FIVE ORIENTATIONS OF A 32-LEAF TREE MISS EXACTLY ONE LEAF PAIR

KOREA SUPERINTELLIGENCE LABS

Abstract. An *orientation cover* of size m of a finite tree T is a family $o_1, \ldots, o_m$ of orientations of the edges of T such that every pair of leaves is joined by a directed path in at least one o_i; the least such m is the orientation cover number $\mathrm{oc}(T)$, and by results of Fitzpatrick–Nowakowski and Brigham–Chartrand–Dutton–Zhang it is the least dimension of an isometric embedding of the path metric of T into $(\mathbb{R}^m, \|\cdot\|_\infty)$; Chalmers extends the identification to every assignment of positive edge lengths. He recently exhibited a cubic tree $\mathcal{T}$ on 62 vertices and 32 leaves with $\mathrm{oc}(\mathcal{T}) = 6$, refuting the conjecture $D(t) = \lceil \log_2 t \rceil$ — the exact form of the sharp-constant conjecture of Brigham, Chartrand, Dutton and Zhang, and the unweighted case of the conjecture of Aksoy–Kılıç–Koçak. His argument shows that five orientations do not suffice; it does not say by how much they fail. We show that they fail by exactly one pair: the maximum number of the $\binom{32}{2} = 496$ leaf pairs of $\mathcal{T}$ that five orientations can jointly join is 495, and it is attained by an explicit family that misses only the pair formed by the two leaves adjacent to one common vertex. The upper bound here is Chalmers' theorem, for which we also give what appears to be the first machine-checked proof: a branch-state recurrence with dominance pruning, run once outside the prover and then replayed inside it against a certificate. The recurrence is one-sided by construction — the states it manipulates over-approximate what five orientations realise — so a defective certificate costs a recomputation and never soundness; it settled the exclusion in under a minute of scripting time — 30 seconds of it producing the certificate verified here — where a direct SAT encoding did not close in 600 seconds. Every assertion below about a specific tree is verified in Lean 4.

1. Introduction

Let T be a finite tree with at least two vertices, with positive lengths on its edges, and let $\dim_\infty(T)$ be the least m for which the resulting path metric embeds isometrically into $(\mathbb{R}^m, \|\cdot\|_\infty)$. Writing t for the number of leaves, set

$$D(t) \;=\; \max\bigl\{\dim_\infty(T) \;:\; T \text{ a finite tree with } t \text{ leaves}\bigr\}.$$

The quantity $\dim_\infty$ is purely combinatorial. An *orientation* of T assigns a direction to each edge; leaves $x \neq y$ are *joined* by the orientation o if the unique x–y path of T is directed by o, in one direction or the other. A family $o_1, \ldots, o_m$ of orientations is an *orientation cover* when every pair of distinct leaves is joined by at least one o_i, and $\mathrm{oc}(T)$ denotes the least size of an orientation cover. For unweighted trees the two invariants are identified by Fitzpatrick and Nowakowski [3, Corollary 26], in a form quantified over all pairs of vertices, and — in the leaf-pair form used here, on Chalmers' attribution [1, §2] — by Brigham, Chartrand, Dutton and Zhang [4]. Chalmers proves the statement for weighted trees as his Proposition 2.1:

$$\dim_\infty(T) \;=\; \mathrm{oc}(T) \qquad \text{for every assignment of positive edge lengths to } T. \tag{1}$$

In particular $\dim_\infty$ does not depend on the edge lengths at all, only on the tree's topology. Nothing in this paper touches (1): we work with oc throughout, and every statement we prove is a statement about orientation covers.

The elementary lower bound is the leaf record. Given orientations $o_1, \ldots, o_m$, record for each leaf x the m-bit vector saying, for each i, whether o_i directs the edge at x towards x or away from it. Two leaves with the same record cannot be joined by any o_i, since a directed path would have to leave one leaf and enter the other. Hence t distinct records are needed and

$$\mathrm{oc}(T) \;\geq\; \lceil \log_2 t \rceil, \qquad \text{so} \qquad D(t) \;\geq\; \lceil \log_2 t \rceil. \tag{2}$$

Fitzpatrick and Nowakowski asked whether (2) can be strict [3, Problem 43]. Brigham, Chartrand, Dutton and Zhang [4] conjectured that the sharp constant is 1 and recorded that every tree with at most 21 leaves attains (2); we have both from Chalmers' account [1, §1], [4] itself being unavailable to us. The exact form of that conjecture, $D(t) = \lceil \log_2 t \rceil$, is also what unit edge lengths give in the conjecture of Aksoy, Kılıç and Koçak [5], that every positively weighted finite metric tree with at most 2^n leaves embeds isometrically into $(\mathbb{R}^n, \|\cdot\|_\infty)$. In August 2026 Chalmers [1] refuted the exact form, and with it the weighted conjecture, by an explicit cubic tree $\mathcal{T}$ on 62 vertices and 32 leaves for which

$$\mathrm{oc}(\mathcal{T}) \;=\; 6 \;>\; 5 \;=\; \lceil \log_2 32 \rceil, \qquad \text{so} \qquad D(32) \;\geq\; 6.$$

The tree is printed in [1] as a parent map (its Table 1) and six covering orientations as 61-bit hexadecimal masks (its Table 2). The hard half is the lower bound: a search over the 2^{305} five-tuples of orientations of the 61 edges is out of reach, and [1] replaces it by a recurrence over the branches of $\mathcal{T}$ with dominance pruning, executed as an uncertified Python and C++ computation archived on Zenodo [2].

What this paper adds. The question [1] answers is whether five orientations suffice. It does not ask how close they come, and neither its text nor its ancillary archive records any such quantity (Remark 8.1). For m orientations of $\mathcal{T}$ let

$$\mathrm{cov}(o_1, \ldots, o_m) \;=\; \#\big\{\{x, y\} \subseteq L(\mathcal{T}) \;:\; x \neq y,\ \text{some } o_i \text{ joins } x \text{ and } y\big\} \;\leq\; \binom{32}{2} \;=\; 496,$$

where $L(\mathcal{T})$ is the leaf set. Our main result is that five orientations come as close as they possibly can without succeeding.

Theorem 1.1 (Theorem 6.2)**.** $\max\big\{\mathrm{cov}(o_1, \ldots, o_5) \;:\; o_1, \ldots, o_5 \text{ orientations of } \mathcal{T}\big\} \;=\; 495$. *An attaining family is exhibited; the single pair it misses is the pair formed by the two leaves adjacent to one common vertex of* $\mathcal{T}$.

The upper bound cov ≤ 495 is exactly the statement $\mathrm{oc}(\mathcal{T}) > 5$ of [1], since a five-tuple with cov $= 496$ is an orientation cover. The lower bound is a witness, found by local search over a reformulation of the problem described in Section 4 and then checked directly against the definition. The content of the theorem is therefore the *exact value*, together with a witness that can be re-verified in seconds.

Alongside it we give a machine-checked proof of Chalmers' lower bound $\mathrm{oc}(\mathcal{T}) \geq 6$ (Theorem 5.1) and of $\mathrm{oc}(\mathcal{T}) = 6$ (Corollary 5.2). To the best of our knowledge no proof assistant has previously been applied to orientation covers or to the strong isometric dimension, and the verification accompanying [1] is an uncertified computation; so Theorem 5.1 appears to be the first formal proof of the refutation — Remark 8.1 says what that "appears" rests on. The mathematics of Section 4 is Chalmers' [1, §3], re-derived here from the definition before his ancillary code was read, and agreeing with it state for state (Remark 4.4).

Section 4 isolates what makes the method usable. The branch states the recurrence manipulates over-approximate the states an actual five-tuple of orientations realises, and every relaxation involved — discarding dominated states, enlarging a projection, guessing which stored state dominates a produced one — moves in the same direction. The consequence is a clean division of labour: the certificate can be produced by any means at all, including buggy ones, because a wrong certificate makes the check fail rather than the theorem false. The same one-sidedness is why the method scales where a general-purpose search does not: the recurrence closed the whole exclusion in 30 seconds of scripting time, while the natural SAT encoding did not close in 600 seconds (Remark 4.5).

Section 7 collects what remains open, including Chalmers' own next question: whether $D(t) \leq \lceil \log_2 t \rceil + 1$ for every $t \geq 2$ (his Problem 5.1), of which the first undecided instance is $D(33)$.

We are careful about one boundary. Nothing here proves $D(32) \leq 6$, nor $D(t) = \lceil \log_2 t \rceil$ for $t \leq 31$; those rest on the censuses of tree topologies reported in [1], which we have not reproduced. What is proved is $\mathrm{oc}(\mathcal{T}) = 6$ for the one tree $\mathcal{T}$, and hence $D(32) \geq 6$.

2. Orientation covers

Throughout, a *cubic tree* is a finite tree in which every vertex has degree 1 or 3; its degree-1 vertices are its *leaves* and its degree-3 vertices are *internal*. A cubic tree with $t \geq 2$ leaves has $t-2$ internal vertices and $2t-3$ edges.

Definition 2.1. Let T be a finite tree with leaf set $L(T)$. An *orientation* of T is a choice of direction for each edge. Leaves $x \neq y$ are *joined* by an orientation o if every edge of the x–y path is directed the same way along that path, i.e. if the path is a directed path from x to y or from y to x. A family $o_1, \ldots, o_m$ is an *orientation cover* of T if every pair of distinct leaves is joined by some o_i, and $\mathrm{oc}(T)$ is the least m admitting one.

Restricting to cubic trees costs nothing: every tree T has an associated tree T' with the same number of leaves, all of whose vertices have degree 1 or 3, and with $\mathrm{oc}(T) \leq \mathrm{oc}(T')$ [3, Lemma 32 and the proof of Theorem 31, pp. 34–35]. So the maximum of oc over t-leaf trees is attained on cubic topologies, which is why the counterexample may be sought there.

Two remarks fix the shape of the problem. First, D is non-decreasing ([4, p. 282], as cited in [1, §4]); combined with $\mathrm{oc}(\mathcal{T}) = 6$ this gives $D(t) \geq 6$ for every $t \geq 32$. Second, the search space for a fixed tree is enormous: $\mathcal{T}$ has 61 edges, so a five-tuple of orientations is one of 2^{305} objects, and even after the reduction of Section 4 the count is 3^{150}. Exclusion statements about oc are therefore not brute-force statements.

The rooted picture. All of Sections 4 and 5 take place in a rooted tree. Root a cubic tree at an internal vertex r; then r has three *root branches*, each hanging below one of the three edges at r, and every other internal vertex v has one parent edge $pe(v)$ and two child edges. We orient bookkeeping so that an orientation is a Boolean function on edges, the value `true` meaning "directed from the child endpoint towards the parent endpoint". For a branch B attached through the edge e and a leaf x of B, we say x is *alive through* e in the orientation o if every edge on the path from x up to and including e carries the same Boolean value; that value being `true` means the whole path is directed upward, `false` that it is directed downward, and in either case the path from x through e is directed. Two leaves lying in different branches at a vertex v are joined by o exactly when both are alive through their respective edges at v and those two edges carry opposite values.

3. The tree $\mathcal{T}$ and its six orientations

The tree of [1] is a cubic tree $\mathcal{T}$ on 62 vertices, printed there as a parent map on the vertex labels $0, \ldots, 61$, rooted at vertex 0. Its edges are indexed $e_0, \ldots, e_{60}$ by listing the pairs (u, v) with $u < v$ in lexicographic order; this convention is what gives the hexadecimal masks of [1, Table 2] their meaning, bit i being set when $e_i = (u, v)$ is directed from u to v.

Since everything below is a statement about a specific 62-vertex object, the first thing to certify is that the object we compute with is the printed one.

Proposition 3.1. *Reading the vertex labels and edge indices off the tree used in the formal development and sorting by edge index reproduces, for each of the* 61 *lexicographic edges* $e_i = (u, v)$ *with* $u < v$*, the child/parent pair that the parent map of* [1, Table 1] *assigns to it. That tree has* 32 *pairwise distinct leaves and* 61 *edges, and its three root branches carry* 4*,* 13 *and* 15 *leaves.*

The counts are exactly those stated in [1], and they pin the tree's coarse shape: 32 leaves and 30 internal vertices, since $32 + 30 = 62$ and $32 \cdot 1 + 30 \cdot 3 = 2 \cdot 61$. We write $L(\mathcal{T})$ for the leaf set and note $|L(\mathcal{T})| = 32$, so $\mathcal{T}$ has $\binom{32}{2} = 496$ leaf pairs. Two leaves of its 4-leaf root branch — vertices 3 and 4, both adjacent to vertex 2 — will reappear in Section 6.

Proposition 3.2. *The six orientations of $\mathcal{T}$ obtained by decoding the hexadecimal masks of* [1, Table 2] *under the convention above join all* 496 *leaf pairs, so* $\mathrm{oc}(\mathcal{T}) \le 6$. *Moreover none of the six is redundant: for each of them there is a leaf pair that the other five fail to join.*

The first assertion is Chalmers' upper bound, re-checked; the second is a finite check on his data which we have not found stated, and which says that the upper bound is not slack in the crude sense. A single orientation does far less on its own: the first of the six joins 112 of the 496 pairs, and the six together join all 496, so the covering relation is neither vacuous nor trivial.

4. Branch states, and why the method is one-sided

This section is the method of [1, §3], in the form we use it. It was re-derived here from Definition 2.1 before the ancillary code of [2] was read; the agreement is recorded in Remark 4.4. Fix a cubic tree rooted at an internal vertex and fix $m = 5$; write $[5] = \{1, \dots, 5\}$ for the index set of a hypothetical five-tuple $o_1, \dots, o_5$ of orientations.

Pivots. The first observation explains the arithmetic that follows and bounds the search space.

Lemma 4.1. *Let v be an internal vertex with incident edges f_1, f_2, f_3 and let o be an orientation. Split the three edges according to whether o directs them into v or out of v. If the split is constant, no leaf-to-leaf path through v is directed at v. Otherwise exactly one edge lies alone on its side of the split — call it the* pivot *at v — and a leaf-to-leaf path through v using $\{f_i, f_j\}$ is directed at v if and only if the pivot is one of f_i, f_j. Consequently a pair of leaves is joined by o if and only if, at every internal vertex on its path, the pivot is one of the two path edges.*

Proof. A path through v using $\{f_i, f_j\}$ is directed at v exactly when one of the two edges points into v and the other out of v, i.e. when they lie on opposite sides of the split. A constant split puts all three on one side. A non-constant split of a three-element set has a unique minority element f_k, and the pairs on opposite sides are exactly $\{f_k, f_i\}$ and $\{f_k, f_j\}$. The last sentence follows because a path is directed if and only if it is directed at each of its internal vertices. $\square$

Every assignment of pivots arises from an orientation: fix the direction of the parent edge at each internal vertex in turn, top down, and the desired minority edge determines the other two directions. Constant splits are never useful, since they block all three pairs at v while every pivot choice blocks only one. So, for covering purposes, an orientation of a cubic tree with c internal vertices *is* an element of a set of size 3^c, and for $\mathcal{T}$ with $c = 30$ the count $3^{30} \approx 2.06 \cdot 10^{14}$ replaces 2^{61}; across five orientations, 3^{150} replaces 2^{305}. This is the form in which all the searches of Sections 6 and 7 were run. Lemma 4.1 is elementary and is surely folklore; it is recorded because it is what makes those searches cheap, and nothing in the formal development depends on it.

States. Fix a branch B attached through the edge e. For a leaf x of B put

$$M(x) \;=\; \big\{\, j \in [5] \;:\; x \text{ is alive through } e \text{ in } o_j \,\big\} \;\subseteq\; [5],$$

the set of coordinates in which the path from x up through e is directed, and let

$$S(B,e) \;=\; \big\{\, M(x) \;:\; x \in L(B) \,\big\}$$

be the *state* of the branch: a set of subsets of [5], of which there are at most 2^{32}. The state is all the rest of the tree can see. Indeed, if B' is another branch meeting B at a vertex v, through the edge e', and

$$D \;=\; \big\{\, j \in [5] \;:\; o_j \text{ gives } e \text{ and } e' \text{ opposite values} \,\big\}$$

is their *sign difference*, then by the last paragraph of Section 2 every pair of leaves $x \in L(B)$, $y \in L(B')$ is joined if and only if

$$a \cap b \cap D \;\neq\; \emptyset \qquad \text{for all } a \in S(B,e),\; b \in S(B',e'). \tag{3}$$

The recurrence. At an internal vertex v with child edges c_a, c_b, child branches A, B and parent edge pe, set

$$P = \{j : o_j(c_a) = o_j(pe)\}, \qquad Q = \{j : o_j(c_b) = o_j(pe)\},$$

Boolean values read in the child-to-parent convention. Then the sign difference of the two children is $D = P \,\Delta\, Q$, a leaf of A alive through c_a stays alive through pe exactly in the coordinates of P, and likewise for B and Q. So the state of the merged branch at v is

$$\big\{\, a \cap P \;:\; a \in S(A, c_a) \,\big\} \;\cup\; \big\{\, b \cap Q \;:\; b \in S(B, c_b) \,\big\}, \tag{4}$$

and the pairs meeting at v are covered exactly under condition (3) for $S(A, c_a)$, $S(B, c_b)$ and D. At the root r, with the three root-branch states A, B, C and the sign differences D_{AB}, D_{AC} — from which $D_{BC} = D_{AB} \,\Delta\, D_{AC}$ — a five-tuple that covers everything must satisfy (3) for all three pairs simultaneously.

Domination. Say that a state S *dominates* a state R when every member of S contains some member of R. If S dominates R then R's instance of (3) implies S's, for any b and any D: given $s \supseteq r$ with $r \cap b \cap D \neq \emptyset$, also $s \cap b \cap D \neq \emptyset$. In words, a dominating state is *weaker*: whatever completes R completes S. So a recurrence that is only ever going to conclude "nothing completes" may replace any produced state by a dominating one, and in particular may keep, at each vertex, only a set of states dominating everything the recurrence can produce there.

Two further relaxations are used, in the same direction. The pair (P, Q) of (4) is replaced by $(P \cup \overline{P \cup Q},\, Q)$; coordinatewise this changes only the case $j \notin P \cup Q$, which by Lemma 4.1 is exactly the case of a constant split at v, and it enlarges both the projected masks and D. Enlarging a mask weakens the state, and enlarging D weakens (3), so the substitution is sound; its purpose is that the enlarged pair satisfies $P \cup Q = [5]$, so that each coordinate falls into one of three cases — exactly the three pivot choices of Lemma 4.1 — and the enumeration per merge is $3^5 = 243$ rather than 2^{10}. Finally, when a produced state has to be matched against a stored dominating state, the matching index is supplied as a hint and the domination is then verified; a wrong hint fails the verification.

Remark 4.2 (one-sidedness). Collecting the three relaxations: the states the recurrence manipulates dominate — are weaker than — the states an actual five-tuple of orientations realises. A run that concludes "no assignment of sign differences at the root satisfies all three cross conditions" therefore establishes the same conclusion for the realised states, and hence that no five orientations cover. The converse fails: a run may be unable to conclude even though no five orientations cover, if the relaxation is too coarse. The method can only *under*-report what five orientations achieve. Practically, this is the reason the certificate may be produced by fast unverified code: a certificate that is wrong — a state missing from the stored list, a bad domination hint, an off-by-one in the enumeration — makes the verification fail and costs a recomputation. It cannot make a false theorem provable, because the verification and the soundness argument, and not the program that produced the certificate, are what the proof rests on.

Remark 4.3 (the certificate). Concretely, the certificate we verify stores, for each vertex of $\mathcal{T}$, the retained (non-dominated) states there together with, for every state the recurrence can produce at that vertex, the index of a retained state dominating it. Over the whole tree this is 2,089 retained states and 242,525 produced states, the largest single vertex carrying 455 and 100,325 of them; the verification walks 1,543,446 guarded combinations of (child state, child state, P, Q). The stored data occupies about 2.7 MB as hexadecimal text.

Remark 4.4 (agreement with the source). Our run retains 10, 360 and 455 states at the roots of the 4-, 13- and 15-leaf branches of $\mathcal{T}$. Chalmers reports 1, 6 and 9 nondominated states for the same three branches [1, §3], his states being reduced to inclusion-minimal masks and taken up to permutation of the five coordinates; the states themselves are printed in the file `lower_bound_recurrence.txt` of the ancillary archive [2], which we re-downloaded on 2026-08-29. Ours are reduced to inclusion-minimal masks as well, so canonicalising them under the symmetric group on the five coordinates is the only step needed, and it returns exactly his three lists, mask for mask — the 4-leaf branch, for instance, retaining the single state $\{00111\}$.

Remark 4.5 (the recurrence against SAT). The natural propositional encoding of "five orientations of $\mathcal{T}$ cover all 496 pairs" uses one variable per (edge, coordinate) pair together with an auxiliary variable per (leaf pair, coordinate, direction): 5,265 variables and 42,976 clauses; fixing the global flip and imposing a lexicographic order on the five coordinates makes it 5,505 variables and 44,417 clauses. CaDiCaL 2.1.2 [8] returned no answer on either within 600 seconds, reaching 4.5 million conflicts on the symmetry-broken one. The branch-state recurrence, by contrast, ran to completion as a Python script: 51 seconds for the full pass and the root test, of which 30 seconds produced the certificate that is verified formally. The reason is structural rather than incidental — the recurrence exploits the separator structure of the tree and the domination order, neither of which the clause set exposes — and it is the reason we regard the recurrence, and not a general-purpose search, as the instrument for orientation-cover non-existence. The comparison is a single measurement on one encoding and one solver, and should be read as such.

5. No five orientations cover $\mathcal{T}$

Theorem 5.1. *No five orientations of the* 61 *edges of* $\mathcal{T}$ *join every pair of its* 32 *leaves by a directed path; that is,* $\mathrm{oc}(\mathcal{T}) \geq 6$.

Proof sketch. Suppose $o_1, \ldots, o_5$ join every pair. Because distinct root branches are disjoint, the hypothesis restricts: any two leaves of a single branch are joined by a path lying inside that branch, so each branch, on its own, satisfies the hypothesis of the induction below.

The induction is on the branch. For a branch B attached through e that passes the verification of Remark 4.3, has pairwise distinct leaves, and all of whose internal leaf pairs are joined, the state $S(B, e)$ realised by $o_1, \ldots, o_5$ is dominated by one of the states stored for the root vertex of B. At a leaf the state is the single full mask [5], which is what the certificate stores there. At an internal vertex the two child branches satisfy the inductive hypothesis, the pairs meeting at the vertex give (3), and (4) with the enlargement of Section 4 exhibits the realised state as one of the states the verification has walked; the stored dominating state then dominates the realised one, by transitivity of domination.

Applying this to the three root branches yields three stored states dominating the realised ones, and the realised sign differences at the root satisfy all three instances of (3). The root part of the verification asserts that no triple of stored states admits sign differences with that property, a contradiction.

What rests on computation is exactly this: the finite verification that the stored certificate is closed under the recurrence and that its root triples are all excluded. That verification is a bounded loop over the 1,543,446 guarded combinations recorded in Remark 4.3 and over all triples of stored root-branch states, and it is evaluated by compiled execution inside the proof assistant. The inductive argument that converts it into the theorem is an ordinary proof and uses no computation at all; see Section 8. ☐

Combining with Proposition 3.2:

Corollary 5.2. 6 *is the least* m *for which* $\mathcal{T}$ *admits an orientation cover of size* m*:* $\mathrm{oc}(\mathcal{T}) = 6$. *Consequently* $D(32) \geq 6 > 5 = \lceil \log_2 32 \rceil$*, so the conjecture* $D(t) = \lceil \log_2 t \rceil$ *is false, and with it the weighted conjecture of* [5].

Proof. Membership is Proposition 3.2. For minimality, an orientation cover of size $m \leq 5$ with $m \geq 1$ yields one of size 5 by repeating one of its members, which Theorem 5.1 forbids; and $m = 0$ fails because $\mathcal{T}$ has two distinct leaves. The consequence for D is (1) together with $\mathcal{T}$ having 32 leaves. ☐

We stress once more what is not claimed. $D(32) \leq 6$, and hence $D(32) = 6$, is proved in [1] (its Theorem 1.2) and depends on a census of small tree topologies reported there; we have not reproduced it, and nothing above establishes it. Nor is the asymptotic form of the sharp-constant conjecture touched: whether $D(t) - \lceil \log_2 t \rceil$ is bounded, and whether $D(t) = (1 + o(1)) \log_2 t$, are both open [1, §5].

6. The exact five-orientation maximum

Theorem 5.1 says five orientations miss at least one of the 496 leaf pairs. They need miss no more.

Proposition 6.1. *There are five explicit orientations of $\mathcal{T}$ jointly joining 495 of its 496 leaf pairs. The single pair they miss is $\{3, 4\}$, the two leaves adjacent to vertex 2 of [1, Table 1].*

Theorem 6.2. *495 is the greatest value of $\operatorname{cov}(o_1, \ldots, o_5)$ over all five-tuples of orientations of $\mathcal{T}$:*

$$\max\{\operatorname{cov}(o_1, \ldots, o_5)\} \;=\; 495 \;=\; \binom{32}{2} - 1.$$

Proof. The value 495 is attained by Proposition 6.1. For the upper bound, $\operatorname{cov}(o_1, \ldots, o_5) \leq 496$ always, and $\operatorname{cov}(o_1, \ldots, o_5) = 496$ would say that every leaf pair is joined by some o_i, i.e. that $o_1, \ldots, o_5$ is an orientation cover of $\mathcal{T}$, contradicting Theorem 5.1. Hence the maximum is at most 495. □

The proof is short because the two halves come from opposite directions and each is hard in its own way. The upper bound is Chalmers' theorem and nothing less will do: no counting argument available to us excludes 496, since by (2) the leaf-record obstruction is silent at $m = 5$ and $t = 32$. The lower bound is a witness, and the witness was not easy to find.

How the witness was found. In the pivot form of Lemma 4.1 a five-tuple of orientations of $\mathcal{T}$ is a vector of 150 ternary variables, and each of the 496 covering constraints is a disjunction over the five coordinates of a conjunction of pivot conditions, one per internal vertex on the pair's path — on average about 7.6 of them. A min-conflicts local search over this space was run from three independent random seeds, each given 20 seconds and performing about 53 restarts in that time; each seed reached configurations leaving exactly one pair uncovered, and none ever reached zero. One such configuration was converted back to five orientations of the 61 edges and re-verified against Definition 2.1 by direct path computation: 495 of the 496 pairs joined, the missing one being $\{3, 4\}$. The published data of Proposition 6.1 is that configuration, as five hexadecimal masks in the convention of Section 3; verifying it is a loop over 496 paths and is what the formal statement checks. The search itself is heuristic and is no part of the proof. □

Remark 6.3. Three independent seeds reaching 1 and never 0 is, of course, not evidence of anything by itself — Theorem 5.1 is what excludes 0. It is recorded because it is how the value 495 was conjectured before it was proved, and because the same search at $m = 6$ reaches 0 uncovered pairs at its first restart, which locates the difficulty of $\mathcal{T}$ sharply at $m = 5$.

A control. An exclusion result invites the question whether the machinery excludes too much. It does not: at 32 leaves, five orientations generically suffice.

Proposition 6.4. *Let $\mathcal{B}$ be a balanced cubic tree on 32 leaves: rooted at an internal vertex, its three branches carry 10, 11 and 11 leaves, and every branch is split as evenly as the parity allows. Then $\mathcal{B}$ has an orientation cover of size five; explicitly, five orientations of $\mathcal{B}$ join all 496 of its leaf pairs.*

Proposition 6.4 is a witness, found at the first restart of the same local search that needed some fifty restarts on $\mathcal{T}$ merely to reach 495. Its role is negative: it is stated and checked under exactly the definitions that make Theorem 5.1 a statement about $\mathcal{T}$, so those definitions are satisfiable at 32 leaves and the failure proved in Theorem 5.1 is a property of $\mathcal{T}$ and not an artefact of the encoding.

Together with the irredundancy half of Proposition 3.2 and the count 112 of Section 3, it also rules out the degenerate readings in which the joining relation is always true or always false.

7. Open questions

$D(33)$. Chalmers asks [1, Problem 5.1] whether $D(t) \le \lceil \log_2 t \rceil + 1$ for every $t \ge 2$. The first open instance is $t = 33$. Monotonicity gives $D(33) \ge 6$, and the leaf-deletion argument of [1, §4] gives $D(t) \le D(t-1)+1$, so $D(33) \le 7$ once $D(32) = 6$ is granted — which is [1]'s Theorem 1.2 and is not proved here. We stress that nothing in this chain bounds D uniformly further up: iterating it gives only $D(t) \le 6 + (t - 32)$, so a uniform bound of 7 on $33 \le t \le 64$ would be Problem 5.1 restricted to that range, not a consequence of it. The open cell is therefore $D(33) \in \{6, 7\}$; both directions are expensive with present methods, and we record the price rather than an attempt. Settling $D(33) = 6$ requires a census of 33-leaf cubic topologies, each tested at $m = 6$; the census reported in [1] at $t = 31$ already runs to 75,021,750 centroid triples, and the 33-leaf count is several times that. Worse, the state space of the recurrence grows badly with m: at $m = 5$ a state is an antichain of subsets of a 5-element set, of which there are 7,581, whereas at $m = 6$ the corresponding count is the next Dedekind number, 7,828,354 [9, A000372], and the enumeration per merge grows from $3^5 = 243$ to $3^6 = 729$. Settling $D(33) = 7$ instead requires exhibiting one 33-leaf tree with $\mathrm{oc} \ge 7$, i.e. a six-orientation exclusion over the same unmeasured space. The inexpensive first experiment is to measure the retained-state counts at $m = 6$ on a single 33-leaf tree.

How rare is $\mathrm{oc} = 6$ at 32 leaves? Trees with $\mathrm{oc} = 6$ appear to be scarce at 32 leaves. We generated 160 random 32-leaf cubic trees by repeated random edge subdivision and screened each with a 12-second budget of the min-conflicts search of Section 6: all 160 yielded a five-orientation cover, while $\mathcal{T}$ yields none under the same search at 20 seconds and three seeds. This is a negative result about a heuristic, not a theorem: it says the uniform random model is the wrong instrument for finding a second example, and suggests structured perturbations of $\mathcal{T}$ instead. We know of no second 32-leaf tree with $\mathrm{oc} = 6$, and exhibiting one non-isomorphic to $\mathcal{T}$ would be a genuinely new result.

Which pair can be the missed one? Theorem 6.2 says five orientations must miss a pair, and Proposition 6.1 exhibits $\{3, 4\}$ as a pair that can be the only one missed. Say that a leaf pair p is *omittable* if some five orientations join every leaf pair except possibly p. Which of the 496 pairs are omittable? An 8-second min-conflicts run per pair found five orientations for 339 of them; for the remaining 157 the search failed, and we do not know whether they are genuinely non-omittable or merely hard. Deciding one of these cases in the negative would need the recurrence of Section 4 extended to carry the masks of two distinguished leaves alongside the branch state — a modest change to the state, but a second certificate format and a second soundness argument.

The maximum at four orientations. The analogue of Theorem 6.2 at $m = 4$ is open. Here the leaf-record argument of (2) does give something: 32 leaves distribute over $2^4 = 16$ records, so at least 16 pairs share a record and are joined by no orientation of the family, whence $\mathrm{cov}(o_1, \ldots, o_4) \le 480$. Whether 480 is attained is unknown to us; note that the upper bound here is a convexity argument over record classes rather than a finite check, so a matching witness would pin a second exact value by genuinely different means.

The weighted statement. Finally, (1) itself is untouched by anything above. Formalising the identity $\dim_\infty = \mathrm{oc}$ would connect the combinatorics proved here to the metric geometry that motivates it, and is real, non-finite mathematics; all our statements concern oc alone.

8. Verification

Every statement of this paper about a specific tree — Propositions 3.1, 3.2, 6.1 and 6.4, Theorems 5.1 and 6.2, and Corollary 5.2 — has been verified in Lean 4 against *Mathlib* [6, 7]; the development contains no `sorry` and no additional axiom declarations. (Lemma 4.1 and the soundness discussion of Section 4 are ordinary mathematics and no part of it.) Repository reference to be added at submission. The division of labour is the point of Remark 4.2 and can be read off the axiom sets. The inductive argument sketched in the proof of Theorem 5.1 — the statement that any state realised by five orientations on a branch passing the verification is dominated by a stored state — depends on no axioms beyond `propext`, `Classical.choice` and `Quot.sound`: it contains no appeal to computation whatever. What is delegated to compiled evaluation, through Lean's `native_decide`, is exactly the finite checks: that the encoded tree agrees with the printed parent map and has the stated counts (Proposition 3.1); that the six published masks join all 496 pairs and that each is irredundant (Proposition 3.2); that the stored certificate is closed under the recurrence at each of the three root branches and that its root triples are all excluded (Theorem 5.1); that the exhibited five masks join exactly 495 pairs (Proposition 6.1); and the corresponding checks for the control (Proposition 6.4). Corollary 5.2 and Theorem 6.2 add no computation of their own: monotonicity in m, the case $m = 0$, and the passage from $\mathrm{cov} = 496$ to a cover are ordinary proofs.

Two remarks on the reliability of that split. First, no property of the certificate is assumed: the search that locates a stored state matching a produced one carries no correctness proof, because its answer is checked and a wrong answer simply fails the check. Second, every number quoted above was produced first by an independent implementation in Python or C and the formal statement was written against it: the coverage of the six published masks, the retained-state counts of Remark 4.4, and the value 495 were all computed outside the prover before any of them was proved, and Remark 4.4 records an agreement with a third, independent implementation — the author's. Checking the mathematical file takes about 20 seconds with the data module precompiled, and about 3 minutes without.

Remark 8.1 (what the two negatives rest on). Two claims above are negatives and are to be read as "not found", not "does not exist". That Theorem 5.1 is the first machine-checked proof of Chalmers' exclusion: as of 2026-08-29 we know of no treatment of orientation covers or of the strong isometric dimension in any proof assistant, on arXiv metadata queries ("strong isometric dimension", and "isometric dimension" with "Lean", return nothing; "orientation cover" with "tree" returns two papers on genome tree spaces) and on web searches for a formalisation of either notion. That the quantity of Theorem 6.2 is new: [1] and every file of the archive [2] were read in full for this paper, as were [3] and [5] (the latter in its preprint version), and none of them records any measure of how close m orientations come. We were unable to obtain [4], so there the negative rests on Chalmers' account of it.

References

[1] L. R. Chalmers, *A 32-leaf tree requiring six coordinates for an isometric ℓ_∞ embedding*, arXiv:2608.16288v1 [math.MG], 17 August 2026. All numbered references to [1] above are to v1, which was still the current version on 2026-08-29.

[2] L. R. Chalmers, *Verification data and code for "A 32-leaf tree requiring six coordinates for an isometric ℓ_∞ embedding"* (ancillary material to [1]), Zenodo, 2026, doi:10.5281/zenodo.21965426.

[3] S. L. Fitzpatrick and R. J. Nowakowski, *The strong isometric dimension of finite reflexive graphs*, Discuss. Math. Graph Theory **20** (2000), no. 1, 23–38, doi:10.7151/dmgt.1104.

[4] R. C. Brigham, G. Chartrand, R. D. Dutton and P. Zhang, *On the dimension of trees*, Discrete Math. **294** (2005), no. 3, 279–283, doi:10.1016/j.disc.2004.11.013.

[5] A. G. Aksoy, M. Kılıç and Ş. Koçak, *Isometric embeddings of finite metric trees into* $(\mathbb{R}^n, d_1)$ *and* $(\mathbb{R}^n, d_\infty)$, in: The Mathematical Legacy of Victor Lomonosov: Operator Theory, Advances in Analysis and Geometry, vol. 2, De Gruyter, Berlin, 2020, pp. 15–24, doi:10.1515/9783110656756-002; preprint arXiv:2002.00062.

[6] L. de Moura and S. Ullrich, *The Lean 4 theorem prover and programming language*, in: Automated Deduction — CADE-28, Lecture Notes in Computer Science, vol. 12699, Springer, 2021, pp. 625–635.

[7] The mathlib Community, *The Lean mathematical library*, in: Proceedings of the 9th ACM SIGPLAN International Conference on Certified Programs and Proofs (CPP 2020), ACM, 2020, pp. 367–381, doi:10.1145/3372885.3373824.

[8] A. Biere, T. Faller, K. Fazekas, M. Fleury, N. Froleyks and F. Pollitt, *CaDiCaL 2.0*, in: Computer Aided Verification (CAV 2024), Part I, Lecture Notes in Computer Science, vol. 14681, Springer, 2024, pp. 133–152, doi:10.1007/978-3-031-65627-9_7. The binary used here is CaDiCaL 2.1.2.

[9] OEIS Foundation Inc., *The On-Line Encyclopedia of Integer Sequences*, `https://oeis.org`; sequence A000372 (Dedekind numbers).