OPTIMIZING PHYLOGENETIC DIVERSITY ACROSS TWO TREES

MAGNUS BORDEWICH, CHARLES SEMPLE, AND ANDREAS SPILLNER

Abstract. We present a polynomial-time algorithm for finding an optimal set of taxa that maximizes the weighted-sum of the phylogenetic diversity across two phylogenetic trees. This resolves one of the challenges proposed as part of the Phylogenetics Programme held at the Isaac Newton Institute for Mathematical Sciences (Cambridge, 2007). It also completely closes the gap between optimizing phylogenetic diversity on one tree, which is known to be in P, and optimizing phylogenetic diversity across three or more trees, which is known to be NP-hard.

1. Introduction

A central task in conservation biology is measuring, predicting, and preserving biological diversity as species face extinction. Dating back to Faith (1992) [2], phylogenetic diversity (PD) is a prominent tool for measuring the biodiversity of a subset of species. This measure is based on the evolutionary distance amongst the species in the subset on an underlying phylogenetic (evolutionary) tree. For a fixed integer k, there are polynomial-time algorithms for finding an optimal k-element subset of species that maximizes the PD score across one tree. However, in practice, the underlying phylogenetic tree of the species under consideration is typically unknown, or there is no 'true tree' relating the species because of evolutionary events such as recombination. Thus one usually obtains two or more different trees for the same set of species, each arising from the analysis of a different gene or section of genome, or simply from analyses that use different models of evolution. Therefore, we would ideally like to optimize PD across a (weighted) set of phylogenetic trees. It has been previously stated that across three or more trees this problem is NP-hard. In this paper, we show that the problem of finding an optimal subset of species that maximizes the PD score across two trees can be solved in polynomial time.

A *phylogenetic X-tree* $T = (V, E)$ is an (unrooted) tree with no degree-2 vertices and whose leaf set X represents a set of species. Suppose the edges of T have non-negative real-valued lengths $\omega : E \to \mathbb{R}^{\geq 0}$. The *phylogenetic diversity* (PD

Date: 8 October 2007.

1991 *Mathematics Subject Classification.* 05C05; 92D15.

Key words and phrases. Phylogenetic diversity.

The first author and third authors were supported by the EPSRC, while the second author was supported by the New Zealand Marsden Fund. The work was carried out while the authors were visiting the Isaac Newton Institute for Mathematical Sciences.

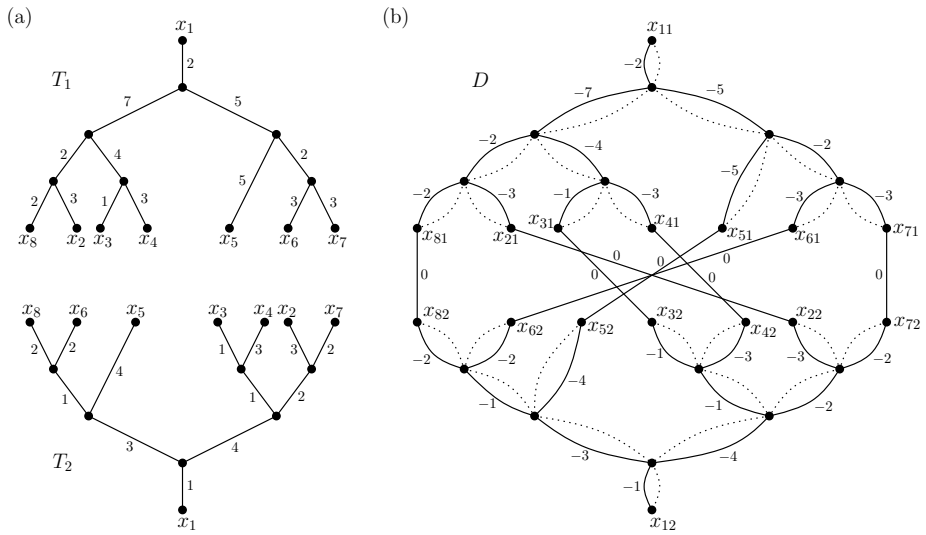

FIGURE 1. (a): Two phylogenetic trees T_1 and T_2 on the same set of species $\{x_1, \ldots, x_8\}$. (b): The network D constructed from T_1 and T_2. All the arcs in D are directed downwards. Each solid arc has the indicated cost and capacity 1. The dotted arcs have cost 0 and capacity $k-2$.

score) of a subset Y of X is the sum of the edge lengths of the minimal subtree of T connecting the elements in Y. Referring to Figure 1(a), if $Y = \{x_1, x_2, x_4\}$, then $PD_{T_1}(Y) = 21$ and $PD_{T_2}(Y) = 14$.

For one tree, the PD optimization problem is to find a subset of X of a given size k that maximizes the PD score amongst all subsets of X of size k. Extending this problem to an arbitrary number of trees, we have the following family of optimization problems:

Problem: Weighted Average PD on t trees (WAPD$_t$)
Instance: A collection $\mathcal{T} = \{T_1, \ldots, T_t\}$ of phylogenetic X-trees whose edges have non-negative real-valued lengths, a collection $\{\lambda_1, \ldots, \lambda_t\}$ of non-negative real-valued weights, and an integer k.
Question: Find a subset Y of X of size k that maximizes

$$PD_{\mathcal{T}}(Y) = \lambda_1 PD_{T_1}(Y) + \cdots + \lambda_t PD_{T_t}(Y)$$

amongst all k-element subsets of X.

The value $PD_{\mathcal{T}}(Y)$ is the *phylogenetic diversity* (PD score) of Y across the trees in $\mathcal{T}$. To allow a weighting scheme on the individual trees, we have additionally included the weights $\lambda_1, \ldots, \lambda_t$. Of course, by multiplying all edge lengths of T_i by λ_i to obtain T_i' for all i, the PD score of Y across $\mathcal{T}$ is $PD_{T_1'}(Y) + \cdots + PD_{T_t'}(Y)$. Thus for computational purposes, no generality is lost by assuming that $\lambda_i = 1$ for all i. We make this assumption in the rest of the paper.

The problem WAPD_1 can be solved by a greedy approach [8, 5] in polynomial time. An implementation of this approach with run time $O(n \log k)$ is available [3], where $n = |X|$. Furthermore, it is even possible to solve WAPD_1 in $O(n)$ time [7].

The problem WAPD_t for $t \geq 2$ appears to have first been raised in [3]. For $t = 3$, Spillner *et al.* [7] noted without proof that WAPD_3, and therefore WAPD_t for all $t \geq 3$, is NP-hard using a reduction from 3-dimensional matching. This left open the problem of determining the computational complexity of WAPD_2, explicitly stated in [7] and subsequently asked again in [9] where a prize was offered for resolving the problem. In this paper, we show that WAPD_2 can be solved by a polynomial-time algorithm by reformulating the problem as a set of minimum-cost flow problems. Furthermore, for completeness, we explicitly show that WAPD_3 is NP-hard using a reduction from vertex cover on cubic graphs.

2. A Polynomial-Time Algorithm for WAPD_2

In this section, we show that WAPD_2 is solvable in polynomial time. To do this, we initially show that a restricted version of WAPD_2 is solvable in polynomial time. In this restriction, we are additionally given a distinguished element, x say, of X in the instance and are asked to find a subset of Y of X of size k that contains x and maximizes

$$PD_{\mathcal{T}}(Y) = PD_{T_1}(Y) + \cdots + PD_{T_t}(Y).$$

It will then immediately follow that WAPD_2 is solvable in polynomial time.

To show that this restricted version of WAPD_2 is solvable in polynomial time, we reformulate it into a network flow problem. Let $X = \{x_1, x_2, \ldots, x_n\}$, and suppose that the edges of T_1 and T_2 are assigned non-negative real-valued lengths $\omega : E(T_1) \cup E(T_2) \to \mathbb{R}^{\geq 0}$. Without loss of generality, choose x_1 to be the distinguished element of X. For the purposes of the reformulation, we distinguish between the vertices of T_1 and T_2 that share a common label in X by relabelling x_i with x_{i1} in T_1 and relabelling x_i with x_{i2} in T_2 for all $i \in \{1, 2, \ldots, n\}$. Furthermore, we view the edges of T_1 (resp. T_2) as arcs directed away from x_{11} (resp. x_{12}).

Now we construct a network D from T_1 and T_2, where the source and sink of D will be x_{11} and x_{12}, respectively. The vertex set V of D is $V(T_1) \cup V(T_2)$ and the arc set A of D is constructed in the following way:

(i) For each arc (u, v) in T_1, add two arcs $(u, v)_1$ and $(u, v)_2$ in parallel directed from u to v with $(u, v)_1$ having capacity 1 and cost $-\omega(u, v)$, and $(u, v)_2$ having capacity $k - 2$ and cost 0.
(ii) For each arc $(u, v) \in T_2$, add two arcs $(v, u)_1$ and $(v, u)_2$ in parallel directed from v to u with $(v, u)_1$ having capacity 1 and cost $-\omega(u, v)$, and $(v, u)_2$ having capacity $k - 2$ and cost 0.
(iii) For each $i \in \{2, 3, \ldots, n\}$, add the arc (x_{i1}, x_{i2}) with capacity 1 and cost 0.

To illustrate this construction, consider the two phylogenetic trees T_1 and T_2 in Fig. 1(a). The flow network for these two trees with x_{11} and x_{12} as the source and sink, respectively, is shown in Fig. 1(b).

Noting that the network resulting from the above construction D has a feasible flow of $k-1$ units, the following lemma states the key property of D.

Lemma 2.1. *Let f be an integer-valued minimum-cost flow of $k-1$ units from x_{11} to x_{12} in D. Then*

$$Y_f = \{x_1\} \cup \{x_i \in X - x_1 : f(x_{i1}, x_{i2}) > 0\}$$

is an optimal solution to the restricted version of WAPD_2 *in which x_1 is the distinguished element of X.*

Proof. Let Y_{opt} be an optimal solution for the restricted version of WAPD_2 in which x_1 is the distinguished element of X. The goal is to show that

$$PD_{T_1}(Y_f) + PD_{T_2}(Y_f) = PD_{T_1}(Y_{\mathrm{opt}}) + PD_{T_2}(Y_{\mathrm{opt}}).$$

For each arc $a = (u, v)$ of T_1 (resp. T_2), let l_a denote the number of leaves in Y_{opt} that can be reached by a directed path in T_1 (resp. T_2) starting at v. Let $f_{Y_{\mathrm{opt}}}$ be the flow of $k-1$ units on D that is defined as follows:

(i) For each arc $a = (u, v) \in T_1$, set

$$f_{Y_{\mathrm{opt}}}((u, v)_1) = \min\{1, l_a\} \text{ and } f_{Y_{\mathrm{opt}}}((u, v)_2) = \max\{0, l_a - 1\}.$$

(ii) For each arc $a = (u, v) \in T_2$, set

$$f_{Y_{\mathrm{opt}}}((v, u)_1) = \min\{1, l_a\} \text{ and } f_{Y_{\mathrm{opt}}}((v, u)_2) = \max\{0, l_a - 1\}.$$

(iii) For each $x_i \in X - x_1$, set $f_{\mathrm{opt}}(x_{i1}, x_{i2}) = 1$ if $x_i \in Y_{\mathrm{opt}}$; otherwise set $f_{Y_{\mathrm{opt}}}(x_{i1}, x_{i2}) = 0$.

It is easily checked that $f_{Y_{\mathrm{opt}}}$ is indeed a flow of $k-1$ units on D and, moreover, the cost of this flow is $cost(f_{Y_{\mathrm{opt}}}) = -(PD_{T_1}(Y_{\mathrm{opt}}) + PD_{T_2}(Y_{\mathrm{opt}}))$.

We next show that the cost of f is

$$cost(f) = -(PD_{T_1}(Y_f) + PD_{T_2}(Y_f)). \tag{1}$$

To establish (1), it suffices to show that the set of arcs used by f of the form $(u, v)_1$ (derived from T_1) or the form $(v, u)_1$ (derived from T_2) are precisely the union of the arcs in the minimal subtree of T_1 connecting the elements in Y_f and the minimal subtree of T_2 connecting the elements in Y_f. Note that if f uses an arc from u to v and both arcs from u to v have cost 0 (*i.e.* a tree edge has length 0), we may assume that f uses the arc of the form $(u, v)_1$. If an arc of D of the form $(u, v)_1$ (derived from T_1) is used by f, then it is clear that there exists an element in $Y_f - x_1$ in the subtree of T_1 below (u, v). Thus, as $x_1 \in Y_f$, the arc (u, v) is in the minimal subtree of T_1 connecting the elements in Y_f. Similarly, if an arc of D of the form $(v, u)_1$ (derived from T_2) is used by f, then (u, v) is in the minimal subtree of T_2 connecting the elements in Y_f. Now assume that (u, v) is in the minimal subtree of T_1 connecting the elements in Y_f. Then (u, v) is on the directed path from x_1 to an element, x_j say, in Y_f. Since x_j is an element of Y_f and since, up to parallel edges, there is a unique directed path from x_{11} to x_{1j} in D, the minimum-cost flow f must use $(u, v)_1$. An analogous argument also holds in the case that (u, v) is in

the minimal subtree of T_2 connecting the elements in Y_f. This establishes (1). It now follows that

$$\begin{aligned} cost(f_{Y_{\text{opt}}}) &= -(PD_{T_1}(Y_{\text{opt}}) + PD_{T_2}(Y_{\text{opt}})) \\ &\leq -(PD_{T_1}(Y_f) + PD_{T_2}(Y_f)) = cost(f). \end{aligned}$$

Hence, as f and $f_{Y_{\text{opt}}}$ are both flows of $k-1$ units on D, but f has minimum cost, we have

$$PD_{T_1}(Y_{\text{opt}}) + PD_{T_2}(Y_{\text{opt}}) = PD_{T_1}(Y_f) + PD_{T_2}(Y_f).$$

This completes the proof of the lemma. □

Lemma 2.2. *The restricted version of WAPD$_2$ can be solved in $O(n^2 \log^2 n)$ time, where $n = |X|$.*

Proof. The construction above can certainly be done within time $O(n^2 \log^2 n)$. Finding a minimum-cost flow in such a network is an old and well-studied problem, and can be solved using an algorithm that runs in time $O((|A| \log |V|)(|A| + |V| \log |V|))$ (see, for example, [1]). Since D has $O(n)$ vertices and arcs this yields a run time of $O(n^2 \log^2 n)$. Furthermore, as all the capacities in D and the target flow of $k-1$ units are integral, there is an integral minimum-cost flow, and this is found by the above algorithm [1]. □

Theorem 2.3. *The problem* WAPD$_2$ *(without restriction) can be solved in $O(n^3 \log^2 n)$ time, where $n = |X|$.*

Proof. Suppose Y_{opt} is an optimal solution to WAPD$_2$ and let x be an element of Y_{opt}. Then Y_{opt} is an optimal solution to the restricted version of WAPD$_2$ in which x is the distinguished element of X. Consequently, by solving the restricted version of WAPD$_2$ for each element of X—thus running the method described above n times—and choosing the solution that maximizes $PD_{T_1}(Y) + PD_{T_2}(Y)$, we obtain an optimal solution to WAPD$_2$. The theorem now follows from Lemma 2.2 □

3. NP-Hardness of WAPD$_3$

In this section, we explicitly show that WAPD$_3$ is NP-hard. The reduction is from a restricted version of the following classic NP-complete problem:

Problem: VertexCover
Instance: A graph $G = (V, E)$ and an integer k.
Question: Does there exist a subset $C \subseteq V$ such that $|C| = k$ and, for every edge $\{u, v\} \in E$, the intersection $\{u, v\} \cap C$ is non-empty.

VertexCover remains NP-complete even if the input graph is restricted to be a 3-connected cubic planar graph [11]. The reduction proceeds as follows. Take a 3-connected cubic planar instance G. Colour the edges of G with three colours $\{1, 2, 3\}$ such that no two adjacent edges receive the same colour. Due to a classic construction of Tait [10], this is equivalent to four-colouring the faces of a planar drawing of G which can be done in quadratic time [6]. For each colour $c \in \{1, 2, 3\}$, let T_c be the phylogenetic V-tree that consists of a (central) vertex z_c of degree

$|V|/2$, where the $|V|/2$ neighbours of z_c each have degree 3 and the $|V|$ leaves are arranged so that, for each edge $\{u, v\}$ of G coloured c, the vertices u and v are adjacent to the same degree-3 vertex. As G is a cubic graph, T_c is well-defined for all c.

For each of T_1, T_2, and T_3, assign length 1 to all edges. It now follows that the PD score of an optimal solution to WAPD_3 across T_1, T_2, and T_3 is equal to $|E|+3k$ if and only if G has a vertex cover of size k, where $k \geq 3$. To see this, consider a vertex cover C of size k. Each element in C appears as a leaf in each tree, so the edges incident with these elements contribute $3k$ to the overall PD score. Since C covers every edge of G, each edge of G corresponds to a unique edge incident with one of z_1, z_2, z_3, and $k \geq 3$, these latter edges contribute exactly $|E|$ to the overall PD score. Thus the PD score of C across T_1, T_2, and T_3 is $|E| + 3k$. The converse is similar.

We remark here that, in the above reduction, T_1, T_2, and T_3 could have been made binary (that is, each internal node has degree 3) if desired by refining z_1, z_2, and z_3 and assigning length ϵ to each new edge such that ϵ is smaller than $3/(|V|/2)$.

4. Concluding Remarks

We end the paper with two remarks. First, the problem WAPD_1 can be considered as a special case of WAPD_2 when one of the initial trees is degenerated to a tree with a single internal vertex. Hence, our algorithm for solving WAPD_2 also applies to WAPD_1. However, the greedy approach for the latter problem mentioned in the introduction, yields a much better asymptotic run time. Note that the greedy approach used to solve WAPD_1 can fail to produce optimal solutions for WAPD_2—this follows from the discussion in [4, Section 7].

Second, a *rooted phylogenetic X-tree* T is a rooted tree with leaf set X and whose root has degree at least 2 and all other internal vertices have degree at least 3. Assuming that the edges of T are assigned non-negative real-valued lengths, the PD score of a subset Y of X on T is the sum of the lengths of the edges in the minimal subtree of T connecting the elements in $Y \cup \rho$, where ρ is the root of T. The family of problems WAPD_t can be interpreted for rooted phylogenetic trees in the obvious way. Indeed, for $t = 1$, $t = 2$, and for all $t \geq 3$, the analogous unrooted computational results hold. In particular, WAPD_2 for rooted phylogenetic trees can be solved in polynomial time. It can be directly interpreted as the restricted version of WAPD_2 described in Section 2, where the distinguished element is ρ, the common root label of the two rooted trees.

References

[1] R. Ahuja, T. Magnanti, and J. Orlin. *Network Flows: Theory, Algorithms, and Applications.* Prentice Hall, London, 1993.

[2] D. P. Faith. Conservation evaluation and phylogenetic diversity. *Biological Conservation*, 61:1–10, 1992.

[3] B. Q. Minh, S. Klaere, and A. von Haeseler. Phylogenetic diversity within seconds. *Systematic Biology*, 55:769–773, 2006.
[4] V. Moulton, C. Semple, and M. Steel. Optimizing phylogenetic diversity under constraints. *Journal of Theoretical Biology*, 246:186–194, 2007.
[5] F. Pardi and N. Goldman. Species choice for comparative genomics: being greedy works. *PLoS Genetics*, 1(6), 2005.
[6] N. Robertson, D. Sanders, P. Seymour, and R. Thomas. A new proof of the four-colour theorem. *Electron. Res. Announc. Amer. Math. Soc.*, 2:17–25, 1996.
[7] A. Spillner, B. T. Nguyen, and V. Moulton. Computing phylogenetic diversity for split systems, 2007. submitted.
[8] M. Steel. Phylogenetic diversity and the greedy algorithm. *Systematic Biology*, 54:527–529, 2005.
[9] M. Steel. Phylognetics: Challenges and conjectures. In *Newton Institute Programme on Phylogenetics*, 2007.
[10] P.G. Tait. On the colouring of maps. In *Proceedings of the Royal Society of Edinburgh, Section A*, volume 10, pages 501–503, 1878-1880.
[11] R. Uehara. NP-complete problems on a 3-connected cubic planar graph and their applications. Technical Report TWCU-M-0004, Tokyo Woman's Christian University, 1996.

Department of Computer Science, Durham University, Durham DH1 3LE, UK

E-mail address: m.j.r.bordewich@durham.ac.uk

Biomathematics Research Centre, Department of Mathematics and Statistics, University of Canterbury, Christchurch, New Zealand

E-mail address: c.semple@math.canterbury.ac.nz

School of Computing Sciences, University of East Anglia, Norwich NR4 7TJ, UK

E-mail address: aspillner@cmp.uea.ac.uk