BUDGETED NATURE RESERVE SELECTION WITH DIVERSITY FEATURE LOSS AND ARBITRARY SPLIT SYSTEMS

MAGNUS BORDEWICH[1] AND CHARLES SEMPLE[2]

Abstract. Arising in the context of biodiversity conservation, the Budgeted Nature Reserve Selection (BNRS) problem is to select, subject to budgetary constraints, a set of regions to conserve so that the phylogenetic diversity (PD) of the set of species contained within those regions is maximized. Here PD is measured across either a single rooted tree or a single unrooted tree. Nevertheless, in both settings, this problem is NP-hard. However, it was recently shown that, for each setting, there is a polynomial-time $\left(1-\frac{1}{e}\right)$-approximation algorithm for it and that this algorithm is tight. In the first part of the paper, we consider two extensions of BNRS. In the rooted setting we additionally allow for the disappearance of features, for varying survival probabilities across species, and for PD to be measured across multiple trees. In the unrooted setting, we extend to arbitrary split systems. We show that, despite these additional allowances, there remains a polynomial-time $\left(1-\frac{1}{e}\right)$-approximation algorithm for each extension. In the second part of the paper, we resolve a complexity problem on computing PD across an arbitrary split system left open by Spillner et al.

1. Introduction

In conservation biology, measures such as phylogenetic diversity are used to quantify the biological diversity of a collection of species. These measures are used to select which species should be conserved and, in this regard, individual species are often the focus of attention. However, as pointed out by Rodrigues et al. (2005), this is not necessarily the best way to conserve diversity:

> Although conservation action is frequently targeted toward single species, the most effective way of preserving overall species diversity is by conserving viable populations in their natural habitats, often by designating networks of protected areas.

Date: 29 December 2010.

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

Key words and phrases. Combinatorial algorithms, phylogenetic diversity, biodiversity conservation, split systems, submodular functions.

The first author was supported by the EPSRC, while the second author was supported by the New Zealand Marsden Fund.

Motivated by this quote and applications of using phylogenetic diversity across areas to make assessments in conservation planning (Moritz and Faith 1998; Rodrigues and Gaston 2002; Smith et al. 2000), Bordewich and Semple (2008) considered a natural computational problem in the context of conserving whole habitats instead of individual species. In this paper, we consider two extensions of this problem.

Dating back to Faith (1992), phylogenetic diversity (PD) has emerged as a leading measure in quantifying the biodiversity of a collection of species. This measure is based on the evolutionary distance among the species in the collection. A formal definition of PD is given in the next section but, for the purposes of the introduction, let $\mathcal{T}$ be either a rooted or unrooted phylogenetic tree whose leaf set X represents a set of species and whose edges have real-valued lengths (weights). The PD score of a subset Y of X is the sum of the weights of the edges of the minimal subtree of $\mathcal{T}$ connecting the species in Y. If $\mathcal{T}$ is rooted, the minimal subtree additionally includes the root. In its most straightforward application to conservation, the task is to find a subset of X of a given size k which maximizes the PD score among all subsets of X of size k. It is now well-known that a greedy algorithm solves this task exactly (Faith 1992; Pardi and Goldman 2005; Steel 2005).

The problem considered in Bordewich and Semple (2008) is the following: In an addition to $\mathcal{T}$, we have a collection $\mathcal{R}$ of regions or areas containing species in X. Each region in $\mathcal{R}$ has an associated cost of preservation. Given a fixed budget B, the task is to find a subset of regions in $\mathcal{R}$ to be preserved which maximizes the PD score of the species contained within at least one preserved region while keeping within budget. This problem is called the Budgeted Nature Reserve Selection (BNRS) and generalizes the analogous unit cost problems described in Moulton et al. (2007), Pardi and Goldman (2007), Rodrigues and Gaston (2002), and Rodrigues et al. (2005). The applications to conservation planning mentioned above are BNRS with unit costs.

Regardless of the setting (whether $\mathcal{T}$ is rooted or unrooted), it follows from a result in Moulton et al. (2007) that BNRS is NP-hard. Nevertheless, it is shown in Bordewich and Semple (2008) that, for each setting, there is a polynomial-time $\left(1 - \frac{1}{e}\right)$-approximation algorithm for it and that this algorithm is tight. In this paper, we consider, for each setting, an extension of BNRS. Formal details are given in the next section, but the extensions include the following:

(i) It is unrealistic to expect that because a species is not contained in at least one of the selected regions for preservation, its probability of survival is zero, or that its probability of survival is one if it is contained in one of the selected regions. The extension in the rooted setting additionally allows for arbitrary survival probabilities with the probability of survival of a species increasing if it is contained in a region selected for preservation. This particular extension of BNRS has been previously described as the Nature Reserve Problem by Pardi and Goldman (2007).

(ii) In many instances, evolutionary relationships cannot be accurately represented by a single tree. In the rooted setting, the relationships may be better represented by a collection of gene-trees (each representing the tree-like evolution of a gene or group of genes) rather than a single species tree. In the unrooted setting, relationships may be better represented by an arbitrary network rather than a tree. We extend BNRS by replacing $\mathcal{T}$ with a collection of weighted trees in the rooted setting and with a so-called split network in the unrooted setting.

(iii) The standard usage of PD assumes that elements of biodiversity, 'features', arise uniformly across a phylogeny and persist to be present in all descendant species. A recent extension (Bordewich et al. 2008) proposes a model in which PD may be measured which includes the gradual disappearance of features over time, so that the features of an ancestral species may not all survive to be present in all descendants of that species. This model only makes sense in the rooted setting, and we extend BNRS to cover this model in this setting.

Despite the additional freedom which comes with such inclusions, there remains (for each extension) a polynomial-time $\left(1 - \frac{1}{e}\right)$-approximation algorithm for solving it. That is, a polynomial-time algorithm that returns a feasible solution whose associated score is at least $\left(1 - \frac{1}{e}\right)$ (approx. 0.63) times the optimal score. The next section formally describes the two extensions and the main results of the paper—including the solution of a related problem left open by Spillner et al. (2008).

2. Main Results

Throughout the paper, X denotes a finite set and represents, for example, a collection of species. A *phylogenetic X-tree $\mathcal{T}$* is an unrooted tree with no degree-two vertices and whose leaf set is X. A *rooted phylogenetic X-tree* is a rooted tree with no degree-two vertices except the root that may have degree two and whose leaf set is X. For the purposes of this paper, we will assume that all the edges of a rooted and unrooted phylogenetic tree are assigned non-negative real-valued lengths. To illustrate, Figure 1 shows an (unrooted) phylogenetic X-tree, where $X = \{a, b, c, d, e, f, g\}$.

Let Y be a subset of X. If $\mathcal{T}$ is an (unrooted) phylogenetic X-tree, then the *phylogenetic diversity of Y on $\mathcal{T}$* is the sum of the edge lengths of the minimal subtree of $\mathcal{T}$ that connects the elements in Y. If $\mathcal{T}$ is a rooted phylogenetic X-tree, then the *phylogenetic diversity of Y on $\mathcal{T}$* is the sum of the edge lengths of the minimal subtree of $\mathcal{T}$ that connects the elements in Y and the root of $\mathcal{T}$. For example, referring to Figure 1, if $Y = \{a, b, f\}$, then $PD(Y)$ is equal to the sum of the weights of the minimal subtree (dashed edges) that connects a, b, and f; in particular, $PD(Y) = 12$.

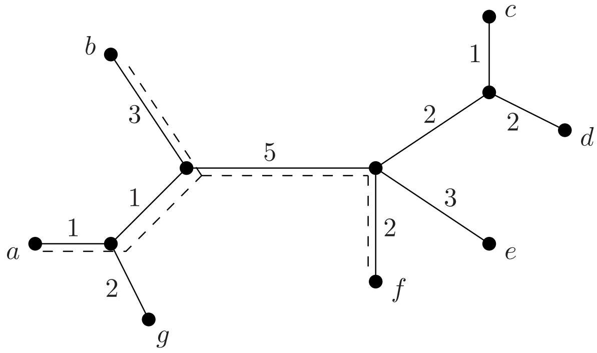

Figure 1. A phylogenetic X-tree with edge lengths, where $X = \{a, b, c, d, e, f, g\}$.

Now let $\mathcal{T}$ be a rooted or unrooted phylogenetic X-tree and let $\mathcal{R}$ be a collection of regions or areas containing species in X. Each $R \in \mathcal{R}$ is a subset of X and has an associated cost $c(R)$ of preservation. Overriding these costs is a fixed budget B, where we may assume, without loss of generality, that $c(R) \leq B$ for all $R \in \mathcal{R}$. The Budgeted Nature Reserve Selection (BNRS) problem is to find a subset $\mathcal{R}'$ of the regions in $\mathcal{R}$ which maximizes the PD score of $\cup_{R \in \mathcal{R}'} R$ on $\mathcal{T}$ such that $\sum_{R \in \mathcal{R}'} c(R) \leq B$. To illustrate, take $\mathcal{T}$ to be the phylogenetic X-tree shown in Figure 1 and $\mathcal{R}$ to be

$$\big\{\{b\}, \{c, f\}, \{c, d\}, \{a, b\}, \{a, g\}, \{e\}, \{e, g\}\big\}.$$

Set c as the cost function on $\mathcal{R}$ defined by $c(\{b\}) = 4$, $c(\{c, f\}) = 8$, $c(\{c, d\}) = 6$, $c(\{a, b\}) = 10$, $c(\{a, g\}) = 4$, $c(\{e\}) = 4$, and $c(\{e, g\}) = 5$, and set $B = 24$. A feasible solution of this instance is $\big\{\{c, d\}, \{a, b\}\big\}$ as $c(\{c, d\}) + c(\{a, b\}) = 6 + 10 = 16$. The PD score of $\big\{\{c, d\}, \{a, b\}\big\}$ is $PD\,(\{c, d\} \cup \{a, b\}) = 15$. However, an optimal solution is

$$\big\{\{b\}, \{c, f\}, \{c, d\}, \{e, g\}\big\},$$

where

$$c(\{b\}) + c(\{c, f\}) + c(\{c, d\}) + c(\{e, g\}) = 4 + 8 + 6 + 5 = 23$$

and

$$PD\,(\{b\} \cup \{c, f\} \cup \{c, d\} \cup \{e, g\}) = 21.$$

For both the rooted and unrooted settings, it is established in Bordewich and Semple (2008) that there is a polynomial-time $\big(1 - \frac{1}{e}\big)$-approximation algorithm for BNRS but, for any $\delta > 0$, BNRS cannot be approximated with an approximation ratio of $\big(1 - \frac{1}{e} + \delta\big)$ unless P=NP.

We next describe the two extensions of BNRS and the associated results.

2.1. **Extension of BNRS in the rooted setting.** In the rooted setting we incorporate all three extensions described in Section 1. The first is to allow varying probabilities of survival. Each taxa $x \in X$ has some probability $a(x, R)$ of surviving in reserve R without conservation efforts. This probability is boosted to $b(x, R) \geq a(x, R)$ if R is selected for conservation.

If x is not present in R, then $a(x,R) = b(x,R) = 0$. For a set of selected reserves $\mathcal{R}' \subseteq \mathcal{R}$, we denote by $p_{\mathcal{R}'}(x)$ the probability that x survives in at least one reserve, where survival in each reserve R is independent and has probability $a(x,R)$ if $R \notin \mathcal{R}'$ and $b(x,R)$ if $R \in \mathcal{R}'$.

The second extension is measure PD in relation not to a single tree, but to a set of weighted trees for the same set of species, each arising, for example, from the analysis of a different gene or section of genome. Thus we extend $\mathcal{T}$ to a collection $\mathcal{P} = \{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_k\}$ of rooted phylogenetic X-trees, where each $\mathcal{T}_j \in \mathcal{P}$ is assigned a non-negative real-valued weight $w(\mathcal{T}_j)$. Then the *phylogenetic diversity of X on $\mathcal{P}$* is the weighted sum of the PD measured against each tree.

The third extension is to use a model of biodiversity which allows disappearing features when calculating PD. Let $\mathcal{T}$ be a rooted phylogenetic X-tree. Under PD, one assumes that features arise during evolution at a constant rate—for two points u and v on $\mathcal{T}$ with u an ancestor of v, the distance from u to v is proportional to the number of new features that arose along the evolutionary path from u to v. Rescaling we assume that for every unit of distance a new feature arises. Furthermore, any feature arising at a point u on $\mathcal{T}$ is present at all points descendant from u.

We extend this model so that, in addition to features arising in this way, features have a constant probability of disappearing on every evolutionary path in $\mathcal{T}$ on which they are present. Mathematically, once a feature is present, it has a constant and memoryless probability $e^{-\lambda}$ of surviving in each time step. The disappearance of features in the context of phylogenetic diversity is considered in Bordewich et al. (2008) and Faith (1994).

For each $x \in X$, let the probability of survival be denoted by $p(x)$. Under this extended model, the *phylogenetic diversity of X on $\mathcal{T}$*, denoted $PD_{(\lambda,\mathcal{T})}(X,p)$, is the expected number of features present amongst the surviving taxa. That is,

$$PD_{(\lambda,\mathcal{T})}(X,p) = \int_{t\in\mathcal{T}} \mathbb{P}(t \to X)\mathrm{d}t,$$

where $(t \to X)$ denotes the event that a feature arising at point t on $\mathcal{T}$ survives to be present in a taxa in X which itself survives. For a collection $\mathcal{P} = \{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_k\}$ of rooted phylogenetic X-trees, where each $\mathcal{T}_j \in \mathcal{P}$ is assigned a non-negative real-valued weight $w(\mathcal{T}_j)$, the *phylogenetic diversity of X on $\mathcal{P}$*, denoted $PD_{(\lambda,\mathcal{P})}(X,p)$, is

$$PD_{(\lambda,\mathcal{P})}(X,p) = \sum_{j=1}^{k} w(\mathcal{T}_j) \int_{t\in\mathcal{T}_j} \mathbb{P}(t \to X)\mathrm{d}t.$$

Thus the full extension of BNRS in the rooted setting, called $\mathrm{BNRS}_{(\lambda,\mathcal{P})}$, is the following:

Problem: Budgeted Nature Reserve Selection ($\mathrm{BNRS}_{(\lambda,\mathcal{P})}$)
Instance: A collection $\mathcal{P}$ of weighted rooted phylogenetic X-trees, a collection $\mathcal{R}$ of subsets of X, a cost function c on the sets in $\mathcal{R}$, a budget B and, for all $(x,R) \in X \times \mathcal{R}$, probabilities $a(x,R)$ and $b(x,R)$, where $b(x,R) \geq a(x,R)$.
Question: Find a subset $\mathcal{R}'$ of $\mathcal{R}$ which maximizes $PD_{(\lambda,\mathcal{P})}(X, p_{\mathcal{R}'})$ such that $\sum_{R\in\mathcal{R}'} c(R) \leq B$.

The problem $\mathrm{BNRS}_{(\lambda,\mathcal{P})}$ extends the rooted setting of BNRS. In particular, by setting $\lambda = 0$, and $a(x,R) = 0$ or $b(x,R) = 1$ for each reserve R in $\mathcal{R}$, and considering a single rooted phylogenetic tree whose weight is 1. Thus, it follows by Bordewich and Semple (2008) that there is no $\delta > 0$ such that $\mathrm{BNRS}_{(\lambda,\mathcal{P})}$ can be approximated with an approximation ratio of $\left(1 - \frac{1}{e} + \delta\right)$ unless P=NP. However, we show in Section 4 that there is a polynomial-time $\left(1 - \frac{1}{e}\right)$-approximation algorithm for $\mathrm{BNRS}_{(\lambda,\mathcal{P})}$.

2.2. **Extension of BNRS in the unrooted setting.** We begin with some preliminary definitions. A bipartition $\{A, B\}$ of X, where $|A|, |B| \geq 1$, is a *split* of X. For simplicity, we write such a bipartition $\{A, B\}$ as $A|B$. A *split system* Σ of X is a collection of splits of X. In addition, Σ is *weighted* if there is a map $w : \Sigma \to \mathbb{R}^{\geq 0}$.

Let Σ be a weighted split system of X, and let Z be a subset of X. The *phylogenetic diversity of* Z *on* Σ, denoted $PD_\Sigma(Z)$, is

$$PD_\Sigma(Z) = \sum_{A|B\in\Sigma;\, A\cap Z, B\cap Z\neq\emptyset} w(A|B).$$

This definition of PD on a split system generalizes the definition of PD on an (unrooted) phylogenetic tree as follows. Let $\mathcal{T}$ be a phylogenetic X-tree. Each edge e of $\mathcal{T}$ induces a unique split $A|B$ of X, where A consists precisely of the subset of X in which, for all $a, a' \in A$, the unique path in $\mathcal{T}$ from a to a' avoids traversing e. For example, in Figure 1, $\{a, b, g\}|\{c, d, e, f\}$ is the split induced by the edge whose length is 5. An arbitrary collection Σ of X-splits is *compatible* if there exists a phylogenetic X-tree whose collection of X-splits arising in this way equates to Σ. Let Y be a subset of X. Assigning, for each edge e of $\mathcal{T}$, the weight of e with the X-split induced by e, it is easily checked that the PD of Y on the resulting collection of weighted X-splits induced by the edges of $\mathcal{T}$ is equivalent to the PD of Y on $\mathcal{T}$. Furthermore, there is a canonical one-to-one correspondence between weighted split systems and split networks analogous to the one-to-one correspondence between weighted compatible split systems and phylogenetic trees. Under this correspondence, computing PD on a split network equates to computing PD on the corresponding weighted split system. For details of split networks and this correspondence, see Bryant and Huson (2006) and Spillner et al. (2008), respectively.

The extension of BNRS in the unrooted setting, called BNRS_Σ, is the following:

Problem: Budgeted Nature Reserve Selection (BNRS$_\Sigma$)
Instance: A weighted split system Σ of X, a collection $\mathcal{R}$ of subsets of X, a cost function c on the sets in $\mathcal{R}$, and a budget B.
Question: Find a subset $\mathcal{R}'$ of $\mathcal{R}$, which maximizes the PD score of $\bigcup_{R\in\mathcal{R}'} R$ on Σ such that $\sum_{R\in\mathcal{R}'} c(R) \leq B$.

Clearly, BNRS$_\Sigma$ extends the unrooted setting of BNRS and so, for any $\delta > 0$, BNRS$_\Sigma$ cannot be approximated with an approximation ratio of $\big(1 - \frac{1}{e} + \delta\big)$ unless P=NP. However, we show in Section 5 that there is a polynomial-time $\big(1-\frac{1}{e}\big)$-approximation algorithm for BNRS$_\Sigma$. We note here that Minh et al. (2010) have recently presented an alternative approach to BNRS$_\Sigma$. They show how to transform the problem into a binary linear programming (BLP) problem so that the standard techniques, and software packages, for solving such problems can be used. This enables them to obtain reasonable running times for exact solutions in some cases, although the worst case running times for BLP algorithms are not polynomially bounded (and cannot be unless P=NP).

It may have been observed by the reader that of the three possible extensions described in Section 1, we have only made the extension to multiple trees or networks in the unrooted setting. As noted earlier, the extended model of biodiversity in which features both appear and disappear during evolution inherently requires a direction to time, and thus a rooted setting. However we could consider extending the unrooted setting to include varying probabilities of survival. Since this would generalise the existing problem, we could not hope to find a better approximation than $\big(1 - \frac{1}{e}\big)$ in this case. It remains an open problem to determine if such an approximation is possible. The approach we have taken for the other problems discussed in this paper, *i.e.* demonstrating submodularity of the core function, does not go through.

2.3. **Maximising PD on a split system.** In the second part of the paper, we resolve a problem left open by Spillner et al. (2008). In particular, consider the following computational problem:

Problem: Maximum PD on Σ (SplitsPD)
Instance: A weighted split system Σ of X, and a positive integer k.
Question: Find a subset Z of X of size k that maximizes $PD(Z)$.

If Σ is compatible, that is, can be realized by a phylogenetic tree, then the (polynomial-time) greedy algorithms in Pardi and Goldman (2005) and Steel (2005) solve SplitsPD. Indeed, there are polynomial-time algorithms for SplitsPD if Σ is a so-called circular split system or, more generally, an affine split system (Minh et al. 2009; Spillner et al. 2008). However, in general SplitsPD is NP-hard (Spillner et al. 2008). Nevertheless, Spillner et al. (2008) observed that a greedy algorithm provides a polynomial-time $\big(1 - \frac{1}{e}\big)$-approximation algorithm for SplitsPD, and that there is some constant $\alpha > 0$ such that, in general, SplitsPD cannot be approximated with an approximation ratio of $(1-\alpha)$ unless P=NP. In the last section of the paper, we show that in fact $\left(1-\frac{1}{e}\right)$ is the best possible.

A brief outline of the paper is as follows. The approximation results for the two extensions of BNRS in Sections 4 and 5 rely on establishing that the function being optimized (or one closely-related) is a submodular function. The next section describes submodular functions and a particular approximation result for such functions. The hardness result for SplitsPD is given in Section 6, the last section.

3. Submodular Functions

For a set $\mathcal{I}$, a function $f : 2^{\mathcal{I}} \to \mathbb{R}$ is *submodular* if, for all subsets $\mathcal{I}', \mathcal{I}'' \subseteq \mathcal{I}$,

$$f(\mathcal{I}') + f(\mathcal{I}'') \geq f(\mathcal{I}' \cup \mathcal{I}'') + f(\mathcal{I}' \cap \mathcal{I}'').$$

Furthermore, such a function is *non-decreasing* if $f(\mathcal{I}') \leq f(\mathcal{I}'')$ whenever $\mathcal{I}' \subseteq \mathcal{I}''$.

Now suppose that f is a non-negative, non-decreasing, submodular function on $2^{\mathcal{I}}$ which is computable in polynomial time. Let c be a function on $\mathcal{I}$ into the non-negative integers, and let B be a non-negative integer. Here, view c as a cost function on $\mathcal{I}$ and B as a budget. For a subset $\mathcal{I}'$ of $\mathcal{I}$, denote $\sum_{I \in \mathcal{I}'} c(I)$ by $c(\mathcal{I}')$. The problem we are interested in is to find a subset $\mathcal{I}'$ of $\mathcal{I}$ which maximizes f such that $c(\mathcal{I}') \leq B$, that is,

$$\max_{\mathcal{I}' \subseteq \mathcal{I}} \left\{ f(\mathcal{I}') : c(\mathcal{I}') \leq B \right\} \tag{1}$$

Sviridenko (2004) showed that the following greedy algorithm (and its subroutine) is a $(1-1/e)$-approximation algorithm for (1).

```
ApproxFunction($\mathcal{I}, f, c, B$)
    Find $\mathcal{I}'$ in $\{\mathcal{I}'' : \mathcal{I}'' \subseteq \mathcal{I}, c(\mathcal{I}'') \leq B, |\mathcal{I}''| \leq 2\}$ that maximizes $f$
    $H_1 \leftarrow \mathcal{I}'$
    $H_2 \leftarrow \emptyset$
    For all $\mathcal{I}_0 \subseteq \mathcal{I}$, such that $|\mathcal{I}_0| = 3$ and $c(\mathcal{I}_0) \leq B$ do
        $U \leftarrow \mathcal{I} \backslash \mathcal{I}_0$
        $\mathcal{I}' \leftarrow$Greedy($\mathcal{I}_0, U$)
        if $f(\mathcal{I}') > f(H_2)$ then $H_2 \leftarrow \mathcal{I}'$
    If $f(H_1) > f(H_2)$, then Return $H_1$, otherwise Return $H_2$

Greedy($\mathcal{I}_0, U$)
    $\mathcal{I}' \leftarrow \mathcal{I}_0$
    Repeat
        select $I \in U$ that maximizes $\frac{f(\mathcal{I}' \cup I) - f(\mathcal{I}')}{c(I)}$
        if $c(\mathcal{I}') + c(I) \leq B$ then
            $\mathcal{I}' \leftarrow \mathcal{I}' \cup \{I\}$
        $U \leftarrow U \backslash I$
```

```
Until $U = \emptyset$
Return $\mathcal{I}'$
```

4. A $(1 - 1/e)$-Approximation Algorithm for $\mathrm{BNRS}_{(\lambda,\mathcal{P})}$

In this section, we show that there is a polynomial-time $\left(1 - \frac{1}{e}\right)$-approximation algorithm for $\mathrm{BNRS}_{(\lambda,\mathcal{P})}$. We begin with two lemmas showing that $PD_{(\lambda,P)}(X, p_{\mathcal{R}'})$ is a non-negative, non-decreasing, submodular function and that it is computable in polynomial time. Throughout the section, we assume that all rooted phylogenetic trees in $\mathcal{P}$ are binary. (A rooted phylogenetic tree is *binary* if its root has degree two and all other internal vertices have degree three.) By allowing edges to have length zero, it is easily checked that no generality is lost by this assumption.

Lemma 4.1. *Let $\mathcal{P}$ be a collection of weighted rooted phylogenetic X-trees and let $\mathcal{R}$ be a collection of subsets of X. For all $(x, R) \in X \times \mathcal{R}$, let $a(x, R)$ and $b(x, R)$ be probabilities, where $b(x, R) \geq a(x, R)$. Then the function $PD_{(\lambda,\mathcal{P})} : 2^{\mathcal{R}} \rightarrow \mathbb{R}^{\geq 0}$ is computable in time $O(|\mathcal{P}||X| + |X||\mathcal{R}|)$.*

Proof. Recall that

$$PD_{(\lambda,\mathcal{P})}(X, p) = \sum_{j=1}^{k} w(\mathcal{T}_j) \int_{t \in \mathcal{T}_j} \mathbb{P}(t \rightarrow X) \mathrm{d}t,$$

where $(t \rightarrow X)$ denotes the event that a feature arising at point t on $\mathcal{T}_j$ survives to be present in a taxa in X which itself survives. We first consider the time to compute $PD_{(\lambda,\mathcal{P})}$ for a single unweighted tree $\mathcal{T}$. That is, the time to compute

$$PD_{(\lambda,\mathcal{T})}(X, p) = \int_{t \in \mathcal{T}} \mathbb{P}(t \rightarrow X) \mathrm{d}t.$$

The first step to computing $PD_{(\lambda,\mathcal{T})}$ is to compute, for each vertex v of $\mathcal{T}$, the probability p_v that a feature which has survived to that point survives from v to be present in a surviving leaf. This is done by beginning with the leaves and working up through $\mathcal{T}$ towards its root. If v is a leaf, then $v \in X$ and so $p_v = p(v)$. For each such v, we can compute $p(v)$ in time polynomial in $|\mathcal{R}|$ via a product of $|\mathcal{R}|$ extinction probabilities. Thus computing for all $v \in X$ takes total time polynomial in $|X||\mathcal{R}|$. Since this computation is independent of $\mathcal{T}$, it can be reused for each of the other trees in $\mathcal{P}$. If v is not a leaf, then v has children, w and w' say, connected by edges with lengths l and l', respectively, and

$$p_v = e^{-\lambda l} p_w + e^{-\lambda l'} p_{w'} - e^{-\lambda(l+l')} p_w p_{w'}.$$

With this in hand, we may now compute the contribution of each edge $e = \{u_e, v_e\}$ of $\mathcal{T}$ towards $PD_{(\lambda,\mathcal{P})}$:

$$\int_{t \in e} \mathbb{P}(t \rightarrow X) \mathrm{d}t = \int_0^{l_e} p_{v_e} e^{-\lambda x} \mathrm{d}x = \frac{p_{v_e}}{\lambda}(1 - e^{-\lambda l_e}),$$

where v_e is the endvertex of the edge e furthest from the root of $\mathcal{T}$ and l_e is the length of e. Thus

$$PD_{(\lambda,\mathcal{T})} = \int_{t\in\mathcal{T}} \mathbb{P}(t \to X)\mathrm{d}t = \sum_{e\in\mathcal{T}} \frac{p_{v_e}}{\lambda}(1 - e^{-\lambda l_e}).$$

Ignoring the computation of $p(v)$ for each $v \in X$, since $\mathcal{T}$ has $O(|X|)$ vertices and edges, the value p_v at all vertices v of $\mathcal{T}$ and the contribution of all edges of $\mathcal{T}$ towards $PD_{(\lambda,\mathcal{T})}$ can be computed in time $O(|X|)$. Thus, up to the computation of $p(v)$ for each $v \in X$, the contribution of each tree in $\mathcal{P}$ towards $PD_{(\lambda,\mathcal{P})}$ can be computed in time $O(|X|)$. It now follows that the full weighted-sum $PD_{(\lambda,\mathcal{P})}(X, p)$ can be computed in time $O(|\mathcal{P}||X| + |X||\mathcal{R}|)$. □

Lemma 4.2. *Let $\mathcal{P}$ be a collection of weighted rooted phylogenetic X-trees and let $\mathcal{R}$ be a collection of subsets of X. For all $(x, R) \in X \times \mathcal{R}$, let $a(x, R)$ and $b(x, R)$ be probabilities, where $b(x, R) \geq a(x, R)$. Then the function $PD_{(\lambda,\mathcal{P})} : 2^{\mathcal{R}} \to \mathbb{R}^{\geq 0}$ is a non-negative, non-decreasing, submodular function.*

Proof. Since $b(x, R) \geq a(x, R)$ for all $(x, R) \in X \times \mathcal{R}$, it follows that, for any point t on a rooted phylogenetic X-tree $T_j \in \mathcal{P}$, the probability that a feature arising at t survives to be present in a surviving taxa is non-decreasing in the set $\mathcal{R}' \subseteq \mathcal{R}$. That is, if we enlarge $\mathcal{R}'$, then the probability of t surviving cannot decrease. Thus, from the definition, $PD_{(\lambda,\mathcal{P})}$ is non-decreasing. Since $PD_{(\lambda,\mathcal{P})}$ is certainly non-negative, it remains to show that it is submodular, that is, for any two subsets $\mathcal{S}, \mathcal{T} \subseteq \mathcal{R}$,

$$\begin{aligned}(2) \qquad PD_{(\lambda,\mathcal{P})}(X, p_{\mathcal{S}}) + PD_{(\lambda,\mathcal{P})}(X, p_{\mathcal{T}}) \geq \\ PD_{(\lambda,\mathcal{P})}(X, p_{\mathcal{S}\cup\mathcal{T}}) + PD_{(\lambda,\mathcal{P})}(X, p_{\mathcal{S}\cap\mathcal{T}}).\end{aligned}$$

To establish (2), it is sufficient to show, by linearity, that, for any point t on an arbitrary rooted phylogenetic tree $\mathcal{T}_j \in \mathcal{P}$, the probability of survival of a feature arising at t is submodular. In turn, by linearity, it is sufficient to show that this holds when t coincides with a vertex of $\mathcal{T}_j$. To this end, for a vertex v of $\mathcal{T}_j$ and a subset $\mathcal{R}'$ of $\mathcal{R}$, let $p_{\mathcal{R}'}(v)$ denote the probability that a feature arising at v survives to be present in some taxon which itself survives when the reserves in $\mathcal{R}'$ are selected for conservation. Thus, to establish (2), it suffices to show that

$$(3) \qquad p_{\mathcal{S}}(v) + p_{\mathcal{T}}(v) - p_{\mathcal{S}\cup\mathcal{T}}(v) - p_{\mathcal{S}\cap\mathcal{T}}(v) \geq 0.$$

We prove (3) by induction on the maximum number of vertices in a path from v to one of its descendants in X. For the base case, suppose that v is itself a leaf x. Let $\mathcal{R}_1$ denote the set of reserves in $\mathcal{S}$ but not in $\mathcal{T}$, let $\mathcal{R}_2$ denote the reserves in $\mathcal{S} \cap \mathcal{T}$, let $\mathcal{R}_3$ denote the reserves in $\mathcal{T}$ but not in $\mathcal{S}$, and let $\mathcal{R}_4$ denote the reserves in $\mathcal{R}$ but not in $\mathcal{S} \cup \mathcal{T}$. For all $i \in \{1, 2, 3, 4\}$, let b_i (respectively, a_i) denote the probability that x survives in some reserve in $\mathcal{R}_i$ when the reserves in $\mathcal{R}_i$ are (respectively, are not) selected for conservation. By the independence of the reserves, it follows that

$$\begin{aligned}
p_{\mathcal{S}}(x) &= 1-(1-b_1)(1-b_2)(1-a_3)(1-a_4)\\
p_{\mathcal{T}}(x) &= 1-(1-a_1)(1-b_2)(1-b_3)(1-a_4)\\
p_{\mathcal{S}\cup\mathcal{T}}(x) &= 1-(1-b_1)(1-b_2)(1-b_3)(1-a_4)\\
p_{\mathcal{S}\cap\mathcal{T}}(x) &= 1-(1-a_1)(1-b_2)(1-a_3)(1-a_4).
\end{aligned}$$

Now

$$\begin{aligned}
p_{\mathcal{S}}(x)+p_{\mathcal{T}}(x)-p_{\mathcal{S}\cup\mathcal{T}}(x)-&p_{\mathcal{S}\cap\mathcal{T}}(x)\\
&= (1-b_2)(1-a_4)(b_3-a_3)(b_1-a_1).
\end{aligned}$$

Since $b_3-a_3, b_1-a_1 \geq 0$, it follows that

$$p_{\mathcal{S}}(x)+p_{\mathcal{T}}(x)-p_{\mathcal{S}\cup\mathcal{T}}(x)-p_{\mathcal{S}\cap\mathcal{T}}(x) \geq 0,$$

thus establishing the base case.

Now assume that (3) holds for vertices w and w', where w and w' are the child vertices of v. Let l and l' be the lengths of the edges $\{v,w\}$ and $\{v,w'\}$, respectively. Then, for a subset $\mathcal{R}'$ of $\mathcal{R}$,

$$p_{\mathcal{R}'}(v) = e^{-\lambda l}p_{\mathcal{R}'}(w)+e^{-\lambda l'}p_{\mathcal{R}'}(w')-e^{-\lambda(l+l')}p_{\mathcal{R}'}(w)p_{\mathcal{R}'}(w').$$

Therefore

$$\begin{aligned}
&p_{\mathcal{S}}(v)+p_{\mathcal{T}}(v)-p_{\mathcal{S}\cup\mathcal{T}}(v)-p_{\mathcal{S}\cap\mathcal{T}}(v)\\
&\quad= e^{-\lambda l}\big(p_{\mathcal{S}}(w)+p_{\mathcal{T}}(w)-p_{\mathcal{S}\cup\mathcal{T}}(w)-p_{\mathcal{S}\cap\mathcal{T}}(w)\big)\\
&\qquad+e^{-\lambda l'}\big(p_{\mathcal{S}}(w')+p_{\mathcal{T}}(w')-p_{\mathcal{S}\cup\mathcal{T}}(w')-p_{\mathcal{S}\cap\mathcal{T}}(w')\big)\\
&\qquad-e^{-\lambda(l+l')}\big(p_{\mathcal{S}}(w)p_{\mathcal{S}}(w')+p_{\mathcal{T}}(w)p_{\mathcal{T}}(w')-p_{\mathcal{S}\cup\mathcal{T}}(w)p_{\mathcal{S}\cup\mathcal{T}}(w')-p_{\mathcal{S}\cap\mathcal{T}}(w)p_{\mathcal{S}\cap\mathcal{T}}(w')\big)
\end{aligned}$$

Without loss of generality, we may assume that $p_{\mathcal{S}}(w) \geq p_{\mathcal{T}}(w)$. Observing that

$$p_{\mathcal{S}\cup\mathcal{T}}(w) \geq p_{\mathcal{S}}(w) \geq p_{\mathcal{T}}(w) \geq p_{\mathcal{S}\cap\mathcal{T}}(w),$$

set $\epsilon, \delta \geq 0$ such that $p_{\mathcal{S}\cup\mathcal{T}}(w) = p_{\mathcal{S}}(w)+\epsilon$ and $p_{\mathcal{T}}(w) = p_{\mathcal{S}\cap\mathcal{T}}(w)+\delta$. By submodularity at w,

$$p_{\mathcal{S}}(w)+p_{\mathcal{T}}(w)-p_{\mathcal{S}\cup\mathcal{T}}(w)-p_{\mathcal{S}\cap\mathcal{T}}(w) \geq 0,$$

and so $\delta \geq \epsilon$. The rest of the induction proof is broken into two cases: (i) $p_{\mathcal{S}}(w') \geq p_{\mathcal{T}}(w')$ and (ii) $p_{\mathcal{S}}(w') < p_{\mathcal{T}}(w')$.

For (i), set $\epsilon', \delta' \geq 0$ such that $p_{\mathcal{S}\cup\mathcal{T}}(w') = p_{\mathcal{S}}(w') + \epsilon'$ and $p_{\mathcal{T}}(w') = p_{\mathcal{S}\cap\mathcal{T}}(w') + \delta'$. Then

$$\begin{aligned}
&p_{\mathcal{S}}(v) + p_{\mathcal{T}}(v) - p_{\mathcal{S}\cup\mathcal{T}}(v) - p_{\mathcal{S}\cap\mathcal{T}}(v)\\
&\quad = e^{-\lambda l}(\delta - \epsilon) + e^{-\lambda l'}(\delta' - \epsilon') - e^{-\lambda(l+l')}\big(p_{\mathcal{S}}(w)p_{\mathcal{S}}(w') - (p_{\mathcal{S}}(w) + \epsilon)(p_{\mathcal{S}}(w') + \epsilon')\\
&\qquad + (p_{\mathcal{S}\cap\mathcal{T}}(w) + \delta)(p_{\mathcal{S}\cap\mathcal{T}}(w') + \delta') - p_{\mathcal{S}\cap\mathcal{T}}(w)p_{\mathcal{S}\cap\mathcal{T}}(w')\big)\\
&\quad = e^{-\lambda l}\delta\big(1 - e^{-\lambda l'}p_{\mathcal{S}\cap\mathcal{T}}(w') - e^{-\lambda l'}\delta'\big) - e^{-\lambda l}\epsilon\big(1 - e^{-\lambda l'}p_{\mathcal{S}}(w') - e^{-\lambda l'}\epsilon'\big)\\
&\qquad + e^{-\lambda l'}\delta'\big(1 - e^{-\lambda l}p_{\mathcal{S}\cap\mathcal{T}}(w)\big) - e^{-\lambda l'}\epsilon'\big(1 - e^{-\lambda l}p_{\mathcal{S}}(w)\big)\\
&\quad = e^{-\lambda l}\delta\big(1 - e^{-\lambda l'}p_{\mathcal{T}}(w')\big) - e^{-\lambda l}\epsilon\big(1 - e^{-\lambda l'}p_{\mathcal{S}\cup\mathcal{T}}(w')\big)\\
&\qquad + e^{-\lambda l'}\delta'\big(1 - e^{-\lambda l}p_{\mathcal{S}\cap\mathcal{T}}(w)\big) - e^{-\lambda l'}\epsilon'\big(1 - e^{-\lambda l}p_{\mathcal{S}}(w)\big).
\end{aligned}$$

Since $\delta \geq \epsilon$ and $p_{\mathcal{T}}(w') \leq p_{\mathcal{S}\cup\mathcal{T}}(w')$,

$$e^{-\lambda l}\delta\big(1 - e^{-\lambda l'}p_{\mathcal{T}}(w')\big) - e^{-\lambda l}\epsilon\big(1 - e^{-\lambda l'}p_{\mathcal{S}\cup\mathcal{T}}(w')\big) \geq 0. \tag{4}$$

Furthermore, $\delta' \geq \epsilon'$ and $p_{\mathcal{S}\cap\mathcal{T}}(w) \leq p_{\mathcal{S}}(w)$, so

$$e^{-\lambda l'}\delta'\big(1 - e^{-\lambda l}p_{\mathcal{S}\cap\mathcal{T}}(w)\big) - e^{-\lambda l'}\epsilon'\big(1 - e^{-\lambda l}p_{\mathcal{S}}(w)\big) \geq 0. \tag{5}$$

Combining (4) and (5),

$$p_{\mathcal{S}}(v) + p_{\mathcal{T}}(v) - p_{\mathcal{S}\cup\mathcal{T}}(v) - p_{\mathcal{S}\cap\mathcal{T}}(v) \geq 0,$$

completing the induction proof for (i).

Consider (ii), where $p_{\mathcal{S}}(w') < p_{\mathcal{T}}(w')$. For this case, set $\epsilon', \delta' \geq 0$ such that $p_{\mathcal{S}\cup\mathcal{T}}(w') = p_{\mathcal{T}}(w') + \epsilon'$ and $p_{\mathcal{S}}(w') = p_{\mathcal{S}\cap\mathcal{T}}(w') + \delta'$. By submodularity at w',

$$p_{\mathcal{S}}(w') + p_{\mathcal{T}}(w') - p_{\mathcal{S}\cup\mathcal{T}}(w') - p_{\mathcal{S}\cap\mathcal{T}}(w') \geq 0,$$

so $\delta' \geq \epsilon'$. Now

$$\begin{aligned}
&p_{\mathcal{S}}(v) + p_{\mathcal{T}}(v) - p_{\mathcal{S}\cup\mathcal{T}}(v) - p_{\mathcal{S}\cap\mathcal{T}}(v)\\
&\quad = e^{-\lambda l}(\delta - \epsilon) + e^{-\lambda l'}(\delta' - \epsilon') - e^{-\lambda(l+l')}\big(p_{\mathcal{S}}(w)(p_{\mathcal{S}\cap\mathcal{T}}(w') + \delta') + (p_{\mathcal{S}\cap\mathcal{T}}(w) + \delta)p_{\mathcal{T}}(w')\\
&\qquad - (p_{\mathcal{S}}(w) + \epsilon)(p_{\mathcal{T}}(w') + \epsilon') - p_{\mathcal{S}\cap\mathcal{T}}(w)p_{\mathcal{S}\cap\mathcal{T}}(w')\big)\\
&\quad = e^{-\lambda l}\delta\big(1 - e^{-\lambda l'}p_{\mathcal{T}}(w')\big) - e^{-\lambda l}\epsilon\big(1 - e^{-\lambda l'}p_{\mathcal{T}}(w') - e^{-\lambda l'}\epsilon'\big)\\
&\qquad + e^{-\lambda l'}\delta'\big(1 - e^{-\lambda l}p_{\mathcal{S}}(w)\big) - e^{-\lambda l'}\epsilon'\big(1 - e^{-\lambda l}p_{\mathcal{S}}(w)\big)\\
&\qquad + e^{-\lambda(l+l')}\big(p_{\mathcal{S}\cap\mathcal{T}}(w)p_{\mathcal{S}\cap\mathcal{T}}(w') - p_{\mathcal{S}}(w)p_{\mathcal{S}\cap\mathcal{T}}(w') - p_{\mathcal{S}\cap\mathcal{T}}(w)p_{\mathcal{T}}(w') + p_{\mathcal{S}}(w)p_{\mathcal{T}}(w')\big)\\
&\quad = e^{-\lambda l}\delta\big(1 - e^{-\lambda l'}p_{\mathcal{T}}(w')\big) - e^{-\lambda l}\epsilon\big(1 - e^{-\lambda l'}p_{\mathcal{S}\cup\mathcal{T}}(w')\big)\\
&\qquad + e^{-\lambda l'}(\delta' - \epsilon')\big(1 - e^{-\lambda l}p_{\mathcal{S}}(w)\big)\\
&\qquad + e^{-\lambda(l+l')}\big((p_{\mathcal{S}}(w) - p_{\mathcal{S}\cap\mathcal{T}}(w)\big)\big(p_{\mathcal{T}}(w') - p_{\mathcal{S}\cap\mathcal{T}}(w')\big).
\end{aligned}$$

Since $\delta \geq \epsilon$ and $p_{\mathcal{T}}(w') \leq p_{\mathcal{S}\cup\mathcal{T}}(w')$,

$$e^{-\lambda l}\delta\big(1 - e^{-\lambda l'}p_{\mathcal{T}}(w')\big) - e^{-\lambda l}\epsilon\big(1 - e^{-\lambda l'}p_{\mathcal{S}\cup\mathcal{T}}(w')\big) \geq 0. \tag{6}$$

Furthermore, as $\delta' \geq \epsilon'$,

$$e^{-\lambda l'}(\delta' - \epsilon')\big(1 - e^{-\lambda l}p_{\mathcal{S}}(w)\big) \geq 0 \tag{7}$$

and, as $p_{\mathcal{S}}(w) \geq p_{\mathcal{S}\cap\mathcal{T}}(w)$ and $p_{\mathcal{T}}(w') \geq p_{\mathcal{S}\cap\mathcal{T}}(w')$,

$$e^{-\lambda(l+l')}\big((p_{\mathcal{S}}(w) - p_{\mathcal{S}\cap\mathcal{T}}(w)\big)\big(p_{\mathcal{T}}(w') - p_{\mathcal{S}\cap\mathcal{T}}(w')\big) \geq 0. \tag{8}$$

Combining (6), (7), and (8), we get that

$$p_{\mathcal{S}}(v) + p_{\mathcal{T}}(v) - p_{\mathcal{S}\cup\mathcal{T}}(v) - p_{\mathcal{S}\cap\mathcal{T}}(v) \geq 0,$$

completing the induction proof of (ii). Hence $PD_{(\lambda,\mathcal{P})}$ is submodular, thereby completing the proof of the lemma. □

Consider $\text{BNRS}_{(\lambda,\mathcal{P})}$. Let $\text{ApproxBNRS}_{(\lambda,\mathcal{P})}$ denote the algorithm obtained from ApproxFunction (see Section 3) by replacing $\mathcal{I}$, f, c, and B with $\mathcal{R}$, $PD_{(\lambda,\mathcal{P})}$, c, and B, respectively. The first part of the next theorem immediately follows from Lemma 4.2 and 4.1, and Sviridenko (2004) (see Section 3), while the second part follows from the fact that BNRS is a special case of $\text{BNRS}_{(\lambda,\mathcal{P})}$.

Theorem 4.3. *$\text{ApproxBNRS}_{(\lambda,\mathcal{P})}$ is a polynomial-time $\big(1-\frac{1}{e}\big)$-approximation algorithm for $\text{BNRS}_{(\lambda,\mathcal{P})}$. Moreover, for any $\delta > 0$, $\text{BNRS}_{(\lambda,\mathcal{P})}$ cannot be approximated with an approximation ratio of $\big(1-\frac{1}{e}+\delta\big)$ unless* P=NP.

5. A $(1-1/e)$-Approximation Algorithm for BNRS_Σ

Bordewich and Semple (2008) showed that there is a polynomial-time $(1-1/e)$-approximation algorithm for when BNRS is restricted to compatible split systems. In this section, we extend this result to arbitrary split systems. We begin by showing that a certain function is submodular.

Lemma 5.1. *Let Σ be a weighted split system of X, let Y be a distinguished non-empty subset of X, and let $\mathcal{R}$ be a collection of subsets of X. Then the function $PD_{(Y,\Sigma)} : 2^{\mathcal{R}} \rightarrow \mathbb{R}^{\geq 0}$ defined, for all subsets $\mathcal{R}'$ of $\mathcal{R}$, by the PD_Σ score of $Y \cup \bigcup_{R\in\mathcal{R}'} R$ is a submodular function.*

Proof. It suffices to show (see, for example, Nemhauser et al. (1978)) that if $\mathcal{R}'' \subseteq \mathcal{R}' \subseteq \mathcal{R}$ and $Q \in \mathcal{R} - \mathcal{R}'$, then

$$PD_{(Y,\Sigma)}(\mathcal{R}' \cup \{Q\}) - PD_{(Y,\Sigma)}(\mathcal{R}') \leq PD_{(Y,\Sigma)}(\mathcal{R}'' \cup \{Q\}) - PD_{(Y,\Sigma)}(\mathcal{R}'').$$

Let $A|B$ be an element of Σ such that $w(A|B)$ contributes a non-zero weighting to $PD_{(Y,\Sigma)}(\mathcal{R}' \cup \{Q\}) - PD_{(Y,\Sigma)}(\mathcal{R}')$. Then either $(Y \cup \bigcup_{R\in\mathcal{R}'} R) \subseteq A$ or $(Y \cup \bigcup_{R\in\mathcal{R}'} R) \subseteq B$ and there is an element $q \in Q$ such that $q \in B$ or $q \in A$, respectively. Since Y is non-empty and $\mathcal{R}'' \subseteq \mathcal{R}'$, it follows that $w(A|B)$ contributes a non-zero weighting to $PD_{(Y,\Sigma)}(\mathcal{R}'' \cup \{Q\}) - PD_{(Y,\Sigma)}(\mathcal{R}'')$, and so the lemma holds. □

Consider BNRS_Σ and let Q be a fixed element in $\mathcal{R}$. Let $\text{ApproxBNRS}_{(Q,\Sigma)}$ denote the algorithm obtained from ApproxFunction by replacing $\mathcal{I}$, f, c, and B with $\mathcal{R} - Q$, $PD_{(Q,\Sigma)}$, c_Q, and $B - c(Q)$, where c_Q is the cost function on the sets in $\mathcal{R} - Q$ defined, for all $R \in \mathcal{R} - Q$, by

$c_Q(R) = c(R)$. The next theorem shows that the following algorithm is a polynomial-time $\big(1 - \frac{1}{e}\big)$-approximation algorithm for BNRS$_\Sigma$.

```
ApproxBNRS$_\Sigma$($\mathcal{R}, PD_\Sigma, c, B$)
    $H \leftarrow \emptyset$
    For all $Z \in \mathcal{R}$ do
        $\mathcal{R}' \leftarrow$ApproxBNRS$_{(Z,\Sigma)}$
        if $PD_\Sigma(\mathcal{R}' \cup \{Z\}) > PD_\Sigma(H)$ then $H \leftarrow \mathcal{R}' \cup \{Z\}$
    Return $H$
```

Theorem 5.2. ApproxBNRS$_\Sigma$ *is a polynomial-time* $\big(1-\frac{1}{e}\big)$*-approximation algorithm for* BNRS$_\Sigma$. *Moreover, for any* $\delta > 0$, BNRS$_\Sigma$ *cannot be approximated with an approximation ratio of* $(1 - \frac{1}{e} + \delta)$ *unless* P=NP.

Proof. In essence, we run through each possible choice of set Q and approximate BNRS$_\Sigma$ assuming Q is in the solution. We must be right for some Q and hence find a good approximation. Let Q be a fixed element in $\mathcal{R}$. Then, by Lemma 5.1, the function $PD_{(Q,\Sigma)} : 2^{\mathcal{R}-Q} \to \mathbb{R}^{\geq 0}$ defined, for all subsets $\mathcal{R}'$ of $\mathcal{R}-Q$, by the PD_Σ score of $Q \cup \bigcup_{R\in\mathcal{R}'} R$ is a submodular function. Furthermore, $PD_{(Q,\Sigma)}$ is certainly non-negative, non-decreasing, and computable in polynomial time. It now follows by Sviridenko (2004) that ApproxBNRS$_{(Q,\Sigma)}$ is a polynomial-time $\big(1 - \frac{1}{e}\big)$-approximation algorithm for BNRS$_\Sigma$ for when the selected set of reserves includes Q.

Let $\mathcal{R}^*$ be an optimal solution to BNRS$_\Sigma$ and now let Q be an element of $\mathcal{R}^*$. Then $\mathcal{R}^*$ is an optimal solution to BNRS$_\Sigma$ for when the selected set of reserves includes Q. Let $\mathcal{R}'$ be the subset of $\mathcal{R} - Q$ returned by ApproxBNRS$_{(Q,\Sigma)}$ applied to $\mathcal{R} - Q$, $PD_{(Q,\Sigma)}$, c_Q, and $B - c(Q)$. It now follows that the PD_Σ score of $Q \cup \mathcal{R}'$ is at least $\big(1 - \frac{1}{e}\big)$ times the PD_Σ score of $\mathcal{R}^*$, and so the theorem holds. □

6. No Better Approximation for Split Systems

In this section, we establish the following theorem, thereby resolving the problem left open by Spillner et al. (2008) at the end of Section 2.

Theorem 6.1. *For any* $\delta > 0$, SplitsPD *cannot be approximated with an approximation ratio of* $\big(1 - \frac{1}{e} + \delta\big)$ *unless* P=NP.

The proof of Theorem 6.1 is via a reduction from Max-k-Cover.

Problem: Maximum k-coverage (Max-k-Cover)
Instance: A finite set D, a collection $\mathcal{C}$ of subsets of D, and a positive integer k.
Question: Find a subset $\mathcal{B} = \{B_1, B_2, \ldots, B_k\}$ of $\mathcal{C}$ of size k that maximizes the size of the set

$$B_1 \cup B_2 \cup \cdots \cup B_k.$$

Feige (1998) showed that no polynomial-time approximation algorithm for Max-k-Cover can have an approximation ratio better than $\left(1-\frac{1}{e}\right)$ unless P=NP.

Proof of Theorem 6.1. Let $(D, \mathcal{C}, k)$ be an instance of Max-k-Cover. We construct an instance of SplitsPD as follows. Let $X = \mathcal{C} \cup \big\{\{\rho\}\big\}$, where ρ is a distinguished element not in D and, for each $d \in D$, let $\sigma_d = A_d|(X - A_d)$, where

$$A_d = \{C \in \mathcal{C} : d \in C\}.$$

Let Σ be the split system $\{\sigma_d : d \in D\} \cup \big\{\{\rho\}|(X - \{\rho\})\big\}$ with weighting $w(\sigma_d) = 1$ for all $d \in D$ and $w(\{\rho\}|(X - \{\rho\}) = \omega$. An appropriate choice for ω will be made shortly. The triple $(X, \Sigma, k+1)$ is our constructed instance of SplitsPD. For simplicity, throughout the proof, we will always assume that the instances of Max-k-Cover and SplitsPD are $(D, \mathcal{C}, k)$ and $(X, \Sigma, k+1)$, respectively.

Let $\mathcal{B}_k$ be an optimal solution to Max-k-Cover, and suppose that it covers b_k elements of D. In terms of SplitsPD, consider the PD score of $\mathcal{B}_k \cup \{\{\rho\}\}$. Since $\{\rho\}$ is an element of $\mathcal{B}_k \cup \{\{\rho\}\}$, this score is the sum of the size of the cover of $\mathcal{B}_k$ and $w(\{\rho\}|(X - \{\rho\})$. That is, the score is $b_k + \omega$. We next determine for what values of ω is $\mathcal{B}_k \cup \{\{\rho\}\}$ guaranteed to be an optimal solution for SplitsPD.

Now there is no set $\mathcal{B}_{k+1} \subseteq \mathcal{C}$ of size $k+1$ which covers more than $b_k \frac{k+1}{k}$ elements of D. To see this, let $\mathcal{B}_{k+1}$ be an arbitrary subset of $\mathcal{C}$ of size $k+1$ that covers b_{k+1} elements of D. First observe that by considering the marginal contribution of each set in $\mathcal{B}_{k+1}$, there is a set in $\mathcal{B}_{k+1}$ whose removal results in a subset of $\mathcal{C}$ of size k that covers at least $b_{k+1} - \frac{b_{k+1}}{k+1}$ elements of D. By the optimality of $\mathcal{B}_k$,

$$b_k \geq b_{k+1} - \frac{b_{k+1}}{k+1} = b_{k+1}\left(1 - \frac{1}{k+1}\right) = b_{k+1}\left(\frac{k}{k+1}\right),$$

and so $b_{k+1} \leq b_k \frac{k+1}{k}$. Since $b_k \frac{k+1}{k} = b_k + \frac{b_k}{k}$, it now follows that we can guarantee $\mathcal{B}_k \cup \{\{\rho\}\}$ is an optimal solution of SplitsPD if $\omega > \frac{b_k}{k}$. Using this fact, we complete the proof by showing that if we can approximate SplitsPD to within a ratio $\left(1 - \frac{1}{e} + \delta\right)$ for some $\delta > 0$, then we can approximate Max-k-Cover to within a ratio better than $\left(1 - \frac{1}{e}\right)$; contradicting Feige (1998).

Suppose that we can approximate SplitsPD to within such a ratio. Since Max-k-Cover can always be solved in polynomial time for constant size k, we may assume that k is large enough so that $\frac{2}{k} < \delta$. By Feige (1998), there is a polynomial-time $\left(1 - \frac{1}{e}\right)$-approximation algorithm for the above instance of Max-k-Cover. Therefore, we can approximate in polynomial time the optimal value b_k with approximation ratio $\left(1 - \frac{1}{e}\right)$, in particular, as $1 - \frac{1}{e} > \frac{1}{2}$, we can compute a weight $\frac{b_k}{k} < \omega \leq \frac{2b_k}{k}$ in polynomial time. It now follows that the optimal solution to SplitsPD is given by the set $\mathcal{B}_k \cup \{\{\rho\}\}$ and has value $b_k + \omega \leq b_k + \frac{2b_k}{k} = b_k\left(1 + \frac{2}{k}\right)$.

Let β be the answer returned by applying our assumed polynomial-time $\big(1-\frac{1}{e}+\delta\big)$-approximation algorithm to the above instance of SplitsPD. Then $\beta \geq \big(1-\frac{1}{e}+\delta\big)(b_k+\omega)$, and so, as $b_k+\omega \geq \beta$,

$$\begin{aligned} b_k &\geq \beta-\omega \\ &\geq \big(1-\tfrac{1}{e}+\delta\big)(b_k+\omega)-\omega \\ &> \big(1-\tfrac{1}{e}+\delta\big)\big(b_k+\tfrac{b_k}{k}\big)-\tfrac{2b_k}{k} \\ &> b_k\big(\big(1-\tfrac{1}{e}+\delta\big)-\tfrac{2}{k}\big) \\ &= b_k\big(1-\tfrac{1}{e}+\big(\delta-\tfrac{2}{k}\big)\big). \end{aligned}$$

But, by our choice of k, we have $\delta-\frac{2}{k}>0$ and so $\beta-\omega$ gives a $\big(1-\frac{1}{e}+\big(\delta-\frac{2}{k}\big)\big)$-approximation to Max-$k$-Cover; a contradiction. This completes the proof of the theorem. □

We end this section with a short remark about the rooted version of SplitsPD. Calling it rSplitsPD, in this problem the instance is a finite set $X\cup\{\rho\}$, a split system Σ of $X\cup\{\rho\}$, and a non-negative integer k, and the question is to find a subset Z of X of size k that maximizes $PD(Z\cup\{\rho\})$.

Using Feige's tight approximation result for Max-k-Cover, it is straightforward to show that, for any δ, rSplitsPD cannot be approximated with an approximation ratio of $\big(1-\frac{1}{e}+\delta\big)$ unless P=NP. Briefly, similar to that in the proof of Theorem 6.1, let $(D,\mathcal{C},k)$ be an instance of Max-k-Cover. We construct an instance of rSplitsPD by setting $X=\mathcal{C}$ and, for each $d\in D$, setting $\sigma_d = A_d|((X\cup\{\rho\})-A_d)$, where

$$A_d=\{C\in\mathcal{C} : d\in C\}.$$

Now take Σ to be the split system $\{\sigma_d : d\in D\}$ with each split in Σ having weight 1. The triple $(X\cup\{\rho\},\Sigma,k)$ is our initial instance of rSplitsPD. If W be a subset of X of size k, then the PD score of $W\cup\{\rho\}$ is the size of the cover of W. Thus, as the reduction from Max-k-Cover to rSplitsPD can be done in time polynomial in the size of $(D,\mathcal{C},k)$, it follows that if there is a polynomial-time approximation algorithm for rSplitsPD with ratio $(1-\alpha)$, where $\alpha>0$, then there is also such an approximation algorithm for Max-k-cover. This establishes the desired outcome. The proof of Theorem 6.1 is a non-trivial modification of this approach, the difficulty lies in the fact that, in SplitsPD, ρ may not be in any optimal solution. In the terminology of this paper, this reduction from Max-k-Cover to rSPlitsPD is also shown by Faller (2010).

References

[1] Bordewich M, Rodrigo AD, Semple C (2008) Selecting taxa to save or sequence: desirable criteria and a greedy solution. Syst Biol 57:1–11

[2] Bordewich M, Semple C (2008) Nature reserve selection problem: a tight approximation algorithm. IEEE/ACM Trans Comput Biol Bioinform 5:275–280

[3] Bryant D, Huson D (2006) Applications of phylogenetic networks in evolutionary studies. Mol Biol Evol 23:254–267

[4] Faith DP (1992) Conservation evaluation and phylogenetic diversity. Biol Conserv 61:1–10

[5] Faith DP (1994) Phylogenetic pattern and the quantification of organismal biodiversity. Philos Trans Biol Sci 345:45–58

[6] Faller B (2010) Combinatorial and probabilistic methods in biodiversity theory. Dissertation, University of Canterbury

[7] Feige U (1998) A threshold of $\ln n$ for approximating set cover. J ACM 45:634–652

[8] Minh BQ, Klaere S, von Haeseler A (2009) Taxon selection under split diversity. Syst Biol 58:586–594

[9] Minh BQ, Klaere S, von Haeseler A (2010) SDA*: A simple and unifying solution to recent bioinformatic challenges for conservation genetics. In: Proceedings of the Second International Conference on Knowledge and Systems Engineering, pp 33–37

[10] Moritz C, Faith DP (1998) Comparative phylogeography and the identification of genetically divergent areas for conservation. Mol Ecol 7:419–429

[11] Moulton V, Semple C, Steel M (2007) Optimizing phylogenetic diversity under constraints. J Theor Biol 246:186–194

[12] Nemhauser GL, Wolsey LA, Fisher ML (1978) An analysis of approximations for maximizing submodular set functions - I. Math Program 14:265–294

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

[14] Pardi F, Goldman N (2007) Resource-aware taxon selection for maximizing phylogenetic diversity. Syst Biol 56:431–444

[15] Rodrigues ASL, Gaston KJ (2002) Maximising phylogenetic diversity in the selection of networks of conservation areas. Biol Conserv 105:103–111

[16] Rodrigues ASL, Brooks TM, Gaston KJ (2005) Integrating phylogenetic diversity in the selection of priority areas for conservation: does it make a difference. In: Purvis A, Gittleman JL, Brooks T (eds) Phylogeny and conservation. Cambridge University Press, Cambridge, pp 101–119

[17] Smith TB, Holder K, Girman D, O'Keefe K, Larison B, Chan Y (2000) Comparative avian phylogeography of Cameroon and Equatorial Guinea mountains: implications for conservation. Mol Ecol 9:1505–1516

[18] Spillner A, Nguyen B, Moulton V (2008) Computing phylogenetic diversity for split systems. IEEE/ACM Trans Comput Biol Bioinform 5:235–244

[19] Steel M (2005) Phylogenetic diversity and the greedy algorithm. Syst Biol 54:527–529

[20] Sviridenko M (2004) A note on maximizing a submodular set function subject to a knapsack constraint. Oper Res Lett 32:41–43

[1]School of Engineering Computing Sciences, Durham University, Durham DH1 3LE, United Kingdom

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

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

E-mail address: charles.semple@canterbury.ac.nz