

Parameterized Algorithms for Diversity of Networks with Ecological Dependencies

Mark Jones 

TU Delft, The Netherlands

Jannik Schestag 

TU Delft, The Netherlands

Abstract

For a phylogenetic tree, the phylogenetic diversity of a set A of taxa is the total weight of edges on paths to A . Finding small sets of maximal diversity is crucial for conservation planning, as it indicates where limited resources can be invested most efficiently. In recent years, efficient algorithms have been developed to find sets of taxa that maximize phylogenetic diversity either in a phylogenetic network or in a phylogenetic tree subject to ecological constraints, such as a food web. However, these aspects have mostly been studied independently. Since both factors are biologically important, it seems natural to consider them together.

In this paper, we introduce decision problems where, given a phylogenetic network, a food web, and integers k , and D , the task is to find a set of k taxa with phylogenetic diversity of at least D under the *maximize all paths* measure, while also satisfying viability conditions within the food web. Here, we consider different definitions of viability, which all demand that a “sufficient” number of prey species survive to support surviving predators.

We investigate the parameterized complexity of these problems and present several fixed-parameter tractable (FPT) algorithms. Specifically, we provide a complete complexity dichotomy characterizing which combinations of parameters—out of the size constraint k , the acceptable diversity loss $\bar{D}$, the scanwidth of the food web $\text{sw}_{\mathcal{F}}$, the maximum in-degree δ in the network, and the network height h —lead to $W[1]$ -hardness and which admit FPT algorithms.

Our primary methodological contribution is a novel algorithmic framework for solving phylogenetic diversity problems in networks where dependencies (such as those from a food web) impose an order, using a color coding approach.

2012 ACM Subject Classification Applied computing → Computational biology; Theory of computation → Fixed parameter tractability

Keywords and phrases Phylogenetic Diversity; Fixed-Parameter Tractability; Phylogenetic Networks; Food Webs; Color Coding

Digital Object Identifier [10.4230/LIPIcs...](https://doi.org/10.4230/LIPIcs...)

Funding *Mark Jones*: Partially funded by the Dutch Organisation for Scientific Research (NWO) grant OCENW.KLEIN.125 and OCENW.M.21.306.

Jannik Schestag: Funded by the Dutch Research Council (NWO), project “Optimization for and with Machine Learning (OPTIMAL)” OCENW.GROOT.2019.015.

1 Introduction

The Sixth Mass Extinction eliminates species and their genera at an unprecedented rate [5], even exceeding rates in previous mass extinction events [26]. The situation is severe enough that approximately a quarter of Earth’s existing species are at threat [10]. Inaction now means jeopardizing further parts of the animal tree of life.

Because resources are not sufficiently available to preserve all species, scientists developed the *phylogenetic diversity* measure [8] to cover the necessity of making an educated decision on which set of species the limited resources should be invested in. Given a phylogenetic tree with edge weights in which leaves represent present-day species, the phylogenetic diversity



© Jones and Schestag;
licensed under Creative Commons License CC-BY 4.0

[Leibniz International Proceedings in Informatics](https://www.leibniz-proceedings.de/)

Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

of a set of species A is the total weight of edges on paths from the root to species in A . Intuitively, a set of species with larger phylogenetic diversity captures a larger variety of genetic material, and is therefore expected to have larger biodiversity. Phylogenetic diversity became the most popular measure of the significance of a set of species [32].

Apart from its biological relevance, phylogenetic diversity probably also found favor in the eyes of many scientists for its highly desirable trait of being easy to compute [8]. Indeed, an optimal solution for maximizing phylogenetic diversity can be found with a greedy algorithm [24, 29] and therefore even large instances can be solved within seconds [21].

To model further relevant aspects of conservation planning, more problems were defined, in which a special focus was placed upon capturing varying costs of saving species [19, 25], finding optimal conservation areas [3, 22], considering species' extinction times [17], or preserving viable sets of species [22]. In the latter problem a food web, modeling predator-prey relationships, is given in addition to the phylogenetic tree. It is asked to find a set A of species that maximizes the phylogenetic diversity and is viable in the sense that each species in A is either a source in the ecosystem, or finds prey within A . Due to concerns that one prey could be insufficient, further definitions of viability have been defined [27].

The notion of phylogenetic trees has been generalized. Ancestry of species is in recent years more often modeled with phylogenetic networks, which in contrast to phylogenetic trees also allow hybridization events or horizontal gene transfer [15]. Consequently, generalizations of phylogenetic diversity on networks have been proposed [4, 16, 30, 31, 33].

As phylogenetic networks represent the connection between species a lot better than phylogenetic trees and considering viability constraints are vital, it is natural to combine these two aspects. Yet, to the best of our knowledge, this has not been done so far. Therefore, we take the step and define problems for the maximization of phylogenetic diversity on networks with different viability definitions. As it was expected that “a combination of these concepts [would] result in very hard problems.” [27], we turn to the toolbox of parameterized complexity to break intractability.

Parameterized complexity is one method to cope with NP-hardness. In this, we consider a problem Π and a parameter p of size κ and say that (Π, p) is FPT if Π can be solved in $\mathcal{O}^*(f(\kappa))$ time, for some computable function f . If Π is W[1]-hard with respect to p or even NP-hard if κ is a constant, then the existence of an FPT-algorithm is unlikely [6, 7].

We consider three definitions of viability, based on whether a non-source species in the food web needs to have one or all of its prey available, or whether a weighted sum of the available prey must reach a certain threshold. We provide a complete complexity dichotomy, for the latter two of these definitions, in the sense that for every combination of the following parameters, we show whether the defined problems are W[1]-hard, or can be solved with an FPT-result: the size constraint k ; the acceptable diversity loss $\overline{D}$; the scanwidth of the food web $\text{sw}_{\mathcal{F}}$; the maximum in-degree δ in the network; and the network height h . For the other definition of viability, we have a near-complete complexity dichotomy that omits only one of the above combinations. In particular, we provide FPT algorithms for the most general version of the problem we define for the parameters $\overline{D} + \text{sw}_{\mathcal{F}}$ and $k + \text{sw}_{\mathcal{F}} + \delta + h$. For the former, we depend on a notion of *anchors* already used in [17], but we improve on their algorithmic idea, in two ways. First, we use a color coding technique which only requires one color per edge. Secondly, we consider a non-linear order. By this, we are able to even provide algorithms for the smaller parameter $\overline{k} + \text{sw}_{\mathcal{F}}$ for the version of the problem on trees.

2 Preliminaries

2.1 Definitions

We use the $\mathcal{O}^*$ -notation which omits factors polynomial in the input size. For a positive integer a , by $[a]$ we denote the set $\{1, 2, \dots, a\}$, and by $[a]_0$ the set $\{0\} \cup [a]$. For functions $f : A \rightarrow B$, where B is a family of sets, we define $f(A') := \bigcup_{a \in A'} f(a)$, and if $B \subseteq \mathbb{Q}$, we define $f_\Sigma(A') := \sum_{a \in A'} f(a)$, for subsets $A' \subseteq A$. We write $\{u, v\}$ for an undirected edge between u and v and uv for a directed edge from u to v . For any graph G , we write $V(G)$ and $E(G)$ for the set of vertices or edges, respectively. For a set of edges E' , we write $V(E')$ for the vertices with at least one endpoint in an edge of E' . For a vertex set $V' \subseteq V(G)$, we let $G[V'] := (V', \{e \in E(G) \mid V(\{e\}) \subseteq V'\})$ denote the subgraph of G induced by V' . We generalize to edge sets $E' \subseteq E(G)$ by $G[E'] := G[V(E')]$.

Phylogenetic Networks and Phylogenetic Diversity. For a given set X , a *phylogenetic X -network* $\mathcal{N} = (V, E, \omega)$ is a directed, acyclic graph (V, E) with *edge-weight* $\omega : E \rightarrow \mathbb{N}_{>0}$, in which there is a single vertex with an in-degree of 0, the *root* ρ , and X is the set of vertices with an in-degree of 1 and an out-degree of 0, called the *leaves*. All other vertices split up into *tree vertices*, which have an in-degree of 1 and an out-degree of at least 2, and *reticulations*, which have an out-degree of 1 and an in-degree of at least 2. Edges incoming at reticulations (tree vertices) are *reticulation edges* (*tree edges*). The set of reticulations, tree vertices, reticulation and tree edges are denoted with $V_R(\mathcal{N})$, $V_T(\mathcal{N})$, $E_R(\mathcal{N})$, and $E_T(\mathcal{N})$, respectively. A *phylogenetic X -tree* $\mathcal{T} = (V, E, \omega)$ is a phylogenetic X -network without reticulations. The set X is a set of *taxa* (species). We interchangeably use the words taxon and leaf. In biological applications, the set X is a set of taxa, and the other vertices of $\mathcal{N}$ correspond to biological ancestors of these taxa. An edge $e = uv$ represents direct biological inheritance from u to v . The weight $\omega(e)$ describes the phylogenetic distance between the endpoints of e . As these endpoints correspond to distinct, (possibly extinct) species, we may assume this distance is greater than zero. Reticulations correspond to species that have direct inheritance from multiple ancestors, such as hybrid species.

For a vertex $v \in V$, the *offspring* $\text{off}(v)$ of v is the set of leaves $x \in X$ for which there is a path from v to x . Given a set of taxa $A \subseteq X$, let $E(A)$ denote the set of edges $uv \in E$ with $\text{off}(v) \cap A \neq \emptyset$. The *phylogenetic diversity* $PD_{\mathcal{N}}(A)$ of A is defined by

$$PD_{\mathcal{N}}(A) := \sum_{e \in E(A)} \omega(e). \quad (1)$$

In other words, the phylogenetic diversity $PD_{\mathcal{N}}(A)$ of a set A of taxa is the sum of the weights of edges that are on a path to a taxon in A .

For phylogenetic trees, this measure is well established [8]. For phylogenetic networks the search for the most relevant measure is still ongoing, but so far the above definition, also called *All-Paths-PD*, is the measure that is “the simplest” [4, 16, 30, 31, 33] and is the only measure of phylogenetic diversity considered in this paper.

Food Webs. A *food web* $\mathcal{F} = (X, E, \gamma)$ on X for a set of taxa X , is a directed acyclic graph (X, E) with an edge weight function $\gamma : E \rightarrow [0, 1]$.

For an edge $xy \in E$, we say that x is *prey* of y and y is a *predator* of x . Thus, edges in $\mathcal{F}$ are directed from prey to predator. The set of prey and predators of x is denoted with $\text{prey}(x)$ and $\text{pred}(x)$, respectively. The set $\text{prey}^{(E)}(x)$ is the set of edges incoming at v and $\text{pred}^{(E)}(x)$ is the set of edges outgoing of v in $\mathcal{F}$. Taxa without prey are *sources*. For the

problems considered in this paper, instances in which the food web has several sources can in quadratic time be transformed into one where there is only one source [20, Observation 2.3]. Therefore, throughout the paper, we assume that $\mathcal{F}$ only has a single source $s_{\mathcal{F}}$.

For a given food web $\mathcal{F}$, a set of taxa $A \subseteq X$ is γ -viable if $\gamma_{\Sigma}(\{ux \mid u \in \text{prey}(x) \cap A\}) \geq 1$ for each non-source $x \in A$. That is, the total weight of edges incoming from taxa in A is at least 1 [27]. Given also a set $E' \subseteq E(\mathcal{F})$ of edges, a set of taxa $A \subseteq X$ is E' -part- γ -viable if $\gamma_{\Sigma}(\{ux \in \text{prey}^{(E)}(x) \mid u \in A \text{ or } ux \in E'\}) \geq 1$ for each non-source $x \in A$. That is, the total weight of edges incoming from taxa in A together with incoming edges in E' is at least 1.

We consider two important special cases of viability. If $\gamma(e) = 1$ for all $e \in E$, then we say a γ -viable set $A \subseteq X$ is ε -viable. That is equivalent to saying that A is ε -viable, if each non-source in A can find prey in A . If $\gamma(ux) = 1/|\text{prey}(x)|$ for every edge in $\text{prey}^{(E)}(x)$ for each $x \in X$, then we say a γ -viable set $A \subseteq X$ is 1-viable. That is, A is 1-viable, if all prey of each taxon in A are in A .

Problem Definitions. In the classical MAXIMIZE PHYLOGENETIC DIVERSITY (MAX-PD) problem, we are given a phylogenetic tree $\mathcal{T}$, and integers k , and D , and it is asked whether a set of k taxa with a phylogenetic diversity of at least D exists [8].

The problems MAP-PD [4], ε -PDD [22], and 1-PDD [27] are generalizations of MAX-PD in the following sense. In MAP-PD, we are given a phylogenetic network instead of a tree and the question stays the same—just with the more general phylogenetic diversity measure. In ε -PDD, 1-PDD and WEIGHTED-PDD, we are, in addition to the input of MAX-PD, given a food web and the solution set is required to be ε -viable, 1-viable or γ -viable, respectively.

In this paper, we consider the following generalizations of these problems.

MAP-WEIGHTED-PDD

Input: A phylogenetic X -network $\mathcal{N}$, a food web $\mathcal{F}$ on X , and integers $k, D \in \mathbb{N}$.

Question: Is there an γ -viable set $S \subseteq X$ such that $|S| \leq k$, and $PD_{\mathcal{N}}(S) \geq D$?

We call the set S a *solution* of the instance. The problems MAP- ε -PDD and MAP-1-PDD are defined analogously, where the set S is required to be ε -viable and 1-viable, respectively.

We note that in ε -PDD, 1-PDD, and WEIGHTED-PDD, as originally defined, the phylogenetic network is required to be a tree.

Throughout the paper, we adopt the convention that n is the number of taxa $|X| = |V(\mathcal{F})|$ and that m is the number of edges of the food web $|E(\mathcal{F})|$.

Scanwidth. For a directed, acyclic graph $G = (V, E)$, a rooted, directed tree $T = (V, E')$ is a *tree extension* of G if for each edge $uv \in E$ there is a path from u to v in T . We say that an edge $uv \in E$ *passes over* an edge $e \in E'$ if the (only) path from u to v in T contains e . For an edge $uv \in E'$, the set of edges that pass over uv is $GW(v)$ and $T_{\mathcal{F}}^{(v)}$ is the set of vertices that can be reached from v in $T_{\mathcal{F}}$. The *scanwidth* of a tree extension T of G is the maximum number of edges of G that pass over an edge of T . The *scanwidth* of G is the minimal scanwidth of any tree extension of G [2]. Computing the tree extension and scanwidth of a directed, acyclic graph is NP-hard [2] and, when considered for phylogenetic networks, FPT when parameterized by the level of the network [11, 13]. In this paper, we will use the parameter *scanwidth* $sw_{\mathcal{F}}$ of the food web, and assume we are given a tree extension of $\mathcal{F}$ of minimum scanwidth.

Other Main Parameters. Here, we define the main parameters which are used in this paper.

For an instance $\mathcal{I} = (\mathcal{N}, \mathcal{F}, k, D)$ of MAP-WEIGHTED-PDD, we define $\bar{k} := |X| - k$ and $\bar{D} := PD_{\mathcal{N}}(X) - D = \sum_{e \in E} \omega(e) - D$. Observe that if a set S of k taxa with diversity

D is preserved, then $\bar{k}$ taxa are not in S and a diversity of $\bar{D}$ is lost. We therefore speak of $\bar{D}$ as the *acceptable loss of diversity*.

We define δ to be the *maximum in-degree of a reticulation* of $\mathcal{N}$. We define h_R and h_T to be the maximum number of reticulations and tree vertices, respectively, which are in a path in $\mathcal{N}$. Further, we define $h := h_R + h_T$ to be the *height of the network*.

Color Coding. In this paper, we use color coding methods. For an in-depth treatment of color coding, we refer the reader to [6, Chapter 5] and [1].

► **Definition 2.1.** *For integers n and k , an (n, k) -perfect hash family $\mathcal{H}$ is a family of mappings $f : [n] \rightarrow [k]$ such that for every subset Z of $[n]$ of size at most k , there is an $f \in \mathcal{H}$ which is injective when restricted to Z .*

Each mapping in an (n, k) -perfect hash family can be seen as coloring of a set of n items into k colors. If we are interested in finding a set of at most k items satisfying some property, then we may assume that each element in the set will have a different color under some coloring. This assumption can make solving a problem easier, and we will use it in the algorithms developed in Sections 3 and 4.

► **Proposition 2.2** ([6, 23]). *For any integers $n, k \geq 1$, a (n, k) -perfect hash family of size $e^k k^{\mathcal{O}(\log k)} \cdot \log n$ can be constructed in time $e^k k^{\mathcal{O}(\log k)} \cdot n \log n$.*

2.2 Related work

ε -PDD, defined in [22], was conjectured to be NP-hard [28]. A formal prove appeared in [9].

► **Theorem 2.3** ([9, Theorem 5.1]). *ε -PDD is NP-hard even if the phylogenetic tree has a height of 2 and the food web is an out-tree.*

ε -PDD has been analyzed within parameterized complexity [20] and it has been shown that ε -PDD is FPT when parameterized with D , but W[1]-hard with respect to $\bar{D}$ [20]. In [27], further definitions for viability were given, among them γ -viable and 1-viable. It has been shown that 1-PDD is W[1]-hard with respect to k , D , $\bar{k}$, and $\bar{D}$ [12]. Further, ε -PDD [20] and WEIGHTED-PDD [27] were analyzed with respect to parameters categorizing the structure of the food web.

- **Theorem 2.4.** (a) *ε -PDD is W[1]-hard when parameterized with $\bar{k}$ or $\bar{D}$, even if the phylogenetic tree is a star [20, Proposition 5.1].*
 (b) *1-PDD is W[1]-hard when parameterized with $\bar{k}$ or $\bar{D}$, even if the phylogenetic tree is a star [12, Theorem 3.3].*
 (c) *1-PDD is W[1]-hard when parameterized with k or D , even if the phylogenetic tree is a star [12, Theorem 3.2].*

In recent years, the question of how to model phylogenetic diversity in networks best has drawn some attention [4, 30, 31, 33]. The measure *All-Paths-PD*, as defined in [4], is hereby the easiest to understand and also computationally slightly less challenging than other definitions. MAP-PD is FPT when parameterized with D [16], and can be solved in $\mathcal{O}^*(2^{\text{ret}_{\mathcal{N}}})$ time for $\text{ret}_{\mathcal{N}}$ being the number of reticulations [16], in $\mathcal{O}^*(2^{\text{sw}_{\mathcal{N}}})$ time for $\text{sw}_{\mathcal{N}}$ the scanwidth [14], and in $\mathcal{O}^*(2^{\text{tw}_{\mathcal{N}}})$ time for $\text{tw}_{\mathcal{N}}$ the treewidth of the network [16].

► **Theorem 2.5** ([4, 16]). *MAP-PD is W[2]-hard, when parameterized with the solution size k , even if every path from the root to a leaf contains exactly one tree vertex and one reticulation.*

$sw_{\mathcal{F}}$	δ	h	parameter alone		k + parameter		$\overline{D}$ + parameter	
$\times$	$\times$	$\times$	no parameter		W[2]-h	Thm. 2.5,[4, 16]	W[1]-h	Thm. 2.4a&b,[20, 12]
$\times$	$\times$	$\checkmark$	p-NP-h	Thm. 2.3,[9]	W[2]-h	Thm. 2.5,[4, 16]	W[1]-h	Thm. 2.4a&b,[20, 12]
$\times$	$\checkmark$	$\times$	p-NP-h	Thm. 2.3,[9]	W[2]-h	Cor. 2.6	W[1]-h	Thm. 2.4a&b,[20, 12]
$\times$	$\checkmark$	$\checkmark$	p-NP-h	Thm. 2.3,[9]	* see caption		W[1]-h	Thm. 2.4a&b,[20, 12]
$\checkmark$	$\times$	$\times$	p-NP-h	Thm. 2.3,[9]	W[2]-h	Thm. 2.5,[4, 16]	FPT	Cor. 3.2a
$\checkmark$	$\times$	$\checkmark$	p-NP-h	Thm. 2.3,[9]	W[2]-h	Thm. 2.5,[4, 16]	FPT	Cor. 3.2a
$\checkmark$	$\checkmark$	$\times$	p-NP-h	Thm. 2.3,[9]	W[2]-h	Cor. 2.6	FPT	Cor. 3.2a
$\checkmark$	$\checkmark$	$\checkmark$	p-NP-h	Thm. 2.3,[9]	FPT	Cor. 4.3	FPT	Cor. 3.2a

■ **Table 1** This table shows parameterized complexity results. Except for the marked cell, MAP-1-PDD and MAP- ε -PDD have the same complexity result. All FPT results are new. For any set of taxa A , in polynomial time $PD_{\mathcal{N}}(A)$ and whether A is γ -viable can be computed. Consequently, by iterating over all subsets of X , MAP-WEIGHTED-PDD is trivially FPT for $n = k + \bar{k} < k + \overline{D}$. Entry *: MAP-1-PDD is W[1]-hard when parameterized with $k + \delta + h$ or even $D + \delta + h$ (Thm. 2.4c) while we conjecture MAP- ε -PDD to be FPT when parameterized with $k + \delta + h$ (Con. 4.9).

Recently, MAP-PD has been considered in semidirected networks. In semi directed networks, MAP-PD can be solved in $\mathcal{O}^*(2^\ell)$ time, where ℓ is the level of the network [14].

Our Contribution. We analyze MAP- ε -PDD and MAP-1-PDD with respect to the parameters k , $\overline{D}$, $sw_{\mathcal{F}}$, δ , and h and show, for any combination of these parameters, whether MAP-1-PDD is W[1]-hard, or admits an FPT algorithm. For MAP- ε -PDD, we only leave one case as a conjecture and prove all others. All algorithms we prove for the generalization MAP-WEIGHTED-PDD. Parameterized results for MAP- ε -PDD and MAP-1-PDD are displayed in Table 1.

In Section 3, we prove that MAP-WEIGHTED-PDD is FPT with respect to $\overline{D} + sw_{\mathcal{F}}$. In Section 4, we show that MAP-WEIGHTED-PDD is FPT with respect to $k + sw_{\mathcal{F}} + h + \delta$. If any of these parameters is dropped, MAP-1-PDD is W[1]-hard.

The algorithm presented in Section 3 is our primary methodological contribution. In this novel approach, only a single color per edge is used in a color coding algorithm. This can be used to prove that ε -PDD and 1-PDD are FPT with respect to $\bar{k} + sw_{\mathcal{F}}$, see Corollary 3.2b. We expect this to be applicable for similar problems parameterized with $\bar{k}$ as well.

Due to space restrictions, proofs of theorems and lemmas marked with $\star$ are partly or fully deferred to the appendix.

2.3 Preliminary Observations

By Theorem 2.5 MAP-PD is W[2]-hard with respect to k even for a network of a small height [16]. In this hardness reduction, however, the maximal in-degree of reticulations is big. It is an easy observation, that, in polynomial time, one can replace vertices of a large degree with a stack of vertices, where the newly created edges have negligible weight compared to the original edges of the network. This works for reticulations and tree vertices alike. We obtain the following result.

► **Corollary 2.6.** MAP-PD is W[2]-hard with respect to k even if the network is binary.

If the food web is an out-tree, each taxon $x \neq s_{\mathcal{F}}$ has exactly one prey. We conclude that the result of Theorem 2.3 also holds for 1-PDD.

► **Corollary 2.7.** 1-PDD is NP-hard even if the phylogenetic tree has a height of 2 and the food web is an out-tree.

3 Parameter $\overline{D} + \text{sw}_{\mathcal{F}}$

By Theorem 2.4a and b, ε -PDD and 1-PDD are W[1]-hard when parameterized by $\overline{D}$. In this section, we prove that MAP- ε -PDD and MAP-1-PDD are FPT when parameterized by $\overline{D} + \text{sw}_{\mathcal{F}}$. Further, we show that when the phylogenetic network is a tree, even an FPT-running time with respect to the smaller parameter $\overline{k} + \text{sw}_{\mathcal{F}}$ is possible. To prove these results, we present our main methodological contribution. This approach uses *anchors*, as introduced in [17]. Yet, it is new to use this approach either on phylogenetic networks or for the smaller parameter $\overline{k}$.

► **Theorem 3.1** ($\star$). Let a tree extension $T_{\mathcal{F}}$ of the food web with scanwidth $\text{sw}_{\mathcal{F}}$ be given.

- (a) MAP-WEIGHTED-PDD can be solved in $\mathcal{O}(2^{7.530 \cdot \overline{D} + \text{sw}_{\mathcal{F}} + \mathcal{O}(\log^2(\overline{D}))} \cdot n \cdot |E(\mathcal{N})| \cdot \log |E(\mathcal{N})|)$ time.
- (b) WEIGHTED-PDD can be solved in $\mathcal{O}(2^{15.059 \cdot \overline{k} + \text{sw}_{\mathcal{F}} + \mathcal{O}(\log^2(\overline{k}))} \cdot n \cdot |E(\mathcal{N})| \cdot \log |E(\mathcal{N})|)$ time.

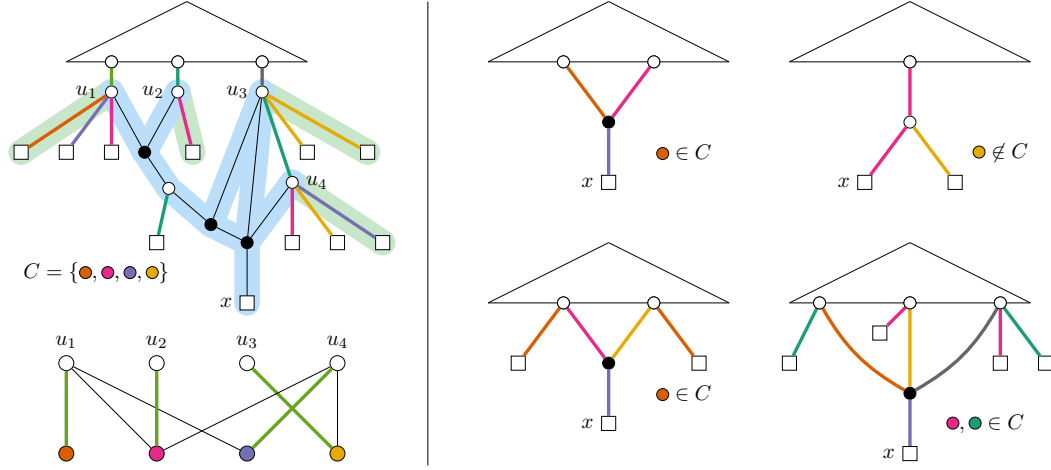
As a consequence of this theorem, we obtain these results.

- **Corollary 3.2.** (a) MAP- ε -PDD and MAP-1-PDD are FPT with respect to $\overline{D} + \text{sw}_{\mathcal{F}}$.
 (b) ε -PDD and 1-PDD are FPT with respect to $\overline{k} + \text{sw}_{\mathcal{F}}$.

For the remainder of this section, fix a tree extension $T_{\mathcal{F}}$ of the food web. Let $x_1, \dots, x_n$ be an ordering of the taxa such that if x_i is a parent of x_j in $T_{\mathcal{F}}$ then $i < j$, and if x_i and x_j share a parent in $T_{\mathcal{F}}$ with $i < j$, then every vertex in the subtree of $T_{\mathcal{F}}$ rooted x_i appears before every vertex in the subtree of $T_{\mathcal{F}}$ rooted x_j in the ordering. Such an ordering can be found by taking a depth-first traversal of $T_{\mathcal{F}}$. For a set of edges F in a directed acyclic graph G , we say an edge $e = uv \in F$ is a *highest edge in F* if no incoming edge of u is in F , and similarly we say $uv \in F$ is a *lowest edge in F* if no outgoing edge of v is in F . For a vertex u with outgoing edges uv and uv' , we say uv' is a *sibling edge of uv* .

The main ideas of our proof is as follows. Our approach uses color coding techniques, wherein we generate a number of colorings on the edges of $\mathcal{N}$, and seek a solution under the assumption that a certain set fulfills that each edge in the set has a different color.

Our dynamic programming algorithm will search for a structure that we define below as *perfect triples*. Roughly speaking, a perfect triple (A, χ_1, χ_2) consists of a set of taxa A such that $X \setminus A$ is a solution for our instance of MAP-WEIGHTED-PDD, and colorings χ_1, χ_2 assigning each leaf in A to a set of colors. Suppose the leaves of A are 'killed' one at a time, in order determined by the depth-first traversal of $T_{\mathcal{F}}$. Then as each leaf is deleted, a certain set of edges will be 'lost' in the sense that they no longer have any offspring that have not been killed. For each $x \in A$, the set $\chi_1(x)$ corresponds to the colors of the edges that are lost when x is deleted. Furthermore, for each highest edge e that is lost when x is killed, there must be a corresponding sibling edge e' that has not yet been lost (otherwise the parent edge(s) of e would also be lost). The colors of these sibling edges are represented by the set $\chi_2(x)$. Note that some of these sibling edges may themselves be lost when a later leaf from A is killed. We may assume, via standard color-coding techniques, that all the lost edges and sibling edges together are multicolored.



■ **Figure 1** Top Left: An example network. The set $F_{x,C}$ is highlighted in $\bullet$ and one possible set $F'_{x,C}$ in $\circ$. For the sake of readability, colors of edges in $F_{x,C}$ are omitted. Bottom Left: The bipartite graph which is constructed in Lemma 3.5. A perfect matching is highlighted. Right: Four examples of networks, where $F_{x,C}$ is not a respecting set of edges.

Our dynamic programming algorithm will keep track of the existence of perfect triples satisfying certain properties. In particular, we store the minimum possible total weight of the set of lost edges for a perfect triple.

We now give some formal definitions.

► **Definition 3.3.** Fix an integer N and a mapping $c : E_T(\mathcal{N}) \rightarrow [N]$. For a taxon $x \in X$ and a set of colors $C \subseteq [N]$, a set of edges $F \subseteq E_T(\mathcal{N})$ is (x, C) -respecting, if

- F contains the edge incoming at x ,
- $c(e) \notin C$ for each $e \in F$,
- for each $uv \in F$ there is a directed path from v to x within $\mathcal{N}[F]$, and
- there is a set F' of edges, which we call anchors of F , such that
 - $c(e) \in C$ for each edge $e \in F'$,
 - $c(e_1) \neq c(e_2)$ for each pair $e_1, e_2 \in F \cup F'$, and
 - for each edge $uv \in F$, exactly one of the following is true
 - (1) F contains all edges incoming at u and $c(e) \notin C$ for each edge e outgoing of u .
 - (2) F does not contain any edge incoming at u and there is an edge e outgoing of u with $c(e) \in C$ and $e \in F'$.

When (x, C) is clear from the context we say an (x, C) -respecting set is simply respecting.

In Figure 1 an example of a respecting set of edges is given and some example networks for which x under a certain coloring has no respecting set of edges.

We continue with proving some essential properties about respecting sets.

► **Lemma 3.4.** For each taxon $x \in X$ and each set $C \subseteq [N]$ of colors, at most one set of edges is (x, C) -respecting.

Proof. Towards a contradiction, assume that F_1 and F_2 are (x, C) -respecting and that there is $u'v' \in F_1$ such that $u'v' \notin F_2$. Fix a set F'_1 of anchors of F_1 , and a set F'_2 of anchors of F_2 . Choose any path from v' to x in $\mathcal{N}[F_1]$ and let uv be the first edge on this path that occurs in F_2 , also. Such an edge exists as the edge incoming at x is in any respecting set of edges.

As F_2 does not contain all edges incoming at u , there is an edge $uw \in F'_2$ with $c(uw) \in C$. But then F_1 is not respecting, as F_1 contains an edge incoming at u . ◀

As a consequence of [Lemma 3.4](#), we write $F_{x,C}$ for the unique set of (x, C) -respecting edges, if existent. If the set $F_{x,C}$ exists, we write $c(F_{x,C})$ for the colors on edges in $F_{x,C}$. By definition, $c(F_{x,C})$ and C are disjoint. If the set $F_{x,C}$ does not exist, we define $\omega(F_{x,C}) = \infty$. We note that the set of anchors is not necessarily unique.

► **Lemma 3.5.** *For a taxon $x \in X$ and a set of colors $C \subseteq [N]$, in $\mathcal{O}(|E(\mathcal{N})| \cdot \sqrt{N})$ time, we can compute $F_{x,C}$, or conclude that it does not exist.*

Proof. We construct a unique set of edges F such that either F is respecting for x and C , or such a set does not exist, as follows. Initially let F contain the unique incoming edge e at x . Now, for each edge uv in F in turn, if u has no outgoing edge uv' with $c(uv') \in C$, then add all incoming edges of u to F . Repeat this process exhaustively to complete the construction of F . If $c(e) \notin C$ for some $e \in F$ or if two edges have the same color, no respecting set exists. By using appropriate data structures, this can be implemented in $\mathcal{O}(|E(\mathcal{N})|)$ time.

It remains to decide whether there exists a valid set of anchors F' for F . In particular, for each highest edge $uv \in F$, we need to choose one outgoing edge uv' with $c(uv') \in C$ to add to F' such that $c(e_1) \neq c(e_2)$ for each pair $e_1, e_2 \in F'$. This can be done by reducing to an instance of PERFECT MATCHING, as follows. Construct a bipartite graph G with vertex set $F_h \cup C$, where F_h corresponds to the set of highest edges in F . For each $e \in F_h$ and $c \in C$, add an edge $\{e, c\}$ to G if e has a sibling edge e' with $c(e') = c$. Now, F is an (x, C) -respecting set if and only if G has a perfect matching covering F_h . PERFECT MATCHING can be solved in $\mathcal{O}(|E(G)| \cdot \sqrt{|V(G)|})$ time with the famous Hopcroft-Karp Algorithm [18]. For each edge in $\mathcal{N}$, there is at most one edge in G . Thus, the overall running time is $\mathcal{O}(|E(\mathcal{N})| \cdot \sqrt{N})$. ◀

With respecting sets defined, we now formally define perfect triples. A triple (A, χ_1, χ_2) , consisting of a set of taxa $A \subseteq X$, and mappings $\chi_1, \chi_2 : A \rightarrow 2^{[N]}$, is *perfect*, if

- the sets $\chi_i(x)$ and $\chi_i(y)$ are pairwise disjoint for $x, y \in A$ and $i \in \{1, 2\}$,
- the sets $\chi_1(x_i)$ and $\chi_2(x_j)$ are pairwise disjoint for $x_i, x_j \in A$ and $i \leq j$, and
- for each $x \in A$ there is a set of respecting edges $F_{x, \chi_2(x)} \subseteq E(\mathcal{N})$ with $\chi_1(x) = c(F_{x, \chi_2(x)})$.

To give some intuition behind the notion of a perfect triple, we observe that the existence of a perfect triple (A, χ_1, χ_2) provides a lower bound on the phylogenetic diversity of $(X \setminus A)$. The key idea is that as we remove the elements of A from X , one at a time, the set of edges lost with the removal of each $x \in A$ is a subset of the edges in $F_{x, \chi_2(x)}$.

► **Lemma 3.6** (★). *$PD_{\mathcal{N}}(X \setminus A) \geq \omega(E(\mathcal{N})) - \sum_{x \in A} \omega(F_{x, \chi_2(x)})$ for every perfect triple (A, χ_1, χ_2) .*

Having these definitions, we now define a colored problem, which we use as an auxiliary problem for solving [Theorem 3.1](#). In EX- N -COLORED-MAP-W-PDD, besides the usual input of MAP-WEIGHTED-PDD $(\mathcal{N}, \mathcal{F}, k, D)$ —where γ is a weighting of $\mathcal{F}$ —we are given a mapping $c : E(\mathcal{N}) \rightarrow [N]$ of a color per edge. We ask whether there exists a set of taxa $A \subseteq X$ of size at least $\bar{k}$ and mappings $\chi_1, \chi_2 : A \rightarrow 2^{[N]}$ such that $X \setminus A$ is γ -viable, the triple (A, χ_1, χ_2) is perfect, and $\sum_{x \in A} \omega(F_{x, \chi_2(x)}) \leq \bar{D}$.

Note that the existence of a perfect triple (A, χ_1, χ_2) does not imply that $X \setminus A$ is γ -viable.

► **Lemma 3.7** (★). *Given a tree-extension $T_{\mathcal{F}}$ of the food web, we can solve instances of EX- N -COLORED-MAP-W-PDD in $\mathcal{O}(5^N \cdot 2^{\text{sw}_{\mathcal{F}}} \cdot n \cdot (N \cdot \bar{k} + |E(\mathcal{N})|))$ time.*

XX:10 Diversity of Networks with Ecological Dependencies

The intuition behind **Lemma 3.7** is as follows: We consider the tree extension of the food web with a dynamic program bottom up. At each vertex v , we determine whether there exists a perfect triple (A, χ_1, χ_2) satisfying certain conditions, where A is a subset of $T_{\mathcal{F}}^{(v)}$, the set of vertices descended from v in $T_{\mathcal{F}}$. We want that $X \setminus A$ is γ -viable; to help determine this we keep track of a subset of edges $\Phi \subseteq GW(v)$, and require that $T_{\mathcal{F}}^{(v)} \setminus A$ is Φ -part- γ -viable.

It remains to show how to reduce instances of MAP-WEIGHTED-PDD to instances of EX- N -COLORED-MAP-W-PDD. This is done using standard color-coding techniques.

Proof of Theorem 3.1. Reduction. Let $\mathcal{I} = (\mathcal{N}, \mathcal{F}, k, D)$ be an instance of MAP-WEIGHTED-PDD. If $\mathcal{N}$ is a tree, then set N to $4\bar{k} - 2$ and otherwise to $2\bar{D}$.

Arbitrarily order the edges $e_1, \dots, e_q$ of $\mathcal{N}$. We may assume $q > N$, as otherwise, we can consider a single instance of EX- q -COLORED-MAP-W-PDD. Let $\mathcal{H}$ be a (q, N) -perfect hash family. For every $f \in \mathcal{H}$ we define a coloring c_f by $c_f(e_j) = f(j)$ for $j \in [q]$ and let $\mathcal{I}_f = (\mathcal{N}, \mathcal{F}, k, D, c_f)$ be the corresponding instance of EX- N -COLORED-MAP-W-PDD. Solve every instance $\mathcal{I}_f$, and return **yes** if and only if $\mathcal{I}_f$ is a **yes**-instance for some $f \in \mathcal{H}$.

The proof of the correctness and running time is deferred to the appendix. ◀

4 Parameter $k + \text{sw}_{\mathcal{F}} + \delta + h$

By **Theorem 2.5**, MAP- ε -PDD and MAP-1-PDD are $W[2]$ -hard with respect to $k + h$, even if the food web does not contain edges. In the following, we therefore add the maximum in-degree of a reticulation δ as a parameter and show that even the more general problem MAP-WEIGHTED-PDD is FPT with respect to $k + \text{sw}_{\mathcal{F}} + h + \delta$.

To do this, we consider a parameter H generalizing the height of a tree. H is the maximum number of tree edges that, in $\mathcal{N}$, are on a path from the root to any taxon. We observe that in a phylogenetic tree, H is the height of the tree minus one. Next, we prove bounds on the value of H in a phylogenetic network.

► **Lemma 4.1** ($\star$). $h_t \leq H$ and $H \leq \delta^{h_r} \cdot h_t \leq \delta^h$.

In this section we prove the following.

► **Theorem 4.2.** *Given a tree extension $T_{\mathcal{F}}$ of the food web with scanwidth $\text{sw}_{\mathcal{F}}$, MAP-WEIGHTED-PDD can be solved in $\mathcal{O}(2^{2.443 \cdot kH + \text{sw}_{\mathcal{F}} + \mathcal{O}(\log^2(kH))} \cdot \text{sw}_{\mathcal{F}} \cdot n \cdot |E(\mathcal{N})|^2 \cdot \log |E_T(\mathcal{N})|)$ time.*

As MAP- ε -PDD and MAP-1-PDD are special cases of MAP-WEIGHTED-PDD, this result transfers to them.

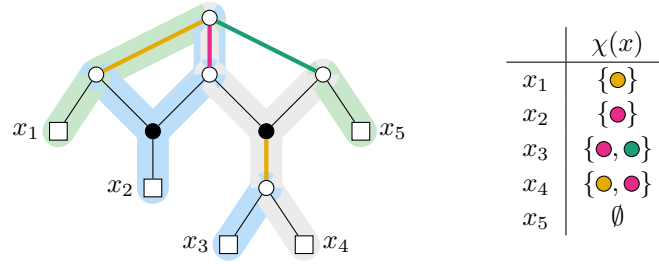
► **Corollary 4.3.** *MAP- ε -PDD and MAP-1-PDD are FPT with respect to $k + \text{sw}_{\mathcal{F}} + h + \delta$.*

First we define some objects necessary in this chapter. For a set A of taxa and a mapping $\chi : X \rightarrow 2^{[k \cdot H]}$, we say that tuple (A, χ) is *colorful* if $\chi(x) \cap \chi(y) \neq \emptyset$ for $x, y \in A$ implies $x = y$.

► **Definition 4.4.** *Fix a mapping $c : E_T(\mathcal{N}) \rightarrow [k \cdot H]$. For a taxon $x \in X$ and a mapping $\chi : X \rightarrow 2^{[k \cdot H]}$, a set of edges $F \subseteq E(\mathcal{N})$ is (x, χ) -suitable, if*

- $c(e) \in \chi(x)$ for each edge $e \in F \cap E_T(\mathcal{N})$,
- $c(e_1) \neq c(e_2)$ for each pair $e_1, e_2 \in F \cap E_T(\mathcal{N})$, and
- for each edge $uv \in F$, there is a path from v to x in $\mathcal{N}[F]$.

When (x, χ) is clear from the context we say an (x, χ) -respecting set is respecting.



■ **Figure 2** A hypothetical network on 3 colors $\{\text{yellow}, \text{pink}, \text{green}\}$. For each taxon the (x, χ) -suitable set $F_{x, \chi}$ is indicated in $\circ$, $\odot$, or $\bullet$. In this network, H takes the value of 3—there are three tree edges on paths to x_3 or x_4 .

Figure 2 shows an example. Note that a respecting set may contain reticulation edges.

To prove Theorem 4.2, we first show how to solve a colored variant of MAP-WEIGHTED-PDD. In kH -COLORED-MAP-W-PDD, besides the usual input $(\mathcal{N}, \mathcal{F}, k, D)$, we are given a mapping $c : E_T(\mathcal{N}) \rightarrow [k \cdot H]$ of a color per tree edge. We ask whether a γ -viable set of taxa $S \subseteq X$ of size at most k and a mapping $\chi : S \rightarrow 2^{[k \cdot H]}$ exist such that (S, χ) is colorful and for each $x \in S$ there is a set of suitable edges $F_{x, \chi}$ and $\sum_{x \in S} \omega(F_{x, \chi}) \geq D$.

► **Lemma 4.5.** *Given an instance $\mathcal{I}$ of kH -COLORED-MAP-W-PDD, a leaf x , and a mapping χ , a suitable edge set F with maximal value $\omega(F)$ can be computed in $\mathcal{O}(2^H \cdot |E(\mathcal{N})| \cdot (kH + |E(\mathcal{N})|))$ time.*

Because of this lemma, in the rest of the section, we simply will write $\omega(F_{x, \chi})$ when referring to the weight of a suitable edge set for x , which has maximal weight.

Proof. *Algorithm.* Let E_x be the set of tree edges uv with $x \in \text{off}(v)$. For a taxon x and a mapping χ , iterate over all subsets F of E_x . If $uv \in F$ and u is a reticulation, then add all incoming edges of u to F , for each $uv \in F$. Add the edge incoming at x to F . Return the biggest value of a so computed F which is suitable.

Correctness. If the algorithm returns value d , then there is a suitable set F with $d = \omega(F)$.

Let F be a suitable set. The set $F_x := F \cap E_x$ appears in the iteration. The edge incoming at x is in F , as it is suitable. Reticulation edges have no color. Thus, adding reticulation edges leading to F keeps the set suitable. Consequently, a set F' with $F' \supseteq F$ has been considered by the algorithm.

Running time. The set E_x has size at most H , by definition. All sets F are computed in $\mathcal{O}(2^H \cdot |E(\mathcal{N})|)$ time. Checking whether a set F is colorful can be done in $\mathcal{O}(kH + |E(\mathcal{N})|)$ time. ◀

► **Lemma 4.6.** *If (S, χ) is colorful, then any (x, χ) -suitable set $F_{x, \chi}$ and any (y, χ) -suitable set $F_{y, \chi}$ are pairwise disjoint for $x, y \in S$, $x \neq y$.*

Proof. Fix vertices $x, y \in S$ and an edge $uv \in F_{x, \chi}$. If uv is a tree edge, then $c(uv) \in \chi(x)$ and because $\chi(x)$ and $\chi(y)$ are disjoint, we conclude $uv \notin F_{y, \chi}$.

Now let uv be a reticulation edge. Let w be the unique *first* tree vertex on a path from v to x —or any other leaf. Then $c(e_w) \in \chi(x)$ for the edge e_w incoming at w . We conclude that $e_w \notin F_{y, \chi}$ and there is no path from v to y in $\mathcal{N}[F_{y, \chi}]$. Consequently $uv \notin F_{y, \chi}$.

If $v = x$, or the only taxon reachable from v is x , then $uv \notin F_{y, \chi}$. ◀

We write $F_{S,\chi}$ for the union $\bigcup_{x \in S} F_{x,\chi}$ for a set S . For a colorful set S we conclude with [Lemma 4.6](#) that $PD_{\mathcal{N}}(S) \geq \omega(F_{S,\chi}) \geq D$.

We are now ready to prove how to solve instances of kH -COLORED-MAP-W-PDD.

► **Lemma 4.7** (*). *Given an instance $\mathcal{I}$ of kH -COLORED-MAP-W-PDD and a tree-extension $T_{\mathcal{F}}$ of the food web, we can solve $\mathcal{I}$ in $\mathcal{O}(2^{k \cdot H + \text{sw}_{\mathcal{F}}} \cdot k^4 \cdot H^2 \cdot \text{sw}_{\mathcal{F}} \cdot n \cdot |E(\mathcal{N})|^2)$ time.*

The intuition behind [Lemma 4.7](#) is as follows: We consider the tree extension of the food web bottom up in a dynamic programming algorithm. We track the existence of colorful tuples whose taxa are all below a given vertex of the tree extension. We index colorful tuples by the sets of colors used, as well as by the set of edges Φ for which the set of taxa is Φ -part- γ -viable, and we store the total weight of the suitable edge sets. We use the fact that the sets $\chi(x)$ and $\chi(y)$ have to be pairwise disjoint. Therefore, if a taxon x is saved and a set of colors has been assigned, these colors can be removed. We define $\chi(x)$ when selecting x .

It remains to show how to reduce instances of MAP-WEIGHTED-PDD to an instances of kH -COLORED-MAP-W-PDD. For this, we use perfect hash families. ([Definition 2.1](#))

Proof of Theorem 4.2. *Reduction.* Let $\mathcal{I} = (\mathcal{N}, \mathcal{F}, k, D)$ be an instance of MAP-WEIGHTED-PDD.

Arbitrarily order the tree edges $e_1, \dots, e_q$ of $\mathcal{N}$. We may assume $q > k \cdot H$. Let $\mathcal{H}$ be a $(q, k \cdot H)$ -perfect hash family. For every $f \in \mathcal{H}$ we define a coloring c_f by $c_f(e_j) = f(j)$ for $j \in [q]$ and let $\mathcal{I}_f = (\mathcal{N}, \mathcal{F}, k, D, c_f)$ be the corresponding instance of kH -COLORED-MAP-W-PDD. Now, solve instance $\mathcal{I}_f$, and return **yes** if and only if $\mathcal{I}_f$ is a **yes**-instance for some $f \in \mathcal{H}$.

Correctness. For any set $E' \subseteq E_T(\mathcal{N})$ of edges with a size of at most $k \cdot H$, there is a function $f \in \mathcal{H}$ such that $c_f(E')$ contains each color at most once.

Now, let $\mathcal{I}$ be a **yes**-instance of MAP-WEIGHTED-PDD with solution $S = \{x_1, \dots, x_k\} \subseteq X$. Further, let $E_T^{(1)}$ be the set of tree edges on paths from the root to x_1 in $\mathcal{N}$, and for $i \in [k-1]$ let $E_T^{(i+1)}$ be the set of tree edges on paths to x_{i+1} which are not in $E_T^{(i)}$. We define $E_T(S)$ as the union of these sets. By definition of H , each set $E_T^{(i)}$ has a size of at most H . By definition of perfect hash families, there is some $f \in \mathcal{H}$, such that c_f is injective on $E_T(S)$. Taking $\chi(x_i) = c_f(E_T^{(i)})$, we conclude that (S, χ) is colorful and $\omega(F_{S,\chi}) = PD_{\mathcal{N}}(S) \geq D$. Thus, (S, χ) is a solution of the **yes**-instance $\mathcal{I}_f$ of kH -COLORED-MAP-W-PDD.

Conversely, whenever (S, χ) is a solution for instance $\mathcal{I}_f$, then S is also a solution for $\mathcal{I}$.

Running Time. The construction of $\mathcal{H}$ takes $e^{kH} (kH)^{\mathcal{O}(\log kH)} \cdot q \log q$ time, and for each $f \in \mathcal{H}$ the construction of instance $\mathcal{I}_f$ of kH -COLORED-MAP-W-PDD takes time linear in $|\mathcal{I}|$. By [Lemma 4.7](#), instances of kH -COLORED-MAP-W-PDD can be solved in $\mathcal{O}(2^{k \cdot H + \text{sw}_{\mathcal{F}}} \cdot k^4 \cdot H^2 \cdot \text{sw}_{\mathcal{F}} \cdot n \cdot |E(\mathcal{N})|^2)$ time, and the number of instances is $|\mathcal{H}| = e^{kH} (kH)^{\mathcal{O}(\log kH)} \cdot \log q$.

Thus, the total running time is $\mathcal{O}(e^{kH} (kH)^{\mathcal{O}(\log kH)} \log q \cdot (q + 2^{k \cdot H + \text{sw}_{\mathcal{F}}} \cdot k^4 \cdot H^2 \cdot \text{sw}_{\mathcal{F}} \cdot n \cdot |E(\mathcal{N})|^2))$. This simplifies to $\mathcal{O}((2e)^{kH} \cdot 2^{\text{sw}_{\mathcal{F}} + \mathcal{O}(\log^2(kH))} \cdot \text{sw}_{\mathcal{F}} \cdot n \cdot |E(\mathcal{N})|^2 \cdot \log |E_T(\mathcal{N})|)$. ◀

By [Theorem 2.4c](#), MAP-WEIGHTED-PDD is not FPT with respect to $k + h$. But this differs when we add $\text{sw}_{\mathcal{F}}$ as a parameter. In a phylogenetic tree, the value of H is h .

► **Corollary 4.8.** *WEIGHTED-PDD is FPT with respect to $k + h + \text{sw}_{\mathcal{F}}$ and can be solved in $\mathcal{O}(2^{2.443 \cdot kh + \text{sw}_{\mathcal{F}} + \mathcal{O}(\log^2(kh))} \cdot \text{sw}_{\mathcal{F}} \cdot n \cdot |E(\mathcal{N})|^2 \cdot \log |E_T(\mathcal{N})|)$ time.*

4.1 Map- ε -PDD with respect to $k + \delta + h$

By [Theorem 4.2](#), MAP-WEIGHTED-PDD is FPT with respect to $k + \text{sw}_{\mathcal{F}} + \delta + h$ and therefore also MAP- ε -PDD and MAP-1-PDD. If any of the four parameters is dropped then MAP-1-PDD is W[1]-hard, as pointed out in [Table 2](#).

■ **Table 2** MAP-1-PDD is W[1]-hard if one of the parameters of $k + \text{sw}_{\mathcal{F}} + \delta + h$ is dropped.

$\text{sw}_{\mathcal{F}} + \delta + h$	$k + \text{sw}_{\mathcal{F}} + \delta$	$k + \text{sw}_{\mathcal{F}} + h$	$k + \delta + h$
Theorem 2.3,[9]	Corollary 2.6	Theorem 2.5,[4, 16]	Theorem 2.4c,[27]

While the first three hardness results hold for both problems, the last explicitly only shows hardness for MAP-1-PDD. We expect this hardness not to hold for MAP- ε -PDD, but believe that with an approach similar to the one presented in [20] to show that ε -PDD is FPT with respect to $k + h$, one can also show that MAP- ε -PDD is FPT when parameterized with $k + \delta + h$. Unfortunately, the proof given in [20] has an incorrect lemma. An example of the error is explain in greater detail in Appendix Section B. We still believe both of the claims to hold. Henceforth, we leave the following as a conjecture.

► **Conjecture 4.9.** MAP- ε -PDD is FPT when parameterized with $k + \delta + h$.

5 Discussion

In this paper, we made the approach to combine the natural problems of maximizing phylogenetic diversity in a network with viability constraints given through a food web. We defined the problem MAP-WEIGHTED-PDD and its special cases MAP- ε -PDD and MAP-1-PDD. We provided several FPT algorithms for these problems and even presented a complete complexity dichotomy by showing for which combination of parameters of k , $\bar{D}$, $\text{sw}_{\mathcal{F}}$, δ , and h , the three problems are in FPT and for which they are W[1]-hard.

Still, several questions remain open. The most obvious is whether Conjecture 4.9 holds.

Further, in Section 3, we showed that MAP-WEIGHTED-PDD is FPT with respect to $\bar{D} + \text{sw}_{\mathcal{F}}$. We showed that this approach is sufficient to show that WEIGHTED-PDD is FPT with respect to the smaller parameter $\bar{k} + \text{sw}_{\mathcal{F}}$. Yet, it is unclear whether MAP-WEIGHTED-PDD admits an FPT algorithm for this parameter.

Another major open question, already pointed out in [20], is whether ε -PDD and 1-PDD are FPT when parameterized with k . This question even remains open if the food web is restricted to an out-tree—or if any vertex in $\mathcal{F}$ has a degree of at most 1.

In this paper, we observed the *All-Paths-PD* measure in phylogenetic networks as a measure of phylogenetic diversity. This measure is computationally the least challenging, but probably not the best measure in capturing phylogenetic diversity. In recent years, several other measures have been proposed [4, 30, 31, 33]. We wonder if there are also efficient algorithms for these problems that respect the viability of the selected set of taxa.

References

- 1 N. Alon, R. Yuster, and U. Zwick. Color-Coding. *Journal of the Association for Computing Machinery (JACM)*, 42(4):844–856, 1995.
- 2 V. Berry, C. Scornavacca, and M. Weller. Scanning phylogenetic networks is NP-hard. In *Proceedings of the 46th International Conference on Current Trends in Theory and Practice of Informatics (SOFSEM 2020)*, pages 519–530. Springer, 2020.
- 3 M. Bordewich and C. Semple. Nature Reserve Selection Problem: A Tight Approximation Algorithm. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 5(2):275–280, 2008.
- 4 M. Bordewich, C. Semple, and K. Wicke. On the complexity of optimising variants of phylogenetic diversity on phylogenetic networks. *Theoretical Computer Science*, 917:66–80, 2022.

- 5 G. Ceballos and P. R. Ehrlich. Mutilation of the tree of life via mass extinction of animal genera. *Proceedings of the National Academy of Sciences*, 120(39):e2306987120, 2023.
- 6 M. Cygan, F. V. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh. *Parameterized Algorithms*. Springer, 2015.
- 7 R. G. Downey and M. R. Fellows. *Fundamentals of Parameterized Complexity*. Texts in Computer Science. Springer, 2013.
- 8 D. P. Faith. Conservation evaluation and Phylogenetic Diversity. *Biological Conservation*, 61(1):1–10, 1992.
- 9 B. Faller, C. Semple, and D. Welsh. Optimizing Phylogenetic Diversity with Ecological Constraints. *Annals of Combinatorics*, 15(2):255–266, 2011.
- 10 M. N. Ferreira. Conservation priorities mapping—a first step toward building area-based strategies. *Frontiers in Science*, 2:1440501, 2024.
- 11 N. Holtgreffe. Computing the Scanwidth of Directed Acyclic Graphs, 2023. Master’s thesis.
- 12 N. Holtgreffe, J. Schestag, and N. Zeh. Limits of Kernelization and Parametrization for Phylogenetic Diversity with Dependencies. Manuscript in preparation, 2025.
- 13 N. Holtgreffe, L. van Iersel, and M. Jones. Exact and Heuristic Computation of the Scanwidth of Directed Acyclic Graphs. *arXiv preprint arXiv:2403.12734*, 2024.
- 14 N. Holtgreffe, L. van Iersel, R. Meuwese, Y. Murakami, and J. Schestag. PANDA: Maximizing All-Path Phylogenetic Diversity in Networks. Manuscript in preparation, 2025.
- 15 D. H. Huson and D. Bryant. Application of Phylogenetic Networks in Evolutionary Studies. *Molecular biology and evolution*, 23(2):254–267, 2006.
- 16 M. Jones and J. Schestag. How Can We Maximize Phylogenetic Diversity? Parameterized Approaches for Networks. In *Proceedings of the 18th International Symposium on Parameterized and Exact Computation (IPEC 2023)*. Schloss-Dagstuhl-Leibniz Zentrum für Informatik, 2023.
- 17 M. Jones and J. Schestag. Maximizing Phylogenetic Diversity under Time Pressure: Planning with Extinctions Ahead. *arXiv preprint arXiv:2403.14217*, 2024.
- 18 M. Karpiński and W. Rytter. *Fast Parallel Algorithms for Graph Matching Problems*. Oxford University Press, 1998.
- 19 C. Komusiewicz and J. Schestag. A Multivariate Complexity Analysis of the Generalized Noah’s Ark Problem. In *Proceedings of the 19th Cologne-Twente Workshop on Graphs and Combinatorial Optimization*, pages 109–121. Springer, 2023.
- 20 C. Komusiewicz and J. Schestag. Maximizing Phylogenetic Diversity under Ecological Constraints: A Parameterized Complexity Study. In *Proceedings of the 44th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2024)*. Schloss-Dagstuhl-Leibniz Zentrum für Informatik, 2024.
- 21 B. Q. Minh, S. Klaere, and A. von Haeseler. Phylogenetic Diversity within Seconds. *Systematic Biology*, 55(5):769–773, 10 2006.
- 22 V. Moulton, C. Semple, and M. Steel. Optimizing phylogenetic diversity under constraints. *Journal of Theoretical Biology*, 246(1):186–194, 2007.
- 23 M. Naor, L. J. Schulman, and A. Srinivasan. Splitters and near-optimal Derandomization. *Proceedings of IEEE 36th Annual Foundations of Computer Science*, pages 182–191, 1995.
- 24 F. Pardi and N. Goldman. Species Choice for Comparative Genomics: Being Greedy Works. *PLoS Genetics*, 1(6):e71, 2005.
- 25 F. Pardi and N. Goldman. Resource-Aware Taxon Selection for Maximizing Phylogenetic Diversity. *Systematic Biology*, 56(3):431–444, 2007.
- 26 S. L. Pimm, C. N. Jenkins, R. Abell, T. M. Brooks, J. L. Gittleman, L. N. Joppa, P. H. Raven, C. M. Roberts, and J. O. Sexton. The biodiversity of species and their rates of extinction, distribution, and protection. *Science*, 344(6187):1246752, 2014.
- 27 J. Schestag. Weighted Food Webs Make Computing Phylogenetic Diversity So Much Harder. *arXiv preprint arXiv:2510.05911*, 2025.
- 28 A. Spillner, B. T. Nguyen, and V. Moulton. Computing Phylogenetic Diversity for Split Systems. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 5(2):235–244, 2008.

- 29 M. Steel. Phylogenetic Diversity and the Greedy Algorithm. *Systematic Biology*, 54(4):527–529, 2005.
- 30 L. van Iersel, M. Jones, J. Schestag, C. Scornavacca, and M. Weller. Average-Tree Phylogenetic Diversity of Networks. In *25th International Workshop on Algorithms in Bioinformatics (WABI 2025)*. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2025.
- 31 L. van Iersel, M. Jones, J. Schestag, C. Scornavacca, and M. Weller. Phylogenetic Network Diversity Parameterized by Reticulation Number and Beyond. In *Proceedings of the 23rd RECOMB International Workshop on Comparative Genomics (RECOMB-CG 2025)*, Seoul, Republic of Korea, 2025.
- 32 M. Vellend, W. K. Cornwell, K. Magnuson-Ford, and A. Ø. Mooers. Measuring phylogenetic biodiversity, 2011.
- 33 K. Wicke and M. Fischer. Phylogenetic diversity and biodiversity indices on phylogenetic networks. *Mathematical Biosciences*, 298:80–90, 2018.

A Appendix—Omitted proofs

A.1 Proof of Theorem 3.1

► **Theorem 3.1** ($\star$). *Let a tree extension $T_{\mathcal{F}}$ of the food web with scanwidth $\text{sw}_{\mathcal{F}}$ be given.*

- (a) MAP-WEIGHTED-PDD can be solved in $\mathcal{O}(2^{7.530 \cdot \bar{D} + \text{sw}_{\mathcal{F}} + \mathcal{O}(\log^2(\bar{D}))} \cdot n \cdot |E(\mathcal{N})| \cdot \log |E(\mathcal{N})|)$ time.
- (b) WEIGHTED-PDD can be solved in $\mathcal{O}(2^{15.059 \cdot \bar{k} + \text{sw}_{\mathcal{F}} + \mathcal{O}(\log^2(\bar{k}))} \cdot n \cdot |E(\mathcal{N})| \cdot \log |E(\mathcal{N})|)$ time.

Proof of Theorem 3.1. *Reduction.* Let $\mathcal{I} = (\mathcal{N}, \mathcal{F}, k, D)$ be an instance of MAP-WEIGHTED-PDD. If $\mathcal{N}$ is a tree, then set N to $4\bar{k} - 2$ and otherwise to $2\bar{D}$.

Arbitrarily order the edges $e_1, \dots, e_q$ of $\mathcal{N}$. We may assume $q > N$, as otherwise, we can consider a single instance of EX- q -COLORED-MAP-W-PDD. Let $\mathcal{H}$ be a (q, N) -perfect hash family. For every $f \in \mathcal{H}$ we define a coloring c_f by $c_f(e_j) = f(j)$ for $j \in [q]$ and let $\mathcal{I}_f = (\mathcal{N}, \mathcal{F}, k, D, c_f)$ be the corresponding instance of EX- N -COLORED-MAP-W-PDD. Solve every instance $\mathcal{I}_f$, and return **yes** if and only if $\mathcal{I}_f$ is a **yes**-instance for some $f \in \mathcal{H}$.

Correctness. For any subset of edges E' with $|E'| \leq N$, there is some $f \in \mathcal{H}$ such that c_f is injective on E' by Definition 2.1.

Let $\mathcal{I}_f$ be a **yes**-instance of EX- N -COLORED-MAP-W-PDD. Thus, there is a perfect triple (A, χ_1, χ_2) such that $X \setminus A$ is γ -viable, A has a size of at least $\bar{k}$ and for each $x \in A$ there is a set of respecting edges $F_{x, \chi_2(x)} \subseteq E(\mathcal{N})$ with $\chi_1(x) = c(F_{x, \chi_2(x)})$ and $\sum_{x \in A} \omega(F_{x, \chi_2(x)}) \leq \bar{D}$. We show that $S := X \setminus A$ is a solution for instance $\mathcal{I}$. By definition, S is γ -viable and has a size of $|X| - |A| \leq |X| - \bar{k} = k$. As (A, χ_1, χ_2) is perfect, $F_{x, \chi_2(x)}$ have pairwise disjoint colors for $x, y \in A$ and are therefore disjoint. We conclude with Lemma 3.6 that $PD_{\mathcal{N}}(S) \geq \omega(E(\mathcal{N})) - \sum_{x \in A} \omega(F_{x, \chi_2(x)}) \geq \omega(E(\mathcal{N})) - \bar{D} = D$.

Now, let $\mathcal{I}$ be a **yes**-instance of MAP-WEIGHTED-PDD with solution $S \subseteq X$. Let $A := X \setminus S$ and since we can assume that S has a size of k , the size of A is $\bar{k}$. Let $E_A \subseteq E(\mathcal{N})$ be the set of edges uv where $\text{off}(v) \subseteq A$. As $D \leq PD_{\mathcal{N}}(S) = \omega(E(\mathcal{N}) \setminus E_A)$, we conclude that $\omega(E_A) \leq \bar{D}$.

Now define F_{x_1} to be the edges uv in E_A with $x_1 \in \text{off}(v)$, and define F_{x_i} to be the edges uv in $E_A \setminus \bigcup_{j=1}^{i-1} F_{x_j}$ with $x_i \in \text{off}(v)$. We define sets F'_{x_i} as follows. For each edge $uv \in F_{x_i}$, for which no incoming edge of u is in F_{x_i} , we add an edge uw outgoing of u to F'_{x_i} , where uw is from $Z_i := ((E(\mathcal{N}) \setminus E_A) \cup \bigcup_{j=1}^{i-1} F_{x_j}) \setminus \bigcup_{j=1}^{i-1} F'_{x_j}$. If F_{x_i} contains more than one edge outgoing of u , adding one edge is sufficient.

▷ **Claim A.1.2.** The set Z_i contains an edge outgoing of u .

Proof. Assume first that $\bigcup_{j=1}^{i-1} F'_{x_i}$ contains an edge uv' . Without loss of generality, let $uv' \in F'_{x_t}$ and $\bigcup_{j=t+1}^{i-1} F'_{x_i}$ does not contain edges outgoing of u . Then there is an edge $uv_t \in F_{x_t}$ and by construction uv_t is not in F'_{x_j} for any $j \in [i]$. Consequently, the set Z_i contains uv_t .

Now, assume that $\bigcup_{j=1}^{i-1} F'_{x_i}$ does not contain edges outgoing of u . If $E(\mathcal{N}) \setminus E_A$ contains an edge e outgoing of u , then $e \in Z_i$. Otherwise, all edges incoming at u are in E_A . As F_{x_i} contains no edge incoming at u , these edges and at least one edges e outgoing of u have to be in F_{x_t} for some $t \in [i-1]$. We conclude $e \in Z_i$ and the set Z_i contains an edge outgoing of u . ◁

The sets F_{x_i} are constructed to be the respecting sets and F'_{x_i} are the auxiliary sets from the definition of respecting sets. We observe that E_A is the union of all F_{x_i} . Thus, if $\mathcal{N}$ is a tree, then $|E_A| \leq 2\bar{k} - 1$, as a forest with $\bar{k}$ leaves has at most $2\bar{k} - 1$ edges. If $\mathcal{N}$ is a network, then $|E_A| \leq \omega(E_A) = \omega(E(\mathcal{N})) - PD_{\mathcal{N}}(S) \leq \bar{D}$. As we added at most one edge to F'_{x_i} per edge of F_{x_i} , we conclude that the union U of all F_{x_i} and F'_{x_i} contains at most $4\bar{k} - 2$ if $\mathcal{N}$ is a tree, and $2\bar{D}$ edges if $\mathcal{N}$ is a network.

Consequently, there is some $f \in \mathcal{H}$, such that c_f is injective on U . We set $\chi_1(x_i) = c_f(F_{x_i})$ and $\chi_2(x_i) = c_f(F'_{x_i})$. By the construction we conclude that (A, χ_1, χ_2) is perfect, S is γ -viable, and $\sum_{i=1}^{\bar{k}} \omega(F_{x_i, \chi_2(x_i)}) = \sum_{i=1}^{\bar{k}} \omega(F_{x_i}) = \omega(E_A) \leq \bar{k}$. Thus, $\mathcal{I}_f$ is a **yes**-instance of EX- N -COLORED-MAP-W-PDD.

Running Time. The construction of $\mathcal{H}$ takes $e^N N^{\mathcal{O}(\log N)} \cdot q \log q$ time (**Proposition 2.2**), and for each $f \in \mathcal{H}$ the construction of instance $\mathcal{I}_f$ of EX- N -COLORED-MAP-W-PDD takes time linear in $|\mathcal{I}|$. By **Lemma 4.7**, solving instances of EX- N -COLORED-MAP-W-PDD takes $\mathcal{O}(5^N 2^{\text{sw}_{\mathcal{F}}} \cdot n \cdot (N^2 \cdot \bar{k}^2 + |E(\mathcal{N})|))$ time, and the number of instances is $|\mathcal{H}| = e^N N^{\mathcal{O}(\log N)} \cdot \log q$.

Thus, the total running time is $\mathcal{O}(e^N N^{\mathcal{O}(\log N)} \log q \cdot (q + 5^N 2^{\text{sw}_{\mathcal{F}}} \cdot n \cdot (N^2 \cdot \bar{k}^2 + E(\mathcal{N}))))$. This simplifies to $\mathcal{O}((5e)^N \cdot 2^{\text{sw}_{\mathcal{F}} + \mathcal{O}(\log^2(N))} \cdot n \cdot |E(\mathcal{N})| \cdot \log |E(\mathcal{N})|)$, as $\bar{k} \leq N$.

Inserting $2\bar{D}$ or $4\bar{k} - 2$ into N , respectively, gives the desired running times. ◀

A.2 Proof of Lemma 3.6

► **Lemma 3.6** ($\star$). $PD_{\mathcal{N}}(X \setminus A) \geq \omega(E(\mathcal{N})) - \sum_{x \in A} \omega(F_{x, \chi_2(x)})$ for every perfect triple (A, χ_1, χ_2) .

Proof. Recall that $x_1, \dots, x_n$ is the ordering of X given by a depth-first traversal of $T_{\mathcal{F}}$. For the sake of notational convenience, assume that $A = \{x_1, \dots, x_{|A|}\}$.

The intuitive idea behind our proof is to show that, as we kill each of the taxa $x_1, \dots, x_{|A|}$ in order, the amount of diversity we lose by killing x_i is at most $\omega(F_{x_i, \chi_2(x_i)})$. To this end, define $\text{EXT}_{\mathcal{N}}(A') := E(\mathcal{N}) \setminus E(X \setminus A')$ for any $A' \subseteq X$. That is, $\text{EXT}_{\mathcal{N}}(A')$ is the set of edges in $\mathcal{N}$ with all offspring in A' .

We prove the following claim by induction. $\text{EXT}_{\mathcal{N}}(\{x_1, \dots, x_s\}) \subseteq \bigcup_{i=1}^s \omega(F_{x_i, \chi_2(x_i)})$, for each $s \in [|A|]$. Note that by letting $s = |A|$, this implies $PD_{\mathcal{N}}(X \setminus A) = \omega_{\Sigma}(E(\mathcal{N}) \setminus \text{EXT}_{\mathcal{N}}(A)) = \omega_{\Sigma}(E(\mathcal{N})) - \omega_{\Sigma}(\text{EXT}_{\mathcal{N}}(A)) \geq \omega_{\Sigma}(E(\mathcal{N})) - \sum_{x \in A} \omega(F_{x, \chi_2(x)})$, as required.

For the base case, observe that $\text{EXT}_{\mathcal{N}}(\{x_1\})$ consists of the edges uv for which v has a path to x , and either $v = x$ or v is a reticulation. But the definition of a respecting set implies that all such edges are also in $F_{x_1, \chi_2(x_1)}$, and so $\text{EXT}_{\mathcal{N}}(\{x_1\}) \subseteq F_{x_1, \chi_2(x_1)}$.

For the inductive step, assume that $\text{EXT}_{\mathcal{N}}(\{x_1, \dots, x_s\}) \subseteq \bigcup_{i=1}^s \omega(F_{x_i, \chi_2(x_i)})$ for some $s < |A|$, and now assume for a contradiction that $\text{EXT}_{\mathcal{N}}(\{x_1, \dots, x_{s+1}\})$ is not a subset of $\bigcup_{i=1}^{s+1} \omega(F_{x_i, \chi_2(x_i)})$. Then consider a lowest edge $e = uv$ in $\text{EXT}_{\mathcal{N}}(\{x_1, \dots, x_{s+1}\}) \setminus$

$\bigcup_{i=1}^{s+1} \omega(F_{x_i, \chi_2(x_i)})$. By definition edges outgoing of v are also in $\text{EXT}_{\mathcal{N}}(\{x_1, \dots, x_{s+1}\})$, and as uv is a lowest edge, all edges outgoing of v are also in $\bigcup_{i=1}^{s+1} \omega(F_{x_i, \chi_2(x_i)})$. At least one child edge vw is not in $\text{EXT}_{\mathcal{N}}(\{x_1, \dots, x_s\})$ (otherwise $uv \in \text{EXT}_{\mathcal{N}}(\{x_1, \dots, x_s\}) \subseteq \bigcup_{i=1}^s \omega(F_{x_i, \chi_2(x_i)})$, and this edge vw must be in $F_{x_{s+1}, \chi_2(x_{s+1})}$. Then as v has an edge incoming which is not in $F_{x_{s+1}, \chi_2(x_{s+1})}$, it follows by definition of a respecting set that vw has a sibling edge vw' with $c(vw') \in \chi_2(s+1)$.

Note however that $vw' \notin \bigcup_{i=1}^{s+1} \omega(F_{x_i, \chi_2(x_i)})$, as $c(vw') \in \chi_2(x_{s+1})$, which is disjoint from $\chi_1(v_i)$ for each $i \leq s$, and $\chi_1(e) \in \chi_1(v_i)$ for each $e \in F_{x_i, \chi_2(x_i)}$. Therefore, vw' is also not in $\text{EXT}_{\mathcal{N}}(\{x_1, \dots, x_{s+1}\})$, as otherwise uv was not a lowest edge in $\text{EXT}_{\mathcal{N}}(\{x_1, \dots, x_{s+1}\}) \setminus \bigcup_{i=1}^{s+1} \omega(F_{x_i, \chi_2(x_i)})$. It follows that w' , and therefore v , has a path to a leaf which is not in $\{x_1, \dots, x_{s+1}\}$, contradicting the assumption that uv was not a lowest edge in $\text{EXT}_{\mathcal{N}}(\{x_1, \dots, x_{s+1}\})$. ◀

A.3 Proof of Lemma 3.7

► **Lemma 3.7** (*). *Given a tree-extension $T_{\mathcal{F}}$ of the food web, we can solve instances of EX-N-COLORED-MAP-W-PDD in $\mathcal{O}(5^N \cdot 2^{\text{sw}_{\mathcal{F}}} \cdot n \cdot (N \cdot \bar{k} + |E(\mathcal{N})|))$ time.*

Proof. Intuition. We consider the tree extension of the food web with a dynamic program bottom up. At each vertex v , we determine whether there exists a perfect triple (A, χ_1, χ_2) satisfying certain conditions, where A is a subset of $T_{\mathcal{F}}^{(v)}$, the set of vertices descended from v in $T_{\mathcal{F}}$. We want that $X \setminus A$ is γ -viable; to help determine this we keep track of a subset of edges $\Phi \subseteq GW(v)$, and require that $T_{\mathcal{F}}^{(v)} \setminus A$ is Φ -part- γ -viable.

Table Definition. For a set of taxa $Q \subseteq X$, sets of colors $C_1, C_2 \subseteq [N]$, a set of edges $\Phi \subseteq E(\mathcal{F})$ between Q and $X \setminus Q$, and an integer $\ell \in [\bar{k}]_0$, let $\mathcal{S}_{(Q, C_1, C_2, \Phi, \ell)}$ be the set of perfect triples (A, χ_1, χ_2) of a set $A \subseteq Q$ of size at least ℓ and mappings $\chi_1, \chi_2 : A \rightarrow 2^C$ with $\chi_1(A) = C_1$ and $\chi_2(A) \setminus \chi_1(A) \subseteq C_2$, for which $Q \setminus A$ is Φ -part- γ -viable.

We define a dynamic programming algorithm with table DP. For a vertex $v \in V(T_{\mathcal{F}})$, sets of colors $C_1, C_2 \subseteq [N]$, a set of edges $\Phi \subseteq GW(v)$, and an integer $\ell \in [k]_0$, we store in $\text{DP}[v, C_1, C_2, \Phi, \ell]$ the minimum value of $\omega(F_{A, \chi_2}) := \sum_{x \in A} \omega(F_{x, \chi_2(x)})$ for any perfect triple $(A, \chi_1, \chi_2) \in \mathcal{S}_{(T_{\mathcal{F}}^{(v)}, C_1, C_2, \Phi, \ell)}$. If v an internal vertex of $T_{\mathcal{F}}$ with children $w_1, \dots, w_t$, then we define further auxiliary tables $\text{DP}_i[v, C_1, C_2, \Phi, \Psi, \ell]$ analogously, with (S, χ_1, χ_2) considered from $\mathcal{S}_{(Q_i, C_1, C_2, P_i, \ell)}$ with $Q_i := \bigcup_{j=1}^i T_{\mathcal{F}}^{(w_j)}$ and $P_i := (\Phi \cup \Psi) \cap \left(\bigcup_{j=1}^i GW(w_j)\right)$; that is—only the first i children of v are considered. We require that the set of edges Ψ is either empty or $\text{pred}^{(E)}(v)$.

Algorithm. We define the table in a bottom-up fashion. Let v be a leaf of $T_{\mathcal{F}}$. (That is a taxon without predators). Fix disjoint color sets $C_1, C_2 \subseteq [N]$. If $\ell > 1$, we store ∞ in $\text{DP}[v, C_1, C_2, \Phi, \ell]$. If $\ell = 0$ and $\gamma_{\Sigma}(\Phi) \geq 1$, we store 0 in $\text{DP}[v, C_1, C_2, \Phi, \ell = 0]$. Otherwise—if $\ell = 1$ or $\gamma_{\Sigma}(\Phi) < 1$ —we store $\min\{\omega(F_{v, C'_2}) \mid C'_2 \subseteq C_2, c(F_{v, C'_2}) = C_1\}$ in $\text{DP}[v, C_1, C_2, \Phi, \ell = 1]$.

Now, let v be an internal vertex of $T_{\mathcal{F}}$ with children $w_1, \dots, w_t$ such that w_i comes before w_{i+1} in the depth-first traversal ordering of X , for $i \in [t-1]$. We define $\text{DP}_1[v, C_1, C_2, \Phi, \Psi, \ell]$ to be $\text{DP}[w_1, C_1, C_2, (\Phi \cup \Psi) \cap GW(w_1), \ell]$. To compute further values for $i \in [t-1]$, we use the following recurrence in which we define $\text{DP}_{i+1}[v, C_1, C_2, \Phi, \Psi, \ell]$ to be

$$\min \quad \text{DP}_i[v, C'_1, C'_2 \cup (C_1 \setminus C'_1), \Phi, \Psi, \ell'] \quad (2)$$

$$+ \text{DP}[w_{i+1}, C_1 \setminus C'_1, C_2 \setminus C'_2, (\Phi \cup \Psi) \cap GW(w_{i+1}), \ell - \ell']. \quad (3)$$

Here, we take the minimum over all $C'_1 \subseteq C_1, C'_2 \subseteq C_2$ and $\ell' \in [\ell]_0$.

Finally, if $v = s_{\mathcal{F}}$ or $\gamma_{\Sigma}(\Phi \cap \text{prey}^{(E)}(v)) \geq 1$, we set $\text{DP}[v, C_1, C_2, \Phi, \ell]$ to be

$$\min\{ \text{DP}_t[v, C_1, C_2, \Phi, \text{pred}^{(E)}(v), \ell] ; \text{DP}'_t \}. \quad (4)$$

Here, DP'_t is $\min_{C'_2 \subseteq C_2} \text{DP}_t[v, C_1 \setminus c(F_{v, C'_2}), C_2 \setminus C'_2, \Phi, \emptyset, \ell - 1] + \omega(F_{v, C'_2})$. Otherwise, we set $\text{DP}[v, C_1, C_2, \Phi, \ell]$ to be DP'_t . Intuitively, DP'_t corresponds to the case that v is one of the taxa to be going extinct, while $\text{DP}_t[v, C_1, C_2, \Phi, \text{pred}^{(E)}(v), \ell]$ corresponds to the case that $\text{DP}_t[v, C_1, C_2, \Phi, \text{pred}^{(E)}(v), \ell]$ corresponds to the case that v is saved.

We return **yes** if $\text{DP}[s_{\mathcal{F}}, C_1, C_2, \emptyset, \bar{k}] \leq \bar{D}$, for some $C_1, C_2 \subseteq [N]$ which are disjoint. Otherwise, we return **no**.

Correctness. Let us quickly consider the basic case. For a leaf v in $T_{\mathcal{F}}$, the set $T_{\mathcal{F}}^{(v)}$ only contains v and so $\mathcal{S}_{(T_{\mathcal{F}}^{(v)}, C_1, C_2, \Phi, \ell)}$ for $\ell > 1$ is empty. If $\ell = 0$, the only possible triples (A, χ_1, χ_2) in $\mathcal{S}_{(T_{\mathcal{F}}^{(v)}, C_1, C_2, \Phi, \ell)}$ have $A \subseteq \{v\}$, i.e. we must let v survive or go extinct. We can only let v survive if $\{v\}$ is Φ -part- γ -viable. Thus we can store 0 if $\gamma_{\Sigma}(\Phi) \geq 1$ and otherwise the minimal value of $\omega(F_{A, \chi_2})$ is just $\omega(F_{v, \chi_2(x)})$, and so we store the minimum weight of a set of respecting edges $F_{v, C}$ such that $c(F_{v, C}) = C_1$ and $C \subseteq C_2$.

To show the correctness of Recurrence (2) and Recurrence (4), we assume that each previous table entries are correct and we prove that if $\text{DP}_i[v, C_1, C_2, \Phi, \Psi, \ell] = d$, respectively $\text{DP}[v, C_1, C_2, \Phi, \ell] = d$, then there is a triple (A, χ_1, χ_2) with $\omega(F_{A, \chi_2}) = d$ from $\mathcal{S}_{(Q_i, C_1, C_2, P_i, \ell)}$, respectively $\mathcal{S}_{(T_{\mathcal{F}}^{(v)}, C_1, C_2, \Phi, \ell)}$. Afterward, we show that for each such triple, it holds that $\text{DP}_i[v, C_1, C_2, \Phi, \Psi, \ell] \geq \omega(F_{A, \chi_2})$, respectively $\text{DP}[v, C_1, C_2, \Phi, \ell] \geq \omega(F_{A, \chi_2})$.

We first show the correctness of Recurrence (2). Assume that $\text{DP}_{i+1}[v, C_1, C_2, \Phi, \Psi, \ell] = d$. Then, by Recurrence (2), there is a $d^1 \in [d]_0$ and $C'_1 \subseteq C_1, C'_2 \subseteq C_2, \ell' \in [\ell]_0$ such that $\text{DP}_i[v, C'_1, C'_2, \Phi, \Psi, \ell'] = d^1$ and $\text{DP}[w_{i+1}, C_1 \setminus C'_1, (C_2 \setminus C'_2) \cup C'_2, (\Phi \cup \Psi) \cap \text{GW}(w_{i+1}), \ell - \ell'] = d - d^1 =: d^2$. Consequently, there is $(A^1, \chi_1^1, \chi_2^1)$ in $\mathcal{S}_{(Q_i, C'_1, C'_2 \cup (C_1 \setminus C'_1), P_i, \ell')}$ such that $\omega(F_{A^1, \chi_2^1}) = d^1$ and there is $(A^2, \chi_1^2, \chi_2^2) \in \mathcal{S}_{(T_{\mathcal{F}}^{(w_{i+1})}, C_1 \setminus C'_1, (C_2 \setminus C'_2), (\Phi \cup \Psi) \cap \text{GW}(w_{i+1}), \ell - \ell')}$ such that $\omega(F_{A^2, \chi_2^2}) = d^2$. As Q and $T_{\mathcal{F}}^{(w_{i+1})}$ are disjoint, so also A^1 and A^2 are disjoint. We therefore define a set $A := A^1 \cup A^2$, and mappings χ_i with $\chi_i(x) = \chi_i^j(x)$ for each taxon $x \in A^j$, $i, j \in \{1, 2\}$. Then $\omega(F_{A, \chi_2}) = \omega(F_{A^1, \chi_2^1}) + \omega(F_{A^2, \chi_2^2}) = d^1 + d^2 = d$. It remains to show that (A, χ_1, χ_2) is in $\mathcal{S}_{(Q_{i+1}, C_1, C_2, P_{i+1}, \ell)}$. Most of the conditions follow because A^1 and A^2 are disjoint and the axioms hold for the individual sets. The difficult part is to show that $\chi_1(x_i)$ and $\chi_2(x_j)$ are disjoint for $i \leq j$, in the case that $x_i \in A^1$ and $x_j \in A^2$. For this, we observe that $\chi_1^1(A^1) \subseteq C'_1 \subseteq C_1$ and $\chi_2^2(A^2) \subseteq C_2 \setminus C'_2 \subseteq C_2$, and these are disjoint.

Now assume (A, χ_1, χ_2) is in $\mathcal{S}_{(Q_{i+1}, C_1, C_2, P_{i+1}, \ell)}$. Let $A^1 := A \cap Q_i$ and let χ_i^1 be the restriction of χ_i to A^1 , for $i \in \{1, 2\}$. Similarly let $A^2 := A \setminus Q_i$, and let χ_i^2 be the restriction of χ_i to A^2 , for $i \in \{1, 2\}$. It is straightforward to check that $(A^1, \chi_1^1, \chi_2^1)$ is in $\mathcal{S}_{(Q_i, C'_1, C'_2 \cup (C_1 \setminus C'_1), P_i, \ell')}$ and $(A^2, \chi_1^2, \chi_2^2) \in \mathcal{S}_{(T_{\mathcal{F}}^{(w_{i+1})}, C_1 \setminus C'_1, (C_2 \setminus C'_2), (\Phi \cup \Psi) \cap \text{GW}(w_{i+1}), \ell - \ell')}$, where $C'_1 = \chi_1(A^1), C'_2 = \chi_2(A^1) \setminus C'_1$, and $\ell = |A^1|$.

We next show the correctness of Recurrence (4). Assume that $\text{DP}[v, C_1, C_2, \Phi, \ell] = d$. By Recurrence (4), we have $\text{DP}_t[v, C_1, C_2, \Phi, \text{pred}^{(E)}(v), \ell] = d$ or $\min_{C'_2 \subseteq C_2} \text{DP}_t[v, C_1 \setminus c(F_{v, C'_2}), C_2 \setminus C'_2, \Phi, \emptyset, \ell - 1] + \omega(F_{v, C'_2}) = d$. In the former case, there is (A, χ_1, χ_2) in $\mathcal{S}_{(Q_t, C_1, C_2, P_t, \ell)}$ with $\omega(F_{A, \chi_2}) = d$. We observe that (A, χ_1, χ_2) is also in $\mathcal{S}_{(T_{\mathcal{F}}^{(v)}, C_1, C_2, \Phi, \ell)}$ which is sufficient for this case. In the latter case, fix C'_2 such that $\text{DP}_t[v, C_1 \setminus c(F_{v, C'_2}), (C_2 \setminus C'_2) \cup c(F_{v, C'_2}), \Phi, \Psi = \emptyset, \ell - 1] + \omega(F_{v, C'_2}) = d$. Consequently, there is a triple (A, χ_1, χ_2) in $\mathcal{S}_{(Q_t, C_1 \setminus c(F_{v, C'_2}), (C_2 \setminus C'_2) \cup c(F_{v, C'_2}), P_t, \ell)}$. Consider (A', χ_1, χ_2) with $A' := A \cup \{v\}$, χ'_i is χ_i on A

with $\chi_2(v) = C'_2$ and $\chi'_1(v) := c(F_{v,C'_2})$. Clearly $\omega(F_{A',\chi_2}) = \omega(F_{A,\chi_2}) + \omega(F_{v,C'_2}) = d$. As $(T_{\mathcal{F}}^{(v)} \setminus A' = Q_t \setminus A$ and $Q_t \setminus A$ is P_t -part- γ -viable, we have also that $T_{\mathcal{F}}^{(v)} \setminus A'$ is ϕ -part- γ -viable (note that the only edges in $P_t \setminus \phi$ are those incoming at v , which are not needed for a set not containing v .) It remains to show that (A', χ'_1, χ'_2) is perfect, assuming that (A, χ_1, χ_2) is perfect.

Note that v appears before any vertex in Q_t in the depth-first traversal of $T_{\mathcal{F}}$, and so we may assume v is the first element of A' . It is clear that $\chi'_1(v) = c(F_{v,C'_2})$ is disjoint from $C_1 \setminus c(F_{v,C'_2})$, and so $\chi'_1(x)$ and $\chi'_1(y)$ are disjoint for all $x, y \in A'$. As $\chi_2(y) \subseteq C_2 \setminus C'_2 \cup (C_1 \setminus c(F_{v,C'_2}))$ for all $y \in A$ and C_1, C'_2 are disjoint, we have that $\chi'_2(v) = C'_2$ and $\chi'_2(y)$ are disjoint. Therefore $\chi'_2(x)$ and $\chi'_2(y)$ are disjoint for all $x, y \in A'$. Similarly as $\chi'_1(v) = c(F_{v,C'_2})$ and $\chi_2(y) \subseteq C_2 \setminus C'_2 \cup (C_1 \setminus c(F_{v,C'_2}))$ for $y \in A$, we have that $\chi'_1(x_i)$ and $\chi'_2(x_j)$ are pairwise disjoint for all $x_i, x_j \in A'$ with $i < j$. The existence of F_{x,C'_2} follows from $\text{DP}'_t + \omega(F_{v,C'_2}) = d$.

On the converse, if (A, χ_1, χ_2) is in $\mathcal{S}_{(T_{\mathcal{F}}^{(v)}, C_1, C_2, \Psi, \ell)}$, then we can show by a case distinction and with arguments similar to the previous paragraph that $\text{DP}[v, C_1, C_2, \Phi, \ell] \leq \omega(F_{A,\chi_2})$.

Running time. By Lemma 3.5, we can compute $F_{x,C}$ or conclude that it does not exist for all $x \in X$ and $C \subseteq [N]$ in $\mathcal{O}(2^N \cdot \sqrt{N} \cdot n \cdot |E(\mathcal{N})|)$ time.

We observe that because C_1 and C_2 are disjoint, $T_{\mathcal{F}}$ is a tree and therefore any vertex has at most one parent, and the field Ψ can only take two values for a fixed vertex v .

In the basic cases, in Recurrence (2), and in Recurrence (4), we iterate over $C'_2 \subseteq C_2$ and in Recurrence (2) we additionally iterate over $C'_1 \subseteq C_1$. Any color $c \in [N]$ can therefore be in $[N] \setminus (C_1 \cup C_2)$, $C_2 \setminus C'_2$, C'_2 , $C_1 \setminus C'_1$ or in the case of Recurrence (2) also in C'_1 .

Thus, all table entries can be computed within $\mathcal{O}(5^N 2^{\text{sw}_{\mathcal{F}}} \cdot n \cdot (N \cdot \bar{k} + |E(\mathcal{N})|))$ time. ◀

A.4 Proof of Lemma 4.1

► **Lemma 4.1** (*). $h_t \leq H$ and $H \leq \delta^{h_r} \cdot h_t \leq \delta^h$.

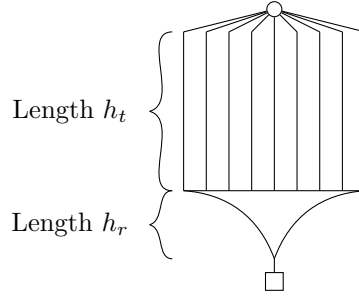
Proof. Recall h_r (and h_t) is the maximum number of reticulations (respectively tree vertices) on a path from the root to a leaf. For each path P from the root to a leaf we observe $|E(P) \cap E_T(\mathcal{N})| \leq H$ and as there is a path of containing h_t tree edges, we conclude $h_t \leq H$.

To see that $H \leq \delta^{h_r} \cdot h_t \leq \delta^h$, let $\mathcal{N}$ be a network maximizing the value of H for values of h_r, h_t, δ , and let $x \in X$ be a leaf for which the maximum number of tree vertices with a path to x is maximized. We show that there is no tree vertex below a reticulation on any path to x . Assume for a contradiction that there is an edge $rv_1 \in E(\mathcal{N})$ with $v_1 \in V_T(\mathcal{N})$ and $r \in V_R(\mathcal{N})$ and a path from v_1 to x . Let $u_1, \dots, u_s$ be the parents of r and w a child of v_1 such that w has a path to x . Remove the edges $u_i r$ for $i \in [s]$, rv_1 , and $v_1 w$, add vertices $v_2, \dots, v_s$ with attached leaves, and add edges $u_i v_i$ for $i \in [s]$, $v_i r$ for $i \in [s]$ and rw . Observe that after this transformation, the values of h_r , h_t , and δ have not changed, but there are $s - 1$ further tree vertices with a path to x . As $s > 1$, this contradicts the maximality.

We may therefore assume that no tree vertices with a path to x are below a reticulation. We conclude that from the root at most δ^{h_r} different paths of length h_t of tree vertices can lead to a leaf. Figure 3 shows this scenario. ◀

A.5 Proof of Lemma 4.7

► **Lemma 4.7** (*). Given an instance $\mathcal{I}$ of kH -COLORED-MAP-W-PDD and a tree-extension $T_{\mathcal{F}}$ of the food web, we can solve $\mathcal{I}$ in $\mathcal{O}(2^{k \cdot H + \text{sw}_{\mathcal{F}}} \cdot k^4 \cdot H^2 \cdot \text{sw}_{\mathcal{F}} \cdot n \cdot |E(\mathcal{N})|^2)$ time.



■ **Figure 3** An example for Lemma 4.1 where H is maximized. This is the case if above the leaf an upside-down pyramid of reticulations is followed by δ^{h_r} paths of tree vertices of length h_t .

Proof. Intuition. We consider the tree extension of the food web bottom up in a dynamic programming algorithm. We track the existence of colorful tuples whose taxa are all below a given vertex of the tree extension. We index colorful tuples by the sets of colors used, as well as by the set of edges Φ for which the set of taxa is Φ -part- γ -viable, and we store the total weight of the suitable edge sets. We use the fact that the sets $\chi(x)$ and $\chi(y)$ have to be pairwise disjoint. Therefore, if a taxon x is saved and a set of colors has been assigned, these colors can be removed. We define $\chi(x)$ when selecting x .

Table Definition. For a set of taxa $Q \subseteq X$, a set of colors $C \subseteq [k \cdot H]$, a set of edges $\Phi \subseteq E(\mathcal{F})$ between Q and $X \setminus Q$, and an integer $\ell \in [k]_0$, let $\mathcal{S}_{(Q,C,\Phi,\ell)}$ be the set of colorful tuples (S, χ) of a set $S \subseteq Q$ of size at most ℓ and a mapping $\chi : S \rightarrow 2^C$ with $\chi(S) = C$ such that S is Φ -part- γ -viable. We define a dynamic programming algorithm with table DP. For a vertex $v \in V(T_{\mathcal{F}})$, a set of colors $C \subseteq [k \cdot H]$, a set of edges $\Phi \subseteq GW(v)$, and an integer $\ell \in [k]_0$, we store in $DP[v, C, \Phi, \ell]$ the maximum value $\omega(F_{S,\chi})$ of any tuple $(S, \chi) \in \mathcal{S}_{(T_{\mathcal{F}}^{(v)}, C, \Phi, \ell)}$. The value of an optimal set can then be found in $DP[s_{\mathcal{F}}, [k \cdot H], \emptyset, k]$. Is v an internal vertex of $T_{\mathcal{F}}$ with children $w_1, \dots, w_t$, then we define further auxiliary tables $DP_i[v, C, \Phi, \Psi, \ell]$ analogously, where tuples (S, χ) are considered from $\mathcal{S}_{(Q,C,P,\ell)}$ with $Q := \bigcup_{j=1}^i T_{\mathcal{F}}^{(w_j)}$ and $P := (\Phi \cup \Psi) \cap \left(\bigcup_{j=1}^i GW(w_j) \right)$; that is—only the first i children of v are considered. We require that the set of edges Ψ is either empty or $\text{pred}^{(E)}(v)$.

Algorithm. We define the table in a bottom-up fashion. Let v be a leaf of $T_{\mathcal{F}}$. (That is a taxon without predators). In $DP[v, C, \Phi, \ell]$, we store $\omega(F_{v,\chi})$ with $\chi(v) = C$ if $\gamma_{\Sigma}(\Phi) = 1$ and $\ell \geq 1$. Otherwise, we store 0.

Now let v be an internal vertex of $T_{\mathcal{F}}$ with children $w_1, \dots, w_t$. We define $DP_1[v, C, \Phi, \Psi, \ell]$ to be $DP[w_1, C, (\Phi \cup \Psi) \cap GW(w_1), \ell]$. To compute further values for $i \in [t-1]$, we use the following recurrence in which we define $DP_{i+1}[v, C, \Phi, \Psi, \ell]$ to be

$$\max_{C' \subseteq C, \ell' \in [\ell]_0} DP_i[v, C', \Phi, \Psi, \ell'] + DP[w_{i+1}, C \setminus C', (\Phi \cup \Psi) \cap GW(w_{i+1}), \ell - \ell']. \quad (5)$$

Finally, if $\ell \geq 1$ and $(v = s_{\mathcal{F}} \text{ or } \gamma_{\Sigma}(\Phi \cap \text{prey}^{(E)}(v)) \geq 1)$, we set $DP[v, C, \Phi, \ell]$ to be

$$\max\{ DP_t[v, C, \Phi, \emptyset, \ell] ; \max_{C' \subseteq C} DP_t[v, C \setminus C', \Phi, \text{pred}^{(E)}(v), \ell - 1] + \omega(F_{v,\chi}) \}. \quad (6)$$

Here, we use $\chi(v) := C'$. Otherwise, we set $DP[v, C, \Phi, \ell]$ to be $DP_t[v, C, \Phi, \emptyset, \ell]$.

We return yes if $DP[s_{\mathcal{F}}, [k \cdot H], \emptyset, k] \geq D$. Otherwise, we return no.

Correctness. We only show the correctness of Recurrence (6) and omit to show the similar case in Recurrence (5) as well as the basic cases. Assume that DP stores the correct value for all children of v in $T_{\mathcal{F}}$ and DP_t stores the correct value for v . We show first that

if $\text{DP}[v, C, \Phi, \ell] = d$, then there is a tuple $(S, \chi) \in \mathcal{S}_{(T_{\mathcal{F}}^{(v)}, C, \Phi, \ell)}$ and $\omega(F_{S, \chi}) = d$. Afterward, we show that $\text{DP}[v, C, \Phi, \ell] \geq \omega(F_{S, \chi})$ for each tuple $(S, \chi) \in \mathcal{S}_{(T_{\mathcal{F}}^{(v)}, C, \Phi, \ell)}$.

So assume that $\text{DP}[v, C, \Phi, \ell] = d$. Consequently, we have $\text{DP}_t[v, C, \Phi, \emptyset, \ell] = d$, or there is some $C' \subseteq C$ such that $\text{DP}_t[v, C \setminus C', \Phi, \text{pred}^{(E)}(v), \ell - 1] = d - \omega(F_{v, \chi})$ for $\chi(v) = C'$. In the former case, there is a colorful tuple (S, χ) with a set $S \subseteq \bigcup_{i=1}^t T_{\mathcal{F}}^{(w_i)} = T_{\mathcal{F}}^{(v)} \setminus \{v\}$ of size ℓ and a mapping χ such that $\omega(F_{S, \chi}) = d$ and S is $(\Phi \cup \emptyset)$ -part- γ -viable. Consequently, S is also Φ -part- γ -viable in $T_{\mathcal{F}}^{(v)}$. In the latter case, $\ell \geq 1$ and there is a colorful tuple (S, χ) with a set $S \subseteq T_{\mathcal{F}}^{(v)} \setminus \{v\}$ of size $\ell - 1$ and a mapping χ with $\chi(S) = C \setminus C'$, such that $\chi(v) = C'$ and $\omega(F_{S, \chi}) = d - \omega(F_{v, \chi})$. Thus, adding v to S yields the desired set.

Conversely, let $(S, \chi) \in \mathcal{S}_{(T_{\mathcal{F}}^{(v)}, C, \Phi, \ell)}$ be given. Assume first that v is not in S . Because S is Φ -part- γ -viable, we conclude $\text{DP}[v, C, \Phi, \ell] = \text{DP}_t[v, C, \Phi, \emptyset, \ell] \geq \omega(F_{S, \chi})$. Now, let v be in S . As S is Φ -part- γ -viable and $\text{prey}^{(E)}(x) \cap \text{GW}(v) = \text{prey}^{(E)}(x) \cap \Phi$ for each $x \in S$, the set $S \setminus \{v\}$ is $(\Phi \cup \text{pred}^{(E)}(v))$ -part- γ -viable. We conclude $(S \setminus \{v\}, \chi) \in \mathcal{S}_{(T_{\mathcal{F}}^{(v)} \setminus \{v\}, C, \Phi, \ell - 1)}$ and therefore $\text{DP}_t[v, C, \Phi, \text{pred}^{(E)}(v), \ell - 1] \geq \omega(F_{S, \chi}) - \omega(F_{v, \chi})$ and further $\text{DP}[v, C, \Phi, \ell] \geq \omega(F_{S, \chi})$.

Running time. Observe that for any vertex v , only two values are possible for Ψ . Therefore, all tables have $\mathcal{O}(2^{k \cdot H + \text{sw}_{\mathcal{F}}} \cdot nk)$ entries, together.

By Recurrence (5), in time $\mathcal{O}(2^{k \cdot H} \cdot 2^{\text{sw}_{\mathcal{F}}} \cdot k^2 \cdot \text{sw}_{\mathcal{F}} \cdot (kH) \cdot n)$ all entries of DP_{i+1} can be computed, using convolutions. By Recurrence (6), we can compute $\text{DP}[v, C, \Phi, \ell]$ in $\mathcal{O}(k \cdot H)$ time and $\omega(F_{v, \chi})$ needs to be computed once per vertex, which by Lemma 4.1 can be done in $\mathcal{O}(2^H \cdot |E(\mathcal{N})| \cdot (kH + |E(\mathcal{N})|))$ time.

This leads to an overall running time of $\mathcal{O}(2^{k \cdot H + \text{sw}_{\mathcal{F}}} \cdot k^4 \cdot H^2 \cdot \text{sw}_{\mathcal{F}} \cdot n \cdot |E(\mathcal{N})|^2)$. ◀

B Appendix—Counter-example for a theorem in [20]

In this section, will briefly point out a mistake in Reduction Rule 6 of [20].

B.1 The Problem Definition

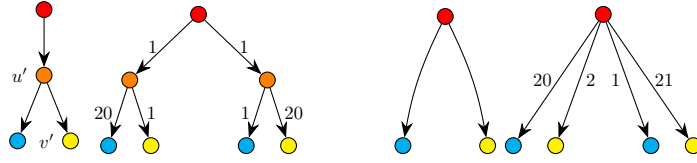
Reduction Rule 6 of [20] is defined on a colored version of ε -PDD, which is used as a subroutine to prove that ε -PDD is FPT with respect to $k + h$. In this colored version of ε -PDD, additionally to a vertex-colored phylogenetic tree $\mathcal{T}$, a food web $\mathcal{F}$, k , and D , one is given a *pattern tree* $\mathcal{T}_P$ —a colorful tree on the same set of colors as the phylogenetic tree has. The task is to find an ε -viable set S of taxa with $|S| \leq k$, $PD_{\mathcal{T}}(S) \geq D$, and the spanning tree of the root and S should be isomorphic with the pattern tree.

Reduction Rule 6 states as follows:

► **Reduction Rule 6.** *Apply previous Reduction Rules exhaustively. Let ρ be the root of $\mathcal{T}$ and let ρ_P be the root of $\mathcal{T}_P$. Let v' be a grand-child of ρ_P and let u' be the parent of v' .*

1. *For each vertex u of $\mathcal{T}$ with $c(u) = c_P(u')$ add edges ρv to $\mathcal{T}$ for every child v of u .*
2. *Set the weight of ρv to be $\omega(uv)$ if $c(v) \neq c_P(v')$ or $\omega(uv) + \omega(\rho u)$ if $c(v) = c_P(v')$.*
3. *Add edges $\rho_P w'$ to $\mathcal{T}_P$ for every child w' of u' .*
4. *Set $\mathcal{T}'_P := \mathcal{T}_P - u'$ and $\mathcal{T}' := \mathcal{T} - u$.*

Reduction Rule 6 is used to reduced the height of the phylogenetic tree until it is only a star and then to use a known algorithm for that case.



■ **Figure 4** An execution of Reduction Rule 6 in [20]. Left is the instance before the execution and right is the instance after the execution. In both instances, to the left is the pattern tree and on the right the phylogenetic tree.

B.2 The Error

Figure 4 shows an example execution of Reduction Rule 6. For $D = 41$ and $k = 2$ the original instance is a **no**-instance, while in the latter, it is a **yes**-instance with the two outermost vertices being a solution (without considering the food web.)

Unfortunately, it is not enough to drop the condition that the spanning tree of the root and S should be isomorphic with the pattern tree. For this example consider $D = 42$ and $k = 2$ for which still the first instance is a **yes**-instance and the latter a **no**-instance.

Nevertheless, we still believe ε -PDD to be FPT with respect to $k + h$ and we further believe that successful attempts for ε -PDD can then be generalized to MAP- ε -PDD when also the parameter δ is considered.