universität wien

DISSERTATION

Titel der Dissertation

Computational Methods in Biodiversity Conservation

angestrebter akademischer Grad

Doktor/in der Naturwissenschaften (Dr. rer.nat.)

Verfasserin / Verfasser:	BUI Quang Minh
Matrikel-Nummer:	0647864
Dissertationsgebiet (lt. Studienblatt):	Molekulare Biologie
Betreuerin / Betreuer:	Univ.-Prof. Dr. Arndt von Haeseler

Wien, am 01. November 2008

Nothing in biology makes sense except in the light of evolution.

Theodosius Dobzhansky (1900-1975)

Biologists must constantly keep in mind that what they see was not designed, but rather evolved.

Francis Crick (*What Mad Pursuit*, 1988)

Abstract

Conservation genetics is an emerging discipline that aims at employing genetic methods to questions in biodiversity conservation. One main achievement of the field is the application of phylogenetic trees to assess the diversity of species (Vane-Wright *et al.*, 1991). *Phylogenetic diversity (PD)* is a quantitative measure proposed by Faith (1992) that assigns to a set of species the sum of the lengths of the branches connecting the species of interest. If the branch lengths reflect evolutionary distances, *PD* will be equivalent to the amount of phylogenetic information accumulated by these species. *PD* therefore aims at preserving as much "evolutionary history" of species as possible.

In this thesis I achieve two main results. Firstly, I develop a novel measure called *split diversity (SD)*. *SD* is motivated by the fact that *PD* relies on having a reliable estimate of the underlying phylogenetic tree. However, conflicting phylogenetic signals are often observed in the data. For example, different genomic regions can provide different trees with different genetic distances. Trees can also differ among reconstruction methods. Given the (possibly incongruent) collection of trees for a fixed set of taxa, how does one evaluate the diversity of a taxon subset? In this context, split diversity is defined as the average of *PD* values computed for each tree. It has been shown that *SD* can be equivalently computed on the union split system of all given trees (Spillner *et al.*, 2008). Hence, various source trees (inferred by different data, e.g., morphological data or genetic data) can be treated simultaneously under the *SD* framework.

Secondly, I develop a new tool called *Phylogenetic Diversity Analyzer (PDA)* to solve several conservation problems. The most simple problem is *taxon selection*: For a fixed number k, find k taxa that maximizes the *PD* or *SD* over all set of k taxa. The resulting maximal set may be considered as of importance for conservation. Under *PD*, I propose two efficient algorithms *gPDA* and *pPDA* based on the greedy strategy that was shown to guarantee an optimal solution (Steel, 2005; Pardi and Goldman, 2005). Under *SD*, I present an efficient dynamic programming algorithm (*SDA*) that computes the optimal *SD* set when the underlying split system is *circular* (that is reconstructed by

e.g., the Neighbor-net method). The more realistic problem is *budgeted taxon selection.* Given that conserving each taxon comes at a specific cost but we are given only a limited budget. We look for a taxon set that maximizes *PD* (or *SD*) over all sets which are affordable within the allotted budget. I will show that the *SDA* algorithm can be extended to cope adequately with budget constraints.

Moreover, I demonstrate the *SD* approach with two datasets. One dataset contains fours genes and the other consists of one gene from bacteria where horizontal gene transfer was detected. The analysis results show some discrepancies between the *PD*-based and *SD*-based taxon selection. This should be taken into account because when such non-treelike events are apparent in the data, considering a single tree for conservation comes at the loss of phylogenetic information. Since non-treelikeness is a major topic in evolutionary biology, it will also be an issue in conservation decision projects. Thus, our proposed method helps close this gap.

PDA is a contribution to the field since it implements all presented algorithms and aided softwares for conservation biologists still remain sparse. Until recently, DIVERSITY, MESA, and WORLDMAP have been the only tools available. Furthermore, the split diversity approach opens some interesting optimization problems. For example, how can one solve the taxon selection under arbitrary split systems? For the extended reserve selection problem, how can one find an optimal collection of k areas with maximal *SD*? In the concept of *SD* alone, it is also interesting to investigate how other combination functions are related to the average of *PD*s as defined above. Answering such questions would further advance the field of "computational biodiversity conservation".

Parts of this thesis have been published in the following articles:

1. B.Q. Minh, S. Klaere, and A. von Haeseler (2006) Phylogenetic Diversity within Seconds. *Systematic Biology*, 55, 769-773.

2. B.Q. Minh, S. Klaere, and A. von Haeseler (2008) Taxon selection under split diversity. *Submitted to Systematic Biology.*

3. B.Q. Minh, F. Pardi, S. Klaere, and A. von Haeseler (2008) Budgeted Phylogenetic Diversity on Circular Split Systems. *Accepted, to appear in IEEE/ACM Transactions on Computational Biology and Bioinformatics.*

The *PDA* (Phylogenetic Diversity Analyzer) package including developments presented in this thesis is freely available from `http://www.cibiv.at/software/pda`.

Acknowledgments

First and foremost, I would like to thank my supervisor, Professor Arndt von Haeseler, for his constant guidance and advice. His brilliant knowledge, invaluable ideas and suggestions have constantly inspired me. I also have to thank the unknown person who printed out Mike Steel's paper and forgot it there. Without him, we would never have started the project on phylogenetic diversity.

I would like to thank Lê Sỹ Vinh for introducing me to Arndt von Haeseler and to the field of bioinformatics.

I would like to thank Steffen Klaere for a fruitful collaboration. Our project proved that a mathematician can work efficiently with a computer scientist.

I would like to thank a lot of colleagues for fruitful discussions including (in alphabetical order): Ross Crozier, Daniel Faith, Tanja Gesell, Lars Jermiin, Fabio Pardi, Heiko Schmidt, Mike Steel, Alicia Toon. I thank Keith Crandall and Félix Forest for providing their data sets.

Financial support from the Wiener Wissenschafts-, Forschungs- and Technologiefonds (WWTF) is greatly appreciated. Friendly environment with all members of the Center for Integrative Bioinformatics Vienna (CIBIV) has made my time full of pleasure and enjoyment.

This thesis would not have been possible without my beloved parents, who were always encouraging me throughout every moment. This thesis is dedicated to my wife Nguyễn Thi Kim Ngân simply for her presence and her continuous support during the midst of pressures.

Contents

List of Figures

List of Tables

Chapter 1

Overview

1.1 Motivation

With the ever looming thread of loss of biodiversity, the localization of biodiversity hotspots and the subsequent establishment of a conservation zone find increased interest among scientists from various disciplines (Wilson, 1997; Myers, 1988, 1990; Myers *et al.*, 2000). Globally, 25 biodiversity hotspots were identified using the endemic *species richness* and the habitat loss (Myers *et al.*, 2000). Species richness assigns to an area the number of species living there (see Gaston and Spicer, 2004, and citations therein), whereas endemic species richness only counts the species that are endemic to the area. Therefore, conservation efforts using (endemic) species richness aim at preserving as many (endemic) species as possible.

One weakness of species richness is the implicit assumption that all species are equal. Such an equal treatment is not justifiable, e.g., “Is the panda equal to one species of rat?” (Vane-Wright *et al.*, 1991). Biodiversity should therefore be evaluated using evolutionary relationships among the species (Vane-Wright *et al.*, 1991). Faith (1992) extended Vane-Wright *et al.*'s approach and introduced the so-called *phylogenetic diversity (PD)* measure. Given a phylogenetic tree of a set of the species, the phylogenetic diversity of a subset of taxa is computed as the sum of the branch lengths of the minimal subtree connecting the species present in that set. Recently, Forest *et al.* (2007) used the flora on a hotspot, the Cape of South Africa, to compare the effect of *PD* and species richness on prioritizing the areas inside the Cape. This study showed that *PD* gives alternative suggestions and should therefore be decoupled from species richness on such dataset.

In bioinformatics, phylogenetic diversity has attracted interest because various conservation questions using *PD* involve typical optimization problems. The most simple problem, the *taxon selection*, asks for a taxon set of a fixed size which maximizes the *PD* over all sets of that size. Such an optimal taxon set can be of important values for conservation. Recently, Steel (2005) and independently Pardi and Goldman (2005) have proven that the taxon selection problem can be solved employing a simple greedy strategy. Taxon selection was furthermore generalized to address more biologically relevant scenarios including a selection of areas instead of taxa (Moulton *et al.*, 2007; Bordewich and Semple, 2008), an introduction of budget constraints (Hartmann and Steel, 2006, 2007; Pardi and Goldman, 2007), an introduction of multiple phylogenetic trees (Minh *et al.*, 2006, 2008a,b), or an introduction of food webs and thus dependencies between species (Moulton *et al.*, 2007). All these scenarios lead to computational challenges. However, efficient algorithms and state of the art implementations remain sparse. Until recently, the packages Diversity (Faith and Walker, 1994), MeSA (Crozier *et al.*, 2005), and Worldmap (Williams and Humphries, 1996) have been the only tools available.

In April 2002, the participants of the *Convention on Biological Diversity* committed themselves to "achieve by 2010 a significant reduction of the current rate of biodiversity loss at the global, regional and national level as a contribution to poverty alleviation and to the benefit of all life on Earth" (Balmford *et al.*, 2005). During the Evolution 2007 conference in Christchurch, New Zealand an SSB Symposium on Phylogenetic Diversity was held uniting most of the authors previously cited. Our conclusion from this meeting is that even though biologists were interested in the scenarios above, the lack of aided software tools tarnishes the application of the theoretical findings. Hence, one important goal is the development of efficient algorithms and softwares to include more relevant scenarios.

1.2 Contributions

We have solved a number of conservation optimization problems. To this end, efficient algorithms have been developed and implemented in a software tool called *Phylogenetic Diversity Analyzer (PDA)*. The results are presented in the subsequent chapters.

Chapter 2 gives a brief introduction to computational biodiversity conservation, phylogenetic diversity, and typical scenarios for optimizing *PD*.

Chapter 3: presents efficient greedy algorithms ($gPDA$, $pPDA$) for the taxon selection on single trees (see also Chapter 2, Problem 1). The algorithm computes an optimal *PD* set of k taxa for a million-taxon tree within a few seconds. This chapter was published in:

B.Q. Minh, S. Klaere, and A. von Haeseler (2006) Phylogenetic Diversity within Seconds. *Systematic Biology*, 55, 769-773.

Chapter 4: introduces the concept of split diversity when evolutionary relationships are better represented in split systems rather than phylogenetic trees. A dynamic programming algorithm (*SDA*) for the taxon selection on circular split systems is presented. Real-data analyses show its usefulness compared to phylogenetic diversity. This chapter was published in:

B.Q. Minh, S. Klaere, and A. von Haeseler (2008) Taxon selection under split diversity. *Submitted to Systematic Biology.*

Chapter 5: extends the *SDA* algorithm to find the optimal taxon set for circular split systems under budget constraints. This chapter was published in:

B.Q. Minh, F. Pardi, S. Klaere, and A. von Haeseler (2008) Budgeted Phylogenetic Diversity on Circular Split Systems. *Accepted, to appear in IEEE/ACM Transactions on Computational Biology and Bioinformatics.*

Chapter 6: gives a brief summary of the results obtained in the thesis.

Chapter 2

Computational Biodiversity Conservation

2.1 Biodiversity Conservation

Biodiversity embraces the variety of life from plants to animals, from micro- to macro-organisms, from genes to genomes and ecosystems (Wilson, 1997). Human extensive activities are damaging the surrounding environment, thus indirectly causing the extinction of various organisms (Wilson, 1997). This projected loss of biodiversity motivates the applied discipline of conservation biology.

To evaluate the severity of this loss, one needs a quantitative measure of biodiversity. Thus, it is not surprising that hundreds of such measures have been proposed in the past since the concept of biodiversity is so broad that it is not obvious how to assign a single number to a species, a genus, or a geographical region (Gaston and Spicer, 2004; Avise, 2005). Wrong diversity assignments would in effect guide policy makers to disastrous conservation decisions. It is therefore of tremendous importance that the impact of measures are evaluated.

The most basic measure of the diversity of an area is *species richness*, the number of species present in the considered area (Gaston and Spicer, 2004). However, since the definition of a species is not clear (Agapow *et al.*, 2004) and since the population size of a species plays an integral part in its survival, more sophisticated measures like the *Simpson-Index* and the *Shannon-Index* are more commonly used as they incorporate the population size into the calculations (see Gaston and Spicer, 2004, and citations therein).

Such measures exhibit an intuitive interpretation of biodiversity and are the most simple but widely used currency for biodiversity assessment. It is apparent that these measures evaluate species solely on their population size and not on their ancestral relationship, e.g. for these measures it is irrelevant whether an area contains 50 chimpanzees and 20 lions or 50 gazelles and 20 elephants. However, due to the lack of other information species richness and its related indices are still considered the first criterion to look at (Gaston and Spicer, 2004).

2.2 Phylogenetic Diversity (PD)

The previously introduced diversity measures are based solely on population size and number of species present. However, the last 20 years have seen a boom in phylogenetic analysis among various organisms (Swofford *et al.*, 1996; Felsenstein, 2004). We are even trying to reconstruct the tree of life (e.g, Maddison and Schulz, 2007). The presence of evolutionary relationships among species, so-called phylogenetic trees, adds another important concept to biodiversity measures. The biodiversity assessment should take phylogenies into account (e.g. Vane-Wright *et al.*, 1991).

Consequently, Faith (1992) suggested phylogenetic diversity (*PD*) as an alternative biodiversity measure. Given a phylogenetic tree of all species of interest, the *PD* of an area is computed as the total sum of lengths of all branches connecting the species living in the given area. Figure 1 illustrates the basic concept. Branch lengths of the tree can be measured by the evolutionary time between two nodes (or number of mutations, or any other dissimilarity measures). If the branch lengths are not present or hard to estimate (i.e., only the tree topology is available), then by assigning each branch a length of 1, the *PD* score is then interpreted as cladistic diversity (Vane-Wright *et al.*, 1991; Crozier, 1997). Recent studies suggested that *PD* should be decoupled from species richness when sufficient genetic data are available (Forest *et al.*, 2007) or when the tree is unbalanced (Rodrigues *et al.*, 2005).

PD has been thoroughly examined and other measures such as *genetic diversity* (Crozier, 1992) also use phylogenetic trees as their basis and are more or less related to *PD*. In the following, we will only focus on *PD* due to its wide-spread use after species richness. Moreover, species richness can also be seen as a special case of *PD* assuming a star tree with equal unit branch lengths. Hence, all the computational approaches

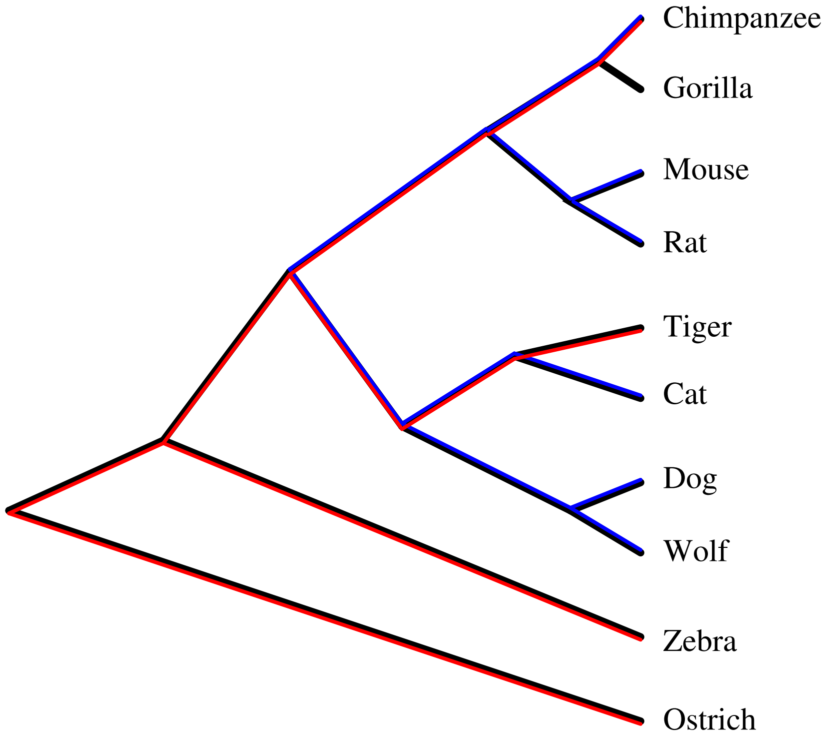


Figure 2.1: An example computation of phylogenetic diversity on a tree with ten species. The *PD* score of the set {Chimpanzee, Mouse, Rat, Cat, Dog, Wolf} is the sum of the lengths of the blue branches, whereas the red branches contribute to the *PD* of {Chimpanzee, Tiger, Zebra, Ostrich}. The first set is more diverse in terms of species richness. However, the second set shows larger phylogenetic diversity.

for *PD* presented here will also apply to species richness. For a discussion of various measures readers are advised to refer to Purvis *et al.* (2005).

PD with respect to algorithmic details has recently gained interest through a cluster of papers (Steel, 2005; Pardi and Goldman, 2005, 2007; Hartmann and Steel, 2006, 2007; Minh *et al.*, 2006, 2008a,b; Moulton *et al.*, 2007; Spillner *et al.*, 2008; Bordewich and Semple, 2008; Bordewich *et al.*, 2008). All authors discuss the following basic optimization, which we call the *taxon selection problem.* Assuming that our resources can only support a fraction of all taxa, the survival of which taxa will maximize the phylogenetic diversity? Or more formally:

Problem 1. (Taxon selection with PD) For a given set X of taxa and its connecting phylogenetic tree, find a subset of k taxa which maximizes the PD over all taxon subsets of size k.

Steel (2005) and Pardi and Goldman (2005) showed that this problem can be solved by employing a greedy strategy. This is surprising since the simple greedy strategy often comes up with a suboptimal solution. However, the proof, that relied on the theory of greedoids, did not automatically lead to an efficient algorithm to compute a solution. In Chapter 3 we proposed a very efficient implementation of the greedy algorithm. Our efficient algorithm computes the PD score for a tree with one million taxa within a few seconds. Moulton *et al.* (2007) proposed several extensions of problem 1, tested the applicability of the greedy algorithm, and analyzed the computational complexity of these scenarios.

2.3 Extended Scenarios for Maximizing Diversity

In this section we introduce three more realistic scenarios that extend Problem 1. For the time being, we use the PD score as the objective function in the optimization problem, although it should be noted that other biodiversity measures can also be applied with the same methodology. We will also use the general “taxa” that, depending on the questions, can be interpreted as species, genus, population, etc.

2.3.1 Budget Constraints

The simple model of taxon selection (Problem 1) implicitly assumes that each taxon requires the same amount of resource for conservation. In fact, people would like to invest more money to svae pandas or polar bears than to monkeys. Therefore, a more realistic scenario is to suppose that preserving each taxon comes at a specific cost. Which selection of taxa will maximize PD such that the total costs do not exceed an allotted budget? The costs and the budget need not only refer to money but also to any other quantifiable human effort such as an affordable size of habitat. This is called the *budgeted taxon selection problem*:

Problem 2. (Budgeted taxon selection with PD) For a given set X of taxa and its connecting phylogenetic tree, non-negative costs c_s for every taxon

$s \in X$ and a total non-negative budget B, find a subset S of taxa to maximize $PD(S)$ subject to $\sum_{s \in S} c_s \leq B$.

If $c_s = 1$ and $B = k$, this formulation will turn out to be Problem 1, and thus is a generalization of Problem 1.

Under budget constraints, the greedy algorithm no longer guarantees an optimal solution. Recently, Pardi and Goldman (2007) introduced a dynamic programming algorithm which ensures to obtain the optimal *PD* set if the costs and budget can be expressed or approximated as integers. The restriction to integral costs and budget is normally not a limitation. For example, people usually mention how many dollars they can afford or how many square kilometers a particular taxon needs for its survival.

2.3.2 Reserve Selection

Usually, measures of diversity are employed to identify geographical regions with high biodiversity. This is also known as the *reserve selection* problem. Again, we are given a set X of taxa of interest. The natural habitat of the taxa is divided into several geographical areas. We call $\mathcal{A}$ the set of these areas. In subsequent descriptions we will identify an area with a subset of taxa. Based on the phylogenetic tree connecting the taxa in X, the *PD* score of an area is defined as the *PD* score of the set of taxa living in this area. Accordingly, the *PD* score of a collection $\mathcal{S}$ of areas is equal to the *PD* score of the set of all taxa living in those areas:

$$PD(\mathcal{S}) = PD\left(\bigcup_{A \in \mathcal{S}} A\right). \tag{2.1}$$

Note, that due to the possibility of overlapping taxon sets for a set of areas, this is not equal to the sum of *PD* scores of the areas considered.

The simplest form of the reserve selection problem is:

Problem 3. (Reserve selection with *PD*) For a given set X of taxa, its connecting phylogenetic tree, a set $\mathcal{A}$ of areas, and a number k, find k areas such that the *PD* score is maximized over all collections of k areas.

Note that with the prioritization of a set of areas all taxa living in these areas benefit from a possible conservation effort, which means that Problem 3 does not restrict the number of taxa. In particular, this scenario can result in the conservation of all taxa considered without prioritizing all areas. Problem 3 will be reduced to Problem 1 when each area contains exactly one taxon.

The problem is called optimizing diversity via regions and proven to be NP-hard in Moulton *et al.* (2007). However, it should be noted that this problem was shown to be equivalent to the maximal covering location problem (MCLP; Church *et al.*, 1996; Rodrigues and Gaston, 2002) that had been proven NP-hard long before (Church and ReVelle, 1974).

One way to find the optimal solution to Problem 3 is to compute the *PD* score of all possible subsets of areas, which will result in a runtime of $O(2^m)$, where m is the number of areas. This is of course not feasible for large m. With $m = 20$ there are already more than one million subsets to look at. So heuristic approaches are needed.

A simple greedy strategy is based on the *complementarity principle* (Vane-Wright *et al.*, 1991; Faith *et al.*, 2004). First, one selects an area A_1 with maximal *PD*. Second, one determines another area A_2 which adds the most "extra" *PD* to A_1. We call this amount the *PD* complementarity of A_2 given A_1 (Faith *et al.*, 2004), formally defined as:

$$PD(A_2|A_1) = PD(A_2 \cup A_1) - PD(A_1). \tag{2.2}$$

Subsequently, we identify the area A_3 which maximize $PD(A_3|A_1 \cup A_2)$ and so on and so forth, until we found k areas.

This greedy algorithm does not guarantee an optimal collection of areas even though it has been usually applied in conservation planning (see Underhill, 1994, for a discussion). An open question is whether the greedy strategy still works if the areas are disjoint.

Another strategy to obtain an exact solution is to transform Problem 3 into an *integer linear programming* problem (Cormen *et al.*, 2001) and then use an available tool such as C-Plex to solve the resulting ILP problem (Rodrigues and Gaston, 2002). This technique was shown to find an optimal collection of k areas efficiently in many cases.

2.3.3 Conflicting Phylogenetic Information

The phylogenetic tree is the basis for the concept of *PD*. However, it is well known that phylogeny reconstruction methods are subject to uncertainties when inferring the tree topology as well as the branch lengths. Methods such as bootstrap and Bayesian sampling are thus often used to assess the reliability of the tree (Felsenstein, 2004). At a genomic level, different genes can give rise to different trees due to varying rates of evolution, genetic recombination, and ancestral polymorphism (Graur and Li, 2000; Nei, 1987). In Minh *et al.* (2006), we introduced this issue and demonstrated it on a simple case with two four-taxon trees.

A way to incorporate the information from more than one tree is to consider the sum of PD over all trees. Let X be a taxon set and $\mathcal{T}$ a collection of m phylogenetic trees $T_1, T_2, \ldots, T_m$ connecting the taxa in X. For a taxon set S we define $PD^{\mathcal{T}}(S) = PD^{T_1}(S) + \ldots + PD^{T_m}(S)$, where $PD^{T_i}(S)$ is the PD score of the set S computed on the tree T_i. The optimization is now guided by $PD^{\mathcal{T}}(S)$. Spillner *et al.* (2008) showed that for a taxon set S, $PD^{\mathcal{T}}(S)$ is equal to "phylogenetic diversity" of S on a *split system* (Bandelt and Dress, 1992a) formed of all splits existing in at least one tree in $\mathcal{T}$. Each split weight is assigned to the sum of the corresponding branch lengths of the trees in $\mathcal{T}$. We call the diversity measure on split systems *split diversity (SD)* (see Chapter 4 for the detailed transformation). Under split diversity, Problem 1 is now restated as:

> **Problem 4.** (Taxon selection with *SD*) For a given set X of taxa and a split system of X, find k taxa which maximizes the *SD* over all taxon subsets of size k.

Problem 4 opens some interesting computational challenges. Spillner *et al.* (2008) have shown that this problem is in fact NP-hard when $\mathcal{T}$ contains more than two trees. For two trees Bordewich *et al.* (2008) have recently shown that Problem 4 can be solved in polynomial time by reducing it to the minimum-cost maximum-flow network problem (Cormen *et al.*, 2001).

Due to the computational difficulty of Problem 4, we propose an approximation in Chapter 4 by reconstructing the neighbor-net split system (Bryant, 2004) from the combined tree-distance matrices and subsequently inferring the *PD* set from this split system. The neighbor-net produces *circular split systems* (Bandelt and Dress, 1992b) in which a dynamic programming algorithm, *SDA*, ensures to obtain the optimal *PD* set. Recently, an attempt to reduce the complexity of the *SDA* algorithm was suggested in Spillner *et al.* (2008) and another efficient algorithm was further presented for affine split systems (Bryant and Dress, 2007).

Apart from maximizing the sum of PD across trees, one can more generally apply any other kinds of objective functions (see also Moulton *et al.*, 2007). One simple extension is the weighted sum of PD where each tree has a different weight regarding how much one believes in this tree. This situation is equivalent to re-scaling each tree's branch lengths with the corresponding weight and subsequently solving Problem 4 across the re-scaled trees (Bordewich *et al.*, 2008). It would be interesting to investigate how other functions are related to Problem 4.

2.4 Combination of the Scenarios

Section 2.3 introduced three possible ways to extend Problem 1 to cope adequately with real-life situations. It is then natural to combine these extensions into more complex models. Theoretically one can combine any two of the variants resulting in three combined scenarios or unify all three into the most general problem. We will go through them in this section and mention their recent computational results.

Problem 3 regards all areas with the same chance to be selected. This is in reality not always true. For example, some areas are difficult to conserve due to degradation by nearby roads, industrial factories or even human populations causing pollution, hunting down animals, or cutting down trees. Hence, the conservation effort for different areas comes at different costs. However, we are given only a limited budget. How can we divide the alloted budget to select several areas such that the total PD score is maximized? This scenario can be seen as a combination of Problem 2 and 3:

> **Problem 5.** (Budgeted reserve selection with PD) For a given set X of taxa, its phylogenetic tree, a set of areas $\mathcal{A}$, a cost function $c(A)$ that describes the expenses to conserve an area $A \in \mathcal{A}$, and a total budget B, find a subset $\mathcal{S}$ of areas so as to maximize $PD(\mathcal{S})$ subject to $\sum_{A \in \mathcal{S}} c(A) \leq B$.

Here we notice three things. First of all, Problem 5 is NP-hard since Problem 3 is a special case and already NP-hard. Secondly, using a similar technique as described before, one can also transform Problem 5 into an ILP. Finally, the type of Problem 5 was mentioned several times in the literature (Church *et al.*, 1996; Pardi and Goldman, 2007) but was often ignored among conservation biologists (Faith and Baker, 2006; Hartmann and Steel, 2006).

Bordewich and Semple (2008) have recently described a greedy algorithm based on the cost-effective complementarity principle and proved that the resulting *PD* score will always be within a $(1-1/e)$ fraction of the optimal score. This works by scanning through all feasible subsets of three areas and greedily adding another area A_j maximizing the ratio between the *PD* complementarity of A_i given the chosen areas and $c(A_j)$. The addition A_j must of course fulfill the budget constraint. This procedure is repeated until no further area is included. The resulting collection of areas will then be compared with the optimal set of at most two areas to determine the final optimal area set.

The second extension combines Problems 2 and 4:

> **Problem 6.** (Budgeted taxon selection with *SD*) For a given set X of taxa and a split system of X, conservation cost c_s for every taxon $s \in X$ and a total budget B, find a subset S of taxa which maximizes $SD(S)$, subject to $\sum_{s \in S} c_s \leq B$.

Problem 6 is of course NP-hard as the special case Problem 4 is already NP-hard. If we apply the approximation using the neighbor-net method as described in Section 2.3.3, then an extension of the algorithm given in Chapter 4 will guarantee an optimal set S. We will present this extended algorithm in Chapter 5.

In a similar way, the combination of Problems 3 and 4 is:

> **Problem 7.** (Reserve selection with *SD*) For a given set X of taxa and a split system of X, a set of areas $\mathcal{A}$, and a number k, find a collection $\mathcal{S}$ of k areas which maximizes the $SD(\mathcal{S})$ over all collections of k areas.

Here $SD(\mathcal{S})$ is defined in a similar way to eq. (2.1):

$$SD(\mathcal{S}) = SD\left(\bigcup_{A \in \mathcal{S}} A\right). \tag{2.3}$$

The most general form is the union of all three extensions:

> **Problem 8.** (Budgeted reserve selection with *SD*) For a given set X of taxa and a split system of X, a set of areas $\mathcal{A}$, conservation cost $c(A)$ for every area $A \in \mathcal{A}$ and a total budget B, find a collection $\mathcal{S}$ of areas which maximizes the $SD(\mathcal{S})$, subject to $\sum_{A \in \mathcal{S}} c(A) \leq B$.

Problem 8 is of course NP-hard. So considerable computer science expertises are required to tackle this unifying problem.

Chapter 3

Phylogenetic Diversity within Seconds

3.1 Introduction

Recently Steel (2005) and Pardi and Goldman (2005) have shown that being greedy works if one is interested in selecting k taxa from a phylogenetic tree that maximize the phylogenetic diversity. The term *phylogenetic diversity* (PD) was coined by Faith (1992) to provide an effective measure of the diversity of a group of taxa. The optimal PD describes the amount of diversity embraced by a properly chosen subset of taxa. Faith (1992) applied PD to place conservation priorities on different taxa, where the taxa to protect reflect a certain value of taxonomic diversity. Thus, some measurable indicator of biodiversity defined on different scales (taxa, group of taxa, ecosystems etc.) is assigned to the corresponding systematic categories. With the advent of molecular genetics, evolutionary divergence on the genomic level may also serve this purpose (Pardi and Goldman, 2005).

For the following, the precise nature of the measure of phylogenetic diversity is not relevant (cf. Humphries *et al.*, 1995; Williams and Araujo, 2002, for a discussion on diversity measures). Phylogenetic diversity should simply describe the overall value of a group of taxa either in terms of genetic diversity, regional diversity, or social diversity. Moreover, it is required that these measures can be mapped onto a phylogenetic tree in a way that the branches of the tree receive non-negative weights.

The problem is then as follows: From a tree with n taxa one wants to identify k taxa that retain the maximal phylogenetic diversity, therefore taking into account the fact that due to restricted resources only a certain percentage of the taxa can be sustained.

Steel (2005) and independently Pardi and Goldman (2005) have proven that a greedy approach yields the optimal set with respect to PD. The greedy strategy repeatedly selects the taxon that adds the most divergence to the already chosen set of taxa. The procedure is repeated until k taxa are found. Both proofs apply – directly or indirectly – the theory of weighted matroids and greedy algorithms (Korte *et al.*, 1991). From this theory it follows that an algorithm with time complexity $O(n \log n)$ is possible.

In the following, we will suggest a time efficient *greedy phylogenetic diversity algorithm* ($gPDA$). Moreover, a different but easier to implement algorithm, the *pruning phylogenetic diversity algorithm* ($pPDA$) will be introduced. Both algorithms compute the optimal k-set for large phylogenies within seconds.

3.2 Notation

Following Steel (2005) we call $\mathcal{T}$ an unrooted phylogenetic X-tree, that is, a tree with leaf set X of taxa and whose remaining interior nodes are of degree at least three. $\mathcal{V}$ denotes the set of all nodes of $\mathcal{T}$ and $\mathcal{E}$ the collection of edges or branches. λ denotes the edge-weight function that assigns to each edge $e = (v, w)$, $(v, w \in \mathcal{V})$ of $\mathcal{T}$ a (non-negative) branch length $\lambda(v, w) \geq 0$.

A path $\mathcal{P}(a, b)$ denotes the collection of distinct nodes $a = v_0, v_1, \ldots, v_{m+1} = b$ in a tree such that v_i, v_{i+1} are adjacent, i.e., connected by an edge. The sum of the edge weights of all edges along the path between two nodes a and b denotes their distance $d(a, b)$ in the tree.

To describe the algorithms, it will be handy to root $\mathcal{T}$ at a node r. Then the remaining leaves are descendents from r. Thus, for each node $v \in \mathcal{V}$ the set $L_{\max}(v)$ is well defined and denotes the descendent(s) farthest away from v. For the sake of clarity we abbreviate the distance $d(v, L_{\max}(v))$ as $d_{\max}(v)$.

For a subset W of X, we consider $\mathcal{T}|W$, the induced phylogenetic W-tree, that connects all taxa in W according to $\mathcal{T}$. Finally, λ_W assigns to each edge e of $\mathcal{T}|W$, the sum of the $\lambda(e)$ values over those edges in $\mathcal{T}$ along the path that corresponds to the new edge e. The *phylogenetic diversity* of W, denoted $PD(W)$, is then

$$PD(W) = \sum_{e} \lambda_W(e),$$

where the summation is over all edges e in the tree $\mathcal{T}|W$ (see Steel, 2005).

3.3 The Time-Efficient Greedy Algorithm: $gPDA$

We briefly describe the implementation of $gPDA$. The phylogenetic tree $\mathcal{T}$ together with its weight-function and the size of k define the input of the algorithm. We want to determine the collection W of k taxa with maximal phylogenetic diversity. In the following, we describe the algorithm for trees with interior nodes of degree three. However, the implementation works for trees with finite interior node degree of at least three. $gPDA$ splits in two steps.

The *initial step* starts with the computation of the longest path in $\mathcal{T}$. This can be achieved in $O(n)$ time by applying a depth-first search (DFS) (cf. Cormen *et al.*, 2001, chap.22). The algorithm starts at an arbitrary leaf c and determines the leaf a furthest away from c in $\mathcal{T}$. It is easy to show, that a is one of the endpoints of the longest path in $\mathcal{T}$. We root the tree at a and based on this root compute for all interior nodes v_i the distance $d_{\max}(v_i)$ and the associated set $L_{\max}(v_i)$. This is again a DFS procedure, i.e., has complexity $O(n)$. Figure 3.1A displays the result of this procedure for a tree with five taxa. The longest path in the tree has distance 20. Thus, the set W is equal to $\{a, b\}$.

To extend W, we note that for each leaf c in $\mathcal{V} - W$ exactly one node v_i, $(i = 1, \ldots, m)$ in $\mathcal{P}(a, b)$ acts as ancestor, i.e. v_i is the node where the paths $\mathcal{P}(a, c)$ and $\mathcal{P}(a, b)$ split. One selects the leaf that is farthest away from its ancestor in $\mathcal{P}(a, b)$. To this end, we generate an ordered list $\mathcal{S}$ with respect to $d_{\max}$ that contains at most $k - 2$ nodes $v_1, v_2, \ldots, v_{k-2}$ from the path set $\mathcal{P}(a, b)$. In $\mathcal{S}$ the nodes are ordered in descending order according to $d_{\max}$, i.e., the following holds

$$d_{\max}(v_{i_1}) \geq d_{\max}(v_{i_2}) \geq \ldots \geq d_{\max}(v_{i_{k-2}}).$$

Before generating $\mathcal{S}$, we must update for each v_i on $\mathcal{P}(a, b)$ the set $L_{\max}(v_i)$ and $d_{\max}(v_i)$ by choosing a leaf c with maximal distance to v_i such that $\mathcal{P}(v_i, c)$ does not have an edge in common with the path $\mathcal{P}(a, b)$. For each node v_i this update can be done in constant time. If $\mathcal{P}(a, b)$ contains more than $k - 2$ nodes and $\mathcal{S}$ has already $k - 2$ elements, then a new node v from $\mathcal{P}(a, b)$ is only added to $\mathcal{S}$ if $d_{\max}(v) > d_{\max}(v_{i_{k-2}})$. The node $v_{i_{k-2}}$

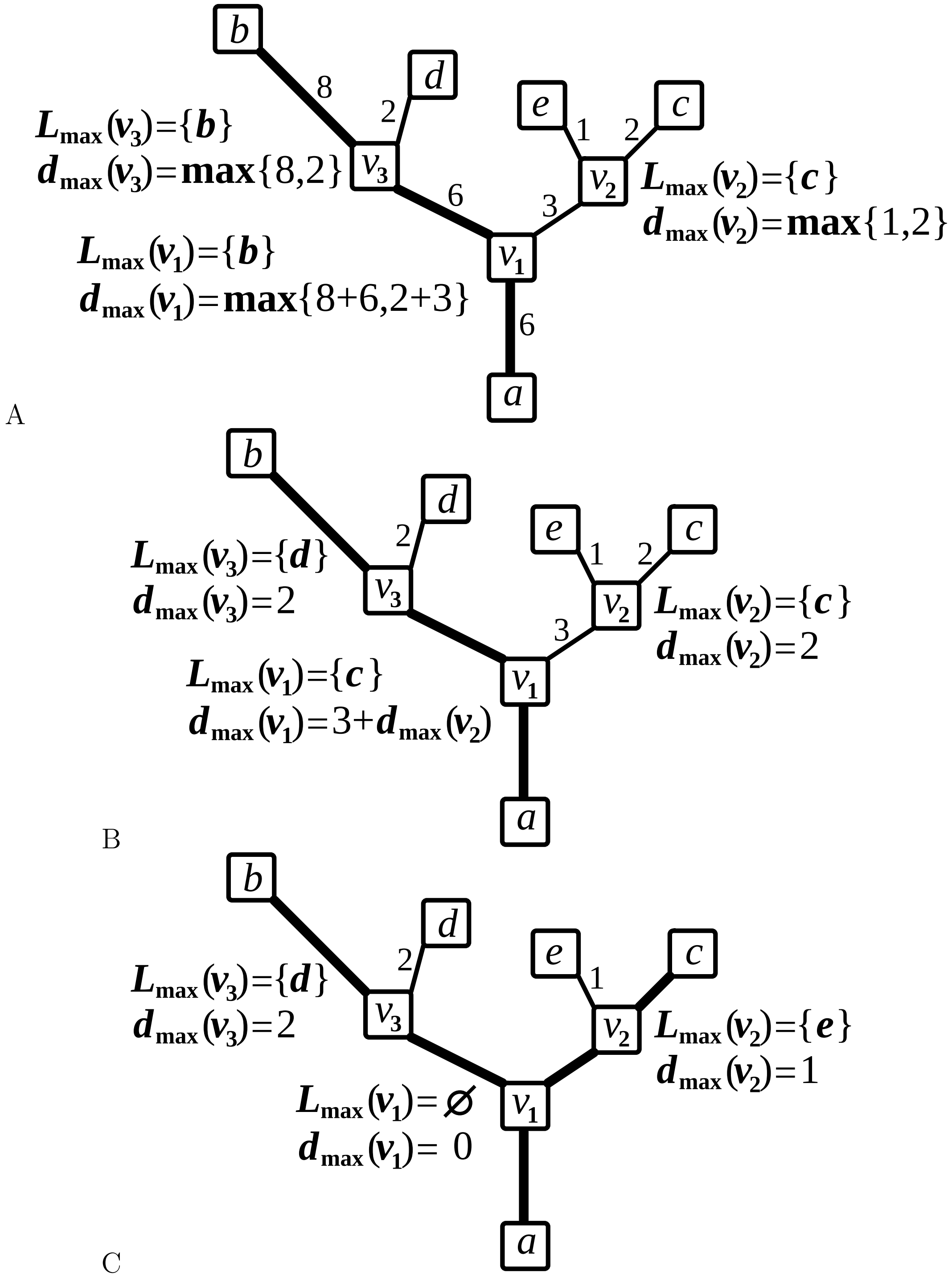


Figure 3.1: Example for the *gPDA*. $d_{\max}(v_i)$ denotes the longest distance between v_i and its descending taxa, and $L_{\max}(v_i)$ denotes the set of taxa with distance $d_{\max}(v_i)$ to v_i. (A) Result of the greedy strategy after selecting the longest path (bold lines). (B) Updating nodes on the longest path in the initial step. (C) Adding leaf c to W and updating the nodes on the partial tree.

is subsequently deleted from $\mathcal{S}$ and v is inserted at its appropriate position in $\mathcal{S}$. This step takes $O(n \log k)$ time in the worst case.

Figure 3.1B displays the result of this update for the five taxon tree. Here, we obtain $\mathcal{S} = (v_1, v_3)$, because $d_{\max}(v_1) = 5 > 2 = d_{\max}(v_3)$. This update procedure will be invoked repeatedly in the following step of $gPDA$.

Having defined W and a sorted list $\mathcal{S}$ we can enter the core of the algorithm, the *greedy step.*

We add a leaf c from $L_{\max}(v_{i_1})$ to W and delete v_{i_1} from $\mathcal{S}$. Then we update the maximal distances and leaves for all nodes on the path $\mathcal{P}(v_{i_1}, c)$ as described for the path $\mathcal{P}(a, b)$. No updates are necessary for interior nodes already in $\mathcal{S}$. Figure 3.1C illustrates this second update for the example tree with $W = \{a, b, c\}$ and $\mathcal{S} = \{v_3\}$. v_1 and v_2 are updated whereas v_3 remains unchanged.

Subsequently, the elements w of the path $\mathcal{P}(v_{i_1}, c)$ are inserted into the ordered list $\mathcal{S}$ according to their distance $d_{\max}(w)$ if $d_{\max}(w) \geq d_{\max}(v_{i_{k-2}})$. In the sample tree v_2 is added and thus $\mathcal{S} = \{v_3, v_2\}$. This completes the greedy step. The greedy step is repeated until W contains k taxa.

To determine the complexity of $gPDA$ recall that computing the longest path and identifying taxa a and b in the initial step consumes $O(n)$ time. The time requirement to generate and update $\mathcal{S}$ is more subtle to establish. Since W will eventually contain k taxa the cardinality of $\mathcal{S}$ is never larger than $k-2$. At any time, the $k-2$ nodes in $\mathcal{S}$ are the most promising for $\mathcal{T}|W$. An insertion of an interior node into $\mathcal{S}$ requires $O(\log k)$ time, because $\mathcal{S}$ is implemented as a red-black search tree data structure (e.g., Cormen *et al.*, 2001, chap. 13). Each interior node is inserted in $\mathcal{S}$ at most once during the $k-2$ greedy steps. Because a bifurcating tree with n taxa has $n-2$ interior nodes generating and updating $\mathcal{S}$ takes $O(n \log k)$ time. Therefore, the overall worst case time complexity of $gPDA$ is $O(n \log k)$.

3.4 An Efficient Pruning Algorithm: $pPDA$

Easier to implement is the *pruning phylogenetic diversity algorithm* ($pPDA$), a special application of the so-called worst-out greedy algorithm (Korte *et al.*, 1991, p. 161). Here, we start with the full tree of n taxa. Based on the length $\lambda(v, x)$ of an exterior edge leading to a leaf $x \in X$, we compute a sorted list $\mathcal{S}$ of the taxa, arranged in ascending order. This completes the *initial step* of the algorithm.

In the following $n-k$ iterations (*pruning steps*), the first taxon s_1 in the list is deleted from $\mathcal{S}$. The degree of the node v that forms the branch (v, s_1) is decreased by one. If the new degree of v equals two, then the incident edges of v are joined and the branch length of the new edge is the sum of the lengths of the joined edges. Moreover, if the new edge is connected to a leaf, the branch length of the leaf is updated. Subsequently, the leaf is put at its appropriate position in $\mathcal{S}$.

After $n-k$ pruning steps $\mathcal{S}$ contains k taxa that constitute the set W with maximal phylogenetic diversity. It is straightforward to prove that $pPDA$ provides trees with maximal phylogenetic diversity. Its optimality follows immediately from the "strong exchange" property of *PD* (Steel, 2005). This algorithm is so simple that it can be carried out on a piece of paper. We conclude the section with the discussion of its complexity.

At each pruning step at most one taxon must be repositioned in $\mathcal{S}$. We also note that the new position of the taxon is always further down the sorted list, because the length of an incident branch always increases. Thus to complete $n-k$ pruning steps, $\mathcal{S}$ needs to store in the worst case $2(n-k)$ taxa. Therefore in the initial step, the selection of those taxa can be done in $O(n)$ time (e.g., Cormen *et al.*, 2001, chap. 10). Then we only have to sort the selected taxa in $O((n-k)\log(n-k))$ time, because $\mathcal{S}$ is implemented as a red-black search tree (e.g., Cormen *et al.*, 2001, chap. 13). Finally, repositioning a taxon in the pruning step needs at most $O(\log(n-k))$ steps. Thus, the complexity of the $n-k$ pruning steps amounts to $O((n-k)\log(n-k))$. This results in an overall complexity of $pPDA$ of $O(n+(n-k)\log(n-k))$.

3.5 Runtime Analysis

We conducted computer simulations to test the wall-clock computing time of $gPDA$ and $pPDA$. Simulations were performed on a 2GHz AMD Opteron 246 with 2GByte RAM. Both algorithms were so fast, that only for huge trees with more than 100,000 taxa a substantial difference in the performance was observed. Therefore we will only compare the results for $n = 100,000$ and 1,000,000 taxa, respectively. The computing times (in

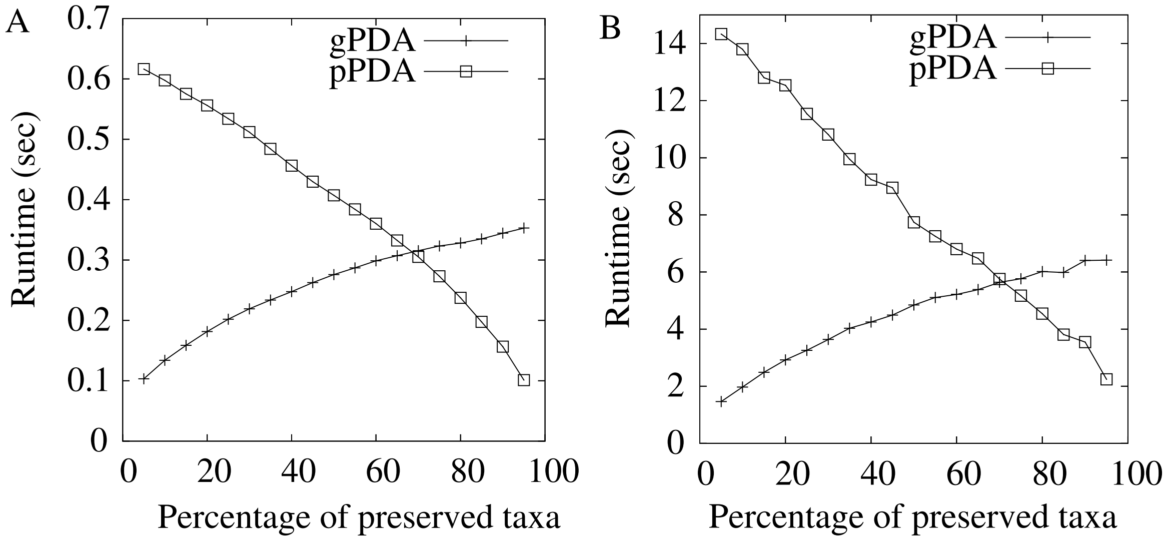


Figure 3.2: Comparison of computing times of $gPDA$ and $pPDA$. Each point represents the average runtime from 100 runs for $n = 100,000$ (A) and $n = 1,000,000$ taxa (B), respectively. Subset sizes ranging from $k = 5\% \cdot n, \ldots, 95\% \cdot n$.

seconds) in Figure 3.2 are based on average times from 100 random trees generated under the Yule-Harding model (Harding, 1971) for each combination of the pair (n, k). The branch lengths are randomly drawn from the interval $(0, 1)$. The size k of W was varied from 5% to 95% of the n taxa in the tree.

For the $n = 10^5$ taxa tree all runs of both algorithms needed less than one second to compute a subtree with maximal PD. In our simulations, $gPDA$ never consumed more than 8 seconds to achieve the subset of maximal phylogenetic diversity in the one million taxa trees, while the longest run for the one million taxa tree with $pPDA$ amounts to 17 seconds. It should be noted that an implementation of the naïve version of the greedy algorithm (as derived from Steel, 2005) needs more than 30 minutes for $n = 10^5$ taxa (data not shown). In our simulations $gPDA$ is faster than $pPDA$ if $k \leq 70\%$ of the taxa, otherwise $pPDA$ outperforms $gPDA$.

Typical applications do not deal with millions of taxa. But recently, Lewis and Lewis (2005) calculated PD for thousands of small trees of 150 taxa. We applied our algorithms to 10,000 trees generated from their data using MrBayes (Ronquist and Huelsenbeck, 2003). Both algorithms took less than 1.5 seconds to extract optimal PD subtrees for all generated trees. Hence, $gPDA$ and $pPDA$ may serve as subroutines in such applications.

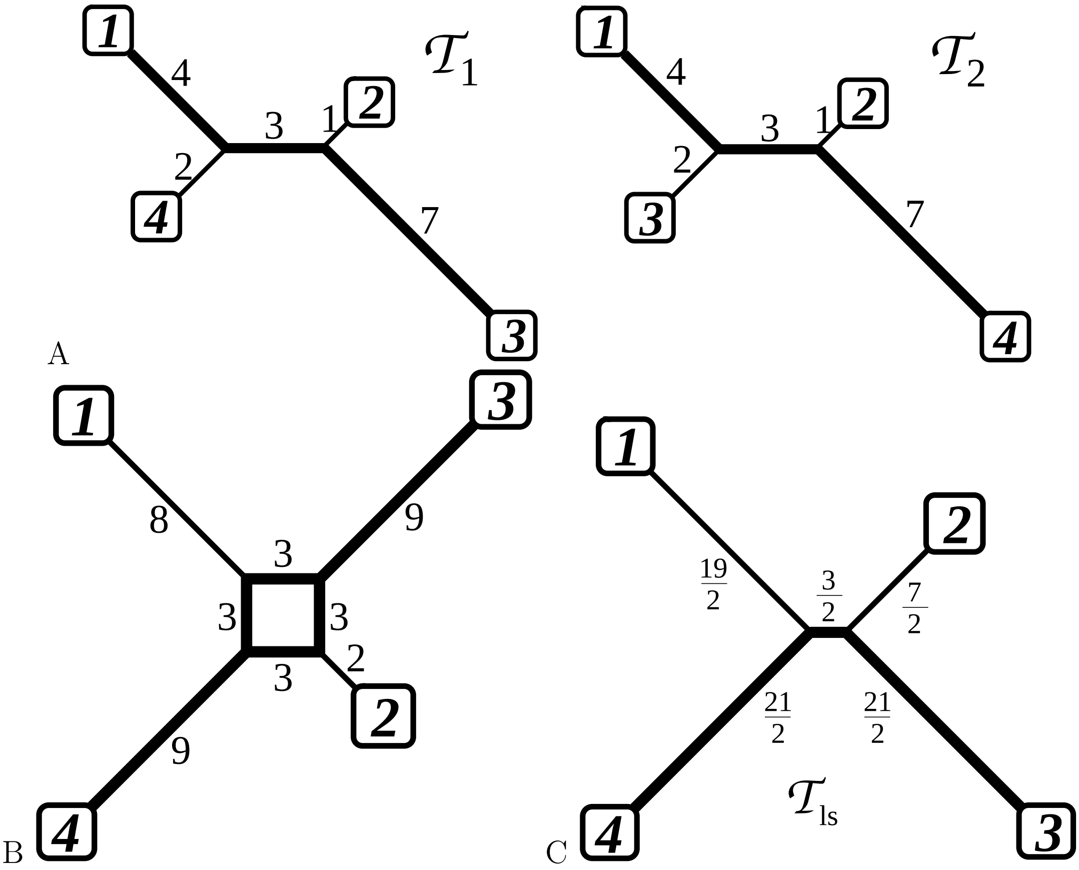


Figure 3.3: For the taxa $1, 2, 3$ and 4 two different gene trees are observed, that lead to two different PD_2 sets $\{1, 3\}$ and $\{1, 4\}$, respectively (A). In contrast, the resulting split graph generated by the sum of pairwise distances between taxa in $\mathcal{T}_1$ and $\mathcal{T}_2$ (B) or the least square fit tree (C) have the PD_2 set $\{3, 4\}$. Bold lines visualize the subgraphs formed by the respective PD_2 sets.

In addition, this example resulted in a different discriminative point of $k = 40\%$ at which $pPDA$ starts outperforming $gPDA$. Thus the superiority of one algorithm over the other crucially depends on the tree shape.

3.6 Discussion

We have presented two versions of the greedy approach, $gPDA$ and $pPDA$. They provide an efficient implementation to compute a subtree of given size k with maximal phylogenetic diversity. Thus $gPDA$ and $pPDA$ may serve as convenient tools to compute subtrees for different sizes of k. The gain in speed is due to the trick that $\mathcal{S}$ does not contain all the interior nodes or taxa. Therefore both algorithms exhibit a worst case performance less than $O(n \log n)$. Our simulations indicated that the tree shape influences the wall clock computing time and the efficiency of the algorithms differently.

Steel (2005) proposed an extension of the *PD* score to accommodate the need to incorporate different measures of diversity, in which each taxon receives a weight depicting its estimated importance. This can be easily integrated into the algorithm by increasing the terminal branches with the weight of the corresponding taxa (Pardi and Goldman, 2005).

Pardi and Goldman also suggested another approach, namely to start with a user defined initial set W. This permits the extension of W to a maximally divergent set starting with a non-optimal seed W. This application may be handy in comparative genomics where one has already some species sequenced and must decide which species to be sequenced next. We included this option in both algorithms.

While the determination of one subset W of X with maximal *PD* is computationally efficient, it would be certainly worthwhile to explore the possibility of different sets $W_1, W_2, \ldots$ with the same maximal *PD*. The number of such sets can theoretically increase dramatically. In view of this theoretically combinatorial explosion the question how to measure phylogenetic diversity becomes important. For the algorithms the precise nature of this measure is irrelevant as long as it can be mapped on the tree relating the taxa under consideration. Combining different measures of diversity may lead to more discriminative branch lengths and therefore reduce the hazard of multiple optimal sets.

In this context confining the measure to genetic distances between the taxa may be helpful (Pardi and Goldman, 2005). However, then different problems arise. Presently, it is not at all clear how to adjust the algorithms for conflicting trees derived from the same set of taxa. It is well known, that different regions of the genome provide trees with drastically different phylogenetic diversities due to violations of the molecular clock or due to varying rates of molecular evolution (Graur and Li, 2000). Sometimes trees derived from different regions may be different due to ancestral polymorphisms (Nei, 1987, pp. 288). The artificial example in Figure 3.3 illustrates the problem. For $k = 2$ we compute $W_1 = \{1, 3\}$, $W_2 = \{1, 4\}$ for trees $\mathcal{T}_1$ and $\mathcal{T}_2$, respectively. If we compute the pairwise distance between taxa as the sum of the pairwise distances in both trees, then the set $W_3 = \{3, 4\}$ displays the largest “cumulative” phylogenetic diversity. We also obtain W_3, if the tree is selected that provides the best least square fit to the distance sum (cf. Felsenstein, 2004, p. 148-153). The crucial point is the fact that W_3 is neither maximal in T_1 nor in T_2. Thus if we construct trees from different genomic regions and combine them naïvely, then the resulting tree and its derived optimal subtree with maximal diversity may not be the representative of the true underlying diversity. One way to address this would be to assign different weights to the different trees and then maximize the weighted average of the PDs calculated for different trees. In the subsequent chapters we will present more sophisticated algorithms to address this issue.

Chapter 4

Taxon Selection under Split Diversity

4.1 Introduction

Practical biodiversity conservation normally focuses on preserving as many species as possible. This is known as the *species richness* concept (Wilson, 1997; Gaston and Spicer, 2004). Despite being widely used due to its easy application, such an approach poses the major problem of treating all species equally (May, 1990). This is not adequate in some respects. For example, “Is the panda equivalent to one species of rat?” (Vane-Wright *et al.*, 1991). Consequently, Vane-Wright *et al.* (1991) suggested the so-called *taxic diversity* that exclusively uses a taxonomic tree connecting the species under consideration for diversity evaluation. Faith (1992) further extended this approach by taking the branch lengths of the tree into account, and introduced *phylogenetic diversity (PD)* in the context of “feature” diversity. Given a set of features (attributes) where each taxon exposes a number of features. The feature diversity of a set of taxa is the number of features that are represented by at least one taxon in the set. Assume that the set of features could be mapped perfectly onto a rooted phylogenetic tree where the branch length depicts the number of features uniquely shared by all descending taxa below this branch. Then the feature diversity (or phylogenetic diversity) can be alternatively computed as the sum of the branch lengths of the minimal subtree connecting the taxa of interest with the root. Faith (1992) also pointed out that other measures of branch lengths could be used too, for example, evolutionary distances estimated from molecular data.

Using *PD*, Faith proposed a taxon selection problem. Given a phylogenetic tree of n

taxa, identify the set of k taxa which maximizes the *PD* where $k < n$. Such an optimal set could be employed to identify taxa important for conservation or to prioritize taxa under sequencing projects (Pardi and Goldman, 2005). Recently, Steel (2005) and independently Pardi and Goldman (2005) proved that a greedy algorithm is sufficient to determine an optimal set of a given size k on a phylogenetic tree. Minh *et al.* (2006) presented an efficient implementation, the greedy phylogenetic diversity algorithm (*gPDA*), capable of handling trees with millions of taxa.

One limitation of *PD* acknowledged by Faith is that "the predictive value of PD depends on having a cladogram that is a reliable estimate of the phylogenetic relationships among the taxa" (Faith, 1992, p. 8-9). However, such a reliable estimate of the phylogenetic tree is in many cases difficult to obtain due to a number of reasons:

(i) At the genetic level, tree reconstruction methods often face statistical uncertainties.

(ii) At the genomic level, it is well known that different regions of the genome provide trees with different genetic distances between taxa due to violations of the molecular clock or due to varying rates of molecular evolution (e.g., Graur and Li, 2000).

(iii) Sometimes, different regions of the genome lead to distinct trees due to ancestral polymorphisms (e.g., Nei, 1987).

The issue of (possibly conflicting) different trees from the same set of taxa has recently been formalized in Minh *et al.* (2006). Given a collection of trees connecting n taxa of interest and a weight for each tree (e.g., the importance of the tree), one needs to identify a subset of k taxa with the maximal weighted average of the phylogenetic diversity calculated for each tree. The *PD* computed in such a way has been proven to be equivalent to computing diversity on a *split system* (Bandelt and Dress, 1992a) formed of all splits existing in at least one tree (Spillner *et al.*, 2008). Split systems generalize phylogenetic trees by allowing for conflicting phylogenetic signals (see the next section for the definition). We call the diversity based on a split system *split diversity (SD)*. A formal definition of *SD* is given in the next section.

The taxon selection with *SD* is not as easy as with *PD* because the simple greedy algorithm no longer guarantees a set of k taxa with maximal *SD* on split systems (Minh *et al.*, 2006; Moulton *et al.*, 2007). In fact, deriving an optimal k-set from arbitrary split systems is NP-hard (Spillner *et al.*, 2008). Here, we present a dynamic programming algorithm (*SDA*) that computes the optimal *SD* set when the underlying split system is *circular* (Bandelt and Dress, 1992a). Circular split systems are reconstructed by the neighbor-net method (Bryant and Moulton, 2004) which has been applied in various phylogenetic analyses (e.g., Sullivan *et al.*, 2006; Henz *et al.*, 2005; de las Rivas *et al.*, 2004; Hertel *et al.*, 2006).

As an illustration we conduct two case studies, on a freshwater crayfish *Euastacus* dataset (Shull *et al.*, 2005) and a bacteria dataset (Sullivan *et al.*, 2006). We compare and contrast the optimal taxon sets when inferred from the neighbor-net split system using *SD* rather than from the tree derived from the same dataset with *PD*. The result of such a comparison may serve as an additional source of information regarding the selection of taxa for conservation.

4.2 Split Systems

This section provides a brief introduction to the concept of split systems. For detailed definitions we refer to Huber and Moulton (2005) or Huson and Bryant (2006). We will use the general term “taxon” that, depending on the question, can be interpreted as species, genus, population, etc.

Let X denote a finite set of n taxa. A *split* $A|B$ is a bipartition of the taxon set, i.e., $A, B \neq \emptyset$, $A \cap B = \emptyset$, and $A \cup B = X$. Two splits $A|B$ and $C|D$ are *compatible* if one of the following intersections is empty: $A \cap C$, $A \cap D$, $B \cap C$, or $B \cap D$. A *phylogenetic tree* connecting X consists of at most $2n-3$ pairwise compatible splits. Every split $A|B$ corresponds to a branch or edge in the tree whose removal will separate A from B. A *split system* Σ is simply a collection of splits of X. The split system Σ is called *weighted* if every split $A|B \in \Sigma$ is assigned a weight $\lambda(A|B)$. Split systems are visualized as *splits graph*, where each split is represented by one edge or several parallel edges. Fig. 4.2 depicts an example of a splits graph for five taxa.

Of particular interest are *circular* split systems. A split system is called circular if there exists a way to enumerate the taxa from 1 to n such that all splits are of the form $\{i, i+1, \ldots, j\} \mid X \setminus \{i, i+1, \ldots, j\}$, $1 \leq i \leq j \leq n$ (Bandelt and Dress, 1992a). $(1, 2, \ldots, n)$ is called the *circular order* of the taxa. Circular split systems can be represented by the so-called *outer-labeled plane splits graphs* (Dress and Huson,

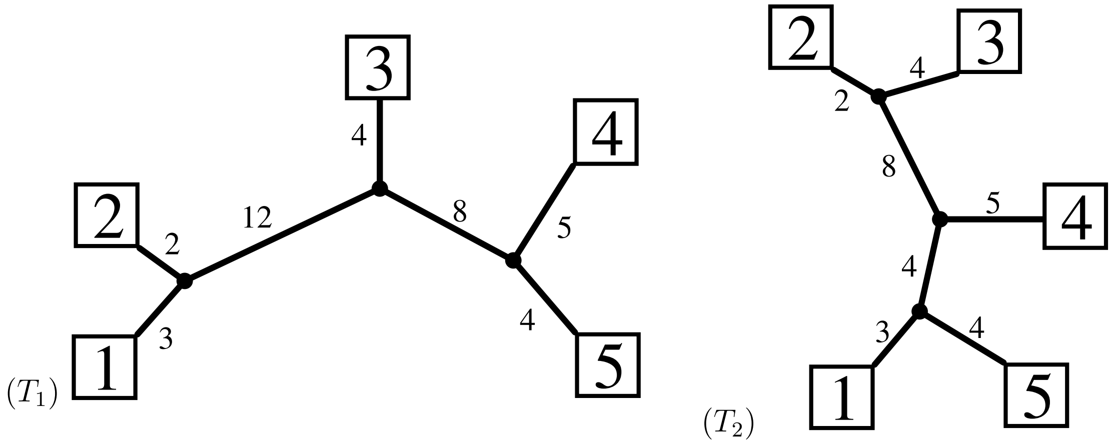


Figure 4.1: Two gene trees with species $1, 2, 3, 4, 5$. Based on tree T_1, PD_2 contains the single taxon set $\{1, 4\}$ and $PD_3 = \{\{1, 3, 4\}, \{1, 4, 5\}\}$. However, the tree T_2 gives an alternative suggestion: $\{3, 5\}$ for PD_2 and $\{3, 4, 5\}$ for PD_3.

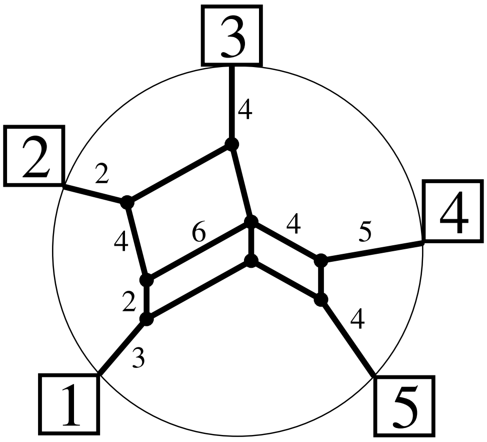
2004). The graph in Fig. 4.2 has a circular taxon order $(1, 2, 3, 4, 5)$. In the graph representation one can draw a circle passing through the taxa in that order, and each split can be depicted by a line bisecting the circle. Note that trees are a special case of circular split systems (Semple and Steel, 2003, Thm. 3.7.1).

4.3 A Measure of Split Diversity

Given an (unrooted) phylogenetic tree, the phylogenetic diversity of a taxon subset $S \subset X$, denoted by $\mathrm{pd}(S)$, is defined as the sum of the lengths of those branches connecting the taxa in S. Using the one-to-one relation between branches and splits in a tree, $\mathrm{pd}(S)$ is equivalently restated as the sum of the weights of those splits separating the taxa of S. This reformulation naturally extends to split systems. Formally, given a weighted split system Σ with a split-weight function λ, the split diversity of a taxon set S, $\mathrm{sd}(S)$, is given by:

$$\mathrm{sd}(S) = \sum_{\substack{A|B \in \Sigma: \\ A \cap S \neq \emptyset \\ B \cap S \neq \emptyset}} \lambda(A|B). \tag{4.1}$$

This definition of split diversity coincides with the concept of feature diversity (Faith, 1992). Each feature can be assigned to a split $A|B$ by proposing that the taxa in A show



Split	Weight
1|2345	3
2|1345	2
3|1245	4
4|1235	5
5|1234	4
12|345	6
23|145	4
45|123	4
15|234	2

Figure 4.2: The splits graph which “unifies” the two trees from Figure 4.1 and its corresponding split system. The circle connecting the taxa of the graph indicates the circular order. SD_2 contains $\{2, 5\}$, and SD_3 contains only $\{1, 3, 4\}$.

the feature and the taxa in B do not. In that case the weight of the split is equal to the number of features agreeing with the split. In general, split weights are not restricted to the number of features but can be infered by any distance measure. Further, with this approach we relax the assumption made by Faith that the set of splits are mapped onto a tree excluding incompatible splits.

Consider the two incongruent trees T_1, T_2 of five taxa depicted in Figure 4.1. The taxon selection with phylogenetic diversity on either T_1 or T_2 return different optimal taxon sets. How does one evaluate this result?

A simple solution is to maximize the average of the phylogenetic diversity computed from each tree, i.e., set S will be assigned a score of $\frac{1}{2}\left(\mathrm{pd}^{T_1}(S) + \mathrm{pd}^{T_2}(S)\right)$ (Minh *et al.*, 2006). This problem is equivalent to maximizing split diversity on the split system in Figure 4.2 (Spillner *et al.*, 2008). This split system summarizes T_1 and T_2 by including all splits from the two trees and assigning to each split $A|B$ a weight of $\lambda(A|B) = \frac{1}{2}(\lambda_1(A|B) + \lambda_2(A|B))$, where λ_1 and λ_2 are the split-weight functions of T_1 and T_2, respectively. If a split does not appear in a tree, its weight in this tree is equal to 0. This particular example demonstrates that one interpretation of split systems is to represent multiple trees (Huson and Bryant, 2006).

Note again that the features exclusively observed in the taxa of A but not in the taxa of B are mapped to the split $A|B$. On the other hand, there can be features exclusively observed in the taxa of B. Since $A|B$ and $B|A$ represent the same split, introducing an "outgroup" taxon ρ that has no features will help distinguish these two groups of features. The features exclusively observed in the taxa of A are assigned to $A|B \cup \{\rho\}$ whereas the features exclusively observed in the taxa of B are $A \cup \{\rho\}|B$. Thus for the computation of $\mathrm{sd}(S)$, one should include ρ in S. This treatment coincides with the original definition of phylogenetic diversity (Faith, 1992) which requires a rooted tree and includes the length of the branches connecting the taxa in S with the root. Split systems are unrooted by definition. Hence, if converting a set of rooted trees into a split system, one should add an outgroup taxon ρ corresponding to the root. In the resulting split system ρ is always included in the set S. Thus the shared evolutionary history of the taxa in S contributes to $\mathrm{sd}(S)$.

4.3.1 Taxon Selection under Split Diversity

As with phylogenetic diversity one is interested in selecting a subset of k taxa which maximizes split diversity on a given weighted split system. To this end, we introduce the maximal split diversity

$$\mathrm{sd}_{\max}(k) = \max_{S \subseteq X, |S|=k} \mathrm{sd}(S)$$

and the collection of all maximal k-sets:

$$SD_k = \{S \subseteq X, |S| = k : \mathrm{sd}(S) = \mathrm{sd}_{\max}(k)\}.$$

If the split system corresponds to a tree, one employs a greedy strategy to obtain $\mathrm{sd}_{\max}(k)$ and an element of SD_k (Steel, 2005; Pardi and Goldman, 2005). The greedy algorithm works by determining an optimal set of two taxa and sequentially adding $k-2$ taxa which contribute the most divergence to the already chosen set. However, if the split system does not represent a tree, the greedy algorithm no longer guarantees an optimal solution (Minh *et al.*, 2006; Moulton *et al.*, 2007). For example, the split system in Figure 4.2 has $\{2, 5\}$ as the only element of SD_2 and $\{1, 3, 4\}$ is the only element of SD_3. As already noted, the taxon selection with split diversity on general split systems falls into the class of NP-hard problems (Spillner *et al.*, 2008).

4.3.2 Computing Split Diversity for Circular Split Systems

For simplicity we define the split diversity of the set of two taxa $\{u, v\}$ the *split-distance* between them, denoted by d_{uv}:

$$d_{uv} = \mathrm{sd}(\{u, v\}) = \sum_{\substack{A|B \in \Sigma \\ u \in A, v \in B \text{ or} \\ v \in A, u \in B}} \lambda(A|B). \tag{4.2}$$

Based on split-distances, a key property of circular split systems with the circular taxon order $(1, 2, \ldots, n)$ is that for any subset $S = \{s_1, s_2, \ldots, s_k\} \subset X$ where $s_1 < s_2 < \ldots < s_k$, $\mathrm{sd}(S)$ can be alternatively computed employing a *circular tour* (Korostensky and Gonnet, 2000). A circular tour visits every taxon $1, 2, \ldots, n$ and returns to taxon 1 while taking the shortest path connecting taxon i and taxon $i + 1$ in the split system. Since each split bisects the circle, a circular tour traverses each split exactly twice. Thus the sum of the weights of all edges encountered during a circular tour equals twice the sum of the weight of all splits. Since circularity is retained for subsets of circular split systems, we have

$$\mathrm{sd}(S) = \frac{1}{2}\left(d_{s_1 s_k} + \sum_{i=1}^{k-1} d_{s_i s_{i+1}}\right). \tag{4.3}$$

4.4 *SDA*: An Efficient Algorithm to Obtain an Element of SD_k for Circular Split Systems

We introduce an efficient algorithm to select a set of k taxa that maximizes the split diversity over all possible sets of k taxa in a circular split system Σ. We then illustrate the algorithm with a small example and show a modification to the algorithm when Σ has an outgroup taxon.

Eq. (4.3) permits a direct computation of $\mathrm{sd}(S)$ for any taxon set S from a split-distance matrix without considering the detailed structure of the underlying splits graph. Based on this observation, the computation of an element of SD_k reduces to the following task:

> *Given n taxa indexed by the circular order $(1, 2, \ldots, n)$ and pairwise split-distances (d_{uv}) for all $u, v \in \{1, 2, \ldots, n\}$. Find the* **longest circular k-tour***, that is the longest circular tour with k taxa.*

For the description of the algorithm we introduce the following notations. An *ordered k-path* from a taxon u to a taxon v is a sequence of k taxa $(u = s_1, s_2, \dots, s_k = v)$ which follow the circular order, $s_1 < s_2 < \cdots < s_k$. Let $L(s_1, s_2, \dots, s_k) = \sum_{i=1}^{k-1} d_{s_i s_{i+1}}$ denote length of the ordered k-path $(s_1, s_2, \dots, s_k)$. For two taxa $u < v$, let L^k_{uv} denote the length of the longest ordered k-path from u to v,

$$L^k_{uv} = \max_{u<s_2<\cdots<s_{k-1}<v} L(u, s_2, \dots, s_{k-1}, v).$$

It is worth noting that a circular k-tour is attained if we add the starting taxon $s_{k+1} = s_1$. Therefore, every circular k-tour is uniquely represented by an ordered k-path and vice versa. We will now present a method to obtain $\mathrm{sd}_{\max}(k)$ and an element of SD_k by computing all entries of L^i_{uv} for $i = 2, \dots, k$.

The key property of the algorithm is that if $(s_1, s_2, \dots, s_k, s_1)$ is the longest circular k-tour then $(s_1, s_2, \dots, s_k)$ is the longest ordered k-path from s_1 to s_k. It then follows that $(s_1, s_2, \dots, s_{k-1})$ is the longest ordered $(k-1)$-path from s_1 to s_{k-1}. Generally, $(s_1, \dots, s_i)$ for $i = 2, \dots, k$ is the longest ordered i-path between s_1 and s_i. Proofs of these propositions are provided in the appendix 4.7.1. The problem exhibits an *optimal sub-structure* (Cormen *et al.*, 2001) for which a *dynamic programming* technique is applicable. As a result, the length ℓ^k_{max} of the longest circular k-tour will be obtained by solving the following iterative maximization:

$$L^i_{uv} = \begin{cases} d_{uv}, & \text{if } i = 2, \\ \max_{u<s<v}\{L^{i-1}_{us} + d_{sv}\}, & \text{if } 3 \le i \le k, \end{cases} \tag{4.4}$$

$$\ell^k_{max} = \max_{1\le u<v\le n} \{L^k_{uv} + d_{uv}\}. \tag{4.5}$$

To resolve these equations we first compute $L^2_{uv}, L^3_{uv}, \dots, L^k_{uv}$ for all pairs of taxa $u < v$ by eq. (4.4), and then calculate ℓ^k_{max} using eq. (4.5).

Based on eq. (4.3), the optimal score is $\mathrm{sd}_{\max}(k) = \ell^k_{max}/2$. To construct an element $S \in SD_k$, we trace back the two taxa u and v which maximize the sum on the right-hand side of eq. (4.5) and then the taxon s from eq. (4.4) with decreasing $i = k, \dots, 3$. In the appendix 4.7.2 we show that the computational complexity of the *SDA* algorithm is $O(kn^3)$.

4.4.1 An Example

Let us consider the circular split system in Figure 4.2 with the circular taxon order of $(1, 2, 3, 4, 5)$. We will construct an optimal 3-set. Eq. (4.2) leads to the pairwise split-distance matrix:

$$(d_{uv}) = \begin{pmatrix} 0 & 11 & 19 & 20 & 17 \\ 11 & 0 & 12 & 21 & 22 \\ 19 & 12 & 0 & 17 & 18 \\ 20 & 21 & 17 & 0 & 11 \\ 17 & 22 & 18 & 11 & 0 \end{pmatrix}$$

With eq. (4.4) the length of the longest ordered 2-path L^2_{uv} equals d_{uv} and therefore:

$$(L^2_{uv}) = \begin{pmatrix} - & 11 & 19 & 20 & 17 \\ & - & 12 & 21 & 22 \\ & & - & 17 & 18 \\ & & & - & 11 \\ & & & & - \end{pmatrix}$$

From L^2_{uv} we derive L^3_{uv} as described in eq. (4.4):

$$(L^3_{uv}) = \begin{pmatrix} - & - & 23 & 36 & 37 \\ & - & - & 29 & 32 \\ & & - & - & 28 \\ & & & - & - \\ & & & & - \end{pmatrix}$$

where the secondary diagonal entries are omitted since there is no ordered 3-path between two neighboring taxa. To trace back the optimal 3-path, we define the index matrix (α^3_{uv}):

$$(\alpha^3_{uv}) = \begin{pmatrix} - & - & 2 & 3 & 3 \\ & - & - & 3 & 4 \\ & & - & - & 4 \\ & & & - & - \\ & & & & - \end{pmatrix},$$

where α^i_{uv} denotes the next to last taxon on the longest ordered i-path between taxon u and v. Thus the longest ordered 3-path from taxon 1 to taxon 3 contains taxon 2, and the longest ordered 3-path from taxon 2 to taxon 5 contains taxon 4, etc.

Finally we calculate the lengths of all longest circular 3-tours by eq. (4.5):

$$(L^3_{uv} + d_{uv}) = \begin{pmatrix} - & - & 42 & 56 & 54 \\ & - & - & 50 & 54 \\ & & - & - & 46 \\ & & & - & - \\ & & & & - \end{pmatrix}.$$

From a maximal entry of this matrix we construct an optimal 3-set. The maximal score is $\mathrm{sd}_{\max}(3) = 56/2 = 28$. The taxa 1 and 4 span the longest circular 3-tour. The stored index α^3_{14} indicates that taxon 3 is on the longest ordered 3-path from 1 to 4. Therefore, the set $\{1, 3, 4\}$ is one element of SD_3 and has an SD score of 28.

4.4.2 Modification for Split Systems with an Outgroup

The above approach will not include the evolutionary history shared by a taxon set. As proposed before, the introduction of an outgroup taxon will help solve this problem. The following modification of *SDA* will ensure its inclusion in the optimal set.

We simply label the outgroup as taxon 1. Then we re-index the remaining taxa according to the circular order of the underlying split system. The computation of $\mathrm{sd}_{\max}(k)$ and an element of SD_k containing the outgroup is accomplished by considering only the ordered k-paths starting at taxon 1. The algorithm proceeds in the same way as before by fixing $u = 1$ in eq.s (4.4) and (4.5). The computational complexity is thus reduced to $O(kn^2)$ (see the appendix 4.7.2).

4.5 Case Studies

In the following we illustrate the proposed method with two datasets. The first dataset contains four different genes from the freshwater crayfish (Shull *et al.*, 2005). The second dataset consists of one gene from marine cyanobacteria and cyanophages where horizontal gene transfer and recombination are detected (Sullivan *et al.*, 2006).

4.5.1 Freshwater Crayfish of Australia

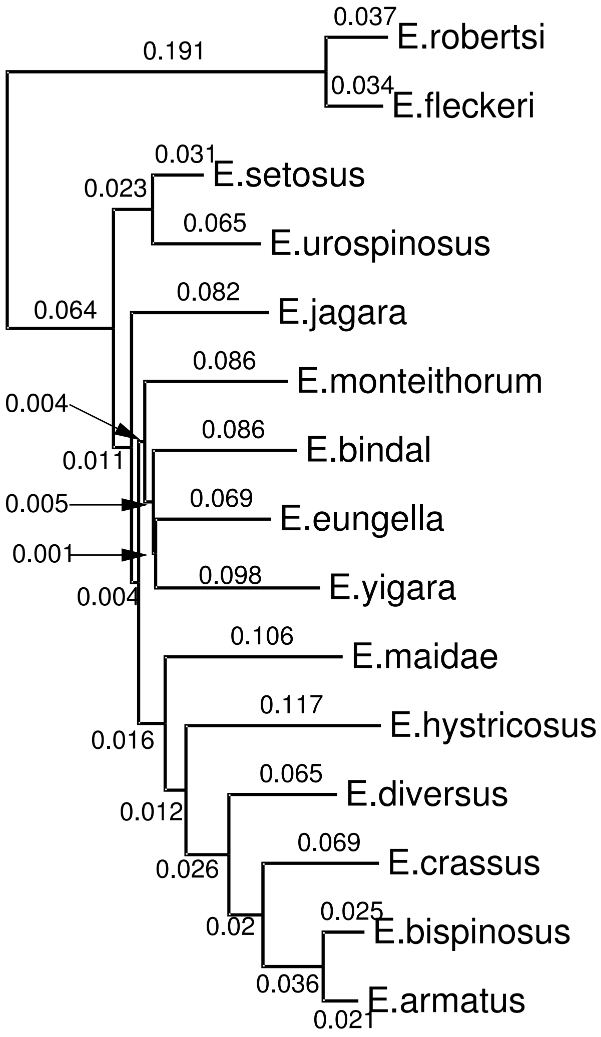


Figure 4.3: The maximum likelihood tree of the 15 threatened crayfish *Euastacus*. The tree is rooted on the branch leading to *E.robertsi* and *E.fleckeri* as suggested (Shull *et al.*, 2005). Branch lengths depict the number of substitution per site.

Freshwater crayfish *Euastacus* of the eastern coast of Australia have been studied with respect to their phylogenetic relationship, biogeographical distribution and conservation status (Whiting *et al.*, 2000; Shull *et al.*, 2005). *Euastacus* are greatly threatened due to various human activities (Alicia Toon, personal communication). 43 *Euastacus* species have been identified so far, of which 16 species were classified as either endangered or

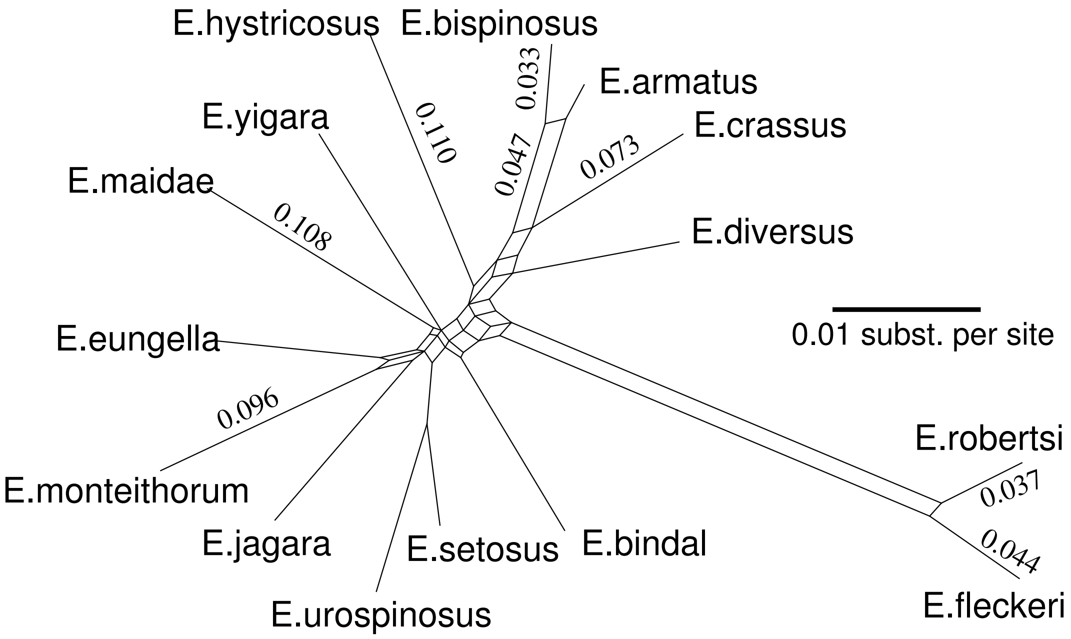


Figure 4.4: The splits graph built by Neighbor-net of 15 endangered or vulnerable crayfish *Euastacus*. Splits with weights smaller than 0.001 are excluded.

vulnerable according to the IUCN Red List (IUCN, 2001).

The conservation priorities for *Euastacus* using *PD* were solely based on the *16S* rDNA gene and a subset of 35 *Euastacus* species (Whiting *et al.*, 2000). Recently, Shull *et al.* (2005) extended the data to 40 *Euastacus* species and to four genes: the mitochondrial *16S* rDNA, *12S* rDNA, *COI* genes, and the nuclear *28S* gene. We use the data from Shull *et al.* (2005) and only focus on 15 threatened *Euastacus* species (*E.neodiversus*, the 16^{th} threatened taxon, is missing due to the lack of data). The species names, IUCN categories and sample IDs are given in Table 4.1. Our aim is to study which species should be selected according to *PD* and *SD*.

A maximum likelihood (ML) tree was reconstructed from the concatenated gene sequences using IQPNNI version 3.2 (Minh *et al.*, 2005) under the GTR+I+Γ model (Felsenstein, 2004). On the other hand, the pairwise genetic distances between sequences, computed by IQPNNI, were used to reconstruct a neighbor-net split system (NNet) (Bryant and Moulton, 2004) using SplitsTree version 4.6 (Huson and Bryant, 2006). The resulting ML tree and NNet are shown in Fig. 4.3 and 4.4, respectively. The ML tree is mainly in agreement with the tree published in Shull *et al.* (2005), whereas the NNet shows several incompatible splits. To compare the *PD* scores computed on the ML tree and the *SD* scores from the NNet, we scaled the split weights of the NNet such that the total sum of split weights equals the length of the ML tree.

Species	Status	Sample	Tree		NNet	
			Rank	pd_{max}	Rank	sd_{max}
E.robertsi	EN	2669	1	-	**14**	1.39
E.hystricosus	VU	2672	2	0.45	3	0.53
E.crassus	EN	2649	3	0.57	**10**	1.17
E.yigara	EN	2664	4	0.68	7	0.94
E.maidae	EN	2658	5	0.78	4	0.64
E.urospinosus	EN	2767	6	0.87	6	0.84
E.bindal	EN	2690	7	0.96	9	1.10
E.monteithorum	EN	2765	8	1.04	**2**	0.40
E.jagara	EN	2763	9	1.12	8	1.02
E.eungella	VU	2663	10	1.19	12	1.31
E.diversus	EN	2773	11	1.26	11	1.24
E.bispinosus	VU	0631	12	1.32	**5**	0.75
E.fleckeri	VU	2668	13	1.35	**1**	-
E.setosus	VU	2693	14	1.38	13	1.35
E.armatus	VU	2653	15	1.40	15	1.40

Table 4.1: List of 15 threatened *Euastacus* species from Shull *et al.* (2005). The second column is the IUCN red list status: EN = Endangered; VU = Vulnerable. Each species was sampled at several locations. Here we use one sample per species. The species sample IDs are depicted in the third column. Remaining columns display the taxa priorities according to the optimal *PD* sets from the ML tree and the optimal *SD* sets from the NNet. The fourth column shows the ranking based on the tree, the fifth column shows the corresponding optimal score pd_{max}. The sixth and seventh column contain the analogous values computed from the NNet. Numbers in bold-face in the sixth column indicate the *SD*-based rankings which are different by at least 6 from the corresponding *PD*-based ones.

Afterwards we applied the *gPDA* (Minh *et al.*, 2006) on the ML tree and the *SDA* algorithm on the NNet to compute the maximal k-sets for $k = 2, \ldots, 14$. First, we observe that only one single optimal set exists for each k on both the ML tree and the NNet. Moreover, since the *gPDA* is a greedy algorithm, we can rank the taxa in the order of their selection by *gPDA*. This ranking is shown in Table 4.1. We also observed that in this particular example, the optimal k-sets from *SDA* are nested with increasing k. Hence, the taxa can also be ranked (Table 4.1).

Most notable is the ranking for *E.robertsi* and *E.fleckeri*. *E.robertsi* is ranked 1^{st} by *gPDA* but 14^{th} by *SDA*. *E.fleckeri* takes 1^{st} place by *SDA* but only the 13^{th} by *gPDA*. Looking at the clade containing these taxa in the tree (Figure 4.3), we see that the length of the external branch leading to *E.robertsi* (0.037) is slightly greater than that leading to *E.fleckeri* (0.034). On the other hand, the NNet (Figure 4.4) shows the opposite. Thus selecting *E.robertsi* will exclude *E.fleckeri* and vice versa. The rankings for *E.crasus* and *E.bispinosus* can be explained similarly if one looks at the clade containing *E.crasus*, *E.bispinosus*, and *E.armatus*. On the ML tree the distances from the root of the clade to *E.crasus* and *E.bispinosus* are 0.069 and 0.061, respectively. On the NNet they are 0.073 and 0.080. *E.crasus* is therefore more preferred by the *gPDA* than the *SDA*. These observations indicate that the outcome depends strongly on variations in either branch-length or split-weight estimates.

The rankings for *E.monteithorum* are also considerably different (Table 4.1): it is 2^{nd} according to *SDA* but 8^{th} by *gPDA*. This is because the NNet displays conflicting splits separating *E.monteithorum* from the first ranked species compared to the tree. These splits, although having small weights, accumulate the larger *SD* "contribution" from *E.monteithorum* on the NNet than on the tree.

4.5.2 Cyanobacteria and Cyanophages

The two marine cyanophage genera *Prochlorococcus* and *Synechococcus* were known as horizontal gene transfer agents for the photosynthesis genes among the host cyanobacteria (Sullivan *et al.*, 2006). Specifically, the two core photosystem genes *psbA* and *psbD* are transferred from cyanobacteria to cyanophages and between phages. Moreover, intragenic recombinations among them were detected. Here the conservation of bacteria or phages is not of interest. We rather want to study what impact such incompatible

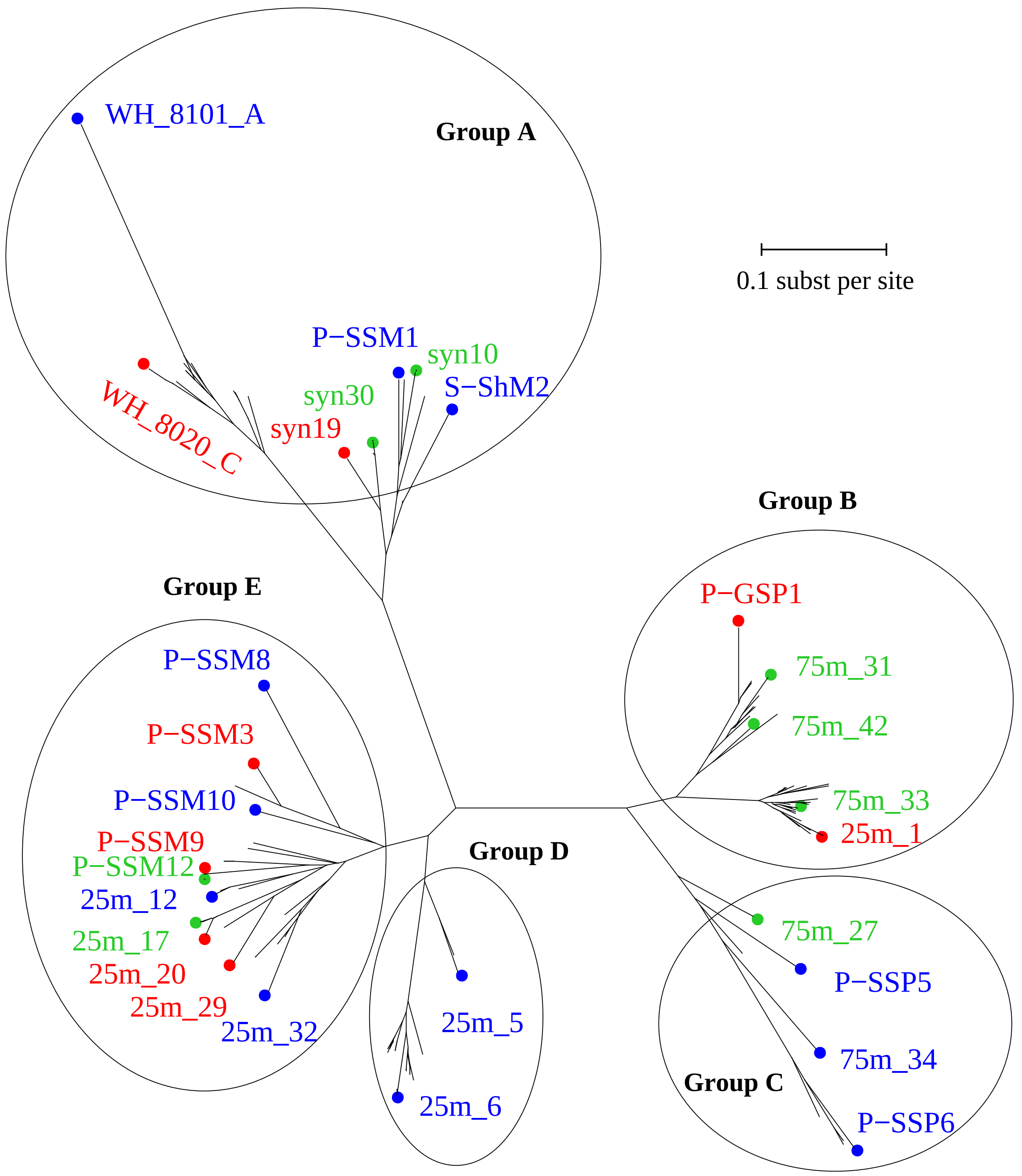


Figure 4.5: The neighbor-joining tree of the CYANO data with taxa in the union of the optimal S^{NJ} and S^{NNet} of size 20. The blue taxa appear in both sets. The red taxa occur exclusively in S^{NJ}. The green taxa are in S^{NNet} and not in S^{NJ}.

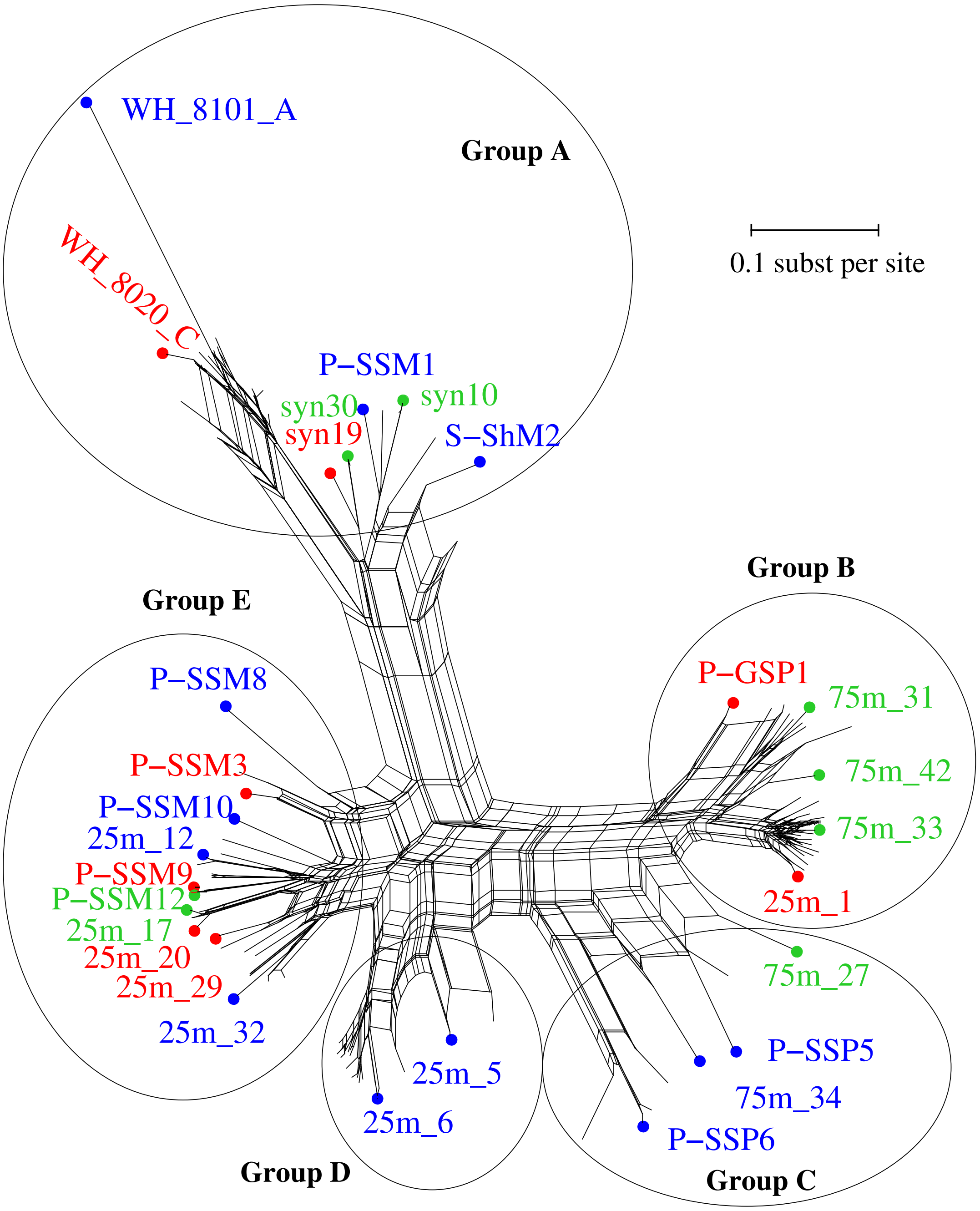


Figure 4.6: The NNet graph of the CYANO data with taxa in the union of S^{NJ} and S^{NNet}. The taxon colors are coded in the same way as in Figure 4.5.

phylogenetic signals have on the taxon selection. To this end, we use the *psbA* gene from these bacteria and phages (Sullivan *et al.*, 2006) to select taxa according to *PD* and *SD*.

A total of 112 *psbA* gene sequences were retrieved from the NCBI GenBank (Benson *et al.*, 2006). The sequences were aligned with ClustalW (Thompson *et al.*, 1994) resulting in an alignment with 729 sites. The pairwise ML distances were computed from the alignment using IQPNNI under the HKY+Γ model of substitution (Felsenstein, 2004). The substitution model parameters were estimated from the ML tree reconstructed by IQPNNI. This model was also used in Sullivan *et al.* (2006). For the resulting distance matrix we inferred the neighbor-joining (NJ) tree (Saitou and Nei, 1987) and the NNet using the SplitsTree program. Finally, we applied the *gPDA* on the NJ tree and the *SDA* on the NNet to compute the maximal S^{NJ} and S^{NNet} sets of size 20, respectively. Thus, we conserve slightly less than 20% of the taxa.

Fig. 4.5 and 4.6 show the NJ tree and the NNet restricted to the taxa that occur at least once in S^{NJ} or S^{NNet}. The blue taxa appear in both sets. The red taxa occur exclusively in S^{NJ}, whereas the green taxa occur exclusively in S^{NNet}. First of all, we notice that the structure of the NNet (Figure 4.6) shows a number of incompatible phylogenetic signals. However, the main taxon groupings of the NJ tree and the NNet do agree. The circular orders of the taxon labels are similar in both "phylogenies", except for the taxon *25m_12*. The corresponding S^{NJ} and S^{NNet} overlap in 12 taxa (core taxa), and eight taxa occur exclusively in one or the other set. Thus the discrepancy of the taxa represented in the two sets is considerable. However, the disagreement is not evenly distributed. Most remarkably, group B (Figure 4.5) is not represented by a blue core taxon. In such a case, the decision depends on the reconstruction method.

Group C in Figure 4.6 nicely displays the influence of the NNet on displaying genetic relatedness. Taxon *75m_27* is included in S^{NNet} but not in S^{NJ}. However, looking at the corresponding position in the NNet it is obvious that *75m_27* occupies a more intermediate position between group B and group C. Because of this intermediate position it considerably contributes to the split diversity and should be included in the data.

The overall results from the two case studies show that the taxon rankings and the optimal taxon sets based on *PD* and *SD* are largely different. In other words, different ways to summarize diversity influence the selection of the taxa. Therefore, for such cases more investigation is needed before an informed conservation decision can be made.

We also measured the performance of the *SDA* algorithm on a big dataset based on the *rbcL* gene containing 736 flora (Forest *et al.*, 2007). We repeated the same procedure as described above. On a 2.2GHz CPU computer, the *SDA* consumed less than 25 seconds to compute all optimal *SD* sets for $k = 2, \dots, 736$. The *SDA* algorithm is therefore suitable for applications with hundreds of taxa.

4.6 Discussion

It is well-known that genes can have different evolutionary histories. Even one gene can exhibit conflicting phylogenetic signals due to horizontal gene transfers or other non-treelike evolutionary events. Therefore, considering single phylogenetic trees for conservation studies comes at the loss of phylogenetic information. We present an alternative approach to incorporate incompatible phylogenetic information into the analysis. The concept of split diversity presented here is the first attempt to model the diversity when phylogenetic relationships cannot be adequately represented in a tree, but in a split system. *SD* will be equivalent to *PD* when the underlying split system corresponds to a tree and therefore consistently generalizes *PD*. Since non-treelikeness is a major topic in evolutionary biology, it will also be an issue in conservation decision projects. Thus, our proposed method helps close this gap.

Split diversity relies on having a meaningful weighted split system. There are at least three ways to obtain a split system and split weights based on the data at hand:

(i) For a set of features (Faith, 1992), a feature can be seen as a split dividing the taxon set into two groups: one group shows the feature and the other does not. Hence, split weights can depict the number of features exhibiting the same split.

(ii) For a collection of trees, a union split system can be constructed from the trees as described previously. If the branch lengths of the trees represent the expected numbers of substitutions, the split weights in the resulting split system can be seen as the average numbers of substitutions across trees. Another way is to combine different tree-distance matrices and then construct a split system from the resulting mixture of distance matrices by methods such as neighbor-net. Note that any distance-combining function other than the weighted average can be applied.

(iii) For molecular data, pairwise genetic distances between molecular sequences can be estimated. Then a weighted split system can be derived from the distance matrix by e.g. neighbor-net. Note that since a phylogenetic tree can also be built from the data, an *SD* analysis only makes sense if the inferred split system is significantly not treelike. Huson and Bryant (2006) provided a good guideline of when a split system should be considered.

The *SDA* algorithm provides an exact solution to the taxon selection problem when the split system is circular. If it is not circular, one can use the pairwise split-distance matrix to infer a circular split system with neighbor-net. An optimal taxon set on this circular split system is an approximation of the best set from the original split system. More complex or even heuristic algorithms are required to deal with arbitrary split systems.

We tested our approach on two datasets, one dataset consisting of multiple genes (Shull *et al.*, 2005), and one dataset where non-treelike events have been verified (Sullivan *et al.*, 2006) as a demonstration. In both cases, the *SDA* returned alternative choices of taxa compared to *gPDA*. We gave some phylogenetic interpretations for this observation and concluded that in terms of taxon selection applying both methods provides a larger set of candidate taxa for conservation.

Split systems and the corresponding splits graphs provide an implicit picture of evolution that simply indicates incompatible phylogenetic signals in the data (Huson and Bryant, 2006). Reticulograms (Legendre and Makarenkov, 2002) and level-k networks (Gusfield *et al.*, 2004) are alternative types of phylogenetic networks that explicitly represent reticulate events. These networks are easier to interpret. It is therefore interesting to define diversity measures on such networks.

Recently, budget constraints as described in the *Noah's Ark Problem* (Weitzman, 1998) receive an increased interest (Hartmann and Steel, 2006; Pardi and Goldman, 2007; Hartmann and Steel, 2007). Here, an overall budget is prescribed to signify the conservation effort. For each taxon a sub-budget is assigned as the requirement for its survival. We now look for a taxon collection whose preservation costs do not exceed the allotted budget. Such a model is clearly not restricted to trees but can also be extended to split systems.

Regarding *PD* alone, Faith (1992, p.10) stated that "PD evaluations based on a single cladogram are sensitive to the quality of the branch length and topology estimation". It would thus be interesting to investigate how the conservation decision based on the optimal *PD* set differs when the tree is reconstructed by different methods or when the tree changes slightly. Crozier and Kusmierski (1994) and Crozier *et al.* (1999) have studied the stability of the *genetic diversity (GD)* measure (Crozier, 1992). They used the non-parametric bootstrap to estimate the mean and the confidence interval of *GD* for different areas. These values were then used to suggest conservation areas. In principle, an analogous study for taxon selection under *GD*, *PD*, or *SD* can be applied. In this context, one is more interested in the stability of the taxon set, i.e., which taxa are in the optimal set of all bootstrap trees, which taxa are frequently observed, and which taxa are never selected. That way, one could identify stable sets of taxa for conservation.

4.7 Appendix

4.7.1 Correctness of the *SDA* Algorithm

A crucial part of the *SDA* is the application of the dynamic programming strategy. We prove the correctness of the dynamic programming in the following two propositions.

Proposition 1. *Let $(s_1, s_2, \dots, s_k, s_1)$ be the longest circular k-tour. Then $(s_1, s_2, \dots, s_k)$ is the longest ordered k-path from s_1 to s_k.*

Proof. Suppose that $(s_1, s_2, \dots, s_k)$ is not the longest ordered k-path from s_1 to s_k. Then there exists a longer ordered k-path $(s_1, s'_2, \dots, s'_{k-1}, s_k)$ from s_1 to s_k. Then

$$L(s_1, s'_2, \dots, s'_{k-1}, s_k) + d_{s_1 s_k} > L(s_1, s_2, \dots, s_k) + d_{s_1 s_k}.$$

Therefore, the circular k-tour $(s_1, s'_2, \dots, s'_{k-1}, s_k, s_1)$ is longer than $(s_1, s_2, \dots, s_k, s_1)$. That contradicts the assumption. □

Proposition 2. *Let $(s_1, s_2, \dots, s_k)$ be the longest ordered k-path from s_1 to s_k. Then $(s_1, s_2, \dots, s_{k-1})$ is the longest ordered $(k-1)$-path from s_1 to s_{k-1}.*

Proof. Similar to the proof of Prop. 1. □

4.7.2 Complexity of the *SDA* Algorithm

Proposition 3. *The* SDA *algorithm has a time complexity of $O(kn^3)$ and a memory complexity of $O(kn)$. If an outgroup is specified, the time complexity is reduced to $O(kn^2)$.*

Proof. For the computation of the matrices (L^i_{uv}) and (α^i_{uv}) one needs to regard all possible combinations of (i, u, v), where $i \in \{2, \dots, k\}$ and $u, v \in \{1, \dots, n\}$. Each entry of (L^i_{uv}) is computed in $O(n)$ time according to eq. (4.4). We get the cumulative time complexity of $O(kn^3)$. The computation of the optimal k-set requires $O(n^2)$ time for the determination of the two taxa u and v maximizing eq. (4.5) and $O(k)$ time for identifying the $k-2$ remaining taxa. In total, the computational complexity of the *SDA* is $O(kn^3)$. For the case with an outgroup u is fixed as the outgroup taxon. Therefore, the time complexity is reduced to $O(kn^2)$.

Considering memory requirement, one observes the following property of eq. (4.4). Each row of the matrix (L^i_{uv}) is computed using only the same row of (L^{i-1}_{uv}) and the split-distance matrix (d_{uv}). Hence, one can compute the first rows (L^i_{1v}) and (α^i_{1v}) and infer the longest circular k-tour originating at taxon 1. Subsequently, one can re-use the memory space to calculate the longest k-tour starting at taxon u, $u = 2, \dots, n-k+1$. With this trick, the memory requirement for the non-outgroup and the outgroup case is $O(kn)$. □

Chapter 5

Budgeted Phylogenetic Diversity on Circular Split Systems

5.1 Introduction

In recent years, biodiversity conservation at a theoretical level has attracted a lot of interest with respect to three issues. The first issue is to decide, among a variety of biodiversity measures, which are best in assessing the diversity of a set of taxa (Wilson, 1997). The traditional *taxonomic richness* concept (Gaston and Spicer, 2004) has been criticized for treating all taxa equally. Since some taxa attain more biological and ecological values and some are facing severe threats of extinction due to damaging human activities, the equal treatment of each taxon may not be adequate (May, 1990; Vane-Wright *et al.*, 1991). In 1991, Vane-Wright *et al.* (1991) pointed out the importance of considering biodiversity based on the evolutionary relationship among taxa and proposed a taxonomy-based measure. Shortly after, Faith (1992) refined it to the so-called *phylogenetic diversity (PD)*: Given a phylogenetic tree, the *PD* of a given set of taxa is the sum of the lengths of the branches connecting them, which takes into account not only the tree topology but also evolutionary distances. Other measures such as *genetic diversity* (Crozier, 1992) also use phylogenetic trees as their basis and are more or less related to *PD* (Crozier, 1997). Throughout this study we will focus on *PD* due to its widespread use. For a discussion of various measures readers are advised to refer to Purvis *et al.* (2005).

The second issue is that due to limited resources only a fraction of taxa can be preserved. Which taxa should be selected? One simple scenario is to assume that only k taxa can be conserved. Thus, one selects those k taxa having maximal *PD* among all possible k subsets of taxa (Faith, 1992). This problem can be efficiently solved by a greedy strategy (Steel, 2005; Pardi and Goldman, 2005) and efficient algorithms were presented in Minh *et al.* (2006) and Spillner *et al.* (2008). A more general and realistic scenario is that conserving each taxon comes at a specific cost (e.g. an amount of money, the size of the habitat or any other quantifiable human effort) but we are given only a limited budget. We need to find a subset of taxa with maximal *PD* such that the total conservation costs do not exceed the allotted budget (Pardi and Goldman, 2007). Formally, given a phylogenetic tree, a function which assigns to each taxon s a non-negative integer cost c_s to preserve it and a total non-negative integer budget B, we aim to

$$\begin{gathered}\text{find a subset } S \text{ of taxa}\\ \text{to maximize } PD(S)\\ \text{subject to } \textstyle\sum_{s\in S} c_s \leq B.\end{gathered}$$

The restriction to integral costs and budget is not a limitation since these values are often expressed in integers. Under budget constraints, the greedy algorithm no longer guarantees an optimal solution, but a dynamic programming algorithm, PD-BUDGET, works (Pardi and Goldman, 2007). The PD-BUDGET can be further applied to solve a subset of the Noah's Ark Problem (Weitzman, 1998; Hartmann and Steel, 2006), when the taxon risk of extinction is also accounted for (Pardi and Goldman, 2007).

The last issue involves the basis for *PD*: the tree. It is well known that different genomic regions can give rise to different trees due to varying rates of evolution, genetic recombination or ancestral polymorphism (Graur and Li, 2000; Nei, 1987). Minh *et al.* (2006) demonstrated a simple case with two four-taxon trees and showed that the two optimal *PD* sets of size 2 inferred from the two trees are different. A way to incorporate the information is to determine the set of taxa maximizing the weighted sum of *PD* over all trees. This problem was shown to be NP-hard and equivalent to determining the set with maximal *split diversity (SD)* (Minh *et al.*, 2008a) on the union split system formed of all splits existing in at least one tree (Spillner *et al.*, 2008). Hence, Minh *et al.* (2007) proposed an approximation by reconstructing a Neighbor-net (Bryant and Moulton, 2004) from the combined tree-distance matrices and subsequently inferring the optimal *SD* set from this split system. The Neighbor-net falls into the group of circular split systems in which another dynamic programming algorithm, *SDA*, ensures to obtain an optimal *SD* set (Minh *et al.*, 2007). A Neighbor-net can also be constructed from genetic distances between molecular sequences and the *SDA* algorithm can be applied. An attempt to reduce the complexity of the *SDA* algorithm was suggested in Spillner *et al.* (2008).

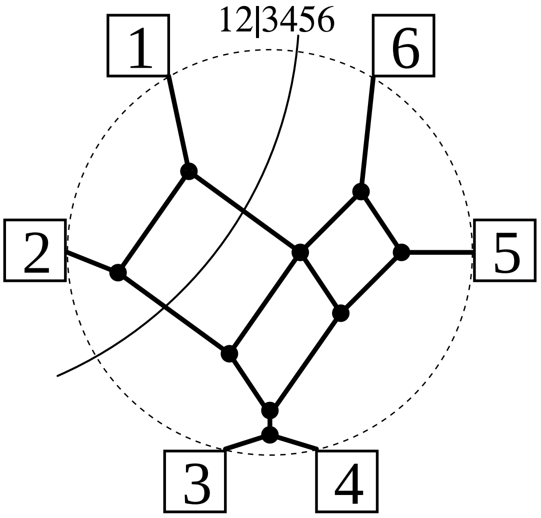
In this paper, we will consider a combination of the last two issues: computing an optimal *SD* set for cicular split systems under budget constraints. The formalism is similar to the problem described for the budget constraints except that the underlying structure is not a tree but a circular split system. Following the work in Minh *et al.* (2007), we will show that the *SDA* algorithm can be extended to cope adequately with budget constraints. The resulting algorithm, *SDA*-BUDGET, can also be applied to the Noah's Ark Problem in the same manner as illustrated for the PD-BUDGET algorithm (Pardi and Goldman, 2007).

5.2 Notations

Following Minh *et al.* (2007), we denote by X the set of n taxa. A *split* $A|B$ is a bipartition of X into two non-empty disjoint sets A and B, i.e., $A \cap B = \emptyset$ and $A \cup B = X$. A *split system* Σ is any collection of splits of X. A tree is a special case of a split system where only *compatible* splits are permitted (Semple and Steel, 2003, pp. 43-44) and each split corresponds to a single branch of the tree. A split system is visualized as a split network, where each split is presented by one or more parallel edges. Fig. 5.1 displays a six-taxon split network and its corresponding split system consisting of 11 splits.

A split system is called *circular* if there exists a way to number the taxa from 1 to n such that all splits are of the form $\{i, i+1, \dots, j\} \mid X - \{i, i+1, \dots, j\}$, $1 \leq i \leq j \leq n$ (Bandelt and Dress, 1992b). $(1, 2, \dots, n)$ is then called a *circular order* of the taxa. The split network in Fig. 5.1 shows an example of a circular split network where one can visually place all taxa onto a circle and each split can be depicted as a line bisecting the circle. Note that a tree is also a special case of circular split system and that the Neighbor-net method (Bryant and Moulton, 2004) always produces a circular split system.

Given a split weight function λ that assigns to each split $A|B \in \Sigma$ a non-negative



Split	Weight
1|23456	4
2|13456	2
3|12456	2
4|12356	2
5|12346	4
6|12345	5
12|3456	7
234|156	6
34|1256	1
345|126	3
56|1234	4

Figure 5.1: A sample split network and its corresponding split system consisting of 11 splits with their split weights. Each split is depicted by a single edge or several parallel edges. For example, the split 12|3456 is depicted by two parallel edges. The circle connecting the taxa of the network indicates the circular order $(1, 2, 3, 4, 5, 6)$.

weight $\lambda(A|B)$, the *split diversity (SD)* of a taxon set S is defined as the sum of the weights of all splits separating the taxa of S:

$$\mathrm{sd}(S) = \sum_{\substack{A|B\in\Sigma: \\ A\cap S\neq\emptyset \\ B\cap S\neq\emptyset}} \lambda(A|B). \tag{5.1}$$

For any two taxa u and v, we define $d_{uv} = \mathrm{sd}(\{u, v\})$ as the *split-distance* between these two taxa.

For a circular split system with circular taxon order $(1, 2, \ldots, n)$, we define a *circular tour* as a sequence of taxa $(s_1, s_2, \ldots, s_k, s_{k+1})$ such that $1 \leq s_1 < s_2 < \ldots < s_k \leq n$ and $s_{k+1} = s_1$ (returning to the first taxon). Minh *et al.* (2007) have shown that on a circular split system, for any subset $S = \{s_1, s_2, \ldots, s_k\}$ with $1 \leq s_1 < s_2 < \ldots < s_k \leq n$, the *SD* score of S is equal to one half of the length of the circular tour $(s_1, \ldots, s_k, s_1)$:

$$\mathrm{sd}(S) = \frac{1}{2}\left(d_{s_1 s_k} + \sum_{i=1}^{k-1} d_{s_i s_{i+1}}\right). \tag{5.2}$$

This crucial property of circular split systems acts as the foundation for the *SDA* algorithm (Minh *et al.*, 2007) as follows. Due to Eq. (5.2) the determination of a subset S of size k with maximum *SD* in a circular split system can be transformed into finding the longest *circular k-tour*, i.e., the longest among those circular tours traversing k taxa. To this end, we define an *ordered k-path* from a taxon u to another taxon v as a sequence of k taxa $(u = s_1, s_2, \ldots, s_k = v)$ following the circular order, i.e., satisfying $1 \leq s_1 < s_2 < \ldots < s_k \leq n$. Note that a circular k-tour is attained by adding a taxon $s_{k+1} = s_1$. Let L^k_{uv} denote the length of the longest ordered k-path from u to v and ℓ^k_{max} the length of the longest circular k-tour. The *SDA* iteratively computes L^i_{uv} between all pairs of taxa u and v for $i = 2, 3, \ldots, k$ and subsequently ℓ^k_{max} by the dynamic programming equations (4) and (5) in Minh *et al.* (2006). Finally, the maximal *SD* score will be simply $\ell^k_{max}/2$ and an optimal set S can be constructed using L^i_{uv}.

5.3 SDA-BUDGET Algorithm

We now explain the core of the *SDA*-BUDGET algorithm, an extension of the *SDA* (*SDA*-BUDGET is essentially *SDA* if every taxon has unit cost and the budget is equal to k). We are given a weighted circular split system, non-negative integer taxon-associated costs c_s and a non-negative integer budget B. We want to identify a taxon subset S with maximal *SD* satisfying the budget constraint $\sum_{s \in S} c_s \leq B$. Like the *SDA* algorithm, Eq. 5.2 turns this problem into:

> *Given n taxa, their circular order $(1, 2, \ldots, n)$, pairwise split-distances (d_{uv}), a taxon preservation cost c_s for each taxon s, and a total budget B. Find the* **longest circular B-tour**, *i.e. the longest among those circular tours whose sum of taxon costs does not exceed B.*

We adapt the notations previously used for the *SDA* algorithm as follows. The term ordered k-path is changed to *ordered b-path* $(s_1, s_2, \ldots, s_i)$ satisfying $\sum_{j=1}^{i} c_{s_j} \leq b$. Hence, there is no condition on the number of taxa along the path, rather the sum of taxon costs on the path should not exceed the allotted budget b. Also, the term circular k-tour is replaced by *circular b-tour*, i.e., a tour $(s_1, s_2, \ldots, s_i, s_1)$ such that $\sum_{j=1}^{i} c_{s_j} \leq b$. Due to these modifications, L^b_{uv} denotes the length of the longest ordered b-path from u to v and ℓ^b_{max} the length of the longest circular b-tour.

The dynamic programming strategy still works: if $(s_1, s_2, \ldots, s_i, s_1)$ is the longest circular B-tour then $(s_1, s_2, \ldots, s_i)$ must be the longest ordered B-path from s_1 to s_i. Similarly, if $(s_1, s_2, \ldots, s_i)$ is the longest ordered B-path from s_1 to s_i then also $(s_1, s_2, \ldots, s_{i-1})$ is the longest ordered $\{B - c_{s_i}\}$-path from s_1 to s_{i-1}. As a result, the length ℓ^B_{max} of the longest circular B-tour will be obtained by solving the following iterative maximization:

$$L^b_{uv} = \begin{cases} -\infty, & \text{if } b < c_u + c_v, \\ d_{uv}, & \text{if } b \geq c_u + c_v \text{ and } \nexists s \text{ with } u < s < v \\ & \text{and } b \geq c_s + c_u + c_v, \\ \max_{u<s<v}\{L^{b-c_v}_{us} + d_{sv}\}, & \text{otherwise,} \end{cases} \tag{5.3}$$

$$\ell^B_{max} = \max_{1 \leq u < v \leq n} \{L^B_{uv} + d_{uv}\}. \tag{5.4}$$

The interpretation is principally as follows. The first line in Eq. 5.3 states that if the costs to conserve only u and v already exceed the budget b, then there exists no ordered b-path from u to v. The second line in Eq. 5.3 states that if $b \geq c_u + c_v$, but the budget b cannot afford any additional taxon s between u and v, then there is a single ordered b-path only containing the two end-point taxa u and v. The third case allows us to include some taxon s in addition to u and v. Then we can invest a budget of $b - c_v$ into the path from u to s, which is reflected in the third line of Eq. 5.3. Finally, Eq. 5.4 simply scans through all the longest ordered B-paths to obtain the length of the longest circular B-tour.

Note that if $c_v = 0$, the third line in Eq. 5.3 becomes $L^b_{uv} = \max\{L^b_{us} + d_{sv}\}$. In such cases, L^b_{uv} is not totally determined by $L^0_{uv}, \ldots, L^{b-1}_{uv}$, and thus we cannot iteratively compute $L^0_{uv}, L^1_{uv}, \ldots, L^B_{uv}$. However, the following simple modification works: compute, for all b, $L^b_{12}, L^b_{13}, \ldots, L^b_{1n}, L^b_{23}, \ldots, L^b_{(n-1)n}$, i.e., iterate on the index of the taxon u, then on v and b.

The last step is analogous to the *SDA* algorithm. The maximal *SD* score is $\ell^B_{max}/2$. To construct an optimal set S, one first determines two taxa maximizing the sum on Eq. 5.4, then traces back the series of taxa s in the third line of Eq. 5.3.

5.3.1 An Example

To illustrate how the *SDA*-BUDGET algorithm proceeds, we use the circular split system in Figure 5.1. The taxon costs are: $c_1 = 3, c_2 = 1, c_3 = 2, c_4 = 4, c_5 = 2, c_6 = 1$. We want to identify an optimal *SD* set with budget $B = 7$. The first step is to determine the pairwise split-distance matrix:

$$(d_{uv}) = \begin{pmatrix} - & 12 & 23 & 23 & 22 & 20 \\ 12 & - & 15 & 15 & 26 & 24 \\ 23 & 15 & - & 4 & 17 & 21 \\ 23 & 15 & 4 & - & 17 & 21 \\ 22 & 26 & 17 & 17 & - & 12 \\ 20 & 24 & 21 & 21 & 12 & - \end{pmatrix}$$

	0	1	2	3	4	5	6	7	$L^7_{uv}+d_{uv}$
L^b_{12}					12	12	12	12	24
L^b_{13}						23	27 (2)	27 (2)	50
L^b_{14}								23	46
L^b_{15}						22	38 (2)	40 (3)	62
L^b_{16}					20	36 (2)	44 (3)	50 (5)	70
L^b_{23}				15	15	15	15	15	30
L^b_{24}						15	15	19 (3)	34
L^b_{25}				26	26	32 (3)	32 (3)	32 (3)	58
L^b_{26}			24	24	38 (5)	38 (5)	44 (5)	44 (5)	68
L^b_{34}							4	4	8
L^b_{35}					17	17	17	17	34
L^b_{36}				21	21	29 (5)	29 (5)	29 (5)	50
L^b_{45}							17	17	34
L^b_{46}						21	21	29 (5)	50
L^b_{56}				12	12	12	12	12	24

Figure 5.2: Solution table of *SDA*-BUDGET algorithm. See the main text for explanation.

Now we come to the core part of the algorithm. The actual procedure is summarized in Figure 5.2. As noted before, *SDA*-BUDGET iterates through $L^b_{12}, L^b_{13}, \ldots, L^b_{(n-1)n}$, as reflected on each row of the table. The columns are for $b = 0, \ldots, 7$. The number to the right of each row is equal to $L^7_{uv} + d_{uv}$, the length of the longest circular 7-tour containing the corresponding two end-points. Basically, one fills out this table from the top row to the bottom row, one row at a time, and from the left column to the right column. We will demonstrate the computation for the first four rows of the table.

The first row regards L_{12}^b. Since $c_1 + c_2 = 4$, all entries with $b < 4$ are equal to $-\infty$, which is here denoted by an empty cell. Otherwise, when $b \geq 4$ the entries are equal to $d_{12} = 12$. So the longest circular 7-tour associated with taxa 1 and 2 has the length of $L_{12}^7 + d_{12} = 12 + 12 = 24$.

The second row is for L_{13}^b. Because $c_1 + c_3 = 5$, $L_{13}^b = -\infty$ if $b < 5$. For $b = 5$ the path can only afford the taxa 1 and 3, thus $L_{13}^5 = d_{13} = 23$. Now if $b = 6$, we see that it can additionally afford the taxon 2, so based on the third line of Eq. 5.3, we have $L_{13}^6 = L_{12}^{6-c_3} + d_{23} = 12 + 15 = 27$, which uses the entry L_{12}^4 in the upper row. At the same time, we record that the taxon 2 contributes to this sum and write it down in the parenthesis on the entry. The same applies for L_{13}^7. So $L_{13}^7 + d_{13} = 27 + 23 = 50$.

The computation for the third row is easy and ignored here. We continue with the fourth row: L_{15}^b. For $b < c_1 + c_5 = 5$, $L_{15}^b = -\infty$. For $b = 5$: $L_{15}^5 = d_{15} = 22$. For $b = 6$ the path from 1 to 4 can afford the taxon 2 in addition, so $L_{15}^6 = L_{12}^{6-2} + d_{25} = 12 + 26 = 38$ and we record the contributing taxon 2. For $b = 7$ we see that the path can cover either taxon 2 or 3. By going through the taxon 2, we have the same score of 38. If one goes through taxon 3: $L_{13}^{7-2} + d_{35} = 23 + 17 = 40$ which is greater than passing through taxon 2. So $L_{15}^7 = 40$ and we memorize taxon 3. Finally, $L_{15}^7 + d_{15} = 40 + 22 = 62$.

The procedure is continued until all the entries of the table are computed. At last, the maximum entry in the right most column will provide us with the longest circular B-tour. In our example, the tour has length 70 and is associated with the longest 7-path from taxon 1 to taxon 6. So the maximal *SD* score is $70/2 = 35$. To recover all taxa in the optimal set, first look at the entry L_{16}^7, which is associated with taking taxon 5. Then look at $L_{15}^{7-c_6} = L_{15}^6$ and we see that taxon 2 was recorded for this entry. Then if we look at $L_{12}^{6-c_5} = L_{12}^4$ we see no other taxon. Therefore an optimal *SD* set is $\{1, 2, 5, 6\}$.

5.3.2 Modification for circular split systems with an outgroup

Like the *SDA* algorithm we can modify *SDA*-BUDGET to compute the best set if an outgroup is given that should always be included in the set (Minh *et al.*, 2007). To this end, we reindex the taxa such that the outgroup taxon is 1. It is easy to see that the *SDA*-BUDGET algorithm now only needs to calculate $L_{12}^b, L_{13}^b, \ldots, L_{1n}^b$, i.e., the first

$n-1$ rows of the solution table as the rows below do not account for paths going through the outgroup taxon 1.

5.3.3 Complexity

The solution table contains $n(n-1)/2$ rows and $B+1$ columns, where each entry is computed in at most $O(n)$ time (Eq. 5.3). The backtracking of the optimal set needs $O(n)$ time. So the time-complexity of the *SDA*-BUDGET algorithm is $O(Bn^3)$. For the case with an outgroup, only the first $n-1$ rows of the table need to be computed. Therefore the time-complexity is reduced by a factor of n, i.e., $O(Bn^2)$.

A trivial implementation would store the whole table, resulting in a memory space requirement of $O(Bn^2)$. However, there is a simple and more efficient way to reduce the memory. We observe that the computation of L^b_{uv} is not related to any values of L^b_{wv} where $w \in \{1,\ldots,u-1\}$. Hence, we first allocate the memory for the first $n-1$ rows and compute $L^b_{12}, L^b_{13}, \ldots, L^b_{1n}$. The longest circular B-tour originating at the taxon 1 is then constructed and recorded. Subsequently, we reuse the allocated memory to compute the next $n-2$ rows, then construct the longest circular B-tour originating at the taxon 2 and compare it to the one previously recorded. If it is longer, the longest B-tour will be updated. We repeat this until the last row. The memory-complexity is therefore reduced to $O(Bn)$.

5.4 Conclusion

We have presented a dynamic programming algorithm *SDA*-BUDGET to compute an optimal *SD* set under budget constraints on circular split sytems. *SDA*-BUDGET is derived from the *SDA* algorithm (Minh *et al.*, 2007) and has the time-complexity of $O(Bn^3)$, where B is the total budget and n is the number of taxa. If an outgroup taxon or a taxon that must be preserved is identifed, the complexity decreases to $O(Bn^2)$. Therefore, users are advised to specify the outgroup/root or the included taxon when it is known in advance.

Chapter 6

Summary

In this doctoral thesis we have proposed the concept of split diversity to measure the diversity of taxa in the presence of incongurent phylogenetic trees or conflicting evolutionary signals. Split diversity considers split systems (Bandelt and Dress, 1992a) as the underlying structure. When the split system corresponds to a tree, split diversity will be equivalent to phylogenetic diversity (Faith, 1992) and therefore consistently generalizes phylogenetic diversity.

Moreover, we have developed a number of (efficient) algorithms to solve various conservation questions listed in Chapter 2 under the concepts of phylogenetic diversity and split diversity. These algorithms are ranging from simple greedy algorithms (Chapter 3 to dynamic programming algorithms (Chapters 4 and 5). The current progress of the field "Computational Biodiversity Conservation" is summarized in Table 6.1. During the course of the development, improved algorithms for the taxon selection on trees and circular split systems (Spillner *et al.*, 2008) and for the budgeted taxon selection on trees (Minh *et al.*, 2008b) were introduced (Table 6.1). Most of the proposed algorithms are implemented in the *PDA* software (Section 6.1).

6.1 The PDA software

The software tool *Phylogenetic Diversity Analyzer (PDA)* that implements the proposed methods is freely available at

`http://www.cibiv.at/software/pda/`.

Problem	**Algorithm's authors**	**Complexity**
Taxon selection on single trees	Minh *et al.* (2006), Chap. 3 Spillner *et al.* (2008)	$O(n \log k)$ $O(n)$
Taxon selection across two trees	Bordewich *et al.* (2008)	$O(n^3 \log^2 n)$ $O(n^2 \log^2 n)^1$
Budgeted taxon selection on single trees	Pardi and Goldman (2007) Minh *et al.* (2008b)	$O(B^2 n)$ $O(Bn^2 \log n)$ $O(Bn \log n)^1$
Taxon selection on circular split systems	Minh *et al.* (2008a), Chap. 4 Spillner *et al.* (2008)	$O(kn^3)$, $O(kn^2)^1$ $O(kn + n \log n)$
Taxon selection on affine split systems	Spillner *et al.* (2008)	$O(kn^3)$
Budgeted taxon selection on circular split systems	Minh *et al.* (2008b), Chap. 5	$O(Bn^3)$, $O(Bn^2)^1$
Reserve selection on single trees	Rodrigues and Gaston (2002)	-^{2}

Table 6.1: Recently developed algorithms for various conservation problems and their computational complexity for unrooted trees or split systems. n - the number of taxa, k - the number of taxa to preserve, B - the total budget. ^{1}The time-complexity when the tree is rooted or an outgroup is provided for the tree or the split system.^{2}The complexity is exponential in the worst-case.

A user-friendly web-interface developed by Tung Lam Nguyen is also available online at `http://www.cibiv.at/software/pda/web-pda/`.

The major features of the program include:

- Accepting an input split system in NEXUS format (e.g., as produced by SplitsTree (Huson and Bryant, 2006)), collection of trees in NEXUS format (e.g., as produced by MrBayes (Ronquist and Huelsenbeck, 2003)), and a tree(s) file in NEWICK format.
- (Taxon selection) Determining the maximal taxon set of a given size k or under budget constraints on trees and general split systems.

- (Reserve selection) Determining the maximal area collection of a given size k or under budget constraints on trees and general split systems.
- Determining the *mimimal* taxon set on trees and circular split systems.
- Identifying multiple optimal taxon sets on trees and circular split systems.
- Supporting unrooted/rooted trees, trees and split systems with an outgroup.
- Specifying the set of taxa/areas to always include into the optimal set.
- (Area analysis) Computing *PD* and *SD* scores of user-defined taxon sets; exclusive *PD*, endemic *PD*, and complementary *PD* of an area.

PDA is written in C++ using the standard template library and runs on all popular platforms (Linux, Windows, MacOS). *PDA* integrates the NEXUS class library (Lewis, 2003) for parsing the NEXUS file and the LP_SOLVE library (`http://sourceforge.net/projects/lpsolve`) for solving (integer) linear programming. Further details about *PDA* can be found in the online user-manual

`http://www.cibiv.at/software/pda/pda-manual/`.

Bibliography

Agapow, P.-M., Bininda-Emonds, O., Crandall, K., Gittleman, J. L., Mace, G., Marshall, J. and Purvis, A. (2004) The impact of species concept on biodiversity. *Q. Rev. Biol.*, pages 161–179.

Avise, J. C. (2005) Phylogenetic units and currencies above and below the species level. In Purvis, A., Gittleman, J. L. and Brooks, T. (eds.), *Phylogeny and Conservation*, pages 76–100, Cambridge University Press, Cambridge, UK.

Balmford, A., Bennun, L., Brink, B. t., Cooper, D., Cote, I. M., Crane, P., Dobson, A., Dudley, N., Dutto, I., Green, R. E., Gregory, R. D., Harrison, J., Kennedy, E. T., Kremen, C., Leader-Williams, N., Lovejoy, T. E., Mace, G., May, R., Mayaux, P., Morling, P., Phillips, J., Redford, K., Ricketts, T. H., Rodriguez, J. P., Sanjayan, M., Schei, P. J., van Jaarsveld, A. S. and Walther, B. A. (2005) The convention on biological diversity's 2010 target. *Science*, **307**, 212–213.

Bandelt, H.-J. and Dress, A. W. M. (1992a) A canonical decomposition theory for metrics on a finite set. *Adv. Math.*, **92**, 47–105.

Bandelt, H.-J. and Dress, A. W. M. (1992b) Split decomposition: A new and useful approach to phylogenetic analysis of distance data. *Mol. Phylogenet. Evol.*, **1**, 242–252.

Benson, D. A., Karsch-Mizrachi, I., Lipman, D. J., Ostell, J. and Wheeler, D. L. (2006) GenBank. *Nucl. Acids Res.*, **34**, D16–D20.

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

Bordewich, M., Semple, C. and Spillner, A. (2008) Optimizing phylogenetic diversity across two trees. in press.

Bryant, D. (2004) The splits in the neighborhood of a tree. *Ann. Combinatorics*, **8**.

Bryant, D. and Dress, A. (2007) Linearly independent split systems. *European J. Combinatorics*, **28**, 1814–1831.

Bryant, D. and Moulton, V. (2004) Neighbor-net: An agglomerative method for the construction of phylogenetic networks. *Mol. Biol. Evol.*, **21**, 255–265.

Church, R. L. and ReVelle, C. (1974) The maximal covering location problem. *Pap. Reg. Sci. Assoc*, **62**, 101–118.

Church, R. L., Stoms, D. M. and Davis, F. W. (1996) Reserve selection as a maximal covering location problem. *Biol. Conserv.*, **76**, 105–112.

Cormen, T. H., Leiserson, C. E., Rivest, R. L. and Stein, C. (2001) *Introduction to Algorithms.* MIT Press and McGraw-Hill, Second edn..

Crozier, R. H. (1992) Genetic diversity and the agony of choice. *Biol. Conserv.*, **61**, 11–15.

Crozier, R. H. (1997) Preserving the information content of species: Genetic diversity, phylogeny, and conservation worth. *Annu. Rev. Ecol. Syst.*, **28**, 243–68.

Crozier, R. H., Agapow, P.-M. and Dunnett, L. J. (2005) Phylogenetic biodiversity assessment based on systematic nomenclature. *Evolutionary Bioinformatics Online*, **1**, 11–36.

Crozier, R. H., Agapow, P.-M. and Pederson, K. (1999) Towards complete biodiversity assessment: an evaluation of the subterranean bacterial communities in the oklo region of the sole surviving natural nuclear reactor. *FEMS Microbiol. Ecol.*, **28**, 325–334.

Crozier, R. H. and Kusmierski, R. M. (1994) Genetic distances and the setting of conservation priorities. In Loeschke, V., Tomiuk, J. and Jain, S. K. (eds.), *Conservation Genetics*, pages 227–237, Birkhauser Verlag, Basel.

Dress, A. and Huson, D. (2004) Constructing splits graphs. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, **1**, 109–115.

Faith, D. P. (1992) Conservation Evaluation and Phylogenetic Diversity. *Biol. Conserv.*, **61**, 1–10.

Faith, D. P. and Baker, A. M. (2006) Phylogenetic diversity (pd) and biodiversity conservation: some bioinformatics challenges. *Evolutionary Bioinformatics Online*, **2**, 70–77.

Faith, D. P., Reid, C. A. M. and Hunter, J. (2004) Integrating phylogenetic diversity, complementarity and endemism for conservation assessment. *Conserv. Biol*, **18**, 255–261.

Faith, D. P. and Walker, P. A. (1994) *DIVERSITY: A Software Package for Sampling Phylogenetic and Environmental Diversity.* Second edn..

Felsenstein, J. (2004) *Inferring Phylogenies.* Sinauer Associates, Sunderland, Massachusetts.

Forest, F., Grenyer, R., Rouget, M., Davies, T. J., Cowling, R. M., Faith, D. P., Balmford, A., Manning, J. C., Proches, S., van der Bank, M., Reeves, G., Hedderson, T. A. J. and Savolainen, V. (2007) Preserving the evolutionary potential of floras in biodiversity hotspots. *Nature*, **445**, 757–760.

Gaston, K. J. and Spicer, J. I. (2004) *Biodiversity: An Introduction.* Blackwell Publishing Professional, Second edn..

Graur, D. and Li, W.-H. (2000) *Fundamentals of Molecular Evolution.* Sinauer Associates, Sunderland, Massachusetts, Second edn..

Gusfield, D., Eddhu, S. and Langley, C. (2004) Optimal, efficient reconstruction of phylogenetic networks with constrained recombination. *J. Bioinform. Comput. Biol.*, **2**, 173 – 213.

Harding, E. F. (1971) The probabilities of rooted tree shapes generated by random bifurcation. *Advances in Applied Probability*, **3**, 44–77.

Hartmann, K. and Steel, M. (2006) Maximizing phylogenetic diversity in biodiversity conservation: Greedy solutions to the noah's ark problem. *Syst. Biol.*, **55**, 644–651.

Hartmann, K. and Steel, M. (2007) Phylogenetic diversity: from combinatorics to ecology. In Gascuel, O. and Steel, M. (eds.), *Reconstructing Evolution: New Mathematical and Computational Advances*, pages 171–196, Oxford University Press, Oxford, UK.

Henz, S. R., Huson, D. H., Auch, A. F., Nieselt-Struwe, K. and Schuster, S. C. (2005) Whole-genome prokaryotic phylogeny. *Bioinformatics*, **21**, 2329–2335.

Hertel, J., Lindemeyer, M., Missal, K., Fried, C., Tanzer, A., Flamm, C., Hofacker, I. L., Stadler, P. F. and of Bioinformatics Computer Labs 2004/2005, T. S. (2006) The expansion of the metazoan microRNA repertoire. *BMC Genomics*, **7**.

Huber, K. T. and Moulton, V. (2005) Phylogenetic networks. In Gascuel, O. (ed.), *Mathematics of Evolution and Phylogeny*, pages 178–204, Oxford University Press, Oxford, UK.

Humphries, C. J., Williams, P. H. and Vane-Wright, R. I. (1995) Measuring Biodiversity Value for Conservation. *Annu. Rev. Ecol. Syst.*, **26**, 93–111.

Huson, D. H. and Bryant, D. (2006) Application of phylogenetic networks in evolutionary studies. *Mol. Biol. Evol.*, **23**, 254–267.

IUCN (2001) *IUCN Red List Categories: Version 3.1.* IUCN Species Survival Commission, Gland, Switzerland.

Korostensky, C. and Gonnet, G. H. (2000) Using traveling salesman problem algorithms for evolutionary tree reconstruction. *Bioinformatics*, **16**, 619–627.

Korte, B., Lovász, L. and Schrader, R. (1991) *Greedoids.* Algorithms and Combinatorics, Springer Verlag Berlin.

Legendre, P. and Makarenkov, V. (2002) Reconstruction of biogeographic and evolutionary networks using reticulograms. *Syst. Biol.*, **51**, 199–216.

Lewis, L. A. and Lewis, P. O. (2005) Unearthing the molecular diversity of desert soil green algae. *Syst. Biol.*, **54**, 936–947.

Lewis, P. O. (2003) NCL: a C++ class library for interpreting data files in NEXUS format. *Bioinformatics*, **19**, 2330–2331.

Maddison, D. R. and Schulz, K.-S. (2007) The tree of life web project. Internet address: `http://tolweb.org`.

May, R. M. (1990) Taxonomy as destiny. *Nature*, **347**, 129–130.

Minh, B. Q., Klaere, S. and von Haeseler, A. (2006) Phylogenetic diversity within seconds. *Syst. Biol.*, **55**, 769–773.

Minh, B. Q., Klaere, S. and von Haeseler, A. (2007) Phylogenetic diversity on split networks. *Technical Report NI07090-PLG, Isaac Newton Institute, Cambridge, UK.*

Minh, B. Q., Klaere, S. and von Haeseler, A. (2008a) Taxon selection under split diversity. *Syst. Biol.*, submitted.

Minh, B. Q., Pardi, F., Klaere, S. and von Haeseler, A. (2008b) Budgeted phylogenetic diversity on circular split systems. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, in press.

Minh, B. Q., Vinh, L. S., von Haeseler, A. and Schmidt, H. A. (2005) pIQPNNI – parallel reconstruction of large maximum likelihood phylogenies. *Bioinformatics*, **21**, 3794–3796.

Moulton, V., Semple, C. and Steel, M. (2007) Optimizing phylogenetic diversity under constraints. *J. Theor. Biol.*, **246**, 186–194.

Myers, N. (1988) Threatened biotas: hotspots in tropical forests. *The Environmentalist*, **8**, 187–208.

Myers, N. (1990) The biodiversity challenge: Expanded hot-spots analysis. *The Environmentalist*, **10**, 243–256.

Myers, N., Mittermeier, R., Mittermeier, C., da Fonseca, G. and Kent, J. (2000) Biodiversity hotspots for conservation priorities. *Nature*, **403**, 853–858.

Nei, M. (1987) *Molecular Evolutionary Genetics*. Columbia University Press, New York.

Pardi, F. and Goldman, N. (2005) Species choice for comparative genomics: Being greedy works. *PLoS Genet.*, **1**, 672–675.

Pardi, F. and Goldman, N. (2007) Resource-aware taxon selection for maximising phylogenetic diversity. *Syst. Biol.*, **56**, 431–444.

Purvis, A., Gittleman, J. L. and Brooks, T. (2005) *Phylogeny and Conservation*. Conservation Biology, Cambridge University Press, Cambridge, UK.

de las Rivas, B., Marcobal, Á. and Muñoz, R. (2004) Allelic diversity and population structure in *oenococcus oeni* as determined from sequence analysis of housekeeping genes. *Appl. Environ. Microb.*, **70**, 7210–7219.

Rodrigues, A. S. L., Brooks, T. and Gaston, K. J. (2005) Integrating phylogenetic diversity in the selection of priority areas for conservation: does it make a difference. In Purvis, A., Gittleman, J. and Brooks, T. (eds.), *Phylogeny and conservation*, Cambridge University Press, Cambridge, UK.

Rodrigues, A. S. L. and Gaston, K. J. (2002) Maximising phylogenetic diversity in the selection of networks of conservation areas. *Biological Conservation*, **105**, 103–111.

Ronquist, F. and Huelsenbeck, J. P. (2003) MrBayes 3: Bayesian phylogenetic inference under mixed models. *Bioinformatics*, **19**, 1572–1574.

Saitou, N. and Nei, M. (1987) The neighbor–joining method: A new method for reconstructing phylogenetic trees. *Mol. Biol. Evol.*, **4**, 406–425.

Semple, C. and Steel, M. (2003) *Phylogenetics.* Oxford Lectures Series in Mathematics and its Applications, J. Ball and D. Welsh (eds.), Oxford University Press.

Shull, H. C., Pérez-Losada, M., Blair, D., Sewell, K., Sinclair, E. A., Lawler, S., Ponniah, M. and Crandall, K. A. (2005) Phylogeny and biogeography of the freshwater crayfish *Euastacus* (Decapoda: Parastacidae) based on nuclear and mitochondrial DNA. *Mol. Phylogenet. Evol.*, **37**, 249–263.

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

Steel, M. (2005) Phylogenetic Diversity and the Greedy Algorithm. *Syst. Biol.*, **54**, 527–529.

Sullivan, M. B., Lindell, D., Lee, J. A., Thompson, L. R., Bielawski, J. P. and Chisholm, S. W. (2006) Prevalence and evolution of core photosystem II genes in marine cyanobacterial viruses and their hosts. *PLoS Biology*, **4**, 1344–1357.

Swofford, D. L., Olsen, G. J., Waddell, P. J. and Hillis, D. M. (1996) Phylogeny reconstruction. In Hillis, D. M., Moritz, C. and Mable, B. K. (eds.), *Molecular Systematics*, pages 407–514, Sinauer Associates, Sunderland, Massachusetts, Second edn..

Thompson, J. D., Higgins, D. G. and Gibson, T. J. (1994) CLUSTAL W: Improving the sensitivity of progressive multiple sequence alignment through sequence weighting, positions–specific gap penalties and weight matrix choice. *Nucl. Acids Res.*, **22**, 4673–4680.

Underhill, L. (1994) Optimal and suboptimal reserve selection algorithms. *Biological Conservation*, **70**, 85–87.

Vane-Wright, R. I., Humphries, C. J. and Williams, P. H. (1991) What to protect? - systematics and the agony of choice. *Biol. Conserv.*, **55**, 235–254.

Weitzman, M. L. (1998) The Noah's Ark problem. *Econometrica*, **66**, 1279–1298.

Whiting, A. S., Lawler, S. H., Horwitz, P. and Crandall, K. A. (2000) Biogeographic regionalization of Australia: assigning conservation priorities based on endemic freshwater crayfish phylogenetics. *Anim. Conserv.*, **3**, 155–163.

Williams, P. and Humphries, C. (1996) WORLDMAP and prioritisation for conservation: integration of systematic data for conservation evaluation. In Jermy, A. C., Long, D., Sands, M. J. S., Stork, N. E. and Winser, S. (eds.), *Biodiversity assessment: a guide to good practice*, pages 98–99, Department of the Environment / HMSO, London.

Williams, P. H. and Araujo, M. B. (2002) Apples, Oranges and Probabilities: Integrating multiple factors into biodiversity conservation with consistency. *Environmental Modelling and Assessment.*, **7**, 139–151.

Wilson, E. O. (ed.) (1997) *Biodiversity.* National Academies Press, Second edn..

Curriculum Vitae

Contact Information

Bui Quang Minh
Center for Integrative Bioinformatics Vienna (CIBIV)
Max F. Perutz Laboratories
Dr. Bohr Gasse 9
A-1030 Vienna, Austria
Phone: ++43 +1 / 79044-4587
Fax: ++43 +1 / 79044-4551
Email: minh.bui(AT)univie.ac.at

Research Interests

- Phylogenetic Analysis
- Biodiversity Conservation
- Parallel Computing

Education

- 1994 - 1997: High School for the Gifted in Informatics, University of Science, Hanoi.
- 1997 - 2001: Bachelor in Computer Science, Vietnam National University, Hanoi.
- Oct 2002 - Oct 2005: Master student in Applied Computer Science, Freiburg University, Germany.

- Jan 2006 - : PhD student at the Center for Integrative Bioinformatics Vienna, Max F. Perutz Laboratories.

Degree

- 'M.Sc.' 2005, Freiburg, Germany Topic: Parallel reconstruction of large maximum likelihood phylogenies.
- 'B.Sc.' 2001, Hanoi, Vietnam Thesis: Solutions to Vietnamese in Linux.

Awards

- 2004: 4th Prize, Imagine Cup 2004, Microsoft Deutschlands, together with Duc Phuong Nguyen. Topic: Mobile Learning with LARA.
- 2001: 2nd VIFOTEC Innovation Prize. Topic: Solutions to Vietnamese in Linux.
- 2000: 3rd Prize Chinese Chess Computer Competition.
- 2000: Young Innovation Medal by Vietnam Youth Union.
- 1998: Special Prize of National Informatics Olympiad for Universities.
- 1997: 1st Prize in National Informatics Contest for High School.
- 1994: 3rd Prize in National Mathematics Contest for Secondary School.
- 1994: 1st Prize for Problem Solving Contest of Newspaper for Mathematics and Youth.

Professional Experience

- 08.2005-10.2005: Guest Student Program at NIC/ZAM, Research Center Juelich.
- 12.2004-12.2005: Master Thesis at the Bioinformatics Department, Dsseldorf University. Thema: Parallelization and Improvements of IQPNNI to reconstruct phylogenetic trees.
- 05.2003-02.2004: "Hiwijob" for Prof.Dr. Stefan Leue at the Freiburg University with topic "Model Checking" with UML-Realtime.

- 09.2001-09.2002: Teaching Assistant at the Department of Technology, Vietnam National University, Hanoi. Member of Project "Linux OS: Researchs and Realization in Faculty of Technology and in Vietnam". Research on Data Mining, Parallel Computing and Vietseek's Search Engine.
- 05.2000-08.2001: Member of national Project: "Operating System Vietnam Linux". Responsible for Solutions to Vietnamese in Linux.