THE EXACT HEIGHT AT WHICH A TREE FORCES A LADDER: $T(2) = 2$ AND $T(3) = 3$ FOR NON-UNIFORM (k,δ)-LADDERS

KOREA SUPERINTELLIGENCE LABS

Abstract. In `arXiv:2607.21761v1`, Conant and Terry introduced a notion of non-uniform ladder for a function $f\colon X \times Y \to [0,1]$ and proved a real-valued analogue of the Shelah–Hodges tree/ladder correspondence: if f admits a $\big(\binom{2k}{k} - 1,\, 2\delta\big)$-tree, then f admits a (k,δ)-ladder. They ask whether the height, which grows like $4^k/\sqrt{\pi k}$, can be brought down to the order of 2^k. Write $T(k)$ for the least height that suffices. We determine it exactly for $k \leq 3$: $T(2) = 2$ and $T(3) = 3$, against the values 5 and 19 supplied by their theorem, and we show $T(k) \geq k$ for every k. The upper bound at $k = 3$ is a four-case analysis on the entries that the tree leaves unconstrained; the lower bounds come from the class of all Boolean concepts on d points. Separately, and as computations rather than as theorems, we record that the two-parameter induction used in the source cannot reach $T(3) = 3$, and that eleven computed values of the auxiliary threshold of that induction all equal $2^{\min(k,\ell)-1}$; if that pattern persists it answers the question affirmatively, with $t = 2^{k-1}$. Every theorem below is formalised in Lean 4 and checked by its kernel; the conjecture, and the measurements behind it, are not.

1. Introduction

1.1. **Trees and ladders in a real-valued function.** Throughout, 2^t denotes the set of binary strings of length t (the *leaves*), $2^{<t}$ the set of binary strings of length $< t$ (the *nodes*), $\sqsubseteq$ the initial-segment relation, and $\sigma^\frown b$ the string σ followed by the bit b. The empty string is $\varnothing$. The source writes these as ${}^t2$, ${}^{<t}2$ and $\trianglelefteq$; the passages quoted below are verbatim except for that change of notation and for the resolution of the source's internal cross-references into numbers.

Conant and Terry [1] study two configurations inside a bounded real-valued function. Their Definition 1.2 reads, verbatim:

> Fix $t \geq 1$, $\delta > 0$, and a function $f\colon X \times Y \to [0,1]$. A (t,δ)-*tree* for f consists of sequences $(x_\sigma : \sigma \in 2^{<t})$ from X, $(y_\eta : \eta \in 2^t)$ from Y, and $(r_\sigma : \sigma \in 2^{<t})$ from $[0,1]$ such that for all $\sigma \in 2^{<t}$ and $\eta, \eta' \in 2^t$, if $\sigma^\frown 0 \sqsubseteq \eta$ and $\sigma^\frown 1 \sqsubseteq \eta'$ then $f(x_\sigma, y_\eta) \leq r_\sigma$ and $f(x_\sigma, y_{\eta'}) \geq r_\sigma + \delta$.

and their Definition 1.4(1), verbatim:

> A (k,δ)-*ladder* for f consists of sequences $x_1, \ldots, x_k \in X$, $y_1, \ldots, y_k \in Y$, and $r_1, \ldots, r_k \in [0,1]$ such that for all $i, j \in [k]$, if $i \leq j$ then $f(x_i, y_j) \geq r_i + \delta$, and if $i > j$ then $f(x_i, y_j) \leq r_i$.

Date: September 3, 2026.

Two features of these definitions do real work below and are easy to lose in paraphrase. First, the thresholds are constrained to $[0,1]$ in both definitions, not merely to $\mathbb{R}$; the *lower* endpoint $r \geq 0$ is what makes the first row of a ladder uniformly large (Lemma 6), and the requirement $r \leq 1$ is what the codomain bound $f \leq 1$ is used for — in every ladder built below, and nowhere else (Lemma 3). Second, the ladder of Definition 1.4(1) is *non-uniform*: the thresholds r_i may differ from row to row. The source says of this notion: "The non-uniform variation of ladders in Definition 1.4(1) has not, to our knowledge, appeared in the literature before." The uniform variation, in which the values $r_1, \dots, r_k$ are all equal to a common value $r \in [0,1]$, is the source's Definition 1.4(2), and the identification of it with what the learning-theory literature calls the fat-threshold dimension is the source's own, not a reading of ours: "The definition of *fat-threshold dimension* for a class of $[0,1]$-valued functions appears later in [4], and essentially corresponds to our notion of uniform ladders" — verbatim, with only the citation key replaced by ours. Its Definition B.2 spells this out, defining the δ-fat-threshold dimension of a class as the supremum of the k for which the class's evaluation function admits a uniform (k,δ)-ladder. Nothing in this paper says anything about the uniform variation.

In learning-theoretic language a (t,δ)-tree for f witnesses that the sequential δ-fat-shattering dimension of the class $\{f(\cdot,y) : y \in Y\}$ is at least t (the source's Definition B.1, in its Appendix B), and when f is $\{0,1\}$-valued and $0 < \delta \leq 1$ it is exactly a Littlestone tree of depth t, so that Definition 1.2 specialises to the Littlestone dimension being at least t.

1.2. **The theorem being sharpened, and its open question.** For relations — equivalently, for $\{0,1\}$-valued f — the correspondence between trees and ladders is classical, going back to Shelah and to Hodges [2]; the source states it as its Theorem 1.6(2): "Given $k \geq 1$, if E admits a $(2^{k+1}-2)$-tree, then E admits a k-ladder." The main real-valued theorem of [1] is its Theorem 1.11, verbatim:

> Given $k \geq 1$ and $\delta > 0$, if $f \colon X \times Y \to [0,1]$ admits a $\left(\binom{2k}{k}-1, 2\delta\right)$-tree, then f admits a (k,δ)-ladder.

The halving of the scale — a tree of scale 2δ yielding a ladder of scale δ — cannot be avoided, by the source's own Proposition 3.4 (reproduced here as Proposition 18). What is not known is whether the height is. The source's Remark 3.2 observes that $\binom{2k}{k}-1$ grows like $4^k/\sqrt{\pi k}$ while the discrete bound of Theorem 1.6(2) is on the order of 2^k, and the Question numbered 3.3 immediately after it asks, verbatim: "In Theorem 1.11, can t be bounded on the order of 2^k?"

That question is about a growth rate. The present paper asks the smaller and more concrete question underneath it: for a fixed k, what *is* the least height that works?

Definition 1. For $k \geq 1$ let $T(k)$ be the least $t \geq 1$ such that for every pair of sets X, Y, every $f \colon X \times Y \to [0,1]$ and every $\delta > 0$: if f admits a $(t, 2\delta)$-tree, then f admits a (k,δ)-ladder.

Definition 1 is legitimate on two counts. The set of admissible t is nonempty because Theorem 1.11 of [1] exhibits one, and it is upward closed because a taller tree contains a shorter one (Lemma 2); so $T(k)$ is a genuine threshold and $T(k) \leq \binom{2k}{k}-1$.

1.3. **Results.**

k	1	2	3	4	5
$\binom{2k}{k} - 1$, from [1], Theorem 1.11	1	5	19	69	251
$2^{k+1} - 2$, the discrete bound of [1], Theorem 1.6(2)	2	6	14	30	62
$T(k)$	1	**2**	**3**	≥ 4	≥ 5

The middle line is included only for scale, and it is *not* an upper bound for $T(k)$: Theorem 1.6(2) bounds the analogous threshold for relations, and since indicator functions embed the discrete situation into the real-valued one, that threshold is at most $T(k)$.

The two upper bounds are Theorems 11 and 12: every $(2, 2\delta)$-tree forces a $(2, \delta)$-ladder, and every $(3, 2\delta)$-tree forces a $(3, \delta)$-ladder, for all X, Y and all $\delta > 0$. Against these, Theorem 1.11 of [1] supplies the heights 5 and 19. The matching lower bounds are instances of a single construction, Theorem 14: the class of all Boolean concepts on d points admits a $(d, 1)$-tree and no $(d+1, \delta)$-ladder whatever $\delta > 0$, whence $T(k) \geq k$ for every k. Corollaries 15 and 16 combine the halves:

$$T(2) = 2, \qquad T(3) = 3.$$

The value $T(1) = 1$ is immediate from the definitions and is not part of the formal development: from a $(1, 2\delta)$-tree, take as the one-element ladder the row $x_\varnothing$, the leaf indexed by the string 1, and the threshold $r_\varnothing$.

Two consequences are worth stating. The bound of Theorem 1.11 of [1] is not close to tight at the first two nontrivial values: it is off by a factor $5/2$ at $k = 2$ and by a factor $19/3$ at $k = 3$. And the least conceivable behaviour — a height logarithmic in k, as in the published discrete relation that a class of Littlestone dimension d has threshold dimension at least $\lfloor \log d \rfloor$ [3] — is excluded outright by $T(k) \geq k$. Whether $T(k) = k$ for every k is open here; a positive answer would answer the source's Question 3.3 with a great deal to spare.

1.4. **What is proved and what is only computed.** Every numbered theorem, proposition, lemma and corollary of Sections 2–6 is a proof: no statement among them rests on an exhaustive search, and none of them was checked only up to some bound. Four finite verifications do occur inside the proof of Proposition 18, all of them over the 3×4 node/leaf table of a height-2 tree, and all of them decided by the Lean kernel rather than by a compiled or external procedure (Section 8).

Exhaustive computation does appear in this work, in two roles, and Section 7 is devoted to it and is clearly separated from the theorems. First, as a source of the proofs: the case analysis of Theorem 12 was *found* by minimising the unsatisfiable core of a difference-logic encoding of the height-3 instance, which returned four of the 336 ordered triples of distinct leaves; the resulting proof is then checked by hand and by machine, and does not refer to the solver. Second, as a source of statements we do *not* claim as theorems: the eleven computed values of the source's own auxiliary threshold, the free-ladder and discrete controls, and the height-4 wall. (The solver also returns $T(4) \geq 4$; that one is not a measurement here, because Theorem 14 proves it.) Those are reported as measurements, with the encoding, the instrument and the wall time attached, and the conjecture of Section 7.4 is labelled a conjecture drawn from eleven values.

2. Preliminaries

We restate the two definitions of §1.1 in the form used below, with the scale of the tree written as a free parameter s (all our applications take $s = 2\delta$, exactly as in Theorem 1.11 of [1]).

Let X, Y be sets and $f\colon X \times Y \to \mathbb{R}$. For $t \geq 1$ and $s > 0$, a (t,s)-*tree* for f is a triple of families $(x_\sigma)_{\sigma\in 2^{<t}}$ in X, $(y_\eta)_{\eta\in 2^t}$ in Y and $(r_\sigma)_{\sigma\in 2^{<t}}$ in $[0,1]$ such that for all $\sigma \in 2^{<t}$ and all $\eta \in 2^t$,

$$\sigma^\frown 0 \sqsubseteq \eta \implies f(x_\sigma, y_\eta) \leq r_\sigma, \qquad \sigma^\frown 1 \sqsubseteq \eta \implies f(x_\sigma, y_\eta) \geq r_\sigma + s.$$

For $k \geq 1$ and $\delta > 0$, a (k,δ)-*ladder* for f is a triple of finite sequences $x_1, \dots, x_k \in X$, $y_1, \dots, y_k \in Y$ and $r_1, \dots, r_k \in [0,1]$ such that

$$i \leq j \implies f(x_i, y_j) \geq r_i + \delta, \qquad i > j \implies f(x_i, y_j) \leq r_i.$$

We say f *admits* such a configuration when one exists.

Lemma 2 (Monotonicity)**.** *Let $f\colon X \times Y \to \mathbb{R}$. If f admits a (t,s)-tree and $t' \leq t$, then f admits a (t',s)-tree. If f admits a (k,δ)-ladder and $k' \leq k$, then f admits a (k',δ)-ladder.*

Proof. For the ladder, keep the first k' rows, leaves and thresholds; every required inequality is one of the original ones. For the tree, keep the same nodes (x_σ) and thresholds (r_σ) for $\sigma \in 2^{<t'} \subseteq 2^{<t}$ and send a leaf $\eta \in 2^{t'}$ to the leaf $\eta^\frown 0^{\,t-t'} \in 2^t$; padding on the right preserves $\sigma^\frown b \sqsubseteq \eta$, and $2^{<t'} \subseteq 2^{<t}$ makes the threshold conditions $r_\sigma \in [0,1]$ inherited. □

Lemma 2 is what makes Definition 1 a threshold rather than a set of admissible heights: if height t forces a (k,δ)-ladder then so does every height $t' \geq t$, because a $(t', 2\delta)$-tree contains a $(t, 2\delta)$-tree.

The next lemma is the only point at which the hypothesis $f \leq 1$ enters; every ladder built in this paper is built through it.

Lemma 3 (Assembly)**.** *Let $f\colon X \times Y \to \mathbb{R}$ satisfy $f \leq 1$, let $\delta \geq 0$, and let $x_1, \dots, x_k$, $y_1, \dots, y_k$ and real numbers $r_1, \dots, r_k \geq 0$ satisfy $f(x_i, y_j) \geq r_i + \delta$ for $i \leq j$ and $f(x_i, y_j) \leq r_i$ for $i > j$. Then these data form a (k,δ)-ladder; in particular $r_i \leq 1$ for every i.*

Proof. Only $r_i \in [0,1]$ needs checking, and $r_i \geq 0$ is assumed. Taking $j = i$ gives $r_i + \delta \leq f(x_i, y_i) \leq 1$, so $r_i \leq 1 - \delta \leq 1$. □

Remark 4. Lemma 3 uses $f \leq 1$ and never $f \geq 0$; accordingly Theorems 11 and 12, which are stated for $f\colon X \times Y \to [0,1]$ as in [1], hold verbatim for any real-valued f bounded above by 1. The lower endpoint of the codomain is not needed there because the thresholds r_σ of the tree are already required to be nonnegative, and every ladder threshold produced below is built from those.

3. Two obstructions

Both lower bounds in this paper, and the reproduction of the source's Proposition 3.4, come from one of the following two observations. The first counts leaves in a fixed row; the second counts rows.

Lemma 5. *Let $\delta > 0$ and let $x_1, \dots, x_k$, $y_1, \dots, y_k$, $r_1, \dots, r_k$ be a (k,δ)-ladder for f. Then $y_1, \dots, y_k$ are pairwise distinct.*

Proof. Suppose $y_i = y_j$ with $i < j$. From $i < j$ we get $f(x_j, y_i) \leq r_j$, and from $j \leq j$ we get $f(x_j, y_j) \geq r_j + \delta$; since $y_i = y_j$ these give $r_j + \delta \leq r_j$, contradicting $\delta > 0$. □

Lemma 6. *Let $x_1, \ldots, x_k, y_1, \ldots, y_k, r_1, \ldots, r_k$ be a (k, δ)-ladder for f, with $k \geq 1$. Then $f(x_1, y_j) \geq \delta$ for every j.*

Proof. $f(x_1, y_j) \geq r_1 + \delta \geq \delta$, using $1 \leq j$ and $r_1 \geq 0$. □

Lemma 6 is the only use in this paper of the lower endpoint of the interval $[0, 1]$ in Definition 1.4(1), and it uses it only at ladder position 1. Combining the two lemmas:

Lemma 7 (Row-count obstruction)**.** *Let $k \geq 1$, $\delta > 0$ and $f\colon X \times Y \to \mathbb{R}$. Suppose that no row of f is at least δ at k pairwise distinct points: that is, for every $a \in X$ and every injective $z\colon \{1, \ldots, k\} \to Y$ there is a j with $f(a, z_j) < \delta$. Then f admits no (k, δ)-ladder.*

Proof. Given a ladder, apply the hypothesis to $a = x_1$ and to the map $j \mapsto y_j$, which is injective by Lemma 5. It returns a j with $f(x_1, y_j) < \delta$, contradicting Lemma 6. □

The second obstruction lives on the $\{0, 1\}$-valued functions, that is, on relations viewed inside the real-valued setting.

Lemma 8. *Let $\delta > 0$ and let $f\colon X \times Y \to \mathbb{R}$ take only the values 0 and 1. In every (k, δ)-ladder for f one has $f(x_i, y_j) = 1$ for $i \leq j$ and $f(x_i, y_j) = 0$ for $i > j$; consequently $x_1, \ldots, x_k$ are pairwise distinct.*

Proof. For $i \leq j$, $f(x_i, y_j) \geq r_i + \delta \geq \delta > 0$, so the value is 1. For $i > j$, $f(x_i, y_j) \leq r_i \leq f(x_i, y_i) - \delta \leq 1 - \delta < 1$, so the value is 0. If now $x_i = x_j$ with $i < j$, then $f(x_j, y_i) = 0$ because $j > i$, while $f(x_i, y_i) = 1$; these contradict each other. □

Lemma 9 (Cardinality obstruction)**.** *Let $\delta > 0$, let X be finite and let $f\colon X \times Y \to \mathbb{R}$ take only the values 0 and 1. If $|X| < k$ then f admits no (k, δ)-ladder.*

Proof. Immediate from the injectivity of $i \mapsto x_i$ in Lemma 8. □

Remark 10. The row-distinctness of Lemma 8 genuinely needs the $\{0, 1\}$ hypothesis. For real-valued f a ladder may repeat a row: with $f(x, y_1) = \delta$, $f(x, y_2) = 2\delta$, $r_1 = 0$ and $r_2 = \delta$ the data $x_1 = x_2 = x$ form a $(2, \delta)$-ladder. The leaves, by contrast, are always distinct (Lemma 5).

4. The upper bounds: height 2 at $k = 2$, height 3 at $k = 3$

Both arguments live inside the right-hand branch of the tree. The root supplies the ladder's first row — every leaf below $\varnothing^\frown 1$ has value at least $r_\varnothing + 2\delta$, so the constant threshold $r_\varnothing + \delta$ works against all of them at once — and the deeper nodes on that branch supply the remaining rows. Nothing outside the 1-cone of the root is used. Throughout this section we write $R_\sigma = r_\sigma$ for the thresholds of the tree, to keep them apart from the thresholds of the ladder being built.

Theorem 11. *Let X, Y be sets, let $f\colon X \times Y \to [0, 1]$ and let $\delta > 0$. If f admits a $(2, 2\delta)$-tree, then f admits a $(2, \delta)$-ladder. Consequently $T(2) \leq 2$, where Theorem 1.11 of* [1] *gives $\binom{4}{2} - 1 = 5$.*

Proof. Let $(x_\sigma),(y_\eta),(R_\sigma)$ be a $(2,2\delta)$-tree, and write the ladder to be built as $(u_i),(v_i),(s_i)$ so that its entries are not confused with the tree's. Consider the two leaves below the node 1, and set

$$u_1 = x_\varnothing, \quad u_2 = x_1, \qquad v_1 = y_{10}, \quad v_2 = y_{11}, \qquad s_1 = R_\varnothing + \delta, \quad s_2 = R_1.$$

The four ladder inequalities are then read off the tree. Since $\varnothing^\frown 1 \sqsubseteq 10$ and $\varnothing^\frown 1 \sqsubseteq 11$,

$$s_1 + \delta = R_\varnothing + 2\delta \le f(x_\varnothing, y_{10}), \qquad s_1 + \delta = R_\varnothing + 2\delta \le f(x_\varnothing, y_{11}),$$

which are the two inequalities of the first row. Since $1^\frown 0 \sqsubseteq 10$ we have $f(x_1, y_{10}) \le R_1 = s_2$, the inequality below the diagonal; and since $1^\frown 1 \sqsubseteq 11$ we have $s_2 + \delta \le R_1 + 2\delta \le f(x_1, y_{11})$. Finally $s_1 = R_\varnothing + \delta \ge 0$ and $s_2 = R_1 \ge 0$ because the tree's thresholds are nonnegative, so Lemma 3 applies and delivers a $(2,\delta)$-ladder. ☐

At $k = 3$ the tree of height 3 has four leaves below the node 1; write

$$a = y_{100}, \qquad b = y_{101}, \qquad c = y_{110}, \qquad d = y_{111},$$

The constraints that Definition 1.2 imposes on the four nodes of the 1-cone, against these four leaves, are exactly:

	a	b	c	d
$x_\varnothing$	$\ge R_\varnothing + 2\delta$	$\ge R_\varnothing + 2\delta$	$\ge R_\varnothing + 2\delta$	$\ge R_\varnothing + 2\delta$
x_1	$\le R_1$	$\le R_1$	$\ge R_1 + 2\delta$	$\ge R_1 + 2\delta$
x_{10}	$\le R_{10}$	$\ge R_{10} + 2\delta$	free	free
x_{11}	free	free	$\le R_{11}$	$\ge R_{11} + 2\delta$

Four entries are left unconstrained by the tree, and the proof splits on three of them.

Theorem 12. *Let X, Y be sets, let $f\colon X \times Y \to [0,1]$ and let $\delta > 0$. If f admits a $(3,2\delta)$-tree, then f admits a $(3,\delta)$-ladder. Consequently $T(3) \le 3$, where Theorem 1.11 of* [1] *gives $\binom{6}{3} - 1 = 19$.*

Proof. Fix a $(3,2\delta)$-tree and use the notation above. In every case the first row of the ladder is $x_\varnothing$ with threshold $R_\varnothing + \delta$; since all four of a,b,c,d lie below $\varnothing^\frown 1$, the first line of the table gives $R_\varnothing + 2\delta \le f(x_\varnothing, \cdot)$ at each of them, which is the whole first row of the ladder whichever three of the four leaves are used and in whichever order. Also, all four thresholds $R_\varnothing, R_1, R_{10}, R_{11}$ are nonnegative, so every threshold constructed below is nonnegative and Lemma 3 applies at the end of each case.

Case A: $R_{10} + \delta \le f(x_{10}, c)$. Take the leaves (a,b,c) in this order, the rows $(x_\varnothing, x_{10}, x_1)$ and the thresholds $(R_\varnothing + \delta,\ R_{10},\ R_1)$. Row 2: $f(x_{10}, a) \le R_{10}$ from the table; $R_{10} + \delta \le R_{10} + 2\delta \le f(x_{10}, b)$; and $R_{10} + \delta \le f(x_{10}, c)$ is the case hypothesis. Row 3: $f(x_1, a) \le R_1$ and $f(x_1, b) \le R_1$ from the table, and $R_1 + \delta \le R_1 + 2\delta \le f(x_1, c)$.

In the remaining cases $f(x_{10}, c) < R_{10} + \delta$.

Case B1: $f(x_{11}, a) \le R_{11} + \delta$. Take the leaves (a,c,d), the rows $(x_\varnothing, x_1, x_{11})$ and the thresholds $\big(R_\varnothing + \delta,\ R_1,\ m\big)$ where $m = \max\{R_{11}, f(x_{11}, a)\}$. Row 2: $f(x_1, a) \le R_1$, and $R_1 + \delta \le R_1 + 2\delta \le f(x_1, c), \le f(x_1, d)$. Row 3: $f(x_{11}, a) \le m$ by definition of m; $f(x_{11}, c) \le R_{11} \le m$; and, since the case hypothesis gives $m \le R_{11} + \delta$,

$$m + \delta \le R_{11} + 2\delta \le f(x_{11}, d).$$

Case B2: $f(x_{11},b) \le R_{11}+\delta$. Identical to Case B1 with b in place of a: the leaves are (b,c,d), the rows $(x_\varnothing, x_1, x_{11})$, the thresholds $\big(R_\varnothing+\delta,\ R_1,\ \max\{R_{11}, f(x_{11},b)\}\big)$, and the only table entry used in row 2 that differs is $f(x_1,b) \le R_1$.

Case B3: $f(x_{11},a) > R_{11}+\delta$ *and* $f(x_{11},b) > R_{11}+\delta$. Take the leaves (c,a,b) in this order, the rows $(x_\varnothing, x_{11}, x_{10})$ and the thresholds $\big(R_\varnothing+\delta,\ R_{11},\ m'\big)$ where $m' = \max\{R_{10}, f(x_{10},c)\}$. Row 2: $f(x_{11},c) \le R_{11}$ from the table, and the two case hypotheses are exactly $R_{11}+\delta \le f(x_{11},a)$ and $R_{11}+\delta \le f(x_{11},b)$. Row 3: $f(x_{10},c) \le m'$ by definition; $f(x_{10},a) \le R_{10} \le m'$; and since both $R_{10} < R_{10}+\delta$ and, in this case, $f(x_{10},c) < R_{10}+\delta$, we get $m' < R_{10}+\delta$ and hence

$$m' + \delta < R_{10} + 2\delta \le f(x_{10},b). \qquad \square$$

Remark 13. The four cases correspond to four ordered triples of distinct leaves, namely (a,b,c), (a,c,d), (b,c,d) and (c,a,b). They were not found by inspection: they are the four triples that survive in a minimised unsatisfiable core of the difference-logic encoding described in Section 7.1, out of the $8\cdot 7\cdot 6 = 336$ ordered triples of distinct leaves of a height-3 tree. The proof above uses no fact about the solver; the search only pointed at which four configurations to try.

5. The lower bounds, and the exact values

One construction handles every k. For $d \ge 1$ let

$$X = \{1,\dots,d\}, \qquad Y = \{0,1\}^X, \qquad f_d(a,b) = b(a),$$

the class of all Boolean concepts on d points, evaluated. Its Littlestone dimension and its threshold dimension are both d; the content of the next theorem is that this standard example answers the present question, and that the row-distinctness step is what makes it work.

Theorem 14. *For every $d \ge 1$ the function f_d takes values in $\{0,1\}$, admits a $(d,1)$-tree, and admits no $(d+1,\delta)$-ladder for any $\delta > 0$. Consequently*

$$T(k) \ \ge\ k \qquad \textit{for every } k \ge 1.$$

Proof. For the tree, put $x_\sigma = |\sigma| + 1$ at each node $\sigma \in 2^{<d}$, let the leaf $\eta \in 2^d$ index the concept y_η whose value at the point i is the i-th bit of η, and take $r_\sigma = 0$ throughout. If $\sigma^\frown 0 \sqsubseteq \eta$ then the $(|\sigma|+1)$-st bit of η is 0, so $f_d(x_\sigma, y_\eta) = 0 \le r_\sigma$; if $\sigma^\frown 1 \sqsubseteq \eta$ then that bit is 1, so $f_d(x_\sigma, y_\eta) = 1 = r_\sigma + 1$. This is a $(d,1)$-tree, that is, a $(d,2\delta)$-tree for $\delta = \frac{1}{2}$.

For the ladder, $|X| = d < d+1$, so Lemma 9 applies.

Finally, fix $k \ge 1$ and take $d = k-1$ (for $k = 1$ there is nothing to prove, as $T(k) \ge 1$ by definition). Then f_{k-1} admits a $(k-1, 2\delta)$-tree at $\delta = \frac{1}{2}$ and no $(k, \frac{1}{2})$-ladder, so $k-1$ is not an admissible height and $T(k) \ge k$. $\square$

Corollary 15. *A $(1,2\delta)$-tree need not force a $(2,\delta)$-ladder: the class of all Boolean concepts on one point admits a $(1,1)$-tree and admits no $(2,\frac{1}{2})$-ladder. With Theorem 11,*

$$T(2) = 2.$$

Proof. The case $d = 1$ of Theorem 14 gives $T(2) > 1$, and Theorem 11 gives $T(2) \le 2$. $\square$

Corollary 16. *A $(2, 2\delta)$-tree need not force a $(3, \delta)$-ladder: the class of all Boolean concepts on two points admits a $(2, 1)$-tree and admits no $(3, \frac{1}{2})$-ladder. With Theorem 12,*

$$T(3) = 3.$$

Proof. The case $d = 2$ of Theorem 14 gives $T(3) > 2$, and Theorem 12 gives $T(3) \leq 3$. $\square$

Remark 17. Corollaries 15 and 16 say in particular that neither upper bound of Section 4 can be lowered, so the case analysis of Theorem 12 is not an artefact of the proof: at height 2 the conclusion is false. The same witness f_2 refutes every longer ladder as well, since a (k, δ)-ladder with $k \geq 3$ contains a $(3, \delta)$-ladder by Lemma 2.

6. The scale in Theorem 1.11 cannot be relaxed

Theorem 1.11 of [1] converts a tree of scale 2δ into a ladder of scale δ, and its Proposition 3.4 shows that this loss is necessary: for $0 < \delta \leq \frac{1}{4}$, every $t \geq 1$ and every $0 < \alpha < 2\delta$ there is a function admitting a uniform (t, α)-tree and no $(3, \delta)$-ladder. Their function on $X = 2^{<t}$, $Y = 2^t$ is

$$f(\sigma, \eta) = \begin{cases} 0 & \sigma^\frown 0 \sqsubseteq \eta, \\ \alpha & \sigma^\frown 1 \sqsubseteq \eta, \\ \alpha/2 & \text{otherwise.} \end{cases}$$

We record the instance $t = 2$, for all admissible parameters at once and by a different argument, namely the row-count obstruction of Lemma 7. The statement is theirs; only the proof and the parameter range differ.

Proposition 18 (Conant–Terry, Proposition 3.4, at height 2)**.** *Let $0 < \alpha \leq 1$ and $\delta > 0$ with $\alpha < 2\delta$, and let f be the function above on $X = 2^{<2}$, $Y = 2^2$. Then f takes values in $[0, 1]$, admits a $(2, \alpha)$-tree with all thresholds equal to 0, and admits no $(3, \delta)$-ladder.*

Proof. The values $0, \alpha/2, \alpha$ lie in $[0, 1]$ because $0 < \alpha \leq 1$. For the tree, take $x_\sigma = \sigma$, $y_\eta = \eta$ and $r_\sigma = 0$: if $\sigma^\frown 0 \sqsubseteq \eta$ then $f(\sigma, \eta) = 0 \leq 0$; if $\sigma^\frown 1 \sqsubseteq \eta$ then $f(\sigma, \eta) = \alpha = 0 + \alpha$. (The two cases are mutually exclusive, so f is well defined; on a height-2 tree this is a finite check over the three nodes and four leaves.)

For the ladder, we verify the hypothesis of Lemma 7 with $k = 3$. Fix a node σ and three pairwise distinct leaves. A value of $f(\sigma, \cdot)$ is at least δ only if it equals α, since $0 < \delta$ and $\alpha/2 < \delta$ by hypothesis; and $f(\sigma, \eta) = \alpha$ requires $\sigma^\frown 1 \sqsubseteq \eta$, which at height 2 holds for at most two leaves η — two when $\sigma = \varnothing$ and one when $|\sigma| = 1$. So among any three distinct leaves at least one has $f(\sigma, \cdot) < \delta$, and Lemma 7 applies. $\square$

Remark 19. The source proves its Proposition 3.4 at every height $t \geq 1$, for $0 < \delta \leq \frac{1}{4}$ and $0 < \alpha < 2\delta$, by a cone-intersection argument. The statement above is restricted to $t = 2$ but holds for every $\delta > 0$ with $\alpha < 2\delta$ and $\alpha \leq 1$; the reason is that the row-count argument does not look at the height at all, only at how many leaves a single node can score high.

7. Computations, and what they do and do not settle

Nothing in this section is a theorem, and nothing in it is part of the formal development. It is reported because the route is reusable, because the height-4 frontier has a measured price, and because one of the computations changes how the source's own proof should be read.

7.1. **The route.** Deciding whether a height is admissible is an $\exists\forall$ problem over the reals, but it collapses to a decidable fragment in two steps.

Rows separate. Call $x \in X$ *i-good* for a tuple $(y_1, \ldots, y_k)$ if some $r \in [0,1]$ has $f(x, y_j) \leq r$ for $j < i$ and $f(x, y_j) \geq r + \delta$ for $j \geq i$; equivalently, taking r to be the left-hand maximum,

$$\max\Big\{0,\ \max_{j<i} f(x, y_j)\Big\} + \delta \ \leq\ \min_{j \geq i} f(x, y_j).$$

Then f admits a (k, δ)-ladder if and only if there is a tuple $(y_1, \ldots, y_k)$ such that for every i some x is i-good for it, because once the leaves are fixed the rows are chosen independently. This removes a factor $|X|^k$ from the search, and by Lemma 5 only tuples of pairwise distinct leaves need be enumerated.

Difference logic. Normalising $\delta = 1$ and replacing the codomain $[0,1]$ by $[0,\infty)$ costs nothing: a counterexample rescales by $1/\delta$, and dividing by $\max f$ returns it to $[0,1]$, the ladder condition $r_i \leq 1$ surviving because $r_i + \delta \leq f(x_i, y_i)$ and the tree condition $r_\sigma \leq 1$ surviving because r_σ may be taken to be the maximum over the 0-cone. What does not rescale away is $f \geq 0$ together with $r_1 \geq 0$, which is the content of Lemma 6 and acts only at position 1. After the normalisation every constraint has the form $u - v \leq c$ or $u - v < c$ with $c \in \{0, 1, 2\}$, so the question "is there a $(t, 2\delta)$-tree admitting no (k, δ)-ladder?" is a quantifier-free difference-logic formula: a conjunction over the distinct leaf tuples of disjunctions over i of conjunctions over the nodes.

The formula was given to an SMT solver [7]. A satisfying assignment at height t is a tree of height t with no (k, δ)-ladder, so it certifies $T(k) > t$; unsatisfiability certifies $T(k) \leq t$. The instances actually run were:

t	k	variables	leaf tuples	verdict	wall
1	2	2	2	satisfiable	0.4 s
2	2	12	12	unsatisfiable	0.2 s
3	2	56	56	unsatisfiable	0.7 s
2	3	12	24	satisfiable	0.04 s
3	3	56	336	unsatisfiable	2.4 s
3	4	56	1 680	satisfiable	5.5 s
4	4	240	43 680	no verdict	2 787 s

Every verdict in this table that supports a statement of Sections 4 and 5 has been replaced by a proof: the satisfiable rows by the explicit witnesses of Theorem 14, the unsatisfiable rows at $k = 2, 3$ by the arguments of Theorems 11 and 12. The encoding itself was checked in two independent ways before any of it was believed: the row-separation reformulation was compared against a direct search over all triples $(\vec{x}, \vec{y}, \vec{r})$ on 60 random rational tables with $|X|, |Y| \leq 3$ and $k \leq 3$, agreeing in every case, and the encoding was run against the class of all Boolean concepts on $d \leq 3$ points computed from Definitions 1.2 and 1.4(1) verbatim, reproducing $T(k) \geq k$ at those d. The source's own published numbers were reproduced first, as a gate: the recursion and the closed form of the bound $B(k, \ell)$ of §7.3, the values

$\binom{2k}{k} - 1 = 1, 5, 19, 69, 251$, and its Proposition 3.4 at $t = 1, 2, 3$ for three admissible pairs (δ, α).

The only place the solver enters the mathematics of this paper is Remark 13: a minimised unsatisfiable core of the row $t = k = 3$ named the four leaf tuples that became the four cases of Theorem 12.

7.2. **The height-4 wall.** $T(4)$ is not decided here. By Theorem 14 it is at least 4, and the last row of the table above is the instance that would decide whether it equals 4: 240 real variables, $16 \cdot 15 \cdot 14 \cdot 13 = 43\,680$ tuples of distinct leaves and about 2.6 million constraint groups, roughly 55 times the height-3 instance at $k = 4$ that is decided in 5.5 seconds. It returned no verdict in 46 minutes and was stopped. So were three further height-4 encodings of the same question: the variant in which the constraint $r_1 \geq 0$ is dropped (stopped at its 3 600 s timeout), the auxiliary statement of §7.3 at $(k, \ell) = (3, 4)$ (killed as the tail of an enclosing 1 690 s run, after the cell $(3, 3)$ had landed), and the purely discrete version over relations (stopped at 3 623 s). Peak solver memory was 210 MB, so the obstacle is the size of the search and not the memory; total solving for this work was about 3.6 hours, of which these four runs account for about 3.3. Height 3 is decided in seconds by all four encodings and height 4 by none of them, which is a fact about the instrument and not evidence about the answer. Two auxiliary quantities were measured in the hope of a shortcut and did not provide one: dropping $r_1 \geq 0$ gives a threshold T_f with $T_f(2) = 1$, $T_f(3) = 2$, $T_f(4) \geq 4$, and since $T(k) \leq T_f(k) + 1$ always — restrict the leaves to the root's 1-cone and let the root supply position 1 — the value $T_f(4) \geq 4$ yields nothing beyond $T(4) \geq 4$; and the discrete thresholds, over relations rather than $[0, 1]$-valued functions, were computed as $D(2) = 2$, $D(3) = 3$, $D(4) \geq 4$ by a Boolean encoding written independently, with every satisfying assignment re-verified by exhaustive search.

7.3. **The source's auxiliary threshold, and a limit on its induction.** The proof of Theorem 1.11 in [1] does not go through $T(k)$ directly. It proves a two-parameter statement, which it calls $P(k, \ell, t)$, asserting that every $(t, 2\delta)$-tree satisfies one of two alternatives: a configuration of length k of one shape, or a configuration of length ℓ of a second shape with the strict inequalities $< r + \delta$ and $\geq r + 2\delta$. Writing $T_{\mathrm{CT}}(k, \ell)$ for the least t for which $P(k, \ell, t)$ holds — a different quantity from our $T(k)$, and the source uses the same letter for it — the source proves

$$T_{\mathrm{CT}}(k, 1) = T_{\mathrm{CT}}(1, \ell) = 1, \qquad T_{\mathrm{CT}}(k, \ell) \leq T_{\mathrm{CT}}(k-1, \ell) + T_{\mathrm{CT}}(k, \ell - 1) + 1,$$

hence $T_{\mathrm{CT}}(k, \ell) \leq B(k, \ell) = 2\binom{k+\ell-2}{k-1} - 1$, and Theorem 1.11 is the case $\ell = k + 1$, where $B(k, k+1) = \binom{2k}{k} - 1$. None of this is a numbered statement of [1]: $P(k, \ell, t)$ and $T_{\mathrm{CT}}(k, \ell)$ are introduced in the body of the proof of Theorem 1.11 (its §3); the two displayed facts are parts (a) and (b) of that proof's Claim 1, which is local to the proof and not one of the paper's numbered statements, and part (b) carries the hypothesis that $T_{\mathrm{CT}}(k-1, \ell)$ and $T_{\mathrm{CT}}(k, \ell - 1)$ exist; the passage to $B(k, \ell)$ is the unnumbered paragraph immediately after the proof of that claim. In particular $T(k) \leq T_{\mathrm{CT}}(k, k+1)$.

The statement $P(k, \ell, t)$ has the same shape as the ladder condition — its rows separate once the leaf map is fixed, and only injective leaf maps matter — so the same encoding computes T_{CT}. The column $\ell = 1$ and the row $k = 1$ reproduce the source's own values, which is the gate for the rest of the table:

$T_{\mathrm{CT}}(k,\ell)$	$\ell=1$	$\ell=2$	$\ell=3$	$\ell=4$
$k=1$	1 $(B=1)$	1 $(B=1)$	1 $(B=1)$	1 $(B=1)$
$k=2$	1 $(B=1)$	2 $(B=3)$	2 $(B=5)$	2 $(B=7)$
$k=3$	1 $(B=1)$	2 $(B=5)$	4 $(B=11)$	≥ 4 $(B=19)$

The cell $(3,4)$ is the one that matters here. $P(3,4,3)$ is satisfiable — there is a height-3 tree violating both alternatives — so $T_{\mathrm{CT}}(3,4)\geq 4$, while $T(3)=3$ by Corollary 16. The route $T(3)\leq T_{\mathrm{CT}}(3,4)$ therefore cannot produce the true value: *the induction of* [1] *bottoms out one level above the answer at* $k=3$, and any improvement of Theorem 1.11 down to $t=k$ has to leave that scheme. This is the reason the argument of Theorem 12 is a direct case analysis rather than an instance of the source's recursion.

One inequality in the table is not merely measured. The same witness as in Theorem 14 gives $T_{\mathrm{CT}}(k,\ell)\geq\min(k,\ell)$: in either alternative, a repetition in the index map forces a single pair (x_σ, r_σ) to satisfy two contradictory inequalities, so the pairs used are pairwise distinct, and the canonical height-d tree of the class of all Boolean concepts on d points offers only d of them. We state this as an observation rather than as a theorem: it is not part of the formal development.

7.4. **A conjecture from eleven values.** Every computed cell of the table above — eleven exact values, the twelfth being the bound $T_{\mathrm{CT}}(3,4)\geq 4$, which is consistent with it — satisfies $T_{\mathrm{CT}}(k,\ell)=2^{\min(k,\ell)-1}$.

Conjecture 20. $T_{\mathrm{CT}}(k,\ell)=2^{\min(k,\ell)-1}$ *for all* $k,\ell\geq 1$.

We stress what this is: a pattern fitted to eleven numbers, not a result, and it is not part of the formal development. It is worth recording because of what its upper half would give. Taking $\ell=k+1$ yields $T_{\mathrm{CT}}(k,k+1)=2^{k-1}$, so $T(k)\leq 2^{k-1}$ and Theorem 1.11 of [1] would hold with $t=2^{k-1}$ — which is exactly "on the order of 2^k", an affirmative answer to the source's Question 3.3, obtained inside the source's own induction rather than around it. The recursion the source proves is consistent with the conjecture but far from forcing it: at $(k,\ell)=(3,3)$ it gives $2+2+1=5$ against the computed 4, and its closed form gives $B(3,3)=11$. The next cells that would test it, $T_{\mathrm{CT}}(3,4)$ and $T_{\mathrm{CT}}(4,4)$ — predicted 4 and 8 — lie at heights 4 and 8, and §7.2 says that the first of them is already out of reach of this instrument.

7.5. **What is open.** Whether $T(4)=4$; whether $T(k)=k$ for every k, which is proved here for $k\leq 3$ and bounded below at every k by Theorem 14; Conjecture 20, whose upper half would settle Question 3.3 of [1]; and the uniform version of all of this, in which the ladder thresholds r_i are required to be equal. The last is a genuinely different question: the ladders produced by Theorems 11 and 12 are non-uniform, so those theorems say nothing about the uniform threshold, and the condition defining it is not quantifier-free, so the route of §7.1 does not apply to it without a quantifier elimination or a split on critical values.

8. Verification

Every theorem, proposition, lemma and corollary of Sections 2–6 is formalised in Lean 4 [5] against Mathlib [6] and checked by the Lean kernel: Definitions 1.2 and 1.4(1) transcribed as stated in Section 2, the assembly lemma, the two obstructions, monotonicity, the two upper bounds with their four-case analysis, the class of all Boolean concepts with its tree and its ladder-freeness, and the height-2 instance of the source's Proposition 3.4. The threshold $T(k)$ of Definition 1 quantifies over all pairs of types and is not itself a definition in the development: what is formalised in each of the two exact values is the pair of statements whose conjunction fixes it, the implication of Section 4 and the witness of Section 5. Mathlib carries the Vapnik–Chervonenkis dimension of a set family but no Littlestone-dimension or ladder machinery, so the two definitions are transcribed from [1] directly rather than instantiated from a library notion. What is *not* formalised is the whole of Section 7: the measured thresholds, the table of solver runs, the values of T_{CT}, the lower bound $T_{\mathrm{CT}}(k,\ell) \geq \min(k,\ell)$, Conjecture 20 and the remark that $T(1)=1$. The toolchain is `leanprover/lean4:v4.32.0` and Mathlib is taken at revision `81a5d257c8e410db227a6665ed08f64fea08e997`. The development is about 640 lines in three modules, which compile in 169, 74 and 68 seconds respectively at a peak of 6.8 GB, measured on a heavily loaded machine and therefore to be read as upper bounds. All thirteen headline declarations depend on exactly the three axioms `propext`, `Classical.choice` and `Quot.sound` of Lean's standard foundation; there is no `sorry`, and *no use of* `native_decide` *occurs anywhere*, so neither a compiler nor a runtime is part of the trusted base. The only decision procedure used is `decide`, evaluated by the kernel, and it is used only in the proof of Proposition 18, on four finite facts about the 3×4 node/leaf incidence table of a height-2 tree — six `decide` calls in all, four of them after a reduction to concrete strings. Four negative controls are formalised alongside the theorems, to guard against statements that hold for the wrong reason: that height 2 does *not* force a $(3,\delta)$-ladder and height 1 does *not* force a $(2,\delta)$-ladder, so neither upper bound can be lowered (Corollaries 15 and 16); that the same height-2 witness refutes $(k,\frac{1}{2})$-ladders for every $k \geq 3$; that the hypothesis of the upper bounds is not vacuous, the class of all Boolean concepts on three points having a height-3 tree; and that it is not automatic either, a constant function admitting no $(1,s)$-tree for any $s > 0$. The repository is private at the time of writing; repository reference to be added at submission.

Acknowledgement of provenance

The objects of this paper are not ours. Definition 1.2, Definition 1.4(1), Theorem 1.11 with its bound $\binom{2k}{k}-1$, Proposition 3.4, Remark 3.2 and the Question 3.3 that motivates everything above are due to Conant and Terry [1], as is the observation that the non-uniform ladder of Definition 1.4(1) appears to be new with that paper; the discrete tree/ladder correspondence behind all of it is due to Shelah and to Hodges [2], quoted here through [1]. What is offered here is the exact values $T(2)=2$ and $T(3)=3$, the general lower bound $T(k) \geq k$, the height-2 proof of Proposition 3.4 by a different mechanism, the observation that the source's own induction cannot reach $T(3)=3$, and Conjecture 20 with the eleven values behind it.

References

[1] G. Conant and C. Terry, *Encoding orders and trees in real-valued functions*, 27 pages, `arXiv:2607.21761v1` (23 July 2026); primary math.CO, cross-listed cs.LG and math.LO. Rechecked on 3 September 2026 against the arXiv abstract page and the arXiv API: v1 is still the only version and there is still no journal reference, so every definition, theorem, corollary, remark, question and proposition number cited above refers to that version. The numbers were not taken from a summary: the e-print was compiled here and they were read off the resulting

document — Definition 1.2 (internal label `def:eptree1`), Definition 1.4 (`def:ladders`), Theorem 1.6 (`thm:hodges`), Theorem 1.11 (`thm:hodgesfn1`), Corollary 3.1 (`cor:hodgesfn1uni`), Remark 3.2 (`rem:2kkbound`), Question 3.3, Proposition 3.4 (`prop:GPT1`), and, in the paper's Appendix B, Definition B.1 (`def:Ldim`) and Definition B.2 (`def:threshdim`). In particular the part label in "Theorem 1.6(2)" is read off the compiled theorem, whose part (2) is the sentence quoted in §1.2; the paper carries appendices A, B and C, so its last section-numbered environment is in §5 and there is no Definition 7.1.

[2] W. Hodges, *Encoding orders and trees in binary relations*, Mathematika **28** (1981), no. 1, 67–71, `doi:10.1112/S0025579300015357`. This is the reference [1] gives for its Theorem 1.6, jointly with Shelah; [1] cites it without an internal theorem number, and Hodges' paper was not consulted directly for the present work, so the statement quoted in §1.2 is quoted from [1]. ([3] attributes the same discrete correspondence instead to Hodges' later book *A shorter model theory*, Cambridge University Press, 1997.)

[3] A. Blondal, S. Gao, H. Hatami and P. Hatami, *Stability and list-replicability for agnostic learners*, `arXiv:2501.05333v3` (16 May 2025; v1 9 January 2025), cs.LG. Consulted for the published state of the discrete correspondence, which its Proposition 3.3 states verbatim as: "If $\mathcal{H} \subseteq \{0,1\}^X$ has $\mathrm{Ldim}(\mathcal{H}) = d$, its threshold dimension is at least $\lfloor \log d \rfloor$." It attributes this to Hodges' 1997 book and refers, for an accessible proof, to N. Alon, R. Livni, M. Malliaris and S. Moran, *Private PAC learning implies finite Littlestone dimension*, STOC'19 — Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, ACM, New York, 2019, 852–860. The proposition number is that of v3 and could move in a later version.

[4] A. Assos, I. Attias, Y. Dagan, C. Daskalakis and M. K. Fishelson, *Online learning and solving infinite games with an ERM oracle*, Proceedings of the Thirty Sixth Conference on Learning Theory (G. Neu and L. Rosasco, eds.), Proceedings of Machine Learning Research, vol. 195, PMLR, 2023, 274–324; `arXiv:2307.01689`. Cited here only for the term *fat-threshold dimension*, which it defines for a real-valued class with a single threshold shared by all rows — the feature that makes it the uniform notion.

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

[6] 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, 367–381.

[7] L. de Moura and N. Bjørner, *Z3: an efficient SMT solver*, in: Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2008), Lecture Notes in Computer Science **4963**, Springer, 2008, 337–340. Used through the `z3-solver` Python distribution, release `5.1.0.0` (16 August 2026), whose bundled library reports version 5.1.0.0 through `Z3_get_version` and the string `5.1.0` through `get_version_string`; single-threaded, for every computation reported in Section 7.