Cut tires form a tree (under depth nesting)

The claim

Let f be a face of H_d in the inherited embedding. By the BFS level-set property (Lemma below), the open interior of f contains only edges of G'_i of depth $< d$ or only edges of depth $> d$. We call f a *high-side* face if its interior contains only depth-$> d$ edges, and a *low-side* face otherwise. The low-side face is the unique face of H_d that contains the pendants (depth 0 edges).

Proposition (Cut tires form a forest, refined)**.** *For each side i of a 6-edge cut of G', the high-side cut tires of G'_i, parameterised by pairs (d, f) with $d \geq 1$ and f a* high-side *face of H_d, form a* forest *under the parent–child relation*

$$\text{parent}\big(T_{d+1}^{(f')}\big) := T_d^{(f)}$$

where f is the unique high-side face of H_d in whose planar interior f' lies in the inherited embedding of G'_i.

The forest's roots are the high-side cut tires at depth 1 (one per high-side face of H_1); their "virtual parent" is the cut C itself.

Remark. *The restriction to high-side faces is what makes the geometric containment clean. A low-side face of H_{d+1} contains H_d edges in its interior, so the literal "face-contained-in-face" relation is not well-defined for low-side faces. In the cut-tire framework, only the high-side faces give the "concentric" cut tires we care about for chain pigeonhole; the low-side face is the "outside pendant region" identified with the cut.*

Proof. We prove the proposition in two stages.

Stage 1: the BFS level-set lemma.

Lemma (BFS depth differs by at most 1 between adjacent edges)**.** *Let $e_1, e_2 \in E(G'_i)$ share a vertex (so they are adjacent in the line graph). Then $|\text{depth}(e_1) - \text{depth}(e_2)| \leq 1$.*

Proof. By definition of BFS depth, $\text{depth}(e) =$ minimum line-graph distance from e to any pendant. If e_1, e_2 are line-graph adjacent, then a shortest line-graph path from a pendant to e_2 can be extended by the one step from e_2 to e_1, yielding a path of length $\text{depth}(e_2) + 1$ from a pendant to e_1. So $\text{depth}(e_1) \leq \text{depth}(e_2) + 1$, and symmetrically. $\square$

Lemma (Level-set property of H_d)**.** *For each $d \geq 1$, every face of H_d in the inherited planar embedding satisfies one of the following:*

- *Every edge of G'_i strictly inside the face has depth $< d$ (a "low-side" face), or*
- *Every edge of G'_i strictly inside the face has depth $> d$ (a "high-side" face).*

Proof. Let f be a face of H_d. Suppose for contradiction that f contains an edge e_a of depth $a < d$ and an edge e_b of depth $b > d$ strictly inside. Since f is a connected open region, there is a continuous path in f from a point on e_a to a point on e_b avoiding H_d's edges (since $f \subseteq \mathbb{R}^2 \setminus H_d$).

Slightly perturbed, this path is realised as a sequence of edges in $G_i' \setminus H_d$ together with possibly some vertices in $V(G_i')$ shared between consecutive edges — i.e. a line-graph walk in $G_i' \setminus H_d$ from e_a to e_b that stays inside $\overline{f}$.

By Lemma , consecutive edges along this line-graph walk differ in depth by at most 1. Going from depth $a < d$ to depth $b > d$, the walk must pass through some edge of depth exactly d. But that edge is in H_d, contradicting that the walk lies in $G_i' \setminus H_d$.

Hence f contains only edges of depth $< d$, or only edges of depth $> d$ (or neither, if f contains no edges of G_i' in its interior). □

Stage 2: high-side faces of H_{d+1} embed in high-side faces of H_d.

Let f' be a high-side face of H_{d+1}. By definition, every edge of G_i' in the open interior of f' has depth $> d+1$.

In particular, no edge of depth d lies in the open interior of f': every depth-d edge of G_i' has depth $d \not> d+1$, so depth-d edges are not in f''s open interior.

Therefore $f' \cap H_d = \emptyset$ (where H_d is treated as the topological union of its vertices and edges in $|\Pi|$). Equivalently, $f' \subseteq \mathbb{R}^2 \setminus H_d$.

Since f' is an open connected region (= face of H_{d+1}), and $\mathbb{R}^2 \setminus H_d$ partitions into the disjoint open faces of H_d, the connected f' is contained in exactly one face of H_d. Call this face f. Then $f' \subseteq f$.

Furthermore, f is high-side: it contains f', which contains depth-$\geq d+2$ edges, which are $> d$ depth. So f is in the "high-side" classification of Lemma .

Hence parent$(T_{d+1}^{(f')}) := T_d^{(f)}$ is well-defined and unique among high-side cut tires.

Conclusion: forest structure.

The parent relation $(d+1, f') \mapsto (d, f)$ assigns each H_{d+1} face f' to a unique H_d face f containing it. Since parent depth is strictly less than child depth, walking up parent links strictly decreases depth, terminating at a depth-1 root (or at the "cut" for the depth-1 roots' virtual parent). No cycles can form. Hence the parent relation defines a forest. □ □

Remark on the proof. Stage 2 is now fully rigorous, thanks to the refinement to *high-side* faces of H_{d+1}. The key step is: a high-side face of H_{d+1} contains, by definition, only depth-$\geq d+2$ edges in its interior. Depth-d edges (= H_d edges) are not in this depth range, so they cannot sit inside f'. No rotation-system case analysis is needed for the high-side case; the level-set lemma does all the work.

The original (unrestricted) proposition was problematic for the *low-side* face of H_{d+1}, which contains the pendants (depth 0) plus all edges of depth $\leq d$ in G_i''s "outside" region. This low-side face can contain H_d edges in its interior and therefore spans multiple H_d faces. By restricting to high-side faces, this difficulty is avoided.

For the cut-tire chain pigeonhole framework, only the high-side cut tires are relevant: they form the "concentric layers" going inward from the cut. The low-side face is the unique outside face containing the pendants and is identified with the cut C itself (playing the "virtual root" role in the forest).

Why this matters for the chain half

Chain pigeonhole asks whether the per-tire S_3-orbit structure composes coherently through the chain. With a tree structure on the cut tires, this becomes a **tree dynamic-programming problem**, not a general graph compatibility problem:

- Process tires from leaves to root.
- At each leaf: $\pi(T_{\text{leaf}})$ has known structure (e.g. S_3-orbits) from the per-tire half.
- Internal node $T_d^{(f)}$ combines:
  - Its own internal $\pi(T)$ structure.
  - Compatibility with each child $T_{d+1}^{(f')}$ via the bijection $\{$in spokes of $T_d^{(f)}\} \leftrightarrow \{$face boundary edges of $T_{d+1}^{(f')}$
- Root: $T_1^{(\cdot)}$ projects its out-spoke colours to $\sigma_i \in \mathcal{R}_i$.

Tree DP is well-understood: $|\mathcal{R}_i|$ can be computed exactly in linear time in the tree size (with size-$|\pi|$ tables at each node). Whether the resulting $\mathcal{R}_0$ and $\mathcal{R}_1$ intersect is a finite check at the cut.

The tree structure is also a **strong topological constraint** on the chain pigeonhole obstruction: any counterexample to chain pigeonhole at the cut must come from a tree-DP failure, which is much narrower than a general-graph obstruction.

Broader empirical sweep

Run on 7 test graphs (script: `tree_structure_sweep.py`; data: `tree_structure_sweep_data.txt`):

graph	$\|V\|$	$\|E\|$	# 6-edge cuts found	trees on both sides
HM #0	38	57	128	128/128
HM #1	38	57	127	127/127
HM #2	38	57	122	122/122
HM #3	38	57	123	123/123
HM #4	38	57	101	101/101
HM #5	38	57	97	97/97
Dodecahedron	20	30	45	45/45

Totals:

- 743 distinct 6-edge cuts examined.
- 1486 $(\text{graph}, \text{cut}, \text{side})$ triples tested.
- 11,477 cut tires examined.
- 0 **tree-structure failures** (no cycles in the parent–child relation under the vertex-overlap heuristic).

The data spans:

- The 6 Holton-McKay non-Hamiltonian 38-vertex cubic plane graphs (their duals are 21-vertex maximal planar graphs of minimal degree 4 and vertex-connectivity 3).

- The dodecahedron (20-vertex cubic plane graph, dual of the icosahedron, which is a 12-vertex 5-regular maximal planar graph with vertex-connectivity 5).

Although neither family is strictly "min degree 5 with vertex connectivity 6" (which is incompatible with the maximal-planar upper bound on average degree of $6 - 12/|V|$), the test covers duals of:

1. Several internally non-trivial maximal planar graphs (HM duals).
2. A min-degree-5 maximal planar graph (icosahedron).

This is broader than the typical chain pigeonhole test bed.

Minimum-counterexample-eligible graphs

By Birkhoff (1913), the primal of any 4CT minimum counterexample is *internally 6-connected*: every 5-vertex cut of the triangulation isolates a single vertex. We verified internal 6-connectivity directly for two test primals (script: `eligible_sweep.py`):

primal triangulation	$\|V\|$	min deg	internal 6-conn?	dual
Icosahedron	12	5	**YES** (verified)	Dodecahedron
Pentakis dodecahedron	32	5	**YES** (verified)	BuckyBall

Both primals confirmed internally 6-connected via exhaustive check over all $\binom{|V|}{5}$ vertex subsets. Tree structure sweep on the corresponding duals:

graph	$\|V\|$	$\|E\|$	# 6-edge cuts	trees on both sides
Dodecahedron	20	30	45	45/45
BuckyBall (truncated icosahedron)	60	90	60	60/60

105/105 **cuts on minimum-counterexample-eligible duals produced trees on both sides** — 0 **failures.**

This is the most direct evidence: cut tires on duals of internally 6-connected triangulations form a forest under depth nesting. No counterexample to the tree structure has been found across the entire test bed.

Empirical demonstration on Holton-McKay #0 (detailed)

G_1' side ($|S| = 28$, depths 0 to 7)

Two depth-1 roots:

- Root $(1,0)$: face length 12, no children (the outer "shell" of H_1).
- Root $(1,1)$: face length 4, with substantial subtree:
  - $(2,0)$ $|f| = 7$
    - $(3,0)$ $|f| = 2 \Rightarrow (4,0)$ $|f| = 4 \Rightarrow (5,0)$ $|f| = 14$
    - $(3,1)$ $|f| = 2 \Rightarrow (4,1)$ $|f| = 8 \Rightarrow (5,1)$ $|f| = 2 \Rightarrow (6,0)$ $|f| = 12 \Rightarrow (7,0)$ $|f| = 2$
    - $(3,2)$ $|f| = 2$
  - $(2,1)$ $|f| = 7$

G_0' side ($|S| = 10$, depths 0 to 2)

Two depth-1 roots:

- Root $(1, 0)$: face length 9, with one child $(2, 0)$ ($|f| = 6$).
- Root $(1, 1)$: face length 9, no children.

Caveats on the empirical parent identification

The empirical demonstration used a vertex-sharing heuristic to identify parents: a face f' of H_{d+1} shares vertices with a face f of H_d, and we picked the parent as the one with smallest face length. This gives ambiguous candidates in some cases (8 ambiguous cases observed in G_1') because vertex sharing does not fully determine geometric containment.

A rigorous parent test would use *point-in-region* containment: pick a point in the open face of H_{d+1} (e.g., the centroid of its boundary walk), determine which face of H_d that point lies in (via the planar embedding's face structure). This always gives a unique answer.

The ambiguity in our empirical run doesn't reflect a violation of the proposition — it's an artifact of the heuristic. Despite the ambiguity, the resulting tree structure looked sensible in both G_0' and G_1'.

Consequence: the chain half becomes tractable

With the tree structure established (or assumed), the chain half of the loose chain pigeonhole conjecture reduces to:

Reformulated chain half (tree DP form). For each leaf cut tire T_{leaf}, $\pi(T_{\text{leaf}})$ is non-empty and S_3-closed. Propagating bottom-up through the parent–child relation preserves S_3-closure and non-emptiness. At the root depth-1 tires, $\mathcal{R}_i$ is the join of the root tires' out-spoke projections. If $\mathcal{R}_i$ is S_3-closed and contains a full S_3-orbit on each side, then $\mathcal{R}_0 \cap \mathcal{R}_1 \neq \emptyset$ (containing a common orbit by S_3-equivariance).

The remaining questions:

1. Is non-emptiness preserved through parent-child propagation?
2. Is S_3-closure preserved? (Yes, by S_3-equivariance of the proper edge 3-colouring constraint.)
3. Does the join of root projections contain a full S_3-orbit?

Each of these is now a finite tree DP claim, much more tractable than the original "compose through the chain" formulation.

Next step

1. Prove Proposition rigorously using the point-in-region containment definition of parent.
2. Implement the tree DP empirically on the Holton-McKay graphs and confirm $\mathcal{R}_0 \cap \mathcal{R}_1 \neq \emptyset$ at the cut.
3. Attempt an analytical bound: $|\mathcal{R}_i| \geq somefunctionoftreesize$, ensuring $\mathcal{R}_0 \cap \mathcal{R}_1 \neq \emptyset$ in general.