Journal of Mathematical Biology manuscript No.
(will be inserted by the editor)

Counting and optimising maximum phylogenetic diversity sets

Kerry Manson · Charles Semple · Mike Steel

Received: date / Accepted: date

Abstract In conservation biology, phylogenetic diversity (PD) provides a way to quantify the impact of the current rapid extinction of species on the evolutionary 'Tree of Life'. This approach recognises that extinction not only removes species but also the branches of the tree on which unique features shared by the extinct species arose. In this paper, we investigate three questions that are relevant to PD. The first asks how many sets of species of given size k preserve the maximum possible amount of PD in a given tree. The number of such maximum PD sets can be very large, even for moderate-sized phylogenies. We provide a combinatorial characterisation of maximum PD sets, focusing on the setting where the branch lengths are ultrametric (e.g. proportional to time). This leads to a polynomial-time algorithm for calculating the number of maximum PD sets of size k by applying a generating function; we also investigate the types of tree shapes that harbour the most (or fewest) maximum PD sets of size k. Our second question concerns optimising a linear function on the species (regarded as leaves of the phylogenetic tree) across all the maximum PD sets of a given size. Using the characterisation result from the first question, we show how this optimisation problem can be solved in polynomial time, even though the number of maximum PD sets can grow exponentially. Our third question considers a dual problem: If k species were to become extinct, then what is the largest possible *loss* of PD in the resulting tree? For this question, we describe a polynomial-time solution based on dynamical programming.

Keywords Phylogenetic tree, phylogenetic diversity, biodiversity measures, optimisation, enumeration, algorithms

K. Manson · C. Semple · M. Steel
Biomathematics Research Centre, School of Mathematics and Statistics,
University of Canterbury, Christchurch, New Zealand
E-mail: kerry.manson@pg.canterbury.ac.nz (corresponding author)

1 Introduction

Advances in molecular genetics and computational techniques over recent decades have allowed biologists to reconstruct evolutionary relationships among thousands of species (Jetz et al., 2014; Upham et al., 2019). However, as fast as this 'Tree of Life' is being assembled, many of these species are heading to extinction because of anthropogenic impacts (Davis et al., 2018). This extinction of species also entails the loss of features and genetic variation through the differential pruning of the underlying tree structure. The impact on this tree is often estimated by the reduced sum of edge lengths measured in evolutionary time (Faith, 1992). For example, if all 575 bird species classified as 'imperilled' were to disappear from the bird phylogeny (from ∼10,000 species), this would result in the loss of 2.7 billion years of evolution (Jetz et al., 2014).

The ancestral relationships between a set of species are generally modelled using phylogenetic trees (Felsenstein, 2004), and one measure of how much of a tree is spanned by a subset of species is the phylogenetic diversity (PD) measure (precise definitions are provided in the next section; here, we give an informal description). In simple terms, every non-empty set of species defines a minimal subtree which connects those species to the root of the tree, and the length of every branch in that subtree is summed to give a PD score for the set overall. The greater the PD score, the more diverse a set of species is assumed to be. To illustrate, Fig. 1 shows the relative ancestry of the species $x_1, x_2, \ldots, x_7$. Solid edges are those used in the calculation of the PD score for species x_3, x_4 and x_7. Thus the PD score of $\{x_3, x_4, x_7\}$ is 16. Note that the PD score of $\{x_4, x_7\}$ is 10.

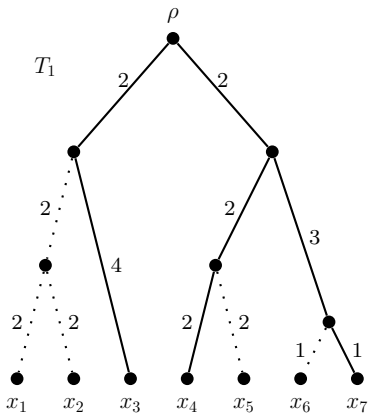

Fig. 1 The minimal subtree connecting species x_3, x_4 and x_7 has a PD score of 16.

An important concern of conservationists is preventing the extinction of species and the subsequent reduction of biodiversity. For a phylogenetic tree,

an extinction is represented by the removal of that species' leaf from the tree. This also removes the edge which connected that species to the rest of the tree, lowering the PD score. In the case where more than one species becomes extinct, the combined effect can be much larger than the sum of individual extinctions. We can interpret Fig. 1 as representing the extinction of x_1, x_2, x_5, and x_6. Notice that the simultaneous extinction of x_1 and x_2 has caused the removal of a third edge, that connecting their least common ancestor to the rest of the tree. These types of dependencies can lead to large differences in PD scores among sets of equal size.

Research has been conducted to assess the usefulness of the PD measure to inform conservation strategy. To this end, sets of species which attain the maximum value of PD (for a given number of species) have been used as a benchmark against a measured response, to be contrasted with random selections of species (Tucker et al., 2019). However, the sets of a given size which maximise PD are not unique. In applications of PD, we see the algorithms which generate such sets being run multiple times to account for this. For example, Molina-Venegas et al. (2021)[p. 586] and Mazel et al. (2018)[p. 7] both performed ten runs on each phylogenetic tree under consideration because:

> "there are multiple subsets of size S that maximises PD in a phylogeny"

and

> "For a given tree there are likely multiple, and possibly very many, sets of species with the same [maximum] PD",

respectively. Furthermore, Mazel et al. (2017)[p. 1021] noted that:

> "this number will vary across simulations and could, in some case, be very large."

Although the non-uniqueness of these sets is known and accounted for, their total number is not well understood. This leaves open questions about the most appropriate number of runs to perform in the above trials, and what the chances are that random selections of species also happen to form sets which maximise PD. In this paper, we investigate mathematical questions concerning the enumeration of maximum PD sets of given size, as well as identifying the sets of species of given size whose extinction would result in the largest loss of phylogenetic diversity.

1.1 Outline of the paper

We begin by stating in the next section the key mathematical definitions required in the paper. In Section 3, we present a new characterisation of those sets which maximise PD for each possible size (Theorem 1). This characterisation allows us to count the number of such sets on any rooted phylogenetic tree, which previous methods could not achieve concisely. Theorem 2 sets out how this process may be achieved efficiently. The conceptual approach from

Theorem 1 is continued in Section 4, leading to Algorithm 1, which selects, in polynomial time, one of these maximising sets that is optimal against a second measure. In Section 5, we consider a dual problem: determining the greatest possible loss of PD if a certain number of species becomes extinct (this turns out to be equivalent to minimising PD for a given number of species). A dynamic programming approach is used to solve this problem in polynomial time for binary (or degree-bounded) rooted phylogenetic trees.

2 Preliminaries

Phylogenetic trees. Let X be a non-empty set of taxa (e.g. species), with $|X| = n$. A *rooted phylogenetic X-tree* is a rooted tree $T = (V, E)$, where X is the set of leaves, and all edges are directed away from a distinguished root vertex ρ, and every non-leaf, non-root vertex has out-degree at least 2. In addition, when $|X| = 1$, the tree consisting of a single vertex is a rooted phylogenetic X-tree. All edges drawn in this paper will be directed down the page. If all non-leaf vertices of T have out-degree 2, we say that T is *binary*.

Three types of restrictions on T will be useful. For $A \subseteq X$, the *A-subtree* of T is the minimal tree which connects the leaves of A to the root vertex ρ. In order for an A-subtree to be a phylogenetic tree, we suppress any non-root vertices with out-degree 1 which arise during its construction. For a set of vertices $V' \subseteq V(T)$, the forest $T[V']$ is the restriction of T to those vertices in V' and (directed) edges $(u', v') \in E(T)$, where $u', v' \in V'$. A subtree of T is *pendant* if it can be disconnected from ρ by deleting a single edge of T. As shorthand, for an arbitrary set A and element x, we write $A \cup x$ in place of $A \cup \{x\}$ and $A - x$ in place of $A \backslash \{x\}$.

For any vertex $v \in V(T)$, we write $x \in c_T(v)$ if $x \in X$ and the unique path from ρ to x includes v. That is, $c_T(v)$ is the set (*cluster*) of leaves descended from v in T. For the (directed) edge $e = (u, v)$, we define $c_T(e) = c_T(v)$. If a cluster has size two or three, we call it, respectively, a *cherry* or a *triple*. A cluster of size four which contains two distinct cherries is called a *fork*. In the rooted phylogenetic tree T_1 of Fig. 1 the set $\{x_1, x_2, x_3\}$ is a triple, the set $\{x_4, x_5, x_6, x_7\}$ is a fork, and each of $\{x_1, x_2\}$, $\{x_4, x_5\}$ and $\{x_6, x_7\}$ is a cherry.

Phylogenetic diversity. The edges of every rooted phylogenetic tree considered in this paper are positively weighted. Let T be a rooted phylogenetic X-tree, and let $\ell : E(T) \to \mathbb{R}^{>0}$ be a function which assigns a positive real-valued length $\ell(e)$ to each edge $e \in E(T)$. Suppose that $u, v \in V(T)$ are two vertices of T connected by a directed path from u to v (this path is unique if it exists). Then the *distance from u to v*, denoted $d(u, v)$, is the sum of the lengths of the edges in this path. If an edge e is subdivided into two edges e_1 and e_2, we require $\ell(e_1) + \ell(e_2) = \ell(e)$. If ℓ is such that for any two distinct leaves x and y we have $d(\rho, x) = d(\rho, y)$, we say that ℓ satisfies the *ultrametric* condition.

For a non-empty subset Y of X, we define the *phylogenetic diversity* of Y on T, denoted by $PD_{(T,\ell)}(Y)$, to be the sum of the edge lengths of the Y-subtree. That is,

$$PD_{(T,\ell)}(Y) = \sum_{\substack{e \in E(T): \\ c_T(e) \cap Y \neq \emptyset}} \ell(e).$$

It will be usual for us to remove the subscript notation and write $PD(Y)$ when it is clear which rooted phylogenetic tree and edge length function we refer to. We also write $PD(T)$ to denote the phylogenetic diversity of the entire X-tree T, in place of $PD_{(T,\ell)}(X)$.

Let T be a rooted phylogenetic X-tree whose edges are assigned a positive real-valued weighting, and let $A \subseteq X$. If $|A| = k$ and $PD(A) \geq PD(Y)$ for all $Y \subseteq X$ with $|Y| = k$, then we call A a *size-k maxPD set*. Similarly, if $|A| = k$ and $PD(A) \leq PD(Y)$ for all $Y \subseteq X$ with $|Y| = k$, then we call A a *size-k minPD set*. To illustrate, Fig. 2 shows an example of a size-3 maxPD and an example of a size-3 minPD sets for the same rooted phylogenetic tree.

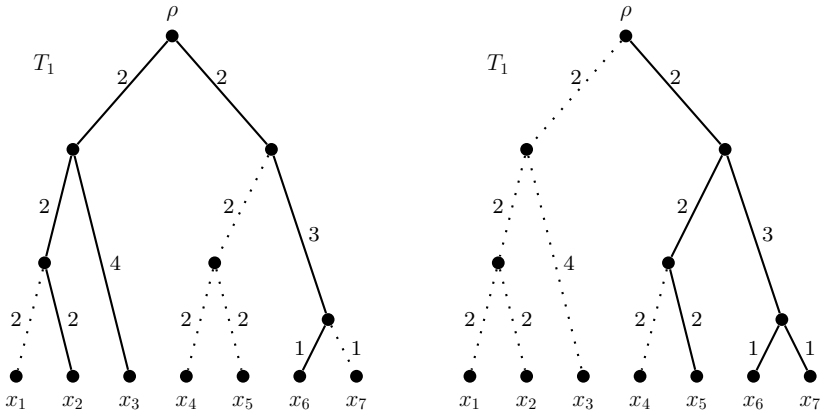

Fig. 2 A size-3 maxPD set $\{x_2, x_3, x_6\}$ and a size-3 minPD set $\{x_5, x_6, x_7\}$ for T_1. Solid lines indicate the $\{x_2, x_3, x_6\}$- and $\{x_5, x_6, x_7\}$-subtrees respectively. Hence $PD(\{x_2, x_3, x_6\}) = 16$, and $PD(\{x_5, x_6, x_7\}) = 11$.

3 The number of maxPD sets on rooted phylogenetic trees

Given a rooted phylogenetic X-tree T, with $|X| = n$ and a weighting on $E(T)$, a natural question is to find a subset Y of X of size t whose extinction minimises PD loss. The solution to this question is to take Y to be $X - W$, where W is a subset of X of size $n - t$ that maximises $PD(W)$. It turns out that a greedy algorithm provably constructs such sets W of $k = n - t$ leaves (Pardi and Goldman, 2005; Steel, 2005). This result relies on an underlying

combinatorial 'strong exchange property' that induces a greedoid structure on maximal PD sets of given size.

Although a greedy algorithm will output a maxPD set, it does not give a clear indication of how many distinct maxPD sets exist for T. Such an algorithm begins with an empty set of leaves and iteratively adds k leaves, based on which leaf adds most to the running total of PD at each iteration. There may be multiple steps at which a choice has to be made between equally-good options. By altering the procedure for breaking ties when they occur, it is possible to discover numerous size-k maxPD sets. This effect is most pronounced for rooted phylogenetic X-trees satisfying the ultrametric condition. For example, Fig. 1 and Fig. 2 show two of the 20 different size-3 maxPD sets for the rooted phylogenetic tree T_1.

All possible maxPD sets can be obtained by using a greedy algorithm, by taking each option separately when presented with ties (Steel, 2005, Theorem 1). However, this process can become quite involved even for small phylogenetic trees. Moreover, each maxPD set could be counted multiple times, as greedy algorithms sometimes select the same set of leaves in different orders.

In this section, we present a more straightforward method for determining exactly how many maxPD sets exist on a given rooted phylogenetic X-tree whose edge lengths satisfy the ultrametric condition. Firstly, by deleting certain edges near the root vertex, we partition the leaf set into disjoint subsets. Then we use a generating function which takes the sizes of these subsets and outputs the number of maxPD sets as a coefficient.

3.1 Counting maxPD sets in an ultrametric context

We restrict our attention to the problem of counting maxPD sets on a rooted phylogenetic X-tree T whose edge lengths satisfy the ultrametric condition. Suppose $T = (V, E)$, and let $1 \leq k \leq |X|$. It turns out that the minimal subtrees of T connecting size-k maxPD sets to the root all contain particular subsets of edges. Furthermore, these common edges induce a subtree of T containing the root vertex. Our approach is to determine which edges of T will be in common to all size-k maxPD sets. From there, we can enumerate these maxPD sets by analysing the forest that results from deleting the common edges.

For example, all twenty size-3 maxPD sets of T_1 from Fig. 1 (with a score of 16) can be found by checking the 35 possible sets of 3 leaves. Comparing these, we see that all of the minimal subtrees of T_1 that connect a size-3 maxPD set to the root of T_1 contain both edges incident with the root, as well as exactly 3 out of the 4 edges descending from the two highest non-root vertices.

We extend the metaphor that the ultrametric condition produces clock-like trees and consider time to run down the page. Vertices at the same height are therefore contemporary and, in particular, the leaves are in the present. Let d be a non-negative real number and let

$$R(d) = \{v \in V : d(v, x) \leq d \text{ for some } x \in X\}$$

be the set of *recent* vertices of T that are at most d units of time from the present. Let $c(d)$ be the number of connected components in $T[R(d)]$. If there exists a distance d such that $c(d) = k$, we define $d_k = \min\{d \in \mathbb{R} : c(d) = k\}$. Note that d_k may not be defined for all $k < n$. However, if d_k is defined, we call k a *branching value*, and d_k a *branching distance*. In other words, d_k is the most recent time for which $T[R(d)]$ has exactly k connected components, if such a time exists. For example, the rooted phylogenetic tree T_2 in Fig. 3 has $\{1, 2, 4, 7, 9, 11\}$ as its set of branching values. The forests $T_2[R(d_4)]$ and $T_2[R(d_7)]$ are shown below T_2 in the same figure.

If k is not a branching value, we will be interested in the nearest integers which are. We write k^+ to denote the smallest branching value of at least k, and k^- to denote the largest branching value of at most k. Note that $k = 1$ and $k = |X|$ are branching values, so that k^+ and k^- are well-defined. If k is a branching value, then $k^- = k = k^+$. Theorem 1 gives a characterisation of maxPD sets of a rooted phylogenetic tree T in terms of the forests $T[R(d_{k^-})]$ and $T[R(d_{k^+})]$.

We first prove Lemma 1. Let $X = \{x_1, \ldots, x_n\}$. We define $T'_k = (V', E')$ to be the rooted tree derived from T (by adding vertices to subdivide edges as necessary) where, for each $x_i \in X$, there is a vertex v_i on the path ρ to x_i for which $d(v_i, x_i) = d_{k^-}$. Since T'_k is derived from T solely by subdivision of edges, $PD_T(A) = PD_{T'_k}(A)$. Let $V'_{Top}(k) = \{v \in V' : d(\rho, v) \leq d_1 - d_{k^-}\}$, and let $\hat{T}_k$ be the rooted tree $T'_k[V'_{Top}(k)]$.

Lemma 1 *Let T be a rooted phylogenetic X-tree whose edge lengths satisfy the ultrametric condition, and let $A \subseteq X$ with $|A| = k$. Then*

$$PD_T(A) \leq PD(\hat{T}_k) + kd_{k^-}.$$

Proof Let $A = \{x_1, \ldots, x_k\}$ be a size-k subset of X. Each element x_i of A contributes at most $d(\rho, x_i)$ to the total of $PD_{T'_k}(A)$. We separate the path from ρ to x_i within T'_k into two parts at vertex v_i. Hence

$$d(\rho, x_i) = d(\rho, v_i) + d(v_i, x_i) = d(\rho, v_i) + d_{k^-}.$$

For all $1 \leq i \leq k$, the path from ρ to v_i lies within $\hat{T}_k$. Therefore the total contribution of these paths to $PD_{T'_k}(A)$ cannot exceed $PD(\hat{T}_k)$. This means $PD_{T'_k}(A)$ must be less than or equal to $PD(\hat{T}_k)$ plus a contribution of (at most) d_{k^-} from each of the k elements of A. Thus $PD_{T'_k}(A) \leq PD(\hat{T}_k) + kd_{k^-}$, and the lemma holds. □

Lemma 2 *Let T be a rooted phylogenetic X-tree whose edge lengths satisfy the ultrametric condition. Let $A \subseteq X$ with $|A| = k$, and let d be a branching distance of T. If one component of $T[R(d)]$ contains no members of A, while a second component of $T[R(d)]$ contains two or more distinct members of A, then A is not a size-k maxPD set.*

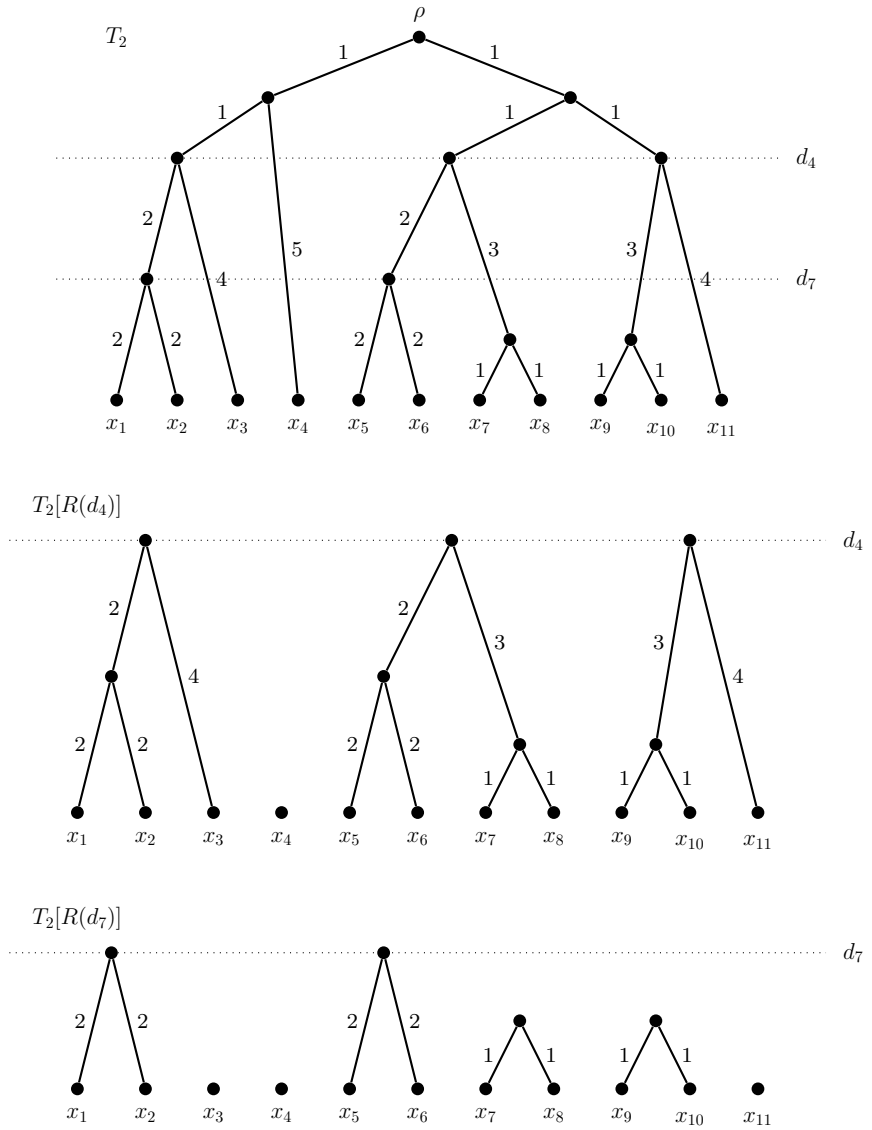

Fig. 3 A rooted phylogenetic tree T, and the forests $T[R(d_4)]$ and $T[R(d_7)]$ corresponding to the branching values 4 and 7. The branching distances d_4 and d_7 are indicated by horizontal dotted lines.

Proof Assume some component of $T[R(d)]$ contains two (distinct) leaves, say x_1, x_2, of A. The PD contribution of adding x_1 to $A - x_1$ cannot exceed d because all edges of T in the path from ρ to x_2 have already been counted towards $PD(A - x_1)$. In particular, $PD(A) - PD(A - x_1) \leq d$.

Now let y be a leaf in a component of $T[R(d)]$ which contains no member of A. The shortest defined distance from a vertex in the $(A - x_1)$-subtree to y must exceed d. (If not, there would be some vertex of the $(A - x_1)$-subtree in the same component of $T[R(d)]$ as y.) Hence $PD((A - x_1) \cup y) > PD(A)$. Since $|(A - x_1) \cup y| = |A|$, the set A is not a size-k maxPD set. □

Theorem 1 *Let T be a rooted phylogenetic X-tree whose edge lengths satisfy the ultrametric condition. Let $A \subseteq X$, and let $|A| = k$. Then A is a size-k maxPD set if and only if A contains at least one leaf from each component of $T[R(d_{k^-})]$, and at most one leaf from each component of $T[R(d_{k^+})]$.*

Proof First suppose that A is a size-k maxPD set. Assume some component of $T[R(d_{k^+})]$ contains two (distinct) leaves, say x_1,x_2, of A. Since $k^+ \geq k$, there must be some component of $T[R(d_{k^+})]$ which has no leaf in A. Then, by Lemma 2, the set A cannot be a maxPD set, contradicting our initial supposition. Thus A contains at most one leaf from each component of $T[R(d_{k^+})]$.

Next, assume that some component of $T[R(d_{k^-})]$ contains no element of A. Since $k^- \leq k$, there is a component of $T[R(d_{k^-})]$ that contains two or more leaves of A. Then, by Lemma 2, the set A cannot be a maxPD set, again contradicting our initial supposition. Thus A contains at least one leaf from each component of $T[R(d_{k^-})]$.

Now suppose that $A = \{x_1, \ldots, x_k\}$ contains at least one leaf from each component of $T[R(d_{k^-})]$, and at most one leaf from each component of $T[R(d_{k^+})]$. By Lemma 1, the value $PD(\hat{T}_k) + kd_{k^-}$ is an upper bound for the PD score of size-k sets. We show that $PD(A)$ achieves this bound.

Notice that the components of $T[R(d_{k^-})]$ match those of $T'_k[R(d_{k^-})]$, and the components of $T[R(d_{k^+})]$ match those of $T'_k[R(d_{k^+})]$, in terms of their constituent leaves. For each $x_i \in A$ there exists a vertex $v_i \in V'$ for which $d(v_i, x_i) = d_{k^-}$. Since each component of $T'_k[R(d_{k^+})]$ contains at most one leaf, the paths from v_i to x_i, and from v_j to x_j contain no common edges, for any distinct $1 \leq i, j \leq k$. So in total, the collection of all paths v_i to x_i for all i contributes exactly kd_{k^-} to $PD(A)$. Furthermore, since A contains at least one leaf from each component of $T[R(d_{k^-})]$, every edge of $\hat{T}_k$ is included in the A-subtree of T. Thus $PD_T(A) = PD_{T'_k}(A) = PD(\hat{T}_k) + kd_{k^-}$, the maximum possible value, and hence A is a size-k maxPD set. □

When k is a branching value for a rooted phylogenetic X-tree T, then $k^- = k = k^+$ and $T[R(d_{k^-})] = T[R(d_k)] = T[R(d_{k^+})]$. Hence, as a direct consequence of Theorem 1, we obtain the following result.

Corollary 1 *Let T be a rooted phylogenetic X-tree whose edge lengths satisfy the ultrametric condition. Let $A \subseteq X$ with $|A| = k$. If k is a branching value of T, then A is a size-k maxPD set if and only if A contains exactly one leaf from each component of $T[R(d_k)]$.*

Theorem 1 can be used to count the number of size-k maxPD sets for a rooted phylogenetic X-tree T whose edge lengths are ultrametric. Note that

this result does not require T to be binary. Let $m(T,k)$ denote the number of size-k maxPD sets on T. The next proposition derives $m(T,k)$ when k is a branching value of T. The case when k is not a branching value of T is covered separately. We express the forest $T[R(d_k)]$ as a union of components $\kappa_i(k)$ for $i \in \{1,2,\ldots,k\}$, and write $\lambda(\kappa_i(k))$ for the number of leaves in $\kappa_i(k)$.

Proposition 1 *Let T be a rooted phylogenetic X-tree whose edge lengths satisfy the ultrametric condition. If k is a branching value for T, then*

$$m(T,k) = \prod_{i\in\{1,\ldots,k\}} \lambda(\kappa_i(k)).$$

Proof By Corollary 1, each maxPD set contains exactly one leaf from component $\kappa_i(k)$, for all $i \in \{1,2,\ldots,k\}$. There are exactly $\lambda(\kappa_i(k))$ ways to choose one leaf from component $\kappa_i(k)$. Since the choice in each component is independent of the choices in all other components, $m(T,k) = \prod_{i\in\{1,\ldots,k\}} \lambda(\kappa_i(k))$. □

Example. Mazel et al. (2018) exhibit a phylogeny of 32 mammal families (reprinted as Fig. 4). Let us call this phylogeny P. We calculate the number of size-8 and size-16 maxPD sets for P.

The family *Leporidae* appears as a single vertex component in both $P[R(d_8)]$ and $P[R(d_{16})]$. For $P[R(d_8)]$, the components appearing clockwise, starting from *Leporidae*, have sizes 1, 8, 3, 2, 7, 3, 5, and 3. The product of these values gives a total of 15120 size-8 maxPD sets for P. Note that this represents 0.14% of the possible sets of 8 leaves. For $P[R(d_{16})]$, the components appearing clockwise, starting from *Leporidae*, have sizes 1, 2, 4, 1, 1, 3, 1, 1, 2, 1, 4, 3, 5, 1, 1, and 1. This gives a total of 2880 size-16 maxPD sets for P.

If k is not a branching value for a rooted phylogenetic X-tree T, calculating the number of size-k maxPD sets is not as immediate. In this case, we use a generating function to determine $m(T,k)$. The following lemma is presented for a more general context.

Lemma 3 *Suppose that $\mathcal{C} = (X_{ij} : i = 1,\ldots,n_j; j = 1,\ldots,r)$ is an array of disjoint sets, and let n_{ij} denote the size of set X_{ij}. Let $N_{\mathcal{C}}(k)$ be the number of sets of size k that can be obtained by selecting at most one element from each set X_{ij} but in such a way that at least one element is selected from $\bigcup_i X_{ij}$ for each value of j. Then $N_{\mathcal{C}}(k)$ is the coefficient of x^k in the polynomial*

$$p_{\mathcal{C},k}(x) = \prod_{j=1}^{r} \left(-1 + \prod_{i=1}^{n_j} (1 + n_{ij}x) \right). \tag{1}$$

Proof For each integer $j \geq 1$, let $p_j(x) = -1 + \prod_{i=1}^{n_j}(1 + n_{ij}x)$, and for each $l \geq 0$, let c_{lj} denote the coefficient of x^l in $p_j(x)$. Then $c_{0j} = 0$ for each j, and for $l > 0$, the coefficient c_{lj} is the number of ways of selecting l elements from $\bigcup_{i=1}^{n_j} X_{ij}$ in such a way that at most one element is selected from the

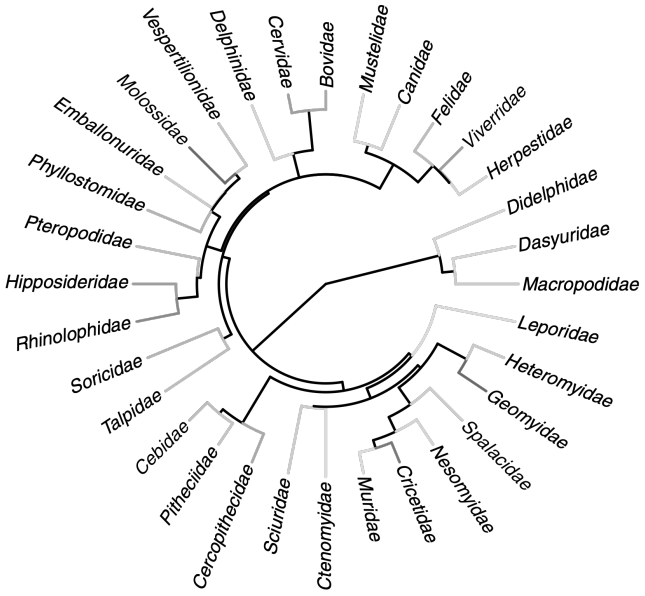

Fig. 4 A phylogeny of 32 mammalian families, appearing originally as Fig. 3a in Mazel et al. (2018).

(pairwise-disjoint) sets $(X_{ij} : i = 1 \ldots, n_j)$ and at least one element is selected (since $l > 0$).

Now $p_{\mathcal{C},k}(x) = \prod_{j=1}^{r} p_j(x)$ and so the coefficient of x^k in $p_{\mathcal{C},k}(x)$ is the sum (call it S_k) of the terms $c_{l_1,1}c_{l_2,2}\cdots c_{l_r,r}$ across all choices of $(l_1, l_2, \ldots, l_r)$ for which $l_1 + l_2 + \cdots + l_r = k$, and $l_m > 0$ for all m (this second condition holds because $c_{0j} = 0$ for all j). Since the sets X_{ij} are pairwise-disjoint (across all choices of i, j), we have $S_k = N_{\mathcal{C}}(k)$ as required. □

Let T be a rooted phylogenetic X-tree, and let k be a positive integer such that $k \leq |X|$. We write $p_{T,k}(x)$ instead of $p_{\mathcal{C},k}$ when $\mathcal{C}$ is constructed from component-connected clusters of T, using the branching values k^- and k^+. If k is a branching value of T, we have $n_j = 1$ for all j, and the result coincides with Proposition 1. In the general case, Lemma 1 gives a polynomial-time algorithm to compute $m(T, k)$.

Theorem 2 *Let T be a rooted phylogenetic X-tree whose edge lengths satisfy the ultrametric condition. Let $|X| = n$, and let $k \leq n$. The components of $T[R(d_{k^-})]$ and $T[R(d_{k^+})]$ can be determined in time $O(n^2)$. The value $m(T, k)$ can be computed in time $O(n^3)$.*

Proof There are at most n different branching values for T (one for every non-leaf vertex, and the value n) from which to select the appropriate k^- and k^+ values. Determining the components of a forest can be achieved in $O(n^2)$ time. Once the components have been determined, the polynomial $p_{T,k}(x)$ is

calculated, and the coefficient of x^k is extracted. With a naïve approach of sequentially multiplying factors, this can be completed in time $O(n^3)$. □

The following example highlights a nice property of the generating function $p_{T,k}(x)$ for a rooted phylogenetic tree T. Calculating the number of size-k maxPD sets for some non-branching value k also gives the number of size-m maxPD sets for every positive integer m in the interval $[k^-, k^+]$.

Example. Consider the rooted phylogenetic tree T_2 in Fig. 3. Firstly, 4 is a branching value for T_2. Thus the number of its size-4 maxPD sets is the product of the number of leaves in each of the four components of $T_2[R(d_4)]$. That is, $m(T_2, 4) = 3 \cdot 1 \cdot 4 \cdot 3 = 36$.

However, 5 is not a branching value for T_2, so we use the generating function approach to find $m(T_2, 5)$. First note that the greatest branching value less than 5 is 4 and the least branching value greater than 5 is 7. The forests $T_2[R(d_4)]$ and $T_2[R(d_7)]$ are shown in Fig. 3. We then construct an array of disjoint sets $\mathcal{C}(T_2)$, with a view to using Eqn. 1. Each entry of $\mathcal{C}(T_2)$ consists of a set of leaves contained in one component of $T_2[R(d_7)]$. That is, the entries of $\mathcal{C}(T_2)$ are $\{x_1, x_2\}, \{x_3\}, \{x_4\}, \{x_5, x_6\}, \{x_7, x_8\}, \{x_9, x_{10}\}, \{x_{11}\}$.

Lastly, we arrange the entries of $\mathcal{C}(T_2)$ so that each column contains precisely those leaves that share a component of $T_2[R(d_4)]$. Thus we have 4 columns in $\mathcal{C}(T_2)$, and set $r = 4$ in Eqn. 1. As T_2 is binary, there are at most two components of $T_2[R(d_7)]$ contained in any component of $T_2[R(d_4)]$. We use an empty set as a placeholder, if required, to ensure that $\mathcal{C}(T_2)$ is a rectangular array. As such, we are able to set $n_j = 2$ for all j in Eqn. 1. The completed array is

$$\mathcal{C}(T_2) = \begin{bmatrix} \{x_1, x_2\} & \{x_4\} & \{x_5, x_6\} & \{x_9, x_{10}\} \\ \{x_3\} & \emptyset & \{x_7, x_8\} & \{x_{11}\} \end{bmatrix}.$$

The generating function for T_2, when $k = 5$, is calculated below.

$$\begin{aligned} p_{T_2,5}(x) &= \prod_{j=1}^{4} \left(-1 + \prod_{i=1}^{2} (1 + n_{ij}x) \right) \\ &= [-1 + (1+2x)(1+x)]^2[-1 + (1+x)(1)][-1 + (1+2x)^2] \\ &= x^4(2x+3)^2(4x+4) \\ &= 16x^7 + 64x^6 + 84x^5 + 36x^4 \end{aligned}$$

Hence T_2 has 84 maxPD sets of size 5. We have also determined that T_2 has 64 size-6 maxPD sets, 16 of size 7, and confirmed that there are 36 maxPD sets of size 4.

3.2 Bounding $m(T, k)$, and its value for a certain family of trees

The shape of a rooted phylogenetic tree impacts the components at each branching distance, and hence the number of maxPD sets which exist. In this section we restrict ourselves to rooted binary phylogenetic trees with the ultrametric constraint on edge lengths. By 'shape', we refer to both the particular branching structure of a tree and the relative distances of the vertices from its leaves (see Steel, 2016, Ch. 3). Here, we begin to address the question of which tree shapes and values of k give the most size-k maxPD sets across a fixed number of leaves. First we consider the lower and upper bounds for the number of size-k maxPD sets when k is a branching value.

Proposition 2 *Let T be a rooted binary phylogenetic X-tree whose edge lengths satisfy the ultrametric condition, and let $|X| = n$. If k is a branching value for T, then*

$$n - k + 1 \leq m(T, k) \leq \left(\frac{n}{k}\right)^k.$$

Moreover, these bounds are sharp.

Proof If k is a branching value for T, then, by Proposition 1, the value $m(T, k)$ is the product of the number of leaves in the k components. Let S be the multiset $\{\lambda(\kappa_i(k)) : 1 \leq i \leq k\}$, that is, the multiset containing the number of leaves of each component.

To find the lower bound for $m(T, k)$, note that if a and b are integers with $1 < a \leq b$, then $(a - 1)(b + 1) < ab$. Hence the product of elements of the multiset $(S - \{a, b\}) \cup \{a - 1, b + 1\}$ will be less than the product of elements of S. This exchange of elements can continue until only one element is greater than 1. Thus the minimum product of k positive integers which sum to n is $n - k + 1$, achieved with one value of $n - k + 1$ and $k - 1$ values of 1. This bound is achieved by rooted caterpillar trees (i.e. rooted phylogenetic X-trees with exactly one cherry).

On the other hand, the maximum such product is bounded above by $(\frac{n}{k})^k$. This follows from the fact that the arithmetic mean, $AM(S)$, of a multiset of positive integers S is greater than or equal to the geometric mean, $GM(S)$, of the same multiset. Thus

$$m(T, k) = \prod_{s \in S} s = (GM(S))^k \leq (AM(S))^k = \left(\frac{n}{k}\right)^k.$$

This maximum is obtained when k is a divisor of n and all components contain $\frac{n}{k}$ leaves. □

Let T be a rooted binary phylogenetic X-tree. If k is not a branching value, it is possible that $m(T, k)$ exceeds the upper bound given in Proposition 2. For example, the tree T_2 from Fig. 3 has $n = 11$, and $m(T_2, 5) = 84 \geq \left(\frac{11}{5}\right)^5 \approx 51.5$.

We have seen above that if T is a rooted caterpillar tree, then $m(T, k)$ is as small as possible. The highly asymmetric structure of caterpillar trees restricts

the possible maxPD sets they contain. In contrast, we now consider $m(T,k)$ values across the family of fully symmetric rooted trees (with constant edge lengths) and include cases when k is not a branching value.

We say T is a *perfect unit-length* tree if the edge lengths of T satisfy the ultrametric condition, and all edges of T have length 1. Perfect unit-length trees have 2^α leaves, where $\alpha \in \mathbb{N}$ is the *height* of the tree (the number of edges between the root and any leaf).

Proposition 3 *Let T be a perfect unit-length tree of height $\alpha \in \mathbb{N}$, and let n denote the number of leaves of T. Let k be a positive integer such that $k \leq n$, and let β be the unique non-negative integer such that $2^{\beta-1} < k \leq 2^\beta$. Then*

$$m(T,k) = \binom{2^{\beta-1}}{k-2^{\beta-1}} \cdot 2^{2^\beta+(\alpha-\beta-1)k}. \tag{2}$$

The values of k that maximise $m(T,k)$ are $k = \lfloor \frac{2n}{3} \rfloor$ for all n and, additionally, $k = \lfloor \frac{2n}{3} \rfloor + 1$ when $n \equiv 1 \pmod 3$.

Proof Firstly, if k is a branching value, then $k = 2^\beta$ and each component has size $2^{\alpha-\beta}$. Therefore, by Proposition 1, $m(T,k) = (2^{\alpha-\beta})^k$, which coincides with Eqn. (2).

Furthermore, if k is not a branching value, we have $k^- = 2^{\beta-1}$ and $k^+ = 2^\beta$. Then by Lemma 1, $m(T,k)$ is the coefficient of x^k in the polynomial

$$p_{T,k}(x) = (-1 + (1 + 2^{\alpha-\beta}x)^2)^{2^{\beta-1}} = (2^{2(\alpha-\beta)}x^2 + 2^{\alpha-\beta+1}x)^{2^{\beta-1}}.$$

Taking the binomial expansion of the last expression we determine that $\binom{2^{\beta-1}}{k-2^{\beta-1}} \cdot 2^{2^\beta+(\alpha-\beta-1)k}$ is the coefficient of x^k in $p_{T,k}(x)$. This establishes Eqn. (2).

To find the value of k which maximises $m(T,k)$, we first show that $m(T,k) \leq m(T,n-k)$ when $k \leq \frac{n}{2}$. Let A be a size-k maxPD set for some $k \leq \frac{n}{2}$. By Theorem 1, A contains at most one leaf from each cherry of T. Then $X-A$ contains at least one leaf from every cherry (which are the components of $T[R(d_{(n-k)^-})]$), and at most one leaf from each component of $T[R(d_{(n-k)^+})]$, as these are all single-leaf components. This implies $X-A$ is a size-$(n-k)$ maxPD set by Theorem 1, and thus there are at least as many size-$(n-k)$ maxPD sets as size-k ones. Hence the value of k that maximises $m(T,k)$ will be greater than or equal to $\frac{n}{2}$.

In the case when $k \geq \frac{n}{2}$, since $2^\alpha = 2^\beta = n$, the expression in Eqn. (2) simplifies to $m(T,k) = \binom{\frac{n}{2}}{k-\frac{n}{2}} \cdot 2^{n-k}$. Computing the ratio $\frac{m(T,k+1)}{m(T,k)}$, we have

$$\frac{m(T,k+1)}{m(T,k)} = \frac{\binom{\frac{n}{2}}{k-\frac{n}{2}+1} \cdot 2^{n-k-1}}{\binom{\frac{n}{2}}{k-\frac{n}{2}} \cdot 2^{n-k}} = \frac{n-k}{2k-n+2}.$$

This ratio is monotonically decreasing as k increases, and equals 1 when $3k = 2n-2$. Our maximal value of $m(T,k)$ will be found at the smallest $k \geq \frac{n}{2}$ for which $\frac{m(T,k+1)}{m(T,k)} \leq 1$, namely when $k = \lfloor \frac{2n}{3} \rfloor$. Note that when $n \equiv 1$

n	k	$m(T,k)$
4	1,2,3	4
8	3,5	32
16	10,11	1,792
32	21	8,945,664
64	42,43	$\sim 2.7 \times 10^{14}$

Table 1 Number of size-k maxPD sets for a perfect unit-length tree T with n leaves

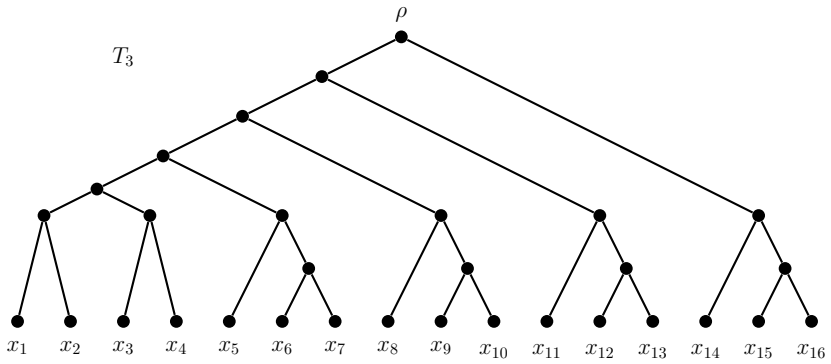

Fig. 5 A rooted binary phylogenetic tree with more maxPD sets (for its optimal value of $k = 8$) than the perfect unit-length tree with the same number of leaves (for its optimal value of $k = 10, 11$)

.

(mod 3) and $k = \lfloor \frac{2n}{3} \rfloor$, we have $\frac{m(T,k+1)}{m(T,k)} = 1$, so we get an equal number of maxPD sets for the two consecutive values k and $k+1$. □

Table 1 shows the growth of $m(T,k)$ as n increases. We note that for $n = 16$, the perfect unit-length tree does not provide the largest value of $m(T,k)$. Figure 5 shows a rooted binary phylogenetic X-tree T_3 on 16 leaves which contains 1809 size-8 maxPD sets (thus having 17 more maxPD sets than the perfect unit-length tree on 16 leaves can achieve for its optimal value of $k = 10$ or $k = 11$). For T_3 we have $p_{T_3,8}(x) = (x^2 + 2x)^2(2x^2 + 3x)^4$.

4 Finding a maxPD set that maximises a linear function on the leaves

Section 3 presented methods for determining the number of size-k maxPD sets for a given rooted phylogenetic tree. These methods confirmed the observations in the literature that, in general, maxPD sets are far from unique. This provides scope for evaluating the collection of maxPD sets against other strategic considerations. In developing strategies for conservation planning, PD is often seen as one measure to be used in conjunction with others (for examples of this, see Cadotte and Tucker (2018); Isaac et al. (2007); Kling

et al. (2019)). For instance, we may wish to incorporate benefit-cost ratios of focussed conservation spending, or employ IUCN categorisations into the analysis. This section provides an algorithm to optimise a further measure across the collection of maxPD sets.

Here, we frame the further measure in terms of a real-valued linear function on the leaves. Each leaf is assigned a function value, and the *linear function score* of a set of leaves is the weighted sum of the function values of the constituent leaves. We seek a size-k set which has as large a linear function score as possible among the size-k maxPD sets. By suitably modifying the linear function, the problem can be rephrased as maximising the unweighted sum across maxPD sets. Thus, for a function $\phi : X \to \mathbb{R}$ we want to determine

$$\max\left\{\sum_{x\in A}\phi(x) : A \text{ is a size-}k \text{ maxPD set}\right\}.$$

We note that it is not always possible to achieve this result by simply adding the function score of each leaf to the length of its incident pendant edge, and then finding a size-k maxPD set of the resulting rooted phylogenetic tree. We provide a counterexample using the tree T_1 from Fig. 2. Consider the function

$$f(x) = \begin{cases} 1, & \text{if } x \in \{x_1, x_2, x_3, x_4\}; \\ 100, & \text{if } x \in \{x_5, x_6, x_7\}. \end{cases}$$

Adding the function values to appropriate pendant edges, results in a tree with a unique size-3 maxPD set $\{x_5, x_6, x_7\}$. However this set is not a maxPD set of the original tree T_1.

For a rooted phylogenetic tree T, MaximiseLinearSum selects a set A consisting of k leaves of T in the following manner. Initially, it determines the components of $T[R(d_{k^-})]$ ('tall' components) and those of $T[R(d_{k^+})]$ ('short' components). For every short component, it keeps (in the set of 'potential' leaves P) one leaf x such that $\phi(x)$ is maximal for that short component. It then discards all other leaves from further consideration. In every tall component, it adds one leaf x to A from the leaves retained in P such that $\phi(x)$ is maximal for that tall component. Finally, from the remaining $k^+ - k^-$ leaves under consideration, it chooses $k - k^-$ with the largest ϕ values. In presenting the algorithm, we make use of the following notation. For a pendant subtree C of T, write X_C for the set of leaves in C. For $S \subseteq X$, let $\phi(S) = \{\phi(x) : x \in S\}$.

Proposition 4 *Let T be a rooted phylogenetic X-tree whose edge lengths satisfy the ultrametric condition. The* MaximiseLinearSum *algorithm outputs a maxPD set of T.*

Proof The for-loop from Lines 4 to 7 ensures that A cannot contain more than one leaf from any short component. The for-loop from Lines 10 to 13 ensures that A contains at least one leaf from every large component. Since Line 15 ensures that $|A| = k$, it follows by Theorem 1, that A is a maxPD set of T. □

Algorithm 1: MaximiseLinearSum

Input: a rooted phylogenetic X-tree T whose edge lengths satisfy the ultrametric condition,
a positive integer $k \leq |X|$,
$\phi : X \to \mathbb{R}$

Output: a size-k maxPD subset $A \subseteq X$, with the largest linear function score among all maxPD sets

```
1  determine T[R(d_{k^-})] and T[R(d_{k^+})];
2  P ← ∅ ;                          /* Potential leaves to include */
3  A ← ∅ ;                                        /* Output set */
4  foreach component C in T[R(d_{k^+})] do
5  |  choose one leaf m from the set {x ∈ X : φ(x) = max φ(X_C)};
6  |  P ← P ∪ m
7  end
8  foreach component C in T[R(d_{k^-})] do
9  |  choose one leaf m from the set {x ∈ P : φ(x) = max φ(X_C ∩ P)};
10 |  A ← A ∪ m;
11 |  P ← P − m
12 end
13 for each of the k − k^- largest elements φ(x) of φ(P) add x to A;
14 return A
```

Proposition 5 *Let T be a rooted phylogenetic X-tree whose edge lengths satisfy the ultrametric condition, and let $\phi : X \to \mathbb{R}$ be a function on the leaves of T. Let k be a positive integer such that $k \leq |X|$. Then* MaximiseLinearSum *applied to T, ϕ, and k correctly outputs a size-k maxPD set with the largest function score among all maxPD sets.*

We give a proof of this result shortly, but first give a short description of our approach. The algorithm MaximiseLinearSum was designed to construct a set containing the largest possible values of ϕ while obeying the constraints imposed by Theorem 1 to ensure the selection of a maxPD set. Suppose that A is a size-k maxPD set of T. The proof considers two possible cases when A is not a valid output of the algorithm, and exhibits a size-k maxPD set with a greater linear function score in each. Finally, outside of these two cases we prove that the linear function score of A must be at least as large as that of any other size-k maxPD set.

Proof Suppose that A is a size-k maxPD set of T. For the result to hold, either A is a valid output of MaximiseLinearSum or there is a size-k maxPD set B, distinct from A, such that $\sum_{x \in A} \phi(x) < \sum_{x \in B} \phi(x)$. One of the following three conditions holds:

1. There is a component of $T[R(d_{k^+})]$ (a short component) which contains leaf $a \in A$ and leaf $b \in B$, where $\phi(a) < \phi(b)$. In this case, a would not

be selected in Line 5 of MaximiseLinearSum, meaning A cannot be a valid output of this algorithm. However, the set $A' = (A - a) \cup b$ is a size-k maxPD set with $\sum_{x \in A} \phi(x) < \sum_{x \in A'} \phi(x)$.

2. Condition 1 fails, and there is a component of $T[R(d_{k^-})]$ (a tall component) which contains leaves $\{a_1, a_2, \ldots, a_s\} \subseteq A$, and $\{b_1, b_2, \ldots, b_t\} \subseteq B$ where, for some $j \in \{1, 2, \ldots, t\}$, the inequality $\phi(b_j) > \phi(a_i)$ holds across all $i \in \{1, 2, \ldots, s\}$. In particular, the strictness of this inequality means that $b_j \notin A$. In this case, the leaf b_j would always be selected by Line 9 of MaximiseLinearSum, precluding A from being a valid output of this algorithm. Moreover, since Condition 1 fails to hold, no element of $\{a_1, a_2, \ldots, a_s\}$ shares a short component with b_j. Thus $A' = (A - a_1) \cup b_j$ is a size-k maxPD set with $\sum_{x \in A} \phi(x) < \sum_{x \in A'} \phi(x)$.
3. Conditions 1 and 2 fail. Thus, in every component of $T[R(d_{k^-})]$, the set A contains a leaf that has the maximal ϕ value for that component. Assume that k is a branching value of T. Then A is a valid output of MaximiseLinearSum, as the choice applied in Line 9 can be the one element of A from within each component. Additionally, since elements of A have the maximal ϕ value in each component, $\sum_{x \in A} \phi(x) \geq \sum_{x \in B} \phi(x)$ for any size-k maxPD set B. Thus the proposition holds when k is a branching value of T.

   Now assume that k is not a branching value. Let $\bar{A} \subseteq A$ consist of k^- elements of A which have the maximal ϕ value in their tall component, one from each tall component. We construct the set $\bar{B} \subseteq B$ to include (i) elements of B that share a short component with some leaf in $\bar{A}$, and (ii) from tall components where no leaf in B satisfies Condition (i), one element of B in each such tall component with the largest ϕ value. The set $\bar{B}$ contains exactly one leaf from each tall component. For $a \in \bar{A}$ and $b \in \bar{B}$ in the same tall component, $\phi(a) \geq \phi(b)$. Thus

$$\sum_{x \in \bar{A}} \phi(x) \geq \sum_{x \in \bar{B}} \phi(x). \tag{3}$$

   Let P be the set of 'potential leaves' as used in Algorithm 1. Then by our construction of $\bar{B}$, we have $B - \bar{B} \subseteq P - \bar{A}$. The set A is a valid output of MaximiseLinearSum if and only if the elements of $A - \bar{A}$ have the $k - k^-$ largest ϕ values among elements of $P - \bar{A}$. The latter condition is equivalent to $\sum_{x \in A - \bar{A}} \phi(x) \geq \sum_{x \in B - \bar{B}} \phi(x)$, that is $\sum_{x \in A} \phi(x) \geq \sum_{x \in B} \phi(x)$.

Hence, under all three conditions, either A is a valid output of MaximiseLinearSum or $\sum_{x \in A} \phi(x) < \sum_{x \in B} \phi(x)$ for some size-k maxPD set B of T, as required. □

Proposition 6 *Let T be a rooted phylogenetic X-tree whose edge lengths satisfy the ultrametric condition, and let $|X| = n$. Then* MaximiseLinearSum *runs in time $O(n^2)$.*

Proof By Theorem 2, Line 1 can be completed in $O(n^2)$. We show that this subroutine dominates the time taken for MaximiseLinearSum to run.

Determining which vertices are in each component can be achieved by a depth-first search in linear time. Both for-loops are completed in $O(n^2)$ time, as there are at most n components and a component contains at most n leaves. Sorting a set and returning the $k - k^-$ largest values can be achieved in $O(n \log n)$ time. Hence, MaximiseLinearSum runs in the same order of time as determining the components of $T[R(d_{k^-})]$ and $T[R(d_{k^+})]$. □

The algorithm MaximiseLinearSum makes use of the component constraints on maxPD sets to solve this problem for rooted phylogenetic X-trees whose edge lengths satisfy the ultrametric condition. For phylogenetic trees whose edge lengths do not satisfy the ultrametric condition, the determination of appropriate connected components requires a further algorithm (Manson, in preparation).

We note that an alternative approach to solving this more general problem comes from the area of lexicographic multi-objective linear programming (Cococcioni et al., 2018, see Section 2). The optimisation can be phrased as a max-flow min-cost problem, in a similar manner to that used in (Bordewich et al., 2009). However this approach relies on first scaling every length of the phylogenetic tree by a suitably large number. Determining an appropriate value for the scaling factor can prove difficult unless the edge lengths are restricted to take only rational values. For some trees with real-valued edge lengths this step requires a pairwise comparison of the PD scores across all sets of k leaves (Manson, in preparation).

5 Maximum possible loss of PD in a tree if k species become extinct ('minPD')

In Section 3, we were interested in finding sets of k species which contained as much diversity as possible. However, it is also worth considering the dual problem: determining how must PD could be lost if k extant species were to become extinct (i.e. a 'worst case scenario' in biodiversity conservation in the face of widespread extinction pressure). More precisely, we consider the problem of determining the *maximum* possible PD loss if a given number species were to become extinct.

Formally, let $T = (V, E)$ be a rooted phylogenetic X-tree and let the function $\ell : E(T) \to \mathbb{R}^{>0}$ assign a positive real-valued length $\ell(e)$ to each edge $e \in E(T)$. Suppose that each species in a subset Y of the leaf set X of T is removed from the tree. The resulting loss of PD, which we denote here as $\Delta_{(T,\ell)}(Y)$ is given by

$$\Delta_{(T,\ell)}(Y) = PD_{(T,\ell)}(X) - PD_{(T,\ell)}(X - Y).$$

This is equivalent to the concept of 'exclusive molecular phylodiversity' as described in Lewis and Lewis (2005).[1]

[1] The function $\Delta_{(T,\ell)}$ is a supermodular (and decreasing) function on the lattice of subsets of X, since PD is a submodular (and increasing) function on this same lattice (Steel, 2016).

Notice that finding a subset Y of X of size k' to maximise $\Delta_{(T,\ell)}(Y)$ is equivalent to finding a subset W $(= X-Y)$ of X of size $k = |X|-k'$ to minimise $PD_{(T,\ell)}(W)$. Unlike the max-PD question, this minimisation question is not solved by the greedy algorithm (Moulton et al., 2007). However, as discussed in Section 6 of (Spillner et al., 2008), minimal PD scores can be found using dynamic programming. In particular, (Blum et al., 1994, Section 3.1) describe an algorithm for an equivalent problem (referred to as the *i-tree problem*). Here we present a detailed description of this algorithm using the terminology of phylogenetic trees.

We call a set of k leaves which has the smallest PD score across all sets of size k, a *size-k minPD set*. In this section, we present a polynomial-time dynamic programming approach to finding minPD scores. For simplicity, we initially restrict our attention to rooted binary phylogenetic trees; however, we show that the same idea extends to rooted phylogenetic trees for which each vertex has bounded out-degree. Note that in this section, we do not require the branch lengths to satisfy the ultrametric condition.

Given a rooted phylogenetic X-tree T, and an integer $0 \leq k \leq |X|$, let $\varphi_T(k)$ be the minimum PD score across all size-k subsets of X. When $k > |X|$, $\varphi_T(k)$ is undefined, and when $k = 0$, we set $\varphi_T(0) = 0$. For the case when T is a single vertex, we define $\varphi_T(k) = 0$. Proposition 7 gives the dynamic programming equation when T is binary.

Proposition 7 *Let T be a rooted binary phylogenetic X-tree and let e_1 and e_2 be the two edges of T incident with the root. Let e_1 have length ℓ_1 and e_2 have length ℓ_2. Finally, let T_1 and T_2 denote the (maximal) pendant subtrees formed by the deletion of e_1 and e_2 respectively.*
For all $k \in \{1, 2, \ldots, |X|\}$,

$$\varphi_T(k) = \min_{\substack{k_1,k_2\geq 0,\\ k_1+k_2=k}} \{\varphi_{T_1}(k_1) + \varphi_{T_2}(k_2) + \ell_1 \cdot \mathbb{I}_{k_1>0} + \ell_2 \cdot \mathbb{I}_{k_2>0}\}, \qquad (4)$$

where $\mathbb{I}_{k_j>0}$ takes the value 1 if $k_j > 0$; otherwise, $\mathbb{I}_{k_j>0} = 0$.

Proof We proceed by induction on the number of vertices in T. For the base case, take the tree T consisting of a single vertex. Since T has no edges, it has a PD score of 0, which corresponds to $\varphi_T(k)$ for all $k \geq 0$ by definition.

Suppose that Eqn. (4) fails to give the minimum PD score for some rooted binary phylogenetic X-tree T. We write φ_i as shorthand for φ_{T_i} for $i = 1, 2$. Furthermore, suppose that $\varphi_i(k')$ equals the size-k' minPD score in T_i for all $k' \leq k$ and $i \in \{1, 2\}$. Since Eqn. 4 fails, there must be a set of k leaves of T which has a lower PD score than any value in the set

$$\{\varphi_1(k_1) + \varphi_2(k_2) + \ell_1 \cdot \mathbb{I}_{k_1>0} + \ell_2 \cdot \mathbb{I}_{k_2>0} : k_1, k_2 \geq 0, k_1 + k_2 = k\}.$$

Let A be such a set of k leaves of T, with k_1 leaves in T_1 and k_2 leaves in T_2.

If $k_2 = 0$, then $PD_T(A) = \ell_1 + PD_{T_1}(A) \geq \ell_1 + \varphi_i(k_1)$ by the inductive assumption. Thus the PD score of A is not lower than the calculated minimum; hence, $k_2 \neq 0$. Similarly, $k_1 \neq 0$. Consequently,

$$PD_T(A) = \ell_1 + \ell_2 + PD_{T_1}(A \cap T_1) + PD_{T_2}(A \cap T_2).$$

For A to have a PD score lower than $\varphi_T(k)$, we must have

$$PD_{T_1}(A \cap T_1) + PD_{T_2}(A \cap T_2) < \varphi_1(k_1) + \varphi_2(k_2).$$

This implies that either $PD_{T_1}(A \cap T_1) < \varphi_1(k_1)$ or $PD_{T_2}(A \cap T_2) < \varphi_2(k_2)$, contradicting our inductive assumption. Therefore, no such set A exists, and $\varphi_T(k)$ calculates a minPD score of size k in T. □

We now present an algorithm which utilises Proposition 7 to calculate a minPD score for a rooted binary phylogenetic X-tree $T = (V, E)$. For a vertex $v \in V(T)$, we use the notation $\varphi_v(k)$ in place of $\varphi_{T_v}(k)$, where T_v is the pendant subtree of T for which vertex v has in-degree 0. Additionally, $\varphi_\rho(k) = \varphi_T(k)$. Note that the root vertex ρ of T will always appear last in the ordered list L defined in the algorithm. For a positive integer i, let $L[i]$ denote the ith entry in list L.

Algorithm 2: MinPDScore

Input : a rooted binary phylogenetic X-tree $T = (V, E)$, with root ρ,
an integer $0 \leq k \leq |X|$.
Output: a real number $\varphi_T(k)$

foreach $x \in X$ **do**
  $\varphi_x(0) \leftarrow 0$;
  $\varphi_x(1) \leftarrow 0$
end
$L \leftarrow$ ordered list of vertices in $V(T) - X$ such that if u is a descendant of v, then u appears before v;
$i \leftarrow 1$;
$j \leftarrow 0$;
while $i < |V(T) - X|$ **do**
  foreach $0 \leq j \leq k$ **do**
    calculate $\varphi_{L[i]}(j)$ according to Eqn. (4) in Proposition 7;
  end
  $i \leftarrow i + 1$;
end
return $\varphi_\rho(k)$

The algorithm MinPDScore computes the minimum PD score for a rooted binary phylogenetic tree T when selecting k of its leaves. This dynamic programming approach calculates the minPD score for pendant subtrees of T, which are then combined to calculate the minPD score for T as a whole. Additionally, by tracking the indicator function values as we go, a corresponding size-k minPD set can be determined.

Proposition 8 *Let $T = (V, E)$ be a rooted binary phylogenetic X-tree, and let $0 \leq k \leq n$, where $n = |X|$. The algorithm* MinPDScore *applied to T and k calculates the minimum PD score for a size-k set of leaves of T in time $O(n^4)$.*

Proof Let $|X| = n$. The ordering of vertices on Line 5 can be completed in $O(|V(T) - X| + |E(T)|) = O(n)$ (Kahn, 1962). The "while" loop from Lines 8 to 13 has order $O(n) \cdot O(n) \cdot O(n^2) = O(n^4)$, since $|V(T) - X| = n - 1$, and $k \leq n$, and we are comparing $k + 1$ values in Eqn. (4). □

5.1 minPD scores for non-binary rooted phylogenetic trees

The algorithm MinPDScore can be adapted for a non-binary rooted phylogenetic tree with bounded out-degree. Specifically, Line 10 of the algorithm is adjusted, and an upper bound on the out-degree of every vertex is required to ensure that the modified algorithm runs in polynomial time.

Let $\{e_1, e_2, \ldots, e_t\}$ denote the set of edges incident with the root of T, and let T_i denote the subtree of T descending from e_i. Set $\ell_i = \ell(e_i)$ for $i \in \{1, 2, \ldots, t\}$, and let

$$K(k,t) = \left\{ \mathbf{k} = (k_1, ..., k_t) : k_i \geq 0 \text{ for all } i \text{ and } \sum_{i=1}^{t} k_i = k \right\}.$$

Then, in place of Eqn. (4), we use Eqn. (5) which applies the same notation as Proposition 7.

$$\varphi_T(k) = \min_{\mathbf{k} \in K(k,t)} \left\{ \sum_{i=1}^{t} (\varphi_i(k_i) + \ell_i \cdot \mathbb{I}_{k_i > 0}) \right\} \tag{5}$$

6 Concluding Remarks

Phylogenetic diversity provides a formal way to quantify recent (and possible future) biodiversity loss, resulting from the current high rate of species extinction. For example, PD has become an integral part of the Zoological Society of London's 'EDGE of Existence' programme for monitoring biodiversity risk (Isaac et al., 2007). PD is more nuanced than simply counting species extinctions, since PD explicitly incorporates the evolutionary relationships among species, and thus provides a proxy for measuring the richness of features that make species unique (Faith, 1992; Wicke et al., 2021).

In this paper, we have investigated new combinatorial questions concerning PD that arise in its application to large data-sets. In particular, we have described a precise way to count the number of maxPD sets of given size on a given tree (in the usual ultrametric setting) and derived some bounds on the growth rate for these numbers. We have also described further mathematical results that establish polynomial-time algorithms to (i) optimise a linear function (across the species at the tips of the tree) over all maxPD sets and (ii) determine the greatest possible loss of PD on a tree if k species were to become extinct (this last question amounts to determining minPD sets of given size).

Our results suggest a number of questions. In future work, we hope to characterise the tree shapes that have the largest number of maxPD sets (of

any given size). A further question is to count the number of minPD sets in the binary ultrametric setting. For caterpillars on n leaves (and ultrametric edge lengths), the number of size-k minPD sets is 1 unless $k = 1$, in which case there are n min PD sets. To see this, observe that a size-k minPD set in a caterpillar is the one that contains the k leaves with the shortest pendant edges (removing any of these leaves to replace it with one of the $n-k$ unchosen leaves would necessarily add more to the PD score than what was lost by not counting the removed pendant edge). A related question is to categorise the trees which have a *unique* minPD set.

7 Acknowledgements

The authors were supported by the New Zealand Marsden Fund (MFP-UOC2005). We thank the referees for helpful suggestions and for alerting us to some relevant previous papers. We also thank Arne Mooers for some helpful comments and suggestions.

References

Blum, A., Chalasani, P., Coppersmith, D., Pulleyblank, B., Raghavan, P., Sudan, M., 1994. The minimum latency problem, in: Proceedings of the twenty-sixth annual ACM symposium on Theory of computing, pp. 163–171.

Bordewich, M., Semple, C., Spillner, A., 2009. Optimizing phylogenetic diversity across two trees. Applied mathematics letters 22, 638–641.

Cadotte, M.W., Tucker, C.M., 2018. Difficult decisions: Strategies for conservation prioritization when taxonomic, phylogenetic and functional diversity are not spatially congruent. Biological Conservation 225, 128–133.

Cococcioni, M., Pappalardo, M., Sergeyev, Y.D., 2018. Lexicographic multi-objective linear programming using grossone methodology: Theory and algorithm. Applied Mathematics and Computation 318, 298–311.

Davis, M., Faurby, S., Svenning, J.C., 2018. Mammal diversity will take millions of years to recover from the current biodiversity crisis. Proceedings of the National Academy of Sciences USA 115, 11262–11267.

Faith, D.P., 1992. Conservation evaluation and phylogenetic diversity. Biological Conservation 61, 1–10.

Felsenstein, J., 2004. Inferring Phylogenies. Sinauer Associates, Sunderland, MA.

Isaac, N.J., Turvey, S.T., Collen, B., Waterman, C., Baillie, J.E., 2007. Mammals on the edge: conservation priorities based on threat and phylogeny. PloS one 2, e296.

Jetz, W., Thomas, G., Joy, J., Redding, D., Hartmann, K., Mooers, A., 2014. Global distribution and conservation of evolutionary distinctness in birds. Current Biology 24, 919–930.

Kahn, A.B., 1962. Topological sorting of large networks. Communications of the ACM 5, 558–562.

Kling, M.M., Mishler, B.D., Thornhill, A.H., Baldwin, B.G., Ackerly, D.D., 2019. Facets of phylodiversity: evolutionary diversification, divergence and survival as conservation targets. Philosophical Transactions of the Royal Society B 374, 20170397.

Lewis, L.A., Lewis, Paul, O., 2005. Unearthing the molecular phylodiversity of desert soil green algae (Chlorophyta). Systematic Biology 54, 936–947.

Manson, K.D., . Mathematical aspects of phylogenetic diversity measures. Ph.D. thesis. University of Canterbury. In preparation.

Mazel, F., Mooers, A.O., Riva, G.V.D., Pennell, M.W., 2017. Conserving phylogenetic diversity can be a poor strategy for conserving functional diversity. Systematic Biology 66, 1019–1027.

Mazel, F., Pennell, M.W., Cadotte, M.W., Diaz, S., Dalla Riva, G.V., Grenyer, R., Leprieur, F., Mooers, A.O., Mouillot, D., Tucker, C.M., et al., 2018. Prioritizing phylogenetic diversity captures functional diversity unreliably. Nature Communications 9, 1–9.

Molina-Venegas, R., Rodríguez, M.Á., Pardo-de Santayana, M., Ronquillo, C., Mabberley, D.J., 2021. Maximum levels of global phylogenetic diversity efficiently capture plant services for humankind. Nature Ecology & Evolution 5, 583–588.

Moulton, V., Semple, C., Steel, M., 2007. Optimizing phylogenetic diversity under constraints. Journal of Theoretical Biology 246, 186–194.

Pardi, F., Goldman, N., 2005. Species choice for comparative genomics: being greedy works. PLoS Genetics 1, e71.

Spillner, A., Nguyen, B.T., Moulton, V., 2008. Computing phylogenetic diversity for split systems. IEEE/ACM Transactions on Computational Biology and Bioinformatics 5, 235–244.

Steel, M., 2005. Phylogenetic diversity and the greedy algorithm. Systematic Biology 54, 527–529.

Steel, M., 2016. Phylogeny: Discrete and Random Processes in Evolution. SIAM, Philadelphia PA.

Tucker, C.M., Aze, T., Cadotte, M.W., Cantalapiedra, J.L., Chisholm, C., Díaz, S., Grenyer, R., Huang, D., Mazel, F., Pearse, W.D., et al., 2019. Assessing the utility of conserving evolutionary history. Biological Reviews 94, 1740–1760.

Upham, N., Esselstyn, J., Walter Jetz, W., 2019. Inferring the mammal tree: Species-level sets of phylogenies for questions in ecology, evolution, and conservation. PLoS Biology 17, e3000494.

Wicke, K., Mooers, A., Steel, M., 2021. Formal links between feature diversity and phylogenetic diversity. Systematic Biology 70, 480–490.