4 Fractional Dimension of Posets from Trees

In this last chapter, we switch gears a little bit, and fractionalize the dimension of posets. We start with a few simple definitions to develop the language of posets. A binary relation on a set $X$ is called a partial order if it is reflexive, antisymmetric, and transitive. A partially ordered set (or “poset”) $P = (X, \leq)$ consists of some ground set $X$ and a partial order $\leq$ on $X$. A pair of elements $x, y \in X$ are incomparable if $x \not\leq y$ and $y \not\leq x$. Such an incomparable (ordered) pair $(x, y)$ is a critical pair if, for all $a, b \in X$ such that $a \leq x$ and $y \leq b$ (but not $(a, b) = (x, y)$), we have $a \leq b$. The idea of critical pairs plays an important role in this chapter, so we will let $C(P)$ (or just $C$) denote the set of critical pairs of $P$.

We may also think of $\leq$ as a subset of $X \times X$. A partial order $L$ on $X$ is a total order if for any $a, b \in X$, either $(a, b) \in L$ or $(b, a) \in L$. A total order $L$ is a linear extension of $\leq$ if $\leq$ is a subset of $L$. A realizer of a poset $P$ is a collection of linear extensions $R = \{L_1, \ldots, L_t\}$ of $\leq$ whose intersection is $\leq$. That is, for any incomparable pair $x, y \in X$, there are $L_i, L_j \in R$ with $(x, y) \in L_i$ and $(y, x) \in L_j$. Finally, the dimension of $P$ is defined as the size of the smallest realizer of $P$, and is denoted $\dim(P)$.$^{21}$

It is easily shown$^{22}$ that a collection of linear extensions $\{L_1, \ldots, L_t\}$ is a realizer if and only if, for every critical pair $(x, y)$, there is some $L_i$ containing $(y, x)$. Even though $x$ and $y$ are incomparable, the critical pair $(x, y)$ behaves as if $x \leq y$ relative to the rest of the poset, and so if $y < x$ in $L_i$, we say that $L_i$ reverses $(x, y)$. In other words, $\{L_1, \ldots, L_t\}$ is a realizer iff it reverses every critical pair.

$^{21}$The concept of dimension of a poset was first introduced by Dushnik and Miller [3].

$^{22}$See [14].
4.1 Fractional Dimension

We may now formulate dimension as an integer programming problem, just as we did chromatic number. For the poset \( P = (X, \leq) \), let \( \mathcal{L} = \{L_1, \ldots, L_m\} \) be the set of all linear extensions of \( \leq \), and \( C = \{c_1, \ldots, c_n\} \) the set of all critical pairs. Let \( M \) be the critical pair/linear extension incidence matrix, with rows indexed by \( C \) and columns indexed by \( \mathcal{L} \). The \( i, j \) entry is a 1 exactly when \( c_i \) is reversed in \( L_j \) (henceforth denoted by \( c_i \preceq L_j \)), and is 0 otherwise. Then, just as with ordinary/fractional chromatic number/clique number, we get pairs of integer and linear programs defining ordinary/fractional dimension and its dual parameter, which we denote \( \kappa(P) \):

\[
\begin{align*}
dim(P) &= \min \mathbf{1} \cdot \mathbf{x} \quad \text{s.t.} \quad M \mathbf{x} \geq \mathbf{1}, \quad \mathbf{x} \geq 0, \quad \mathbf{x} \in \mathbb{Z}^m \\
\kappa(P) &= \max \mathbf{1} \cdot \mathbf{y} \quad \text{s.t.} \quad M^T \mathbf{y} \leq \mathbf{1}, \quad \mathbf{y} \geq 0, \quad \mathbf{y} \in \mathbb{Z}^n \\
dim_f(P) &= \min \mathbf{1} \cdot \mathbf{x} \quad \text{s.t.} \quad M \mathbf{x} \geq \mathbf{1}, \quad \mathbf{x} \geq 0, \quad \mathbf{x} \in \mathbb{R}^m \\
\kappa_f(P) &= \max \mathbf{1} \cdot \mathbf{y} \quad \text{s.t.} \quad M^T \mathbf{y} \leq \mathbf{1}, \quad \mathbf{y} \geq 0, \quad \mathbf{y} \in \mathbb{R}^n
\end{align*}
\]

Feasible solutions to the LPs are referred to as fractional realizers and fractional critical pair packings (or, henceforth, simply “fractional packings”)\(^ {23} \), respectively. As before, \( \kappa(P) \leq \kappa_f(P) = \dim_f(P) \leq \dim(P) \) for any finite poset \( P \), and the middle “=” is only “\( \leq \)” if \( P \) is infinite.

Again, to facilitate our discussion, particularly with respect to infinite posets, we reframe these definitions. Define a fractional realizer of \( P \) to be a mapping \( f : \mathcal{L} \to [0, 1] \) such that for each \( c \in C \) we have \( \sum_{L \in \mathcal{L} : \rho(c) L} f(L) \geq 1 \). The weight of this realizer is

\(^{23} \)“Fractional critical pair packing” is, for lack of a better term, used in reference to the hypergraph covering/packing formulation presented in [11].
\[ w(f) = \sum_{L \in \mathcal{L}} f(L) \], and fractional dimension is

\[ \dim_f(P) = \inf \{ w(f) : f \text{ a fractional realizer of } P \} \].

Again, this definition matches the linear programming formulation when \( P \) is finite, but is still well-defined when \( P \) is infinite.

We similarly modify the definition of fractional packing number: a fractional packing of \( P \) is a mapping \( g : C \to [0,1] \) such that for each \( L \in \mathcal{L} \) we have \( \sum_{e \in L} g(e) \leq 1 \). The weight of this mapping is \( w(g) = \sum_{e \in C} g(e) \), and fractional packing number is

\[ \kappa_f(P) = \sup \{ w(g) : g \text{ a fractional packing of } G \} \].

which is just the dual linear program formulation if \( P \) is finite.

We also have a \( b \)-fold version of fractional dimension, as follows: a collection of linear extensions \( \{L_1, \ldots, L_i\} \) is a \( b \)-fold realizer if and only if every critical pair is reversed in (at least) \( b \) of the \( L_i \)'s. Then the \( b \)-fold dimension of \( P \), denoted \( \dim_b(P) \), is the size of the smallest \( b \)-fold realizer of \( P \). And, as before, we may also write \( \dim_f(P) = \lim_{b \to \infty} \frac{\dim_b(P)}{b} \). The proof that the two definitions agree is analogous to that for \( \chi_f \), even in the case of infinite posets. A more complete treatment of fractional dimension may be found in Brightwell and Scheinerman [2].

### 4.2 Posets of Graphs

We may derive a poset \( P(G) \) from a graph \( G \) as follows. The ground set is \( X = V(G) \cup E(G) \), and the only (non-equality) relations are of the form \( v < e \), where vertex \( v \) is an endpoint of edge \( e \). What are the critical pairs of this poset?
For \( u, v, w \in V(G) \), \( vw \in E(G) \), if \((u, vw)\) is a critical pair, then \( a \leq u \) and \( vw \leq b \) implies \( a \leq b \). But \( a \leq u \) implies \( a = u \), and \( vw \leq b \) implies \( vw = b \), so \((a, b) = (u, vw)\). So if \((u, vw)\) is an incomparable pair, there are no candidate \( a \) and \( b \), and \((u, vw)\) vacuously satisfies the definition of a critical pair. So any \((\text{vertex, edge})\) pair is a critical pair so long as the vertex isn’t in the edge.

For \( u, v \in V(G) \), if \((u, v)\) is a critical pair, then \( a \leq u \) and \( v \leq b \) implies \( a \leq b \). But \( a \leq u \) implies \( a = u \), so we require that \( v \leq b \) implies \( u \leq b \), that is, everything above \( v \) is also above \( u \). Since all edges incident to \( v \) are above \( v \), all these edges must also be above \( u \). This can only happen if \( v \) is a leaf (vertex of degree 1) and \( u \) the other end of \( v \)’s edge (\( u \) is henceforth referred to as a “branch”). This is the only possible type of \((\text{vertex, vertex})\) critical pair.

There are no \((\text{edge, vertex})\) critical pairs, for this would require each of the edge’s vertices to be \( \leq \) the critical pair vertex, which can’t happen. Also, there are no \((\text{edge, edge})\) critical pairs, for this would require both of the first edge’s vertices to be below the second edge, which only occurs if they are the same edge, and then not incomparable.

In order to establish the fractional dimension of one of our posets (or class of posets), we need to utilize the same approach we’ve seen before: bounding \( \dim_f \) from above with fractional realizers, and bounding it below with fractional packings. Of course, for any given poset of a graph, there are a huge number of distinct linear extensions that must be considered in building a fractional realizer\(^{24}\), so this is not a calculation we wish to undertake for most specific posets. We shall content ourselves with the limiting value of \( \dim_f \) for certain classes of trees (including general trees). This is facilitated by the

\(^{24}\text{We shall see that there are, roughly, } |V(G)|! \text{ maximal linear extensions.}\)
Lemma 4.1 Given graphs $G_1 \subseteq G_2$, it follows that

$$\dim_f(P(G_1)) \leq \dim_f(P(G_2)).$$

Proof. Any incomparable pair in $P(G_1)$ is still an incomparable pair in $P(G_2)$ (adding vertices and edges can’t add or alter the $v < uv$ relations between existing vertices and edges). So for an incomparable pair $(a, b)$ in $P(G_1)$, this pair must have $b < a$ in linear extensions of total weight at least 1 in any fractional realizer of $P(G_2)$, and this fact holds when these linear extensions are restricted to elements of $P(G_1)$. Thus any fractional realizer of $P(G_2)$, when restricted to elements and relations is $P(G_1)$, is also a fractional realizer of $P(G_1)$. Given this, our desired result is immediate. \qed

Because of this, we limit our attention to posets of arbitrarily large graphs, and simply prove tight upper bounds. For instance, showing that $\lim_{q \to \infty} \dim_f(P(S_q)) = 1 + \sqrt{2}$ (where $S_q$ is the $q$-star) proves that $1 + \sqrt{2}$ is a tight upper bound on $\dim_f$ for the posets of all finite stars.

A maximal linear extension is one for which the set of critical pairs it reverses is not a proper subset of the set of critical pairs reversed by any other linear extension. Just as we only needed to consider maximal independent sets for the fractional coloring problem (see Lemma 1.2), we need only consider maximal linear extensions for the fractional dimension problem:

Lemma 4.2 The values of $\dim_f$ and $\kappa_f$ don’t change if we reformulate their definitions taking $\mathcal{L}$ to be the set of all maximal linear extensions.
The proof is analogous to that of Lemma 1.2. Further, in our current environment, we have:

**Lemma 4.3** The critical pairs reversed in a maximal linear extension of the poset of any graph are fully determined by the ordering of \( V(G) \) within it.

**Proof.** We wish to show that, in constructing a maximal linear extension, once we’ve specified the order of the vertices, we can then specify where to place the edges without losing maximality. Given an edge \( uv \in X \), the only two relations in \( P \) involving \( uv \) are \( u < uv \) and \( v < uw \), so \( uv \) must be placed somewhere above its endpoints in any linear extension. On the other hand, for the sake of reversing critical pairs, we want edges as low as possible, since edges are only in critical pairs of the form (vertex,edge), and this critical pair is only reversed if the edge is below the vertex in the linear extension. So the best place we can put \( uv \) is right above the higher of \( u \) and \( v \) in our linear extension. If several edges are placed directly above the same vertex, their relative order does not affect the reversal of any critical pairs. Thus, once we have ordered \( V(G) \) in a linear extension, we know where we must place the edges if we wish to make this linear extension maximal. Further, the critical pair \( (u, vw) \) gets reversed exactly when \( u \) comes above both \( v \) and \( w \) in the ordering of \( V(G) \).

Henceforth, we will describe (maximal) linear extensions in terms of permutations of \( V(G) \).

Schnyder [12] showed that \( \text{dim}(P(G)) \leq 3 \) iff \( G \) is planar. For fractional dimension, on the other hand, Brightwell and Scheinerman [2] proved that \( \text{dim}_f(P(G)) \leq 3 \) for any finite graph \( G \), and that equality holds iff \( G \) contains a triangle. Taking trees to be the
most obvious examples of triangle-free graphs, they went on to show that $\dim_f(P(T)) \leq 1 + \phi = 2.61803$ for any tree $T$, where $\phi = \frac{1}{2}(1 + \sqrt{5})$ is the golden mean, and conjectured that this upper bound was tight. In the remainder of this chapter, we will show that the correct value of this upper bound is (approximately) 2.44504, and present other specific results pertaining to stars, binary trees, and infinite trees.

### 4.3 Posets of Trees and $\dim_f$ for Posets of Stars

We shall consider only complete, rooted $q$-ary trees, that is, trees where every non-leaf vertex has $q$ children, every non-root vertex has one parent, and all leaves are at the same level of the tree. Any tree may be considered rooted, and if $T$ has maximum vertex degree $\Delta$, then it is a subtree of some complete, rooted $(\Delta - 1)$-ary tree. By Lemma 4.1, this is sufficient to establish an upper bound on $\dim_f(P(T))$. Henceforth, $n$ will denote the depth of any tree under consideration.

We warm up with the relatively simple calculation of a tight upper bound for $\dim_f$ of the posets of finite stars. This result was noted, but not proved, by Brightwell and Scheinerman [2].

**Theorem 4.4** \lim_{q \to \infty} \dim_f(P(S_q)) = 1 + \sqrt{2}

**Proof.** Since we can describe our linear extensions of $P(S_q)$ with permutations of $V(S_q)$, there is, up to isomorphism, only one decisions to make in constructing them: how far down to put the root. Let us put a fraction $p = 2 - \sqrt{2}$ of the leaves above the root, and let $L_p$ be the random variable which chooses a linear extension uniformly at random from the set of all such linear extensions. In the star with root $r$, there are only two types of critical
pairs: \((r, u)\) and \((u, rv)\).

\[
\Pr\{L_p \text{ reverses } (r, u)\} = \Pr\{r > u \text{ in } L_p\} = 1 - p = \sqrt{2} - 1,
\]

\[
\Pr\{L_p \text{ reverses } (u, rv)\} = \Pr\{u > r, v \text{ in } L_p\} 
\approx p((1 - p) + p/2) = p - p^2/2 = \sqrt{2} - 1.
\]

Note that the quantity \(p((1 - p) + p/2)\) assumes that each leaf is being put over the root independently with probability \(p\). Although this is never the case with finite stars, this approximation becomes arbitrarily close to correct as \(n \to \infty\), and so will serve in our limit calculations.

Now, if we distribute total weight \(1/p\) evenly among all linear extensions in the sample space of \(L_p\), then the total weight on any critical pair is

\[
(\text{total weight on } L_p) \cdot (\text{fraction of } L_p \text{ containing the critical pair}) = (1/p) \cdot (p) = 1.
\]

Thus this weighting creates a valid fractional realizer of weight \(1/p = 1 + \sqrt{2}\). Of course, we can never actually take an exact portion \(2 - \sqrt{2}\) of leaves, but we may get arbitrarily close to this as \(q\) gets large, so we have our desired upper bound on the limit.

For a lower bound, let \(L_p\) be as above, but let \(p\) take on any value in \([0,1]\). Now, the probability that a given critical pair will be reversed by \(L_p\) is the same as the fraction of that type of critical pair that are reversed by any one member of \(L_p\)'s sample space. So if we distribute total weight \(\alpha\) evenly among all \((r, v)\) critical pairs, and \(\beta\) among all \((u, rv)\) critical pairs, then the total weight put on any linear extension from \(L_p\) is

\[
w_{\alpha,\beta}(p) = \alpha(1 - p) + \beta(p - p^2/2), \quad \text{and}
\]
\[
\frac{d}{dp} w_{\alpha,\beta}(p) = \beta(1 - p) - \alpha = 0 \quad \text{at} \quad p = 1 - \frac{\alpha}{\beta}
\]

\[
\frac{d^2}{dp^2} w_{\alpha,\beta}(p) = -\beta
\]

So \(w_{\alpha,\beta}(p)\) attains its maximum value at \(p = 1 - \alpha/\beta\). As in our upper bound calculations, the quantity \((p - p^2/2)\) is only actually correct if we are placing each leaf above the root independently with probability \(p\), but again, it becomes arbitrarily close to correct in the limit, which is all we are presently concerned with.

If we set \(\alpha = \sqrt{2}/2\) and \(\beta = (\sqrt{2} + 1)/\sqrt{2}\), then \(w\) attains its maximum value for \(p = 2 - \sqrt{2}\), and

\[
w_{\alpha,\beta}(2 - \sqrt{2}) = \frac{\sqrt{2}}{2}(\sqrt{2} - 1) + \frac{\sqrt{2} + 1}{\sqrt{2}}(\sqrt{2} - 1) = 1
\]

So our weighting of critical pairs never puts weight more than 1 on any linear extension, and so is a valid fractional packing. The value of this fractional packing is \(\alpha + \beta = 1 + \sqrt{2}\), giving the desired lower bound. Note that it is the inaccuracy in the \((p - p^2/2)\) value which keeps this from actually being a valid fractional packing in any finite case. It is, however, sufficient in showing that, as \(q \to \infty\), we can create fractional packings arbitrarily close to this value, which is all we need to establish our limit. \(\square\)

Note that, since \(S_{q-1} \subset S_q\), Lemma 4.1 tells us that \(\dim_f(P(S_q))\) is an increasing function of \(q\), and so actually increases to the limiting value of \(1 + \sqrt{2}\).

We now move on to trees of arbitrary depth. Henceforth, for the sake of our optimal fractional realizers and packings, it suffices to consider very specific classes of linear extensions and critical pairs. A linear extension of a poset of a tree is contiguous if, for
any vertex $x$ with children $y_1, \ldots, y_q$, all the vertices of any subtree rooted at some $y_i$ appear consecutively in the linear extension. That is, if $T_i$ is the subtree rooted at $y_i$, then a contiguous linear extension has

$$\ldots, V(T_1), V(T_2), \ldots, V(T_k), x, V(T_{k+1}), \ldots, V(T_q), \ldots$$

appearing consecutively in it, for some ordering of $x$’s children and some $k \in \{0, 1, \ldots, q\}$. Any vertex not in the subtree rooted at $x$ comes before or after these vertices in the linear extension, and the vertices of each $V(T_i)$ appear within this collection in a similarly contiguous fashion. See Figure 7(b).

A more constructive description comes from building the tree recursively. If we build the tree by recursively replacing each leaf by a $q$-star with that leaf as the center, then we similarly update the linear extension by replacing the leaf with the entire star contiguous within the linear extension.

Since we are talking about complete $q$-ary trees, we may now fully describe contiguous linear extensions, up to isomorphism, by specifying what fraction of each vertex’s children appear above it in the linear extension. We shall be more precise about this below.

In the case of critical pairs, we don’t limit our consideration to a select few, as we did with linear extensions, but instead observe that every critical pair can be put into one of just a few categories. In any contiguous linear extension, this category fully describes a critical pair’s behavior. Of course, all (branch,leaf) critical pairs are identical up to isomorphism. The (vertex,edge) critical pairs may be described by where the vertex falls in the rooted tree relative to the edge; specifically, what is the lowest common ancestor of the vertex and the lower endpoint of the edge. For the critical pair $(u, vw)$, where $v$ is above $w$ in the tree,
Figure 7: (a) The top three levels of a complete ternary tree. (b) A contiguous linear extension thereof, where each $T_{ij}$ represent the entire subtree rooted at a level $Y$ vertex. Note that the subtree rooted at each $x_i$ appears contiguously within this linear extension.

we have four choices for the lowest common ancestor of $u$ and $w$: 
Figure 8: The five types of critical pairs and their intermediate vertices

(i) \(u\): \(vw\) is in the subtree rooted at \(u\) (\(vw\) is below \(u\) in \(T\))

(ii) \(w\): \(u\) is in the subtree rooted at \(w\) (\(u\) is below \(v\) and \(w\) in \(T\))

(iii) \(v\): \(u\) is below \(v\) and one of \(w\)’s “siblings” (not beneath \(w\))

(iv) none of the above: \(u\) and \(vw\) are in different branches of some subtree

(v) we shall henceforth use this designation for (branch,leaf) critical pairs.

We may now fully characterize the conditions necessary for each such critical pair type to be reversed in a contiguous linear extension (putting vertex>edge or branch>leaf; see Figure 8):

(i) if \(y_v\) is the child of \(u\) whose subtree contains \(vw\) (possibly \(y_v = v\)), we must have \(u > y_v\) (the entire subtree rooted at \(y_v\), including \(vw\), is then beneath \(u\))
(ii) if $y_u$ is the child of $w$ whose subtree contains $u$ (possibly $y_u = u$), we must have $w > v$ (to get $w$’s subtree, including $u$, above $v$) and $y_u > w$ (to get $y_u$’s subtree, including $u$, above $w$).

(iii) if $y_u$ is the child of $v$ whose subtree contains $u$ (possibly $y_u = u$), we need $y_u > v$ and either $v > w$ or $y_u > w > v$.

(iv) if $x$ is $u$ and $v$’s lowest common ancestor, and $y_u$ and $y_v$ are $x$’s children whose subtrees contain $u$ and $v$, respectively (possibly $y_u = u$ or $y_v = v$), we must have $y_u > y_v$ ($x$’s relative position is irrelevant).

(v) we must simply have branch $> \text{leaf}$.

If we chose a contiguous linear extension “at random”, then the probability of any of the above events can be described in terms of the fraction of vertices’ children which appear above them.

We now present two results. Because the proofs of each are similar, we state the results first, then dedicate separate sections to all upper bound and tightness calculations.

**Theorem 4.5** For any (finite or infinite) binary tree $T$ (i.e. tree with maximum degree 3), $\dim_f(P(T)) \leq \frac{7}{5}$, and this bound is best possible.

**Theorem 4.6** For any (finite or infinite) tree $T$ of bounded degree, $\dim_f(P(T)) \leq z_0 \approx 2.44504$, where $z_0$ is a root of $z^3 - 7z^2 + 14z - 7 = 0$. This bound is, at least to within 2000 decimal places of accuracy, best possible.
4.4 Upper Bound Calculations

As mentioned earlier, we may fully describe contiguous linear extensions simply by stating what fraction of a vertex’s children are above it in the linear extension. The simplest way to do this is to let this fraction be the same for all vertices. In particular, for a \( q \)-ary tree, we may define a contiguous linear extension \( L_i \), up to isomorphism, by allowing every vertex to have exactly \( i \) of its children above it in \( L_i \), for any \( i \in \{0, \ldots, q\} \). We may also treat \( L_i \) as a random variable: we choose a linear extension uniformly at random from the set of all such \( L_i \)'s. Note that the arrangement of a vertex’s children around it is independent of that for any other vertex. Then the probability that any vertex is above its parent in \( L_i \) is \( \frac{i}{q} \).

Further, we can calculate the exact probability that any of our five types of critical pairs is reversed in \( L_i \). Again, for the critical pair \((u,vw)\), we have:

(i) \( \Pr\{u>y_v\} = \frac{q-i}{q} \)

(ii) \( \Pr\{y_u>w>v\} = \frac{i^2}{q^2} \)

(iii) \( \Pr\{y_u>v \text{ and } y_u>w\} = \frac{\sum_{k=1}^{i}(q-k)}{q(q-1)} = \frac{i(2q-i-1)}{2q(q-1)} \)

(iv) \( \Pr\{y_u>y_v\} = \frac{1}{2} \)

(v) \( \Pr\{\text{branch>leaf}\} = \frac{q-i}{q} \)

The optimal usage of such \( L_i \)'s in constructing fractional realizers requires a mix of different \( i \) values. If we use \( L_i \) a fraction \( a_i \) of the time, where \( \sum_i a_i = 1 \), then the above becomes

(i) \( \Pr\{u>y_v\} = \sum_{i=0}^{q} \left(\frac{q-i}{q}\right) a_i \)

(ii) \( \Pr\{y_u>w>v\} = \sum_{i=0}^{q} \left(\frac{i^2}{q^2}\right) a_i \)
(iii) \( \Pr\{y_u > v \text{ and } y_u > w\} = \sum_{i=0}^{q} \left( \frac{q}{2q(q-1)} \right) a_i \)

(iv) \( \Pr\{y_u > y_v\} = \frac{1}{2} \)

(v) \( \Pr\{\text{branch} > \text{leaf}\} = \sum_{i=0}^{q} \left( \frac{q-i}{q} \right) a_i \)

Suppose that the smallest value above is \( p \); that is, every critical pair gets probability weight at least \( p \). Then if we actually put weight \( a_i/p \) on \( L_i \)\(^{25}\) for each \( i \), the total weight used is \( 1/p \). Further, each of the values in (i)-(v) above gets multiplied by \( 1/p \), and so is at least 1. We then have a fractional realizer of weight \( 1/p \), and so we wish to maximize \( p \) to get the best upper bound possible. Note that (i) and (v) are identical, and since we will not be able to get all of (i)\(\text{(ii)}\)\(\text{(iii)} \) simultaneously above \( \frac{1}{2} \), (iv) will never be our smallest value. Trying to get the minimum of (i)-(iii) as large as possible is a balancing act, where making one larger makes another smaller. So the best we can do is when they’re all equal\(^{26}\). So we may write each of (i)-(iii) above as \( \sum_i c_i a_i = p \) for the appropriate values of the \( c_i \)'s. Along with \( \sum_i a_i = 1 \), we have a linear system with 4 equations and \( q + 1 \) unknowns. Therefore, so long as the system is consistent, a solution will exist with only 4 non-zero variables. We clearly need \( p \) to be one of these; let the others be \( a_i, a_j \), and \( a_k \). Since \( a_k = 1 - a_i - a_j \), we can make this substitution in (i)-(iii) and remove the last equation. Rearranging, (i)-(iii) take the form

\[
(c_i - c_k)a_i + (c_j - c_k)a_j - p = -c_k .
\]

Putting this in matrix form, substituting in the appropriate \( c \) values for (i)-(iii) and

\(^{25}\)More specifically, if we distribute total weight \( a_i/p \) evenly among all linear extensions from \( L_i \)'s sample space

\(^{26}\)This claim is made without proof; such a proof adds no real content, as what follows would be valid even if it were false. We still establish a valid upper bound.
collecting these in a single matrix equation, we get

$$
\begin{bmatrix}
\frac{k-i}{q} & \frac{k-i}{q} & -1 \\
\frac{j^2-k^2}{q^2} & \frac{j^2-k^2}{q^2} & -1 \\
\frac{2q(i-k)+i(i+1)+k(k+1)}{2q(q-1)} & \frac{2q(j-k)+j(j+1)+k(k+1)}{2q(q-1)} & -1
\end{bmatrix}
\begin{bmatrix}
a_i \\
a_j \\
p
\end{bmatrix}
= \begin{bmatrix}
-\frac{q-k}{q} \\
-\frac{k^2}{q^2} \\
-\frac{k(2q-k-1)}{2q(q-1)}
\end{bmatrix}
$$

If we represent the above as $Ax = b$, we may find $x$ explicitly as $A^{-1}b$, since $A^{-1} = \frac{1}{\det(A)} \text{adj}(A)$ may be calculated symbolically. Interestingly, regardless of our choice of $(i, j, k)$, $p$ always solves as $\frac{2q-1}{2q-3}$, giving a fractional realizer weight of $\frac{5q-3}{2q-1}$. We need only check that we can find some $(i, j, k)$ where $a_i, a_j, a_k \geq 0$. It turns out that a number of choices of $(i, j, k)$ work. In particular, $(i, j, k) = (0, \lceil q/2 \rceil, q)$ always works. In the case of $q$ even, for example, we get

$$(a_i, a_j, a_k) = \left( \frac{1}{5q-3}, \frac{4q-4}{5q-3}, \frac{q}{5q-3} \right)$$

Multiplying these values by $1/p = \frac{5q-3}{2q-1}$ gives the actual amount of weight we wish to assign to each type of linear extension. This assignment constitutes a valid fractional realizer of weight $\frac{5q-3}{2q-1}$, which proves $\dim_f(P(T)) \leq \frac{5q-3}{2q-1}$ for any complete, finite $q$-ary tree $T$, and thus for any finite tree with maximum degree $q + 1$.

In a finite tree, there are only a finite number of linear extensions with parameter $i$, and so it makes sense to take the weight $a_i/p$ and divide it evenly among all linear extensions in the class $L_i$. In the case of infinite trees, however, this clearly does not work. We may, however, through a clever choice of a finite collection of members of $L_i$, create the desired

---

27So that all linear extensions receive non-negative weights, and we actually do have a valid fractional realizer.
randomness and independence in putting vertices above or below their parents, even in an infinite tree. Let $\Pi$ be the set of permutations on $[q]$. For $\pi, \sigma \in \Pi$, define the contiguous linear extension $L_i(\pi, \sigma)$ as follows. If vertex $v$ is on an odd level of $T$, arrange its children in $L_i(\pi, \sigma)$ according to $\pi$; if $v$ is on an even level, arrange its children according to $\sigma$; in either case, put $v$ below exactly $i$ of its children. We now chose the random variable $L_i$ uniformly from $\{L_i(\pi, \sigma) : (\pi, \sigma) \in \Pi \times \Pi\}$. Now not only are all possible arrangements of a vertex’s children equally likely, but they are independent of the arrangement of that vertex’s siblings. Note that the reversal of any critical pair is determined by the ordering of vertices from (at most) three consecutive levels of $T$. For example, whether or not a type(ii) critical pair $(u, vw)$ is reversed is fully determined by the location of $v$, $w$ and $y_u$ within $L_i(\pi, \sigma)$, and all these vertices lie within three consecutive levels of $T$. Since the arrangement of the higher of these levels is independent of the arrangements within the lower level, all our previous calculations are valid for this construction. Thus our result also applies to infinite $q$-ary trees.

We notice that, as $q \to \infty$, $1/p = \frac{5q-3}{2q-1}$ approaches 2.5, which is not the best possible upper bound for posets of trees. However, this does provide the best known upper bound for instances of $q$ where this value is less than then general bound of 2.44504 (specifically, $q = 2, 3, 4, 5$). In the case of binary trees ($q = 2$), this bound is best possible.

**Proof of Theorem 4.5 (first part).** Setting $q = 2$ in the above informs us that, for all (finite or infinite) binary trees $T$, $\dim_f(P(T)) \leq \frac{2}{3}$. □

The proof that this bound is best possible appears in the following section.

Since these bounds are not, in general, best possible, how may we improve on them?

---

28Note that $q = 5$ gives $1/p = 2.4$ but $q = 6$ gives $1/p = 2.45$. 
So far, we have taken the probability that a vertex appears below one of its children to be the same for all vertices. If instead we make this probability conditional on whether or not a vertex is itself above or below its parent, we may improve our limiting bound.

**Proof of Theorem 4.6 (first part).** Given the poset of a complete \(q\)-ary tree, we wish to define a single class of contiguous linear extensions, all of which are equivalent. We let \(L\) be a random variable which chooses one such linear extension uniformly at random. We now build our contiguous linear extensions “top down”, i.e. recursively, starting at the top. For any vertex \(y\) with parent \(x\) and child \(z\), let

\[
\begin{align*}
    a &= \Pr\{z > y \text{ in } L\} \\
    b &= \Pr\{z > y \mid y > x \text{ in } L\} \\
    c &= \Pr\{z > y \mid y < x \text{ in } L\}
\end{align*}
\]

In order to make sure that these three are consistent, we will require that

\[
\Pr\{z > y\} = \Pr\{z > y \mid y > x\} \Pr\{y > x\} + \Pr\{z > y \mid y < x\} \Pr\{y < x\}
\]

i.e. that \(a = ab + (1 - a)c\). Also, while every other vertex will have either a fraction \(b\) or \(c\) of its children above it, the root will actually have a fraction \(a\) of its children above it. Now, as before, we may calculate the probability that each type of critical pair gets reversed in \(L\). For the critical pair \((u, vw)\), we have:

(i) \(\Pr\{u > y_v\} = 1 - a\), or equivalently, \(a(1 - b) + (1 - a)(1 - c)\)

(ii) \(\Pr\{y_u > w > v\} = ab\)
(iii) \( \Pr\{y_u > v \text{ and } y_u > w\} = a(b - b^2/2) + (1-a)(c - c^2/2) \), or just \( a - a^2/2 \) if \( v \) is the root

(iv) \( \Pr\{y_u > y_v\} = \frac{1}{2} \)

(v) \( \Pr\{\text{branch} > \text{leaf}\} \) = same as (i)

As before, we wish to maximize the minimum of these quantities (which we call \( p \)), and we do so by setting (i)=(ii)=(iii). Using both quantities in (i), we get three equations, which solve as

\[
\begin{align*}
b &= \frac{1-a}{a} \\
c &= \frac{2a-1}{1-a} \\
0 &= 7a^3 - 7a^2 + 1
\end{align*}
\]

the last of which has two roots in \([0,1]\), but the larger gives \( c = 1.8019 \), which is not in \([0,1]\). We are left with the following solution:

\[
\begin{align*}
a &\doteq 0.59101, \quad b \doteq 0.69202, \quad c \doteq 0.44504, \quad p \doteq 0.40899, \quad 1/p \doteq 2.44504.
\end{align*}
\]

And, as desired, this solution satisfies our requirement that \( a = ab + (1-a)c \). As before, we distribute total weight \( 1/p \) evenly over the sample space of \( L \), and since every critical pair is in a fraction at least \( p \) of these, each critical pair receives total weight at least 1. We thus have a valid fractional realizer of weight \( 1/p \). This is exactly the value we want for \( z_0 \). Noting that \( z_0 = 1/p = 1/(1-a) \), we have \( a = (z_0 - 1)/z_0 \), and substituting this into \( 0 = 7a^3 - 7a^2 + 1 \) gives \( z_0^3 - 7z_0^2 + 14z_0 - 7 = 0 \). So \( \dim_f(P(T)) \leq z_0 \) for any tree in question.

Of course, \( a, b \) and \( c \) are irrational, so we can never put these exact fractions of children
over their parents. However, as $q$ gets large, we can get arbitrarily close to these values. Further, since $T_1 \subset T_2$ implies $\dim_f(P(T_1)) \leq \dim_f(P(T_2))$, $\dim_f$ has to be a non-decreasing function of $q$, and so this upper bound applies to all complete rooted trees. We can use the same technique previously described to reduce the size of $L$’s sample space to $(q!)^2$ for infinite trees, so long as $q$ is finite. Thus we have our upper bound on all (finite or infinite) complete $q$-ary trees for any finite $q$, and thus the bound also applies to all trees of bounded degree. □

Again, the proof that this bound is best possible appears in the following section.

4.5 The Tightness of Upper Bounds

We now establish that the given upper bounds are best possible. To this end, we consider the dual invariant, fractional packing number, since the value of any valid fractional packing serves as a lower bound on fractional dimension. We proceed by assigning weights to all critical pairs, and then establishing that we have a fractional packing by means of a dynamic program. We are still working with complete $q$-ary trees ($q$ fixed), and we wish to construct the depth $n$ tree $T_n$ recursively as follows: $T_2$ is just the $q$-star (with its center the root); to make $T_{n+1}$, we take $q$ copies of $T_n$, create a new root, and draw edges between the new root and each of the old roots. We only assign weight to “close-as-possible” critical pairs; that is, for the (vertex, edge) critical pairs of any particular type, we require the vertex to be as close as possible to the edge. In terms of our previous language, we only use critical pairs where $y_u = u$ or $y_v = v$. So type (i) and (ii) critical pairs are contained within three consecutive levels of the tree, and type (iii) are within two. “Close-as-possible” is redundant for type (v), and we never assign weight to type (iv). At each iteration, the “new” critical pairs are
exactly the ones that use the new root (either as a vertex or edge endpoint). All other critical pairs are (copies of) ones found in previous iterations. For convenience, we refer to the top three levels of the tree as \( r \) (the root), \( X \) and \( Y \), in downwards order (see Figure 7(a)), and thus we only directly discuss critical pairs involving vertices in these levels. For each type of critical pair except (iv), we associate a fixed weight:

\[
\begin{align*}
(i): & \quad \gamma \\
(ii): & \quad \delta \\
(iii): & \quad \beta \\
(v): & \quad \alpha
\end{align*}
\]

This weight is distributed evenly among all new critical pairs of the indicated type. So, for example, since there are \( q(q - 1) \) new type (iii) critical pairs at each iteration, we assign to each such new critical pair a weight of \( \beta/(q(q - 1)) \). When we iterate, all old weights are divided by \( q \), since we have made \( q \) copies of all critical pairs. This way, while the weight on any particular (copy of an) old critical pair is divided by \( q \), the total weight on all old critical pairs remains unchanged. Specifically, if we define

\[
w_n = \text{total weight on all critical pairs in the tree } T_n.
\]

we may recursively calculate\(^{29}\) \[ w_n = w_{n-1} + \beta + \delta + \gamma. \] Of course, at any step, we would have to scale down all weights so that no linear extension received total weight more than 1, but for now we address this issue only indirectly. At each stage we wish to know, “What is the most weight that this critical pair weighting puts on any linear extension of \( P(T_n) \)?”

We condition this answer on the number of sub-roots (children of the root) that appear above the root in the linear extension in question, and therein lies our dynamic program.

\(^{29}\)Because they involve leaves, there are never any new type (v) critical pairs after \( T_2 \), so we do not add \( \alpha \).
$f_n(i) = \text{maximum possible weight on any linear extension of } P(T_n) \text{ that has } i \text{ subroots above the main root.}$

Looking at the different types of critical pairs (excepting type (iv), which always get weight 0), we see that only type (iii) uses vertices from more than one $T_{n-1}$ subtree. So within a linear extension, once we have ordered the root and its children, we only care about the ordering of the other vertices relative to other vertices in their own $T_{n-1}$ subtree; how vertices from different subtrees are mixed is immaterial. Further, for any new type (i) or (ii) critical pairs, we need only consider a single $T_{n-1}$ subtree, and whether its subroot is placed above or below the main root. We define

$$g_{n}^x = \text{maximum possible total weight that can be put on critical pairs in any } (T_{n-1} + \text{root})$$

by a linear extension where the subtree’s root is above the main root.

$$g_{n}^r = \text{maximum possible total weight that can be put on critical pairs in any } (T_{n-1} + \text{root})$$

by a linear extension where the subtree’s root is below the main root.

These values are, roughly, $f_{n-1}(i)/q$ plus weight added by new type (i) and (ii) critical pairs. Specifically, for each such subtree, we wish to use the $i$ which gives the largest possible weight. For a linear extension of (a copy of) $T_{n-1}$, if $i$ of its subroots (in $Y$) are above its main root (in $X$), then we can put at most $f_{n-1}(i)/q$ total weight on it. If its root is above the root $r$ of $T_n$, then no new type (i) critical pairs may be reversed (reversing these requires $r$ to be placed above the subtree’s root). Further, a type (ii) critical pair is reversed only if its $Y$ vertex is above both its $X$ parent and $r$. We have specified that $i$ such $Y$ vertices are above their parent (in this copy of $T_{n-1}$), which is in turn above $r$, so...
exactly $i$ type (ii) critical pairs (from a total of $q$) are reversed by this linear extension of the $T_{n-1} + r$ subtree. Since each of the $q^2$ new type (ii) critical pairs gets weight $\delta/q^2$, we have

$$g^x_n = \max_{i=0,\ldots,q} \left\{ \frac{1}{q} f_{n-1}(i) + \frac{i}{q^2} \delta \right\}$$

Applying a similar analysis to the case where the root of $T_{n-1}$ (in $X$) is below $r$, we see that $(q-i)$ of the $Y$ vertices go below their parent, so for each of these, the corresponding type (i) critical pair is reversed. For the $i$ vertices in $Y$ that go above their parent in $X$, we may freely put them between $r$ and their parent, or above $r$. The former reverses a type (i) critical pair, while the latter reverses a type (ii). Since these are the only critical pairs in $T_{n-1} + r$ that are affected by the positioning of $r$ relative to $V(T_{n-1})$, we make this decision based on which of $\delta$ or $\gamma$ is larger. Then we have

$$g^r_n = \max_{i=0,\ldots,q} \left\{ \frac{1}{q} f_{n-1}(i) + \frac{q-i}{q^2} \gamma + \frac{i}{q^2} \cdot \max\{\delta, \gamma\} \right\}$$

We may now compute $f_n$ by considering which type (iii) critical pairs are reversed. For each $X$ vertex $u$ above the root, every type (iii) critical pair of the form $(u, rw)$ in which $u$ is above $w$ is reversed. There are $\sum_{k=1}^{i} (q-k) = (iq - i(i+1)/2)$ of these reversed (from a total of $q(q-1)$ type (iii)'s), and so

$$f_n(i) = \left( \frac{iq - i(i+1)/2}{q(q-1)} \right) \beta + i \cdot g^x_n + (q-i) \cdot g^r_n$$

Finally, we have

$$f_2(i) = \left( \frac{iq - i(i+1)/2}{q(q-1)} \right) \beta + \frac{q-i}{q} \alpha$$

$$w_2 = \beta + \alpha$$

$$w_n = w_{n-1} + \beta + \delta + \gamma$$
and so we have established the mechanics of our dynamic program. For convenience, define \( f_n = \max_{i=0, \ldots, q} f_n(i) \). Now, after the \( n \)th iteration, when we have used the indicated weighting scheme and determined that no linear extension of \( P(T_n) \) can have total weight more than \( f_n \), if we divide all the weights of all the critical pairs by \( f_n \), then we have a valid fractional packing of \( P(T_n) \) with value \( w_n / f_n \). This value is a lower bound on \( \dim_f(P(T_n)) \). Further, it is a lower bound for any non-negative values of \( \alpha, \beta, \delta \) and \( \gamma \), so the best lower bound comes with the best selection of these values. With this in mind, we are ready to establish a lower bound for \( \dim_f \) of all posets of binary trees.

**Proof of Theorem 4.5 (second half).** We have \( q = 2 \), and all notation as above. Set

\[
\alpha = 2, \quad \beta = 2, \quad \delta = 2, \quad \gamma = 3
\]

Since \( \gamma > \delta \), we have

\[
f_2(0) = \alpha = 2, \quad f_2(1) = \beta / 2 + \alpha / 2 = 2, \quad f_2(2) = \beta / 2 = 1
\]

\[
g^x_n = \max_{i=0,1,2} \left\{ \frac{1}{2} f_{n-1}(i) + \frac{i}{2} \right\}
\]

\[
g^r_n = \max_{i=0,1,2} \left\{ \frac{1}{2} f_{n-1}(i) + \frac{3}{2} \right\}
\]

\[
f_n(0) = 2g^x_n, \quad f_n(1) = 1 + g^x_n + g^r_n, \quad f_n(2) = 1 + 2g^x_n
\]

\[
w_2 = 4, \quad w_n = w_{n-1} + 7
\]

Notice that, if \( f_{n-1}(0) = f_{n-1}(1) = f_{n-1}(2) + 1 \), then \( f_n(1) - f_n(2) = g^r_n - g^x_n = 1 \) and \( f_n(0) - f_n(1) = g^r_n - g^x_n - 1 = 0 \). Since these relations hold for the \( n = 2 \) base case, they must hold for all \( n \) inductively. So \( f_n = f_n(0) = 2g^x_n = f_{n-1}(0) + 3 = f_{n-1} + 3 \) for all \( n \geq 3 \). So we have
\[ w_n = w_2 + 7(n - 2) = 7n - 10 \text{ and } f_n = f_2 + 3(n - 2) = 3n - 4 \]

Since \( w_n/f_n \) serves as a lower bound on \( \text{dim}_f \), we have that \( \text{dim}_f(P(T_n)) \geq \frac{7n-10}{3n-4} \) for a complete \( n \)-level binary tree \( T_n \), and so the previously derived upper bound of 7/3 is best possible. \( \square \)

**Proof of Theorem 4.6 (2nd half).** As with the upper bound, in order to establish a lower bound for all trees, we want \( q \to \infty \). Notice that, although the argument \( i \) of \( f_n(i) \) is a number of vertices, in the formulas it generally appears as part of a fraction of critical pairs. So let us reformulate \( f_n(i) \) slightly, so that now \( i \) represents the fraction of the root’s children which appear above it in a linear extension. Now, the domain of \( f_n(i) \) is \( \{0, \frac{1}{q}, \frac{2}{q}, \ldots, \frac{q}{q}\} \), so as \( q \to \infty \), we may choose arguments arbitrarily close to any real value in \([0, 1]\). Thus we may approximate this situation simply by taking \( f_n(i) \) to be a function on the real interval \([0, 1]\). We next reformulate our entire dynamic program in this continuous form, which will demonstrate the limiting behavior of our previous system as \( q \to \infty \).

We assume that we may choose any real fraction \( i \) of a vertex’s children to go above it in a linear extension. We still talk about \( T_n \) as if \( q \) were finite, for if we actually had \( q = \infty \), it would make no sense to talk about a specific fraction of an infinite number of children. We still iterate our tree and functions in the same manner, and still distribute weights \( \alpha, \beta, \delta \) and \( \gamma \) evenly on the various types of “new” critical pairs. \( f_n(i) \) is still the most weight put on any linear extension of \( P(T_n) \) with parameter \( i \) by our weighting. However, we must slightly reformulate the meanings of \( g_n^r \) and \( g_n^s \):

\[ g_n^r = \text{maximum possible total weight that can be put on all critical pairs in all } (T_{n-1} + \]
root)'s by a linear extension where every subtree’s root is above the main root.

\( g_n^r = \) maximum possible total weight that can be put on all critical pairs in all \((T_{n-1} + \text{root})'s\) by a linear extension where every subtree’s root is below the main root.

These values still represent weight that lies only within \(T_{n-1}+\text{root}\) subtrees, but now each represents the weight on all such subtrees, were the root to fall as indicated for each of them. But since the weight on any collection of such subtrees is independent of the weight on any other disjoint collection, if exactly a fraction \(i\) of \(r\’s\) children come above it, then exactly a fraction \(i\) of the total weight represented by \(g_n^r\) will be on this linear extension (as well as \(1 - i\) of the total weight of \(g_n^r\)). We may now formulate our expressions much the same as before, though it is much easier to represent the fraction of a particular type of critical pair which is being reversed\(^{30}\):

\[
\begin{align*}
   g_n^w &= \max_{i \in [0,1]} \{ f_{n-1}(i) + i \delta \} \\
   g_n^r &= \max_{i \in [0,1]} \{ f_{n-1}(i) + (1 - i) \gamma + i \cdot \max \{ \delta, \gamma \} \} \\
   f_2(i) &= (i - i^2/2) \beta + (1 - i) \alpha \\
   f_n(i) &= (i - i^2/2) \beta + ig_n^x + (1 - i)g_n^r \\
   w_2 &= \beta + \alpha \\
   w_n &= w_{n-1} + \beta + \delta + \gamma
\end{align*}
\]

We henceforth take \(\gamma \geq \delta\), so that \(g_n^r = \gamma + \max_i \{ f_{n-1}(i) \}\). By choosing \(\alpha\) carefully, we can force \(f_n(i)\) to behave nicely.

\(^{30}\)The \(i - i^2/2\) of the \(f\) formulations is exactly the same as the expression \(p - p^2/2\) found in the proof for stars. It is the fraction of new type (iii) critical pairs that are reversed when a fraction \(i\) of the subroots are placed above the main root.
Lemma 4.7 If we choose \( \alpha = \frac{\beta - \alpha + \delta^{2}/2}{\beta - \delta} \), then \( f_n(i) = f_{n-1}(i) + c \) for all \( i \in [0, 1] \), where \( c = g_3^x - \alpha \).

Proof. Let \( j = \text{argmax}(g_3^x) \) and \( k = \text{argmax}(g_3^r) \). Substituting \( f_2 \) and checking derivatives gives

\[
g_3^x = \max_{i \in [0, 1]} \left\{ -\frac{\beta}{2} + i(\beta - \alpha + \delta) + \alpha \right\}, \quad j = \frac{\beta - \alpha + \delta}{\beta}
\]

\[
g_3^x = \frac{(\beta - \alpha + \delta)^2}{2\beta} + \alpha
\]

\[
g_3^r = \gamma + \max_{i \in [0, 1]} \left\{ -\frac{\beta}{2} + i(\beta - \alpha) + \alpha \right\}, \quad k = \frac{\beta - \alpha}{\beta}
\]

\[
g_3^r = \frac{(\beta - \alpha)^2}{2\beta} + \alpha + \gamma
\]

Then

\[
f_3(i) = (i - \frac{\beta}{2}) \beta + ig_3^x + (1 - i)g_3^r = f_2(i) + i(\alpha + g_3^x - g_3^r) + (g_3^r - \alpha).
\]

Since \( \alpha \) was specifically chosen to solve

\[
\alpha = g_3^r - g_3^x = \gamma - \frac{2(\beta - \alpha)\delta + \delta^2}{2\beta},
\]

and with \( c = g_3^x - \alpha \), we have our result for the base case of \( n = 3 \). The general case follows by induction: if we assume that \( f_{n-1}(i) = f_2(i) + (n - 3)c \), then clearly \( g_n^x = g_3^x + (n - 3)c \) and \( g_n^r = g_3^r + (n - 3)c \), and then

\[
f_n(i) = (i - \frac{\beta}{2}) \beta + ig_n^x + (1 - i)g_n^r
\]

\[
= f_2(i) + i(\alpha + g_3^x - g_3^r) + (n - 3)c + (g_3^r - \alpha)
\]

\[
= f_2(i) + (n - 2)c.
\]
which proves our lemma. □

As before, \( \dim_f(P(T_n)) \) is bounded below by \( w_n / \max_i \{ f_n(i) \} \). (More precisely, the bound will approach this value as \( q \to \infty \).) But \( w_n = w_2 + (\beta + \delta + \gamma)n \) and \( f_n(i) = f_2(i) + cn \). So as \( n \to \infty \), the limiting value of this lower bound is just \( (\beta + \delta + \gamma)/c \). Our goal now becomes to choose the values of \( \beta, \delta \) and \( \gamma \) which maximize this quantity. More specifically, we must solve the following non-linear optimization problem:

\[
\max_{\alpha, \beta, \delta, \gamma} \frac{\beta + \delta + \gamma}{c} \quad \text{s.t.} \quad c = \frac{(\beta - \alpha)^2}{2\beta} + \gamma, \quad \alpha = \frac{\beta \gamma - \beta \delta + \delta^2/2}{\beta - \delta}, \quad \gamma \geq \delta.
\]

Although we cannot hope for a clean analytic solution, we may reduce the problem somewhat. Notice that our objective function is not affected if we proportionally scale each of \( \beta, \delta \) and \( \gamma \), since \( c \) and \( \alpha \) would also be scaled by the same proportion. Thus we may arbitrarily choose \( \beta = 1 \), giving

\[
\alpha = \frac{\gamma - \delta + \delta^2/2}{1 - \delta}
\]

\[
c = g_3 - \alpha = (1 - \alpha)^2/2 + \gamma = \frac{1}{2} \left( \frac{1 - \gamma - \delta^2/2}{1 - \delta} \right)^2 + \gamma.
\]

Substituting these into our objective function gives

\[
\max_{0 \leq \delta \leq \gamma} \frac{2(1 - \delta)^2(1 + \delta + \gamma)}{\delta^4/4 + \delta^2 \gamma + \delta^2 + \gamma^2 - 4\delta \gamma + 1}
\]

The value of this optimization problem will be the best possible value of \( w_n / \max_i \{ f_n(i) \} \), and will thus be a lower bound on \( \dim_f(P(T_n)) \) as \( n, q \to \infty \).
At this time, we are still attempting an analytic proof that the solution to this optimization problem is the same as the upper bound $z_0$. We have, however, used Mathematica to determine that the two quantities are equal out to 2000 decimal places of accuracy, so there can be little doubt that they are, in fact, the same number. □

4.6 Infinite Trees

We have already established that 2.44504 is a tight upper bound on the fractional dimension of posets of infinite trees so long as their maximum degree is bounded. What if an infinite tree is of unbounded maximum degree? The answer to this mystery is as close as the stars.

The following is a Corollary of a result that was discovered by Tom Trotter and John Moore, but never published.

**Lemma 4.8** Any tree $T$ has $\dim(P(T)) \leq 3$.

**Proof.** In other words, any poset $P(T)$ of a tree $T$ has a size three realizer. We already know this result to be true when $T$ is finite, since finite trees are planar and Schnyder [12] showed that $\dim(P(G)) \leq 3$ when $G$ is planar. The following proof works equally well for finite and infinite trees.

We must specify the three linear extensions in our proposed realizer, and then check that any critical pair of types (i)-(v) gets reversed by at least one of these. For any tree $T$, draw $T$ rooted, and impose a left-to-right ordering on each vertex’s children. We then create the order of $V(T)$ in $L_1$ based on a left-to-right depth-first search, but we add a vertex only at the last time the search passes through it. Therefore a vertex comes below all its descendants in the ordering of $L_1$. In other words, for $u, v \in V(T)$, we have $v > u$ in
$L_1$ if either (a) $v$ is a descendant of $u$ in $T$, or (b) if $x$ is $u$ and $v$’s lowest common ancestor in $T$, and $y_u$ and $y_v$ are $x$’s children which contain $u$ and $v$, respectively, in their subtrees, then $y_v$ is left of $y_u$ in the ordering of $x$’s children within $T$. $L_2$ is constructed similarly, except that we apply a right-to-left depth-first search. $L_3$ is simply a top-down ordering, where the root is first, followed by all the root’s children, etc.; the order of vertices from within any given level of $T$ is unimportant.

We now check that $\{L_1, L_2, L_3\}$ is a realizer of $P(T)$ by checking that any critical pair of types (i)-(v) gets reversed by at least one of these linear extensions. Recall that, in order to reverse $(u, vw)$, we must have $u > v, w$ in $L_i$.

(i) For critical pair $(u, vw)$, $v$ and $w$ are descendants of $u$ in $T$, and so $u > v, w$ in $L_3$.

(ii) For critical pair $(u, vw)$, $v$ and $w$ are ancestors of $u$, and so are recorded after $u$ in either of the searches defining $L_1$ and $L_2$; that is, $u > v, w$.

(iii) For critical pair $(u, vw)$, let $y_u$ be the child of $v$ whose subtree contains $u$ ($y_u \neq w$). If $w$ is right of $y_u$ in the ordering of $v$’s children within $T$, then $u \geq y_u > w > v$ in $L_1$; otherwise, $w$ is left of $y_u$, and we see this ordering within $L_2$.

(iv) For critical pair $(u, vw)$, let $x$ be $u$ and $v$’s lowest common ancestor, and $y_u$ and $y_v$ be the children of $x$ whose subtrees contain $u$ and $vw$, respectively. If $y_v$ is right of $y_u$ in the ordering of $x$’s children within $T$, then $u \geq y_u > w > v \geq y_v$ in $L_1$; otherwise, $y_v$ is left of $y_u$, and we see this ordering within $L_2$.

(v) For critical pair (branch, leaf), we always see $\text{branch} > \text{leaf}$ in $L_3$.

So $\{L_1, L_2, L_3\}$ reverses every critical pair of $P(T)$, and we’re done.$\Box$
Theorem 4.9 If $T$ is an infinite tree of unbounded degree, then $\dim_f(P(T)) = 3$.

**Proof.** The previous Lemma establishes 3 as an upper bound. Now, another way of saying that $T$ has unbounded degree is to say that, for every positive integer $n$, $T$ has the $n$-star $S_n$ as a subgraph. Suppose this is true for $T$. For fixed positive integer $b$, let $\dim_b(P(T)) = d$.

Let $n = 2^d + 1$, and consider $S_n \subset T$ and any smallest $b$-fold realizer $\mathcal{R}$ of $P(S_n)$. In each of the $d$ linear extensions in $\mathcal{R}$, a leaf of $S_n$ is either above or below the root (center), so for all of $\mathcal{R}$, we may describe this vertex’s position relative to the root by a length $d$ binary sequence (1=above root, 0=below root). There are only $2^d$ possible sequences, but $2^d + 1$ leaves. So by the pigeonhole principle, there must be two leaves $u$ and $v$ that have the same position relative to the root in every linear extension in $\mathcal{R}$. Well, $(r, u), (r, v), (u, rv)$ and $(v, ru)$ are all critical pairs that must each be reversed $b$ times by $\mathcal{R}$. To reverse $(r, u)$ and $(r, v)$, we must have $b$ linear extensions where $u$ and $v$ are below $r$. To reverse $(u, rv)$, we must have $b$ linear extensions where $u > v > r$, and $b$ more with $v > u > r$ to reverse $(v, ru)$. These three sets of $b$ linear extensions are necessarily disjoint, so we must have $d \geq 3b$. Thus

$$\dim_f(P(T)) = \lim_{b \to \infty} \frac{\dim_b(P(T))}{b} \geq \lim_{b \to \infty} \frac{\dim_b(P(S_n))}{b} \geq \lim_{b \to \infty} \frac{3b}{b} = 3.$$

Thus there is a gap in fractional dimension between posets of infinite trees with bounded and unbounded degree: the former is bounded above by $z_0 \approx 2.44504$, while the later is always 3. In particular, note that, while $\dim_f$ of the poset of an infinite star is 3, $\dim_f$ of the poset of any finite subgraph (finite star) is bounded above by $1 + \sqrt{2}$. This result is analogous to that of $\chi_f$ versus $\overline{\chi_f}$ from Chapter 2, and in fact, the graph $G_{1,1}^2$ (for which
\( \chi_f = 3 \) and \( \overline{\chi_f} = 1 + \sqrt{2} \) was originally constructed based on this observation about posets.