+2)(n-1)! | Lean Kernel | `SweepAcycloid.claim_window_cardinality` | Axioms: Classical.choice, Quot.sound, propext; fidelity: exact | Pure-core Lean 4 kernel and the Box 1 definitions only; the hypotheses are those of the Box 2 signature. |
| Thm. 8: for n >= 4, m >= 1, 2m <= n, W(n, m) has geodesic adjacent-swap walks of length d(sigma, tau), no shorter walks, an edge isolating each pair, and transitive sign vectors | Lean Kernel | `SweepAcycloid.claim2_isometric_simple_sweep_acycloid` | Axioms: Classical.choice, Quot.sound, propext; fidelity: exact | Pure-core Lean 4 kernel and the Box 1 definitions only; the hypotheses are those of the Box 2 signature. |

| Paper Claim | Proof Tier | Lean Decl / Artifact | Actual Axioms & Fidelity | Assumptions & What You Must Trust |
|---|---|---|---|---|
| Thm. 9: for n >= 4, m >= 1, 2m <= n, W(n, m) is a simple allowable graph of permutations and T_{W(n, m)} is a simple acycloid and a sweep acycloid | Lean Kernel | `SweepAcycloid.claim_simple_allowable_graph_and_sweep_acycloid` | Axioms: Classical.choice, Quot.sound, propext; fidelity: exact | Pure-core Lean 4 kernel and the Box 1 definitions only (tope reading of Padrol-Philippe Def. 7.13(i); extension to all covectors cited from Sec. 7.4 and Lem. 7. 17); the hypotheses are those of the Box 2 signature. |
| Thm. 10: for m >= 1 and 2m <= n, a family P of pairs is shattered by T_{W(n, m)} if and only if P is a forest with \|C_P(n)\| + m <= n + 1 | Lean Kernel | `SweepAcycloid.claim_shattered_characterization` | Axioms: Classical.choice, Quot.sound, propext; fidelity: exact | Pure-core Lean 4 kernel and the Box 1 definitions only; the hypotheses are those of the Box 2 signature. |
| Thm. 11: for n >= 4, m >= 2, 2m <= n, the VC dimension of T_{W(n, m)} is exactly n-2 | Lean Kernel | `SweepAcycloid.claim3_vc_dimension_eq` | Axioms: Classical.choice, Quot.sound, propext; fidelity: exact | Pure-core Lean 4 kernel and the Box 1 definitions only; the hypotheses are those of the Box 2 signature. |
| Prop. 12: for n >= 2, the VC dimension of T_{W(n, 1)} = T_{S_n} is exactly n-1 | Lean Kernel | `SweepAcycloid.claim_vc_m1` | Axioms: Classical.choice, Quot.sound, propext; fidelity: exact | Pure-core Lean 4 kernel and the Box 1 definitions only; the hypotheses are those of the Box 2 signature. |
| Thm. 13: for m >= 3 and 2m <= n, the shattered complex of T_{W(n, m)} fails matroid augmentation (I3) via explicit star A (size n-m) and tree B (size n-m+1) | Lean Kernel | `SweepAcycloid.claim_shattered_not_matroid` | Axioms: Classical.choice, Quot.sound, propext; fidelity: exact | Pure-core Lean 4 kernel and the Box 1 definitions only; the hypotheses are those of the Box 2 signature. |
| Lem. 15: restriction and repaired contraction of sign systems satisfying (V0)-(V3), with a counterexample to the unrepaired rule | Lean Kernel | `SweepAcycloid.claim4_restriction_and_repaired_contraction` | Axioms: Quot.sound, propext; fidelity: exact | Pure-core Lean 4 kernel and the Box 1 definitions only; the hypotheses are those of the Box 2 signature. |
| Lem. 16: every oriented matroid on a finite ground set Fin k possesses at least one tope | Lean Kernel | `SweepAcycloid.claim_exists_tope_fin` | Axioms: Classical.choice, Quot.sound, propext; fidelity: exact | Pure-core Lean 4 kernel and the Box 1 definitions only; the hypotheses are those of the Box 2 signature. |
| Thm. 17: for m >= 3 and 2m <= n, no oriented matroid on E_n has T_{W(n, m)} as its tope set | Lean Kernel | `SweepAcycloid.claim5_not_oriented_matroid` | Axioms: Classical.choice, Quot.sound, propext; fidelity: exact | Pure-core Lean 4 kernel and the Box 1 definitions only; the hypotheses are those of the Box 2 signature. |
| Thm. 18: for m >= 3 and 2m <= n, the ordered partition I^dS is a potential sweep of W(n, m) and not a sweep | Lean Kernel | `SweepAcycloid.claim6_dasilva_potential_not_sweep` | Axioms: Classical.choice, Quot.sound, propext; fidelity: exact | Pure-core Lean 4 kernel and the Box 1 definitions only; the hypotheses are those of the Box 2 signature. |
| Thm. 19: for m >= 3 and 2m <= n, star restriction plus 2m-5 halfspace restrictions of T_{W(n, m)} yield identically the Knauer-Marc excluded minor Q_{n-2m+4}^{−−}(n-2m+4) | Lean Kernel | `SweepAcycloid.claim_knauer_marc_minor` | Axioms: Classical.choice, Quot.sound, propext; fidelity: exact | Pure-core Lean 4 kernel and the Box 1 definitions only; the hypotheses are those of the Box 2 signature. |
| Prop. 21: the star restriction of T_{W(6, 3)} equals the weight-{2,3} band on five elements and Tamura's acycloid C_0 after reorienting coordinate 3 | Lean Kernel | `SweepAcycloid.claim7_tamura_c0_band63` | Axioms: Classical.choice, Quot.sound, propext; fidelity: exact | Pure-core Lean 4 kernel and the Box 1 definitions only; the hypotheses are those of the Box 2 signature. |
| Thm. 22: for n >= 4, W(n, 1) and W(n, 2) are exact sweep-order sets of explicit point configurations over every linearly ordered commutative ring with 0 < 1 | Lean Kernel | `SweepAcycloid.claim8_realizable_m1_m2` | Axioms: Classical.choice, Quot.sound, propext; fidelity: exact | Pure-core Lean 4 kernel and the Box 1 definitions only; the hypotheses are those of the Box 2 signature. |

[a] Allowed pure-kernel axioms: `propext`, `Classical.choice`, `Quot.sound`. Actual axioms printed above are read directly from verified outputs. Provenance nonce: 838022b350ff | Lean mode: committed_receipt | Closure SHA-256: e43754653562 | Git anchor: ded7fe7259b4 | Axioms SHA-256: ce0e8b4f068e | Fidelity SHA-256: 2bd54b57ccfb.

## 6. PRIOR WORK AND OPEN QUESTIONS

**The question and labelled poset strictness.** Question 7.21 appears with identical wording in all three arXiv versions of [1] (as Question 7.20 in v1 and 7.22 in v2) and as Question 3.7.21 of Philippe's thesis [2, Section 3.7.4, pp. 125–127]. It sits in the chain of inclusions of [1, Theorem 7.19]: posets of sweeps of point configurations are strictly contained in posets of sweeps of sweep oriented matroids, which are contained in posets of sweeps of sweep acycloids. Viewed as posets of labelled ordered partitions of $[n]$, the maximal (singleton-block) sweeps of $W(r+2,3)$ recover $W(r+2,3)$; if this labelled poset were that of a sweep oriented matroid, that oriented matroid would have tope set $T_{W(r+2,3)}$, contradicting Theorem 1(d). So the second inclusion of [1, Theorem 7.19] is strict for posets of labelled ordered partitions in every rank $r \geq 4$ (we make no claim up to abstract poset isomorphism).

**Handa, Tamura, and Knauer–Marc.** By Proposition 7.4 of [3] (p. 261), every non-matroidal acycloid has at least five elements, so every sweep acycloid on $n \leq 3$ letters ($|E_3| = 3$) is an oriented matroid. His Proposition 7.7 (p. 262) lists all four non-matroidal acycloids on five elements up to reorientation, including Tamura's $C_0$, whose source is an unpublished private communication (Reference [48], p. 265: *A. Tamura, private communication, 1985*). By Proposition 21, the star restriction of $T_{W(6,3)}$ equals $C_0$ after reorienting coordinate 3, while differing without reorientation (for instance, $124635 \in W(6,3)$ has star vector $++-+-$, to which the fifth original circuit conforms). Handa's Theorem 3.7 (p. 245, citing Fukuda and Handa [6]) characterizes tope graphs of rank-$\leq$ 3 oriented matroids as antipodal planar partial cubes, framing the open rank-3 case. Knauer and Marc [4] (Theorem 1.1, Section 5, p. 21, and Corollary 7.2) show that a finite antipodal partial cube is the tope graph of an oriented matroid if and only if it avoids $Q^-$, and note (Section 8, Problem 1, p. 32) that the four non-matroidal acycloids on five elements arise from $Q_4^{--}(1), \ldots, Q_4^{--}(4)$. Theorem 19 exhibits $Q^{--}_{n-2m+4}(n-2m+4)$ inside $T_{W(n,m)}$ for every $m \geq 3, 2m \leq n$ by star restriction and $2m-5$ halfspace restrictions, whereas the contraction chain of Theorem 17 ends at the constant-weight set $B(2m-2, m-1, m-1)$ on $2m-2$ coordinates (6 vectors on 4 coordinates at $(6,3)$). General oriented-matroid theory is in [7].

**Search record.** Searches covered Google Scholar, arXiv, zbMATH, Semantic Scholar, OEIS, MathOverflow, Mathematics Stack Exchange, GitHub, Zenodo, and HAL. Semantic Scholar listed 13 citing works of [1] (2 being preprint and journal forms of the same paper); keyword searches across the 12 distinct preprint texts returned 0 hits for "Question 7." and "sweep acycloid", only bibliography hits for "allowable graph", and passing mentions of "sweep oriented" and "acycloid". Those searches did not read the citing texts in full and do not rule out an answer phrased in different terminology. The primary full texts of [1], [2], [3] (including scanned pages 261–263 and 265), and [4] (arXiv v3) were read in full. Not read or not searched (marked `[to check]` in the bibliography): the monograph [7]; the paywalled journal full texts of [5] and [6] and the journal pagination of [4]; three open-access papers on antipodal partial cubes in *Discussiones Mathematicae Graph Theory* whose host could not be verified; at least 7 further citing works counted by Google Scholar (20 in all); OpenAlex and CORE; publisher web pages and Beaur's thesis; and conference proceedings or Japanese-language reports on acycloids.

**Open questions.**

1. Is every sweep acycloid of VC dimension 3 the tope set of an oriented matroid? By Handa [3, Theorem 3.7], this asks whether every rank-3 sweep acycloid has a planar tope graph.
2. Is $W(6,3)$ the smallest non-matroidal sweep acycloid, by number of letters ($n = 6$) or number of words (240)? Handa [3, Proposition 7.4] rules out $n \leq 3$, leaving $n = 4$ and $n = 5$ open; we make no minimality claim.

## 7. DATA AVAILABILITY, REPRODUCIBILITY, AND USE OF AI

This draft directory contains the manuscript, `recompute.py` with `recompute.out`, the independent Lean-source reader `certificate.py` with `certificate.out`, the figure scripts under `figures/`, the claim and reference ledgers, the trust-boundary and fidelity manifests, and the pinned Lean 4 library under `lean/`. `certificate.py` shares no code with `recompute.py`: it parses the explicit data of the Lean library directly from the `.lean` sources (Tamura's circuit table, the da Silva block function, the two-element counterexample of Lemma 15(c), the closed sanity equalities of `Examples.lean`, and the declaration lists of `Verify.lean` and `axioms.out`), re-evaluates them with its own code, and checks that every claim of the trust-boundary manifest names a declaration whose recorded axioms are standard. Upstream SAT, DRAT, and C search files from the initial $n = 6$ exploration are not included in this bundle and are not needed to verify any claim in this paper. Independent replay commands are `python3 recompute.py` (with `OUROTRIX_CLAIMS_FAST` unset to include literal brute-force checks), `python3 certificate.py`, `lake build` followed by `lake env lean SweepAcycloid/Verify.lean` inside `lean/`, and `ourotrix paper trust -p . -g`.

AI research agents built and directed by Tyler B. Hughes discovered the family, developed and verified the proofs, constructed and audited the Lean 4 library, conducted the literature searches, and authored this manuscript. No human mathematician has yet refereed the proof.

## A. PROOFS OF SUPPORTING LEMMAS AND COMPANION RESULTS

This appendix collects the proofs deferred from Section 3; each result is also proved in the Lean 4 library (Section 5).

*Proof of Corollary 7.* Induct on $k$ using Proposition 6(b) at $m = 1$ (where $W(k+1,1) = S_{k+1}$) together with the row-major bijection $\mathrm{Fin}(a) \times \mathrm{Fin}(b) \leftrightarrow \mathrm{Fin}(ab)$ to construct an explicit bijection $S_k \leftrightarrow \mathrm{Fin}(k!)$. Compose with Proposition 6(b); uniqueness of $c$ follows from the pigeonhole principle on finite types. □

*Proof of Lemma 15.* (a) The axioms (V0)–(V3) survive precomposition. Fixing a tope $T_0$ of $\mathcal{L}$, for any $X \in \mathcal{L}$ the vector $X \circ T_0 \in \mathcal{L}$ has full support, is therefore a tope, and satisfies $X \preceq X \circ T_0$; thus the maximal elements of $\mathcal{L}|_t$ are precisely its full-support vectors, which are the restrictions of topes. (b) Coordinate $e$ remains 0 under negation, composition, and elimination, so $\mathcal{L}/e$ satisfies (V0)–(V3). If $(Y,+)$ and $(Y,-)$ are topes, elimination at $e$ yields $(Y,0) \in \mathcal{L}$, so $Y \in \mathcal{L}/e$ has full support and is a tope. Conversely, if $Y$ is a tope of $\mathcal{L}/e$, since $Y_0 \in \mathcal{L}/e$ has full support, $Y \preceq Y \circ Y_0$ forces $Y = Y \circ Y_0$ (full support); with $X = (Y,0) \in \mathcal{L}$, both $X \circ (Y_0, \pm) = (Y, \pm)$ lie in $\mathcal{L}$ and have full support, so both are topes. (c) Direct check: contracting the second coordinate yields $\{0\}$. □

*Proof of Lemma 16.* Every covector $X \in \mathcal{L}$ has support size at most $k$, and $0 \in \mathcal{L}$ by (V0). Choose $T \in \mathcal{L}$ maximizing the support size. If $U \in \mathcal{L}$ satisfies $T \preceq U$ and $U_i \neq T_i$ for some $i$, then $T_i = 0$ and $U_i \neq 0$, giving strictly larger support size than $T$, a contradiction. Hence $U = T$ and $T$ is a tope. □

*Proof of Theorem 19.* Since $2m \le n$, $N = n - 2m + 4 \ge 4$ and $k_+ + k_- + N = n - 1$. By the star-restriction characterization proved inside Theorem 17, a full sign vector $Y \in \{+,-\}^{n-1}$ occurs as $(X^{\sigma}_{(i,n)})_{1 \le i \le n-1}$ for some $\sigma \in W(n,m)$ if and only if $m - 1 \le \#\{i : Y_i = +\} \le n - m$. Fixing the first $k_+ = m - 3$ coordinates of $Y$ to $+$ and the next $k_- = m - 2$ coordinates to $-$ leaves the remaining $N$ coordinates $v$ contributing $\#\{i : Y_i = +\} = k_+ + \#\{i : v_i = +\}$. The inequalities $m - 1 \le (m-3) + \#\{i : v_i = +\} \le n - m$ simplify identically to $2 \le \#\{i : v_i = +\} \le n - 2m + 3 = N - 1$, which defines $Q_N^{--}(N)$. □

*Proof of Proposition 21.* Equivalence of (i) and (ii) holds because the letters before 6 in $\sigma \in W(6,3)$ can be any 2- or 3-element subset of $[5]$. Negating coordinate 3 turns the five circuits of $C_0$ into the five vectors with one 0 and four $+$; none of these five vectors or their negatives conforms to a full vector $v \in \{+,-\}^5$ if and only if $v$ has at most three of each sign, that is, two or three plus signs. □

*Proof of Theorem 22.* (a) Set $P(1) = 0$ and $P(k+1) = e_k$ for $1 \le k \le n-1$. For $\sigma \in S_n$, the direction $u_k = \sigma^{-1}(k+1) - \sigma^{-1}(1)$ yields $\langle u, P(i) \rangle = \sigma^{-1}(i) - \sigma^{-1}(1)$. (b) Set $P(1) = 0$, $P(k+1) = (n-1)e_k$ for $1 \le k \le n-2$, and $P(n) = (1, \dots, 1) \in R^{n-2}$, so $(n-1)\langle u, P(n) \rangle = \sum_{i=1}^{n-1} \langle u, P(i) \rangle$ for every $u \in R^{n-2}$. Thus $\langle u, P(n) \rangle$ can never be the strict minimum or strict maximum, so every sweep order lies in $W(n,2)$. Conversely, given $\sigma \in W(n,2)$ with $p = \sigma^{-1}(n)$ and $q = n - p$ (so $p - 1 \ge 1$ and $q \ge 1$), set $M = n^2$ and for each $i < n$ at position $s = \sigma^{-1}(i)$ set $v_i = -Mq + s$ if $s < p$ and $v_i = M(p-1) + s$ if $s > p$. The large terms cancel identically: $(p-1)(-Mq) + qM(p-1) = 0$, leaving $0 < \sum_{i=1}^{n-1} v_i \le (n-1)n < M$. With $u_k = v_{k+1} - v_1$ $(1 \le k \le n-2)$, we have $\langle u, P(i) \rangle = (n-1)v_i - (n-1)v_1$ for $i < n$ and $\langle u, P(n) \rangle = \sum_{j=1}^{n-1} v_j - (n-1)v_1$. Comparing $\sum_{j=1}^{n-1} v_j$ directly with $(n-1)v_i$ avoids division in $R$: for $s < p$ we have $v_i \le -M + n - 1 < 0$ so $(n-1)v_i < 0 < \sum_j v_j$, whereas for $s > p$ we have $v_i \ge M + 1 > M$ so $(n-1)v_i > M > \sum_j v_j$, and within each group $(n-1)v_i$ increases strictly with $s$. Thus $u$ sweeps $P$ in order $\sigma$. □

## REFERENCES


(1) A. Padrol and E. Philippe, Sweeps, polytopes, oriented matroids, and allowable graphs of permutations, *Combinatorica* **44** (2024), 63–123. doi:10.1007/s00493-023-00062-3; arXiv:2102.06134.

(2) E. Philippe, *Geometric realizations using regular subdivisions: construction of many polytopes, sweep polytopes, s-permutahedra*, PhD thesis, Universitat de Barcelona and Sorbonne Université (2024). hdl:2445/217199.

(3) K. Handa, Topes of oriented matroids and related structures, *Publ. Res. Inst. Math. Sci.* **29** (1993), 235–266. doi:10.2977/prims/1195167272; Zbl 0777.05039.

(4) K. Knauer and T. Marc, On tope graphs of complexes of oriented matroids, *Discrete Comput. Geom.* **63** (2020), 377–417. doi:10.1007/s00454-019-00111-z; arXiv:1701.05525.

(5) I. P. F. da Silva, Axioms for maximal vectors of an oriented matroid: a combinatorial characterization of the regions determined by an arrangement of pseudohyperplanes, *European J. Combin.* **16** (1995), 125–145. doi:10.1016/0195-6698(95)90052-7; Zbl 0824.52018. [to check]

(6) K. Fukuda and K. Handa, Antipodal graphs and oriented matroids, *Discrete Math.* **111** (1993), 245–256. Zbl 0782.05069; doi:10.1016/0012-365X(93)90159-Q. [to check]

(7) A. Björner, M. Las Vergnas, B. Sturmfels, N. White and G. M. Ziegler, *Oriented Matroids*, 2nd ed., Cambridge University Press (1999). doi:10.1017/CBO9780511586507. [to check]