

ISMB 2007 - Tutorial: Introduction to Phylogenetic Networks

Daniel H. Huson

Center for Bioinformatics, Tübingen University,
Sand 14, 72075 Tübingen, Germany
www-ab.informatik.uni-tuebingen.de

July 21, 2007

Contents

Contents	1
0 Introduction	3
1 Phylogenetic trees	5
1.1 Phylogenetic trees	5
1.2 Aligned sequences	6
1.3 A simple model of evolution	7
1.4 The Jukes-Cantor model of evolution	7
1.5 The tree reconstruction problem	8
1.6 Tree reconstruction methods	9
1.7 Maximum parsimony methods	9
1.8 Maximum Likelihood and Bayesian methods	10
1.9 Distance-based methods	10
1.10 Software	11
2 Consensus networks and super networks	12
2.1 Additional evolutionary events	12
2.2 Gene trees can differ	13
2.3 The split encoding of a tree	13
2.4 Trees and splits	14
2.5 Representing incompatible splits	14
2.6 Consensus of trees	15
2.7 Consensus networks	15
2.8 Consensus super networks	16
2.9 Distance-based network methods	17
2.10 Software	18

- 3 Hybridization and reticulate networks 19**
 - 3.1 Speciation by hybridization 19
 - 3.2 A simple model of reticulate evolution 20
 - 3.3 Rooted reticulate network 21
 - 3.4 Network reconstruction problem 22
 - 3.5 Independent reticulations 22
 - 3.6 SPR's and independent reticulations 23
 - 3.7 Reticulate and split networks 23
 - 3.8 Application to plant data 24
 - 3.9 Software 25

- 4 Recombination networks 26**
 - 4.1 Recombination networks 26
 - 4.2 Example 1 28
 - 4.3 Example 2 29
 - 4.4 A branch-and-bound approach 30
 - 4.5 Example 3 30
 - 4.6 Software 31

- 5 Other types of networks 32**
 - 5.1 Reticulograms 32
 - 5.2 Augmenting species trees by gene trees 32

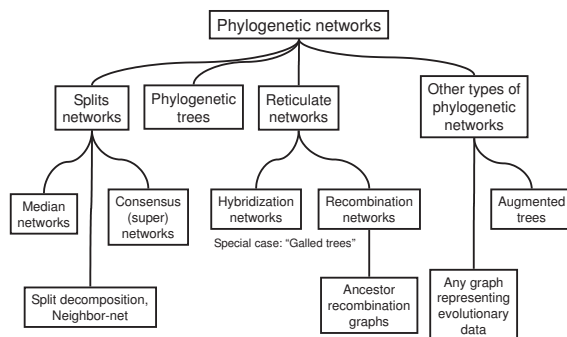
- Bibliography 34**

Chapter 0

Introduction

Phylogenetic networks are becoming an important tool in molecular evolution, as the role of reticulate events such as hybridization, horizontal gene transfer and recombination becomes more evident, and as the available data increases in quantity and quality [43].

Here is an illustration of some of the existing concepts:



The literature on phylogenetic networks is confusing, and this confusion has three sources:

1. There are many types of phylogenetic networks, including trees, split networks, median networks, median-joining networks, neighbor-nets, recombination networks, ARGs, hybridization networks, netting, reticulograms, etc.
2. The term “phylogenetic network” is appealing and so researchers are tempted to equate it to the specific types of networks that they are interested in. For example,
 - Gusfield [17] uses the general term *phylogenetic network* to mean a *recombination network*,
 - Linder and Rieseberg [42], use the same term to mean a *hybridization network*, and
 - Huber and Moulton equate phylogenetic network = *reticulate network with multi-edges* [24].

All three concepts can be derived from the more general definition of a *reticulate network*, which, in turn, is just one type of *phylogenetic network*.

3. More interestingly, there are two fundamentally different types of phylogenetic networks, and we propose to distinguish between:
 - *implicit networks*, that aim at visualizing incompatible data, and
 - *explicit networks*, that aim at providing an explicit model of “reticulate evolution”.

In this tutorial, we will first review *phylogenetic trees*.

We will then consider the concept of *consensus networks*, as a particular type of *split network*, which are a type of *implicit network* that are used to display incompatible phylogenies, and form the computational basis for other types of networks.

We then discuss how to analyze *hybridization* using *reticulate networks* based on multiple gene trees. These are a good example of *explicit networks* as describe evolutionary scenarios involving hybridization events.

Finally, we will look at obtaining *recombination networks* from binary sequence data.

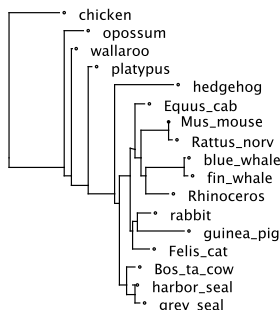
Related topics that are not covered include *ancestor recombination graphs (ARGs)* used in population genetics [25, 20, 15, 53, 54, 55], algorithms for detecting recombination sites in sequences [51, 59, 13], and methods that operate by first constructing a tree and then adding supplementary edges [48, 40, 19].

A book chapter based on this tutorial will appear in [29].

Chapter 1

Phylogenetic trees

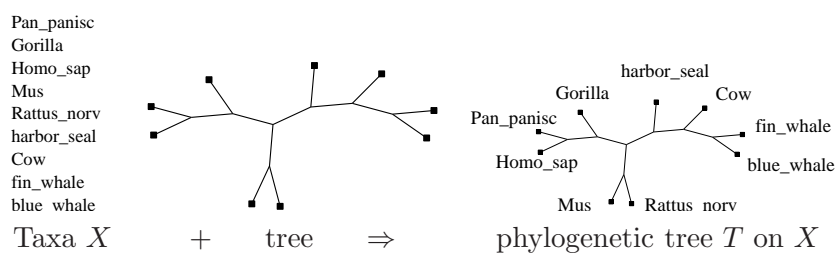
In this chapter we first introduce some basic definitions concerning trees and sequences. We then describe a very simple model of sequence evolution along a tree. We finally discuss some of the methods that are used to reconstruct a phylogenetic tree from a set of extant sequences.



1.1 Phylogenetic trees

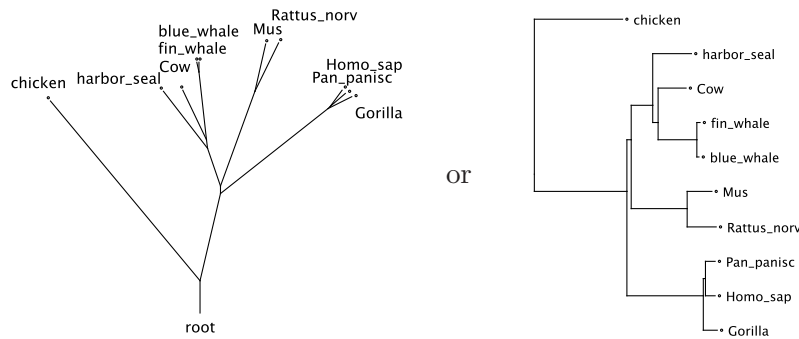
Throughout, let $X = \{x_1, \dots, x_n\}$ denote a set of *taxa*, in which each *taxon* x_i represents some species or organism whose evolutionary history is of interest to us.

For example, $X = \{x_1, x_2, \dots\}$ might denote a set of mammals, with x_1 representing a gorilla, x_2 a seal etc. A *phylogenetic tree* T on X (or X -tree) is obtained by labeling the leaves of a tree by the set X :

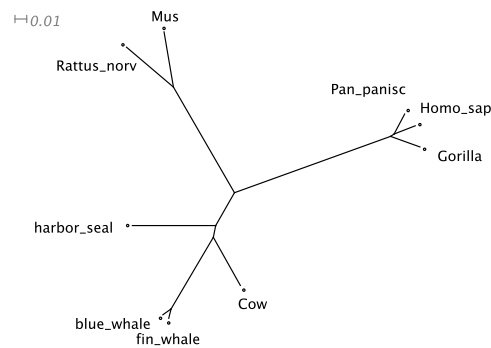


The above is an example of an *unrooted* tree. From a theoretical and algorithmic point of view, unrooted trees are easier to work with than “rooted” trees. In biology the latter are of more interest, as they define *clades* of related taxa.

One way to determine where to *root* a tree is to include an appropriate *outgroup* in the analysis and to place the root on the branch leading to the outgroup:



Each branch e of a phylogenetic tree T may be scaled to represent $r \times t$, the “rate of evolution” r multiplied by the time t along e :



A phylogenetic tree T is called *bifurcating* or *resolved*, if all its internal nodes (except the root) have degree 3, and *multifurcating* or *unresolved*, else.

1.2 Aligned sequences

In *molecular phylogenetics*, a set of taxa $X = \{x_1, \dots, x_n\}$ may be given as an alignment of molecular sequences of the form:

$$A = \begin{cases} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ & & \dots & \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{cases}$$

The sequences are usually obtained from some gene or locus that all taxa have in common. One popular sequence is the SSU rRNA molecule, which has proven to carry a robust phylogenetic signal.

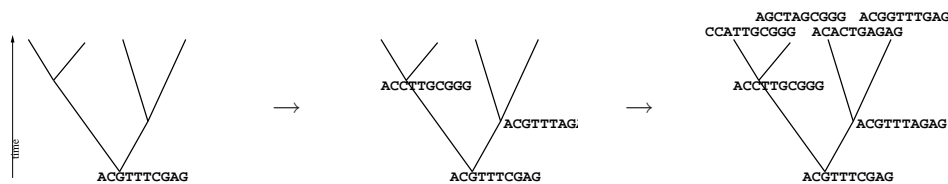
The problem of aligning sequences is non-trivial, but this question is beyond the scope of this tutorial.

Example:

Homo sap	fqtpmviilqaimgsatlamtliiftiiiiltvhdnttvptmitpmlt
Pan panisc	fqtpmiiifqaimgsatlaltliiftiiviltvhdnttavpttitpmlt
Gorilla	lqtpmviifqaimgsatlamtliiftvimiltvhetnttvptmiapmlt
harbor seal	fqlpmviifqaiiggatlalafitftiiiifltvhdtdtstlimilsmilt
Cow	fqtpmviifqaiiggatlalalalitftiiiifmtvhdtdtstlmlsmflt
fin whale	lqtfmviifqaimgettlalafitftiaifltvhdtdtstlmltilsmllt
blue whale	lqtfmviifqaimgettlvlaitftiaifltvhdtdtstlmltilsmllt
Rattus norv	fqismiiifqaimggatlvlaitftiilvfltvhdtdtstftitiissmat
Mus	fqismiiifqaimggatlvlaitftiilifltvhdtdtstftitiissmit

1.3 A simple model of evolution

Start with an ancestor sequence of length n at the root of a given tree. The sequence evolves up the tree, experiencing point-mutations along the way, at a fixed rate:



This model allows for only two types of events, namely *mutations* and *speciation events* (at the nodes of the tree).

1.4 The Jukes-Cantor model of evolution

T. Jukes and C. Cantor [35] formalized such a simple model of DNA sequence evolution:

Definition Let T_0 be a rooted phylogenetic tree. The *Jukes-Cantor* model of evolution makes the following assumptions:

1. The possible states for each site are A, C, G and T.
2. The initial sequence length is an input parameter and for each site the state at the root is drawn from a given distribution (typically uniform).
3. The sites evolve identically and independently (i.i.d.) up the branches of the tree from the root at a fixed rate u .
4. With each branch $e \in E$ we associate a duration $t = t(e)$ and the expected number of mutations per site along e is $ut(e)$. The probabilities of change to each of the 3 remaining states are equal.

How do we “evolve” a sequence up a branch e under the Jukes-Cantor model?

Let $a = a_1a_2 \dots a_n$ and $b = b_1b_2 \dots b_n$ denote the source and target sequences associated with e . We assume that a has already been determined and we want to determine b .

Under the Jukes-Cantor model, the evolutionary event

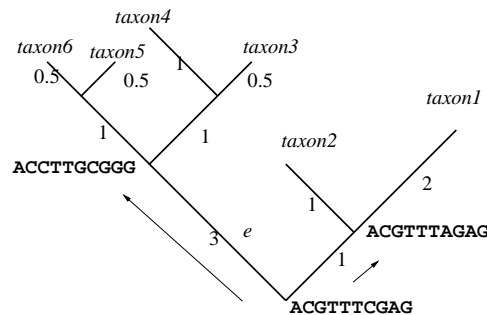
nucleotide changes to one of the other three bases

occurs at a fixed rate u .

From this, we obtain a *probability-of-change formula* for the probability of an *observable* change occurring at any given site in time t :

$$\text{Prob}(\text{change} \mid t) = \frac{3}{4} \left(1 - e^{-\frac{4}{3}ut} \right).$$

This model can be used to “evolve” sequences along a model tree T_0 . Consider the following example with $u = 0.1$:



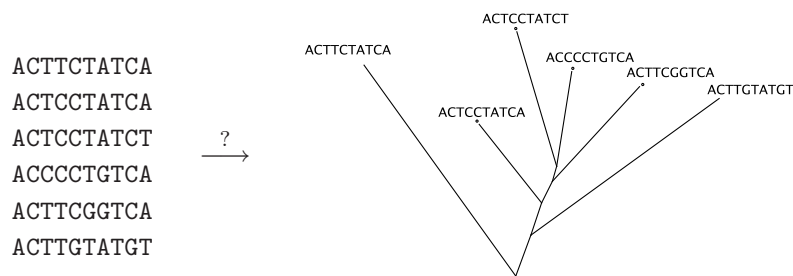
The root node is assigned a random sequence. Then the sequences are evolved up the branches, using the probability-of-change formula to decide whether to “mutate” a given base.

E.g., the probability of change along the branch labeled e is

$$0.75(1 - e^{-\frac{4}{3} \times 0.1 \times 3}) = 0.75(1 - e^{-0.4}) = 0.247.$$

1.5 The tree reconstruction problem

Given a set of sequences that were generated along some model tree T_0 according to some model, can the model tree be reconstructed?



More precisely, the challenges are:

- determine the unrooted topology of T_0 ,
- estimate the branch lengths of T_0 , and to
- correctly determine the position of the root in T_0 .

1.6 Tree reconstruction methods

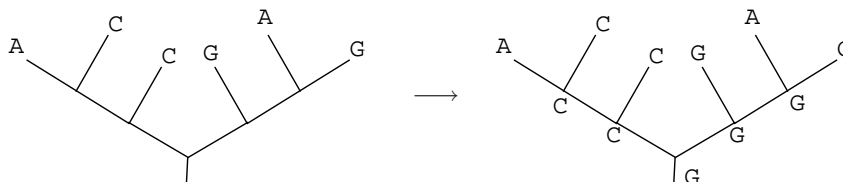
There exist many different approaches to this problem [12]:

- *Sequence-based* methods search for a tree that optimally explains the given sequence data, such as:
 - maximum parsimony [9],
 - maximum likelihood [10], and
 - Bayesian inference [27].
- *Distance-based* methods infer a distance matrix from the input data and then *construct* a tree from the matrix, such as:
 - UPGMA [52]
 - Neighbor-Joining [50] and its variants Bio-NJ [14] and Weighbor [3].

1.7 Maximum parsimony methods

Maximum parsimony methods search for a phylogenetic tree T on X that “explains” an alignment A using a minimum number of evolutionary events.

For any fixed tree T , a most parsimonious explanation of any column of the alignment A is easily computed:



However, all possible trees on X must potentially be considered to find the optimal one!

1.8 Maximum Likelihood and Bayesian methods

Any maximum-likelihood or Bayesian method is based on an explicit model of evolution, such as the Jukes-Cantor model.

In the maximum-likelihood approach, one computes the “likelihood” $P(T | A)$ that the true tree is T , given that the alignment A was observed. The method returns:

$$T_{ML} = \arg \max_T P(A | T).$$

More desirable is the tree T that maximizes the probability of generating the data A (computed using Bayes’ Theorem):

$$T_{Bayesian} = \arg \max_T P(T | A).$$

Both approaches are computationally very expensive.

1.9 Distance-based methods

First compute a distance matrix D from the alignment A . The simplest approach for DNA is to use the *Hamming distance* $Ham(a, b)$, i.e. the proportion of positions at which two sequences a and b differ.

This *underestimates* the true evolutionary distance (number of mutations that took place), as back mutations and multiple mutations at the same position are not counted. Thus, correction formula based on some model of evolution are used.

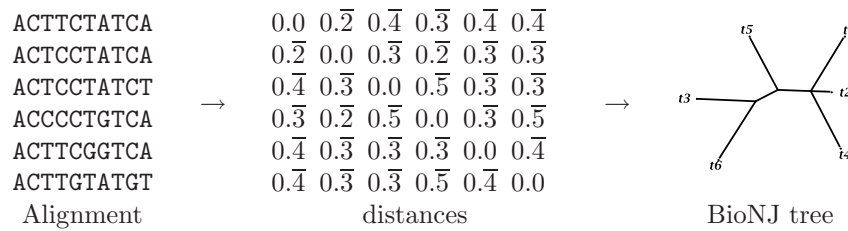
For example, in the case of the Jukes-Cantor model, observed Hamming distances between sequences are transformed thus:

$$JC(a, b) = -\frac{3}{4} \ln \left(1 - \frac{4}{3} Ham(a, b) \right).$$

Let D be a distance matrix for taxa X obtained from an alignment A . The goal is to construct a phylogenetic tree T in which the path lengths between taxa in T approximate the distances between taxa in D .

The most popular methods for computing such a tree are UPGMA [52] and Neighbor-Joining (NJ).

Both are fast algorithms that use a hierarchical clustering approach. UPGMA is most suitable, when the sequences evolved under the assumption of a “molecular clock”. NJ and its variants are more widely applicable and are popular due to their speed.



1.10 Software

Here is a small selection of software that build phylogenetic trees:

- PAUP* [56], a program for performing phylogenetic analysis using parsimony, maximum likelihood and other methods,
- Phylip [11], a package for phylogenetic inference,
- MrBayes [26], a program for Bayesian inference of trees,
- Mesquite [46], a modular system for evolutionary analysis,
- PAL [8], an object-oriented programming library for molecular evolution and phylogenetics, and
- SplitsTree4 [28, 30], an integrated program for estimating phylogenetic trees and networks.

Chapter 2

Consensus networks and super networks

In this chapter we first discuss additional evolutionary events that are not considered in simple models such as the one proposed by Jukes and Cantor.

This will lead us to the fundamental observation that:

gene trees differ.

Hence, it may not be adequate to represent a set of gene trees by a single consensus tree, as is sometimes done, and we will discuss how to represent the conflicting signals using a “consensus network” or “super network”.

Finally, we will briefly look at some other methods that use a network to represent conflicting signals.

2.1 Additional evolutionary events

Models such as the Jukes-Cantor one are usually understood to represent the evolution of a *single* gene. They don't consider insertions and deletions, or more complicated events.

If one studies more than one gene simultaneously, additional evolutionary events must be taken into account. E.g.:

- individual genes may be *born*, *duplicated* or *lost*.

Moreover, biological mechanisms such as

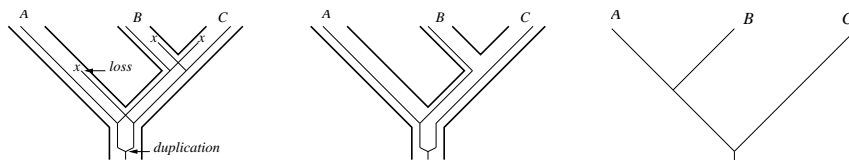
- *recombination*,
- *hybridization*, or
- *horizontal gene transfer*

may be involved.

2.2 Gene trees can differ

Now, suppose we are given one or more genes for X . Consider a model in which the sequence of a gene evolves via mutations, but we also allow gene *duplication* or *loss*.

The true phylogeny of a gene can differ from the model phylogeny. here we depict a species phylogeny using bold parallel lines and the history of a single gene by thin lines:



So, true “gene trees” can differ from the true “species phylogeny” and also from each other.

2.3 The split encoding of a tree

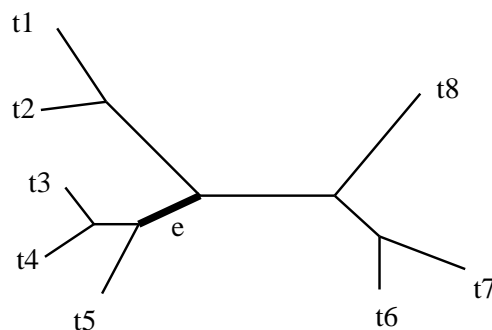
Let $X = \{x_1, \dots, x_n\}$ be a set of taxa and g_1, \dots, g_k a set of genes that are present in all taxa. For each gene g_i we are given a sequence alignment A_i .

Assume that we have reconstructed a phylogenetic tree T_i for each gene g_i . The goal is to compute a *consensus* of these trees. To this end, we introduce the following concepts.

An X -split $S = \frac{A}{B} (= \frac{B}{A})$ is a bipartitioning of X with [2]:

$$A, B \neq \emptyset, A \cap B = \emptyset \text{ and } A \cup B = X.$$

Any edge e of T defines a split $S = \frac{A}{B}$, where A and B are the sets of taxa contained in the two sub-trees defined by e . E.g.:

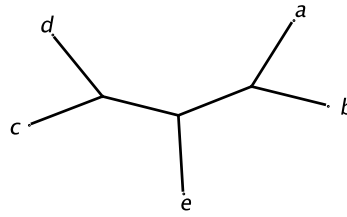


For the edge labeled e we get:

$$A = \{t_3, t_4, t_5\} \text{ and } B = \{t_1, t_2, t_6, t_7, t_8\}.$$

Let $\Sigma(T)$ denote the *split encoding* of T , i.e. the set of all splits obtained from T .

Consider the tree T :



The split encoding $\Sigma(T)$ contains 5 *trivial* splits and 2 *non-trivial* ones. The trivial splits are:

$$\frac{\{a\}}{\{b, c, d, e\}}, \frac{\{b\}}{\{a, c, d, e\}}, \frac{\{c\}}{\{a, b, d, e\}}, \frac{\{d\}}{\{a, b, c, e\}} \text{ and } \frac{\{e\}}{\{a, b, c, d\}},$$

and the non-trivial ones are:

$$\frac{\{a, b\}}{\{c, d, e\}} \text{ and } \frac{\{a, b, e\}}{\{c, d\}}.$$

2.4 Trees and splits

Two different X -splits $S = \frac{A}{B}$ and $S' = \frac{A'}{B'}$ are *compatible*, if “one is a refinement of the other”, that is, if one of the four following inclusions holds:

$$A \subset A', A \subset B', B \subset A', \text{ or } B \subset B'.$$

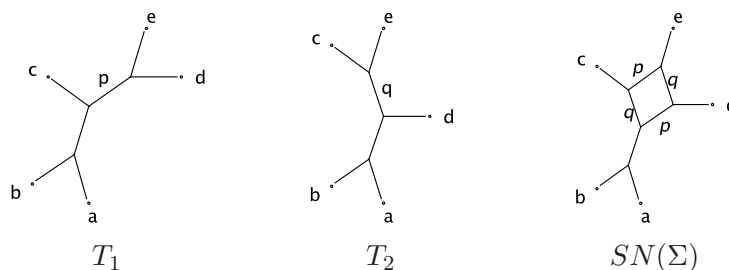
This is an important concept, as we have:

Lemma Let Σ be a set of X -splits. Then there exists an unique X -tree T with $\Sigma(T)$ iff Σ is compatible [5].

2.5 Representing incompatible splits

Any *compatible* set of X -splits can be represented by a phylogenetic tree. What about incompatible splits sets?

Consider the following two trees T_1 and T_2 , for which the splits $S_p = \frac{\{a,b,c\}}{\{d,e\}} \in \Sigma(T_1)$ and $S_q = \frac{\{a,b,d\}}{\{c,e\}} \in \Sigma(T_2)$ are incompatible:



The “split network” $SN(\Sigma)$ represents the incompatible set of splits $\Sigma := \Sigma(T_1) \cup \Sigma(T_2)$, using “bands of parallel edges” to represent splits that are incompatible with others [7].

2.6 Consensus of trees

A collection of trees $\mathcal{T} = \{T_1, \dots, T_K\}$ is often summarized using a consensus tree.

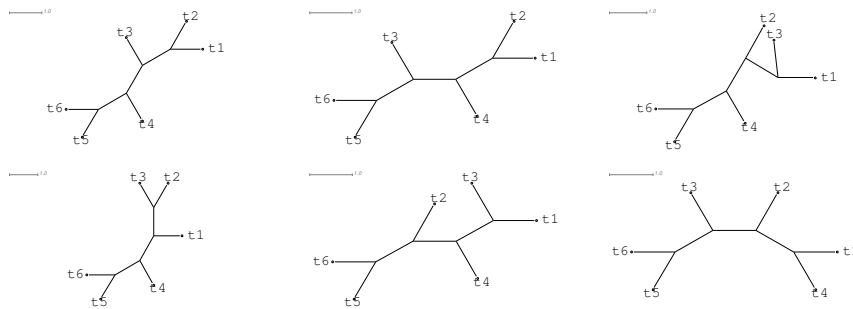
Let $\Sigma_{all} = \cup_{T \in \mathcal{T}} \Sigma(T)$ be the set of all present splits.

Let $\Sigma(p) = \{S \in \Sigma_{all} : |\{T \in \mathcal{T} : S \in \Sigma(T)\}| > pK\}$ be the set of splits that occur in more than a proportion p of all trees. Then,

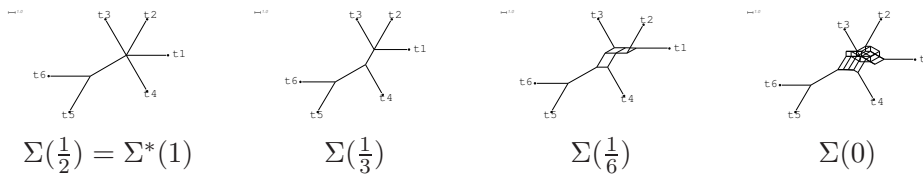
- $\Sigma^*(1) := \bigcap_i \Sigma(T_i)$ defines the *strict consensus*,
- $\Sigma(\frac{1}{2})$ defines the *majority consensus*, and, more generally,
- $\Sigma(\frac{1}{d+1})$ ($d \geq 0$) defines a set of *consensus splits*.

Note that both $\Sigma^*(1)$ and $\Sigma(\frac{1}{2})$ are always compatible and thus correspond to trees, whereas $\Sigma(\frac{1}{d+1})$ with $d \geq 0$ may be incompatible and thus is usually represented by a network.

For example, given these trees as input:



We get these consensus trees and networks:



2.7 Consensus networks

Often, a set of trees $\mathcal{T} = \{T_1, \dots, T_K\}$ is summarized using a consensus *tree*.

This may not always be appropriate, as gene trees are not necessarily different estimations of the same true phylogeny, but may differ substantially for biological reasons.

A *consensus network* is obtained by computing the consensus splits $\Sigma(\frac{1}{d+1})$ for some fixed value $d \geq 0$.

The parameter d sets the *maximum dimensionality* of the corresponding network: for $d = 1$ the network will be 1-dimensional, hence a tree, for $d = 2$ the network may contain parallelograms, and in general it may contain cubes of dimension $\leq d$ [22, 21].

2.8 Consensus super networks

Consider a set of taxa $X = \{x_1, \dots, x_n\}$ and a set of genes $G = \{g_1, \dots, g_t\}$.

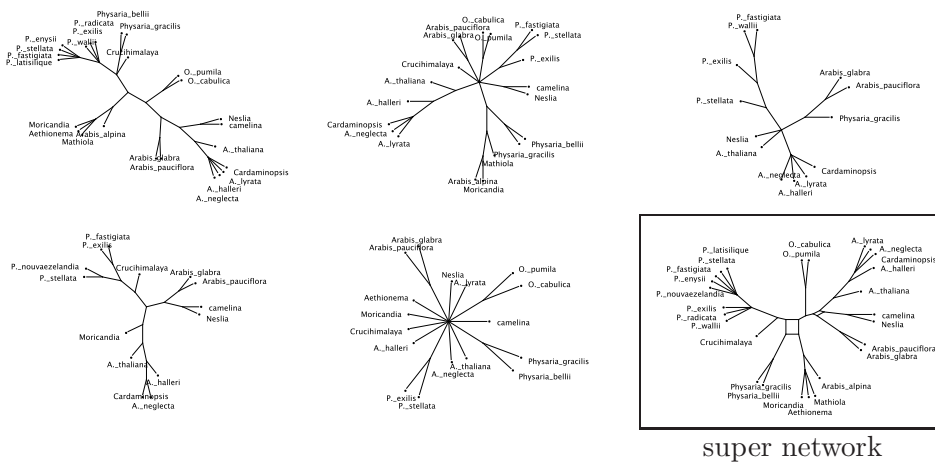
It is often the case that a given gene is not available for all taxa and the alignment A_i associated with some gene g_i only contains sequences for a subset $X' \subset X$. Then, any X' -tree inferred from A_i is called a *partial X-tree*.

For a collection of *partial trees* $\mathcal{T} = \{T_1, \dots, T_K\}$, the consensus methods above do not apply.

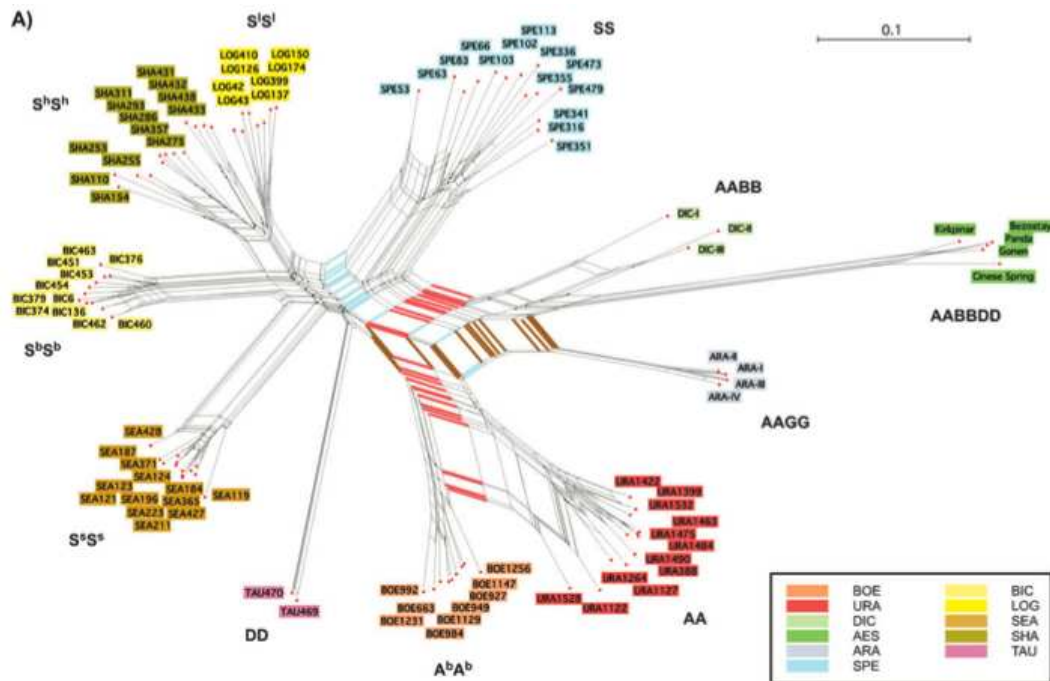
One alternative is to compute an optimal *super tree* T that “optimally” summarizes the set of input trees.

A second approach is to summarize the input trees in terms of a *super network* that attempts to represent as many of the input “partial” splits as possible.

The *Z-closure method* [31] takes as input a set of partial X -trees $\mathcal{T} = \{T_1, \dots, T_K\}$ and produces as output a set of X -splits Σ . Here is an example of five partial gene trees and a summarizing super network:



Here is a super network that was computed to identify multiple events of pseudogene evolution in the Brassicaceae [37]:



2.10 Software

- SplitsTree4 [30] provides implementations of *all* methods described in this chapter, including a number of different algorithms for constructing networks from splits.
- SpectroNet [23] provides an algorithm for constructing a split network (a special case, namely the median network) and some related methods

Chapter 3

Hybridization and reticulate networks

In this chapter we first discuss the concept of hybrid speciation. We then describe a simple model of evolution that incorporates gene trees and reticulation events.

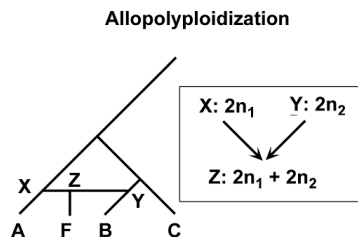
We then introduce the concept of a reticulate network and discuss some approaches for inferring such networks from gene trees.

Finally, we give an overview over the available software.

3.1 Speciation by hybridization

There are two main mechanisms of speciation by hybridization.

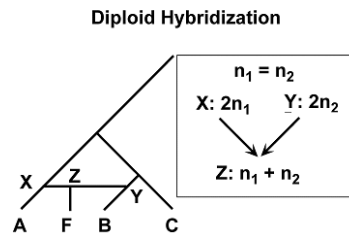
In *allopolyploidization*, the hybrid speciation occurs when two different lineages produce a new species that has the complete nuclear genomes of both parental species:



Thus, two parents X and Y each pass on their whole diploid genomes, with $2n_1$ and $2n_2$ chromosomes, respectively, to produce a polyloid offspring Z with $(2n_1 + 2n_2)$ chromosomes.

Subsequently, it can happen that the genome reduces to half its size and is then a mosaic of genes from both ancestors.

In *diploid* (or *homoploid*) hybrid speciation, each of the parents produces normal gametes (haploid) to produce a normal diploid hybrid:



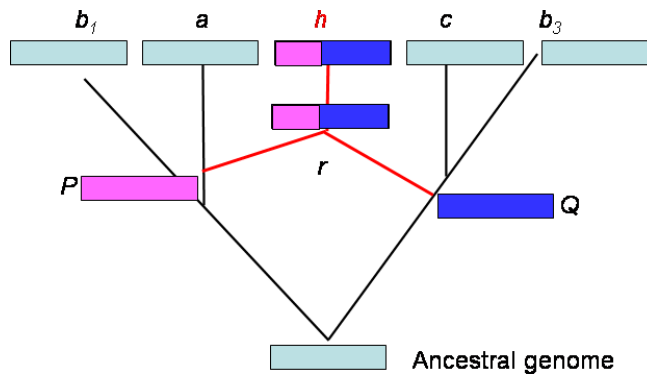
Although diploid hybridization is more common, the ability of the hybrid to backcross with the parent species usually prevents that a new species will arise. Although less common, allopolyploidization is believed to produce more new species.

Hybridization is usually restricted to plants, frogs and fish.

3.2 A simple model of reticulate evolution

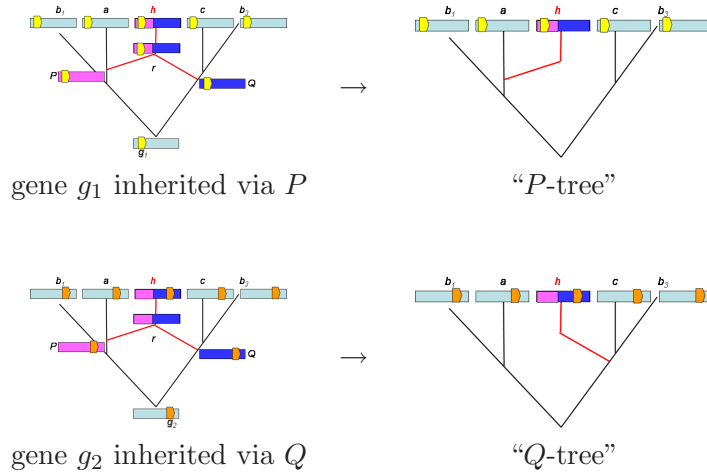
We will describe a simple model of evolution that allows *reticulate* events such as hybridization, and, later, recombination.

Consider the following network:



In such a *reticulate network* N , a *reticulate node* r inherits sequence from two different ancestors P and Q .

We will assume that genes are “atomic” w.r.t. reticulation and thus that the evolutionary history of any given gene is a tree:



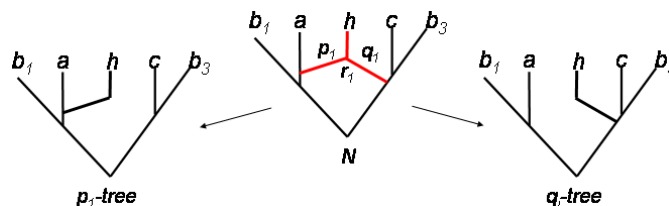
3.3 Rooted reticulate network

Definition Let X be a set of taxa. A (*rooted*) *reticulate network* N on X is a connected, directed acyclic graph with:

- there exists precisely one node of indegree 0, called the *root*,
- all other nodes are *tree nodes* of indegree 1, or *reticulation nodes* of indegree 2,
- every edge is either a *tree edge* incident to precisely two tree nodes, or a *reticulation edge* leading from a tree node to a reticulation node, and
- the set of *leaves* (nodes of outdegree 0) consists only of tree nodes and is labeled by X .

Let N be a reticulate network on X with k reticulation nodes r_1, \dots, r_k . For any such node r_i , let p_i and q_i denote the two associated reticulation edges.

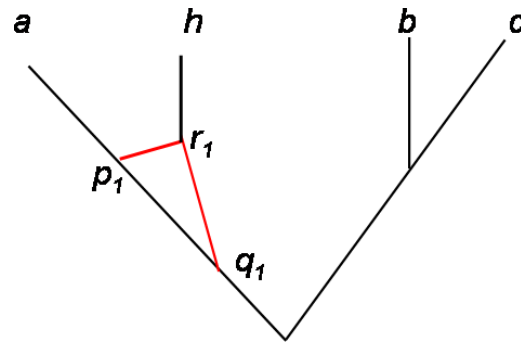
We can obtain an X -tree from N by choosing and removing one reticulation edge p_i or q_i for each r_i .



The set of trees $\mathcal{T} = \mathcal{T}(N)$ obtainable in this way is called the set of *induced trees* or trees that *can be sampled* from N .

Lemma The number of different trees that can be sampled from a network N with k reticulations is $|\mathcal{T}(N)| = 2^k$.

Note, however, that “different” trees may have the same topology and only differ in their branch lengths:



Here, both induced trees are of the form: $((a, h), (b, c))$.

3.4 Network reconstruction problem

Given a set of trees $\mathcal{T} = T_1, \dots, T_m$, we would like to determine the reticulate network N from which the trees were sampled.

This form of the problem is not always solvable, e.g. if some of the 2^k possible trees are missing. Thus we consider the following:

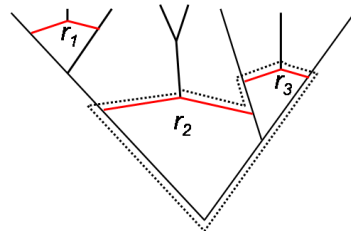
Most Parsimonious Network Problem:

Determine a reticulate network N such that $\mathcal{T} \subseteq \mathcal{T}(N)$ and N contains a minimum number of reticulation nodes.

In fully generality, this is known to be a hard problem [57]. We now discuss a special case that can be solved efficiently.

3.5 Independent reticulations

Two reticulation nodes r_i, r_j in N are *independent* of each other, if they are not contained in any common undirected cycle. Consider:

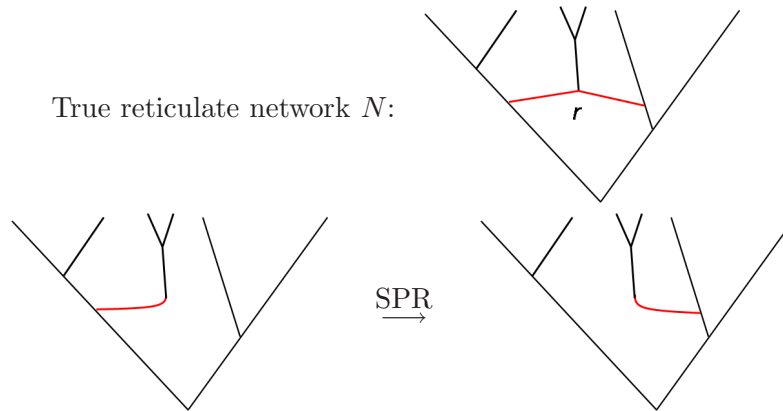


Here, r_1 is independent of r_2 and r_3 , whereas r_2 and r_3 are not independent of each other, as the highlighted cycle shows.

A reticulation that is independent of all others is called a *gall* and a network N in which all reticulations are galls is called a *galled tree* [17] or *gt-network* [49].

3.6 SPR's and independent reticulations

Wayne Maddision [47] considered the situation in which the true reticulate network N contains only a single reticulation and observed that an independent reticulation corresponds to a *sub-tree prune and regraft* (SPR) operation:



Maddision's algorithm:

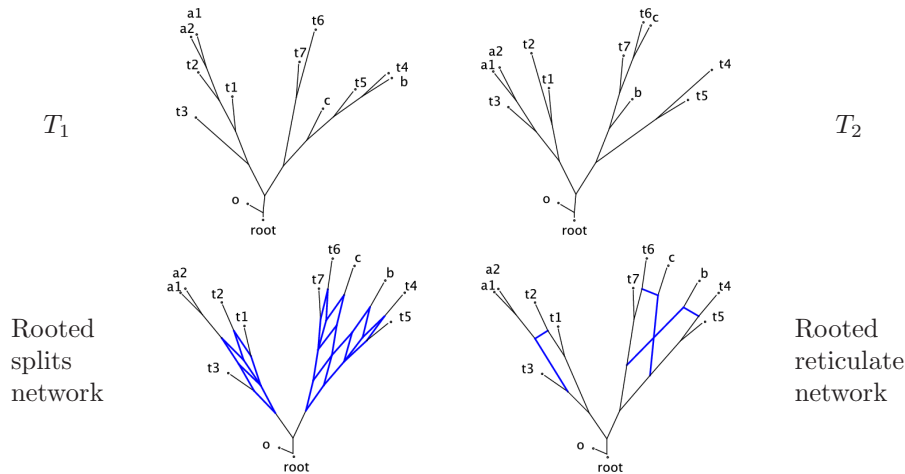
- Given two bifurcating trees, compute their SPR distance
- If the distance is 0, return a tree
- If the distance is 1, return a network
- Else, fail

This approach has been generalized to networks with multiple independent reticulations ("galled trees") [49].

Unfortunately, on real data, such algorithms will often return "fail". One challenge is to produce useful output in the case of imperfect data.

3.7 Reticulate and split networks

There is a nice relationship between a reticulate network N and the network of all splits of all trees sampled from N [16, 32]:



More precisely, there exists a one-to-one correspondence between the “netted regions” of the split network and the “tangles” of dependent reticulations of the reticulate network.

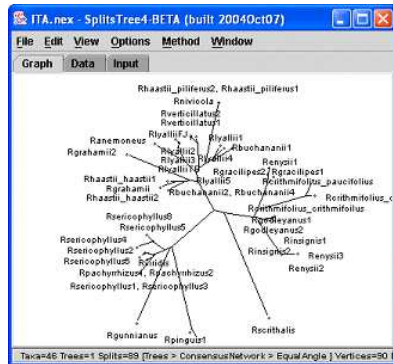
This leads to the following approach:

- Determine the set of all input splits
- Determine the netted components of the split network
- Analyze each component C separately:
- If C can be explained by a reticulate network $N(C)$, then locally replace C by $N(C)$

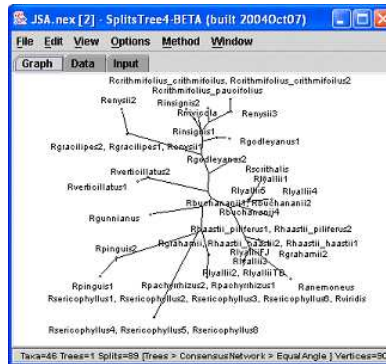
Using an algorithm that allows “overlapping” reticulations [32], this approach is implemented in the program SplitsTree4.

3.8 Application to plant data

Given two trees on *Ranunculus* (buttercup) data [44]:

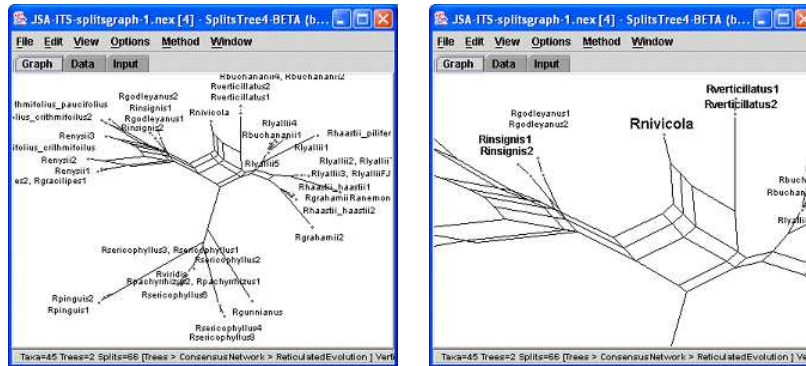


Nuclear ITS gene



chloroplast J_{SA} region

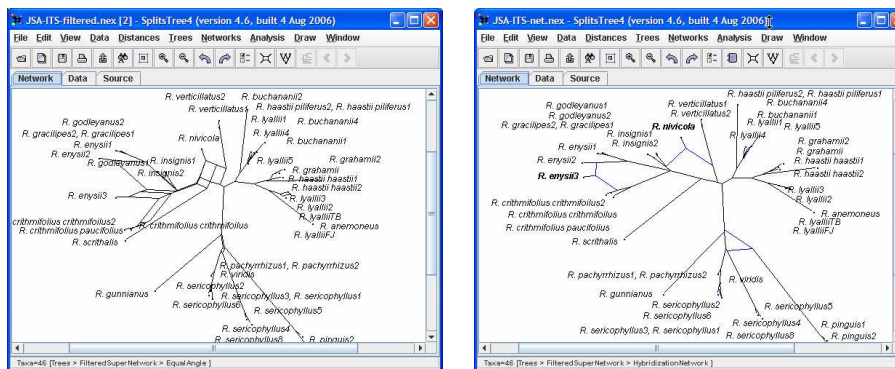
We obtain the following split network:



This split network suggests that *R. nivicola* may be a hybrid of the evolutionary lineages on the left- and right-hand sides.

Current algorithms are sensitive to false branches in the input trees and here initially no reticulation is detected.

However, a new filtering technique aims at removing all incompatibilities that cannot be explained by a simple hybridization scenario [34]:



This agrees with earlier suggestions [44] that *R. nivicola* is an allopolyploid formed between *R. insignis* and *R. verticillatus*, and that *R. enysii 3* is a diploid hybrid between *R. enysii 2* and *R. crithmifolius*.

3.9 Software

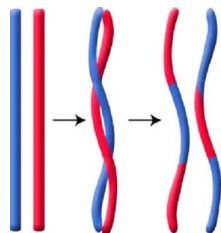
Here is an overview over publicly available software for constructing reticulation networks from trees:

- SplitsTree4 [30] provides a method *HybridizationNetwork* that takes as input a list of trees or partial trees and produces as output a reticulation network, in which any unresolvable tangles are represented by their split network, as illustrated above.
- In [49] a program SPNet is described, but it is not available for download.

Chapter 4

Recombination networks

In this chapter we will look at the problem of reconstructing a reticulate network from an alignment of binary sequences that have evolved under a model of mutation-, speciation- and *recombination* events:



This has been much studied in population genetics [25, 20, 15, 53, 54, 55] and *ancestor recombination graphs* (ARGs) are used in that context.

4.1 Recombination networks

For ease of exposition, we will concentrate on the combinatorial aspects of the problem and thus consider *recombination networks* rather than ARGs.

We will make some simplifying assumptions:

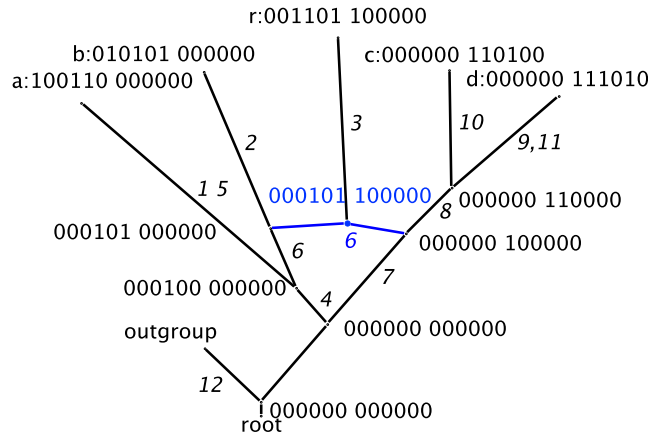
- all sequences have a common ancestor,
- any position can mutate at most once, and
- recombinations are always single crossovers.

Given an alignment A of binary sequences of length n , a *recombination network* R can be viewed as a *reticulation network* N , together with [6]:

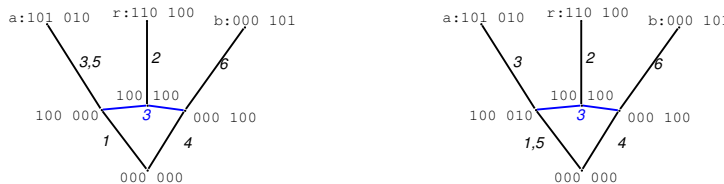
- a labeling of all nodes by binary sequences of length n , such that the leaves of R are labeled by A ,
- a corresponding labeling of each tree edge e by those positions that mutate along e , and

- a corresponding labeling of each reticulation node r indicating the crossover position for the recombination at r .

Example of six sequences a, b, c, d, r and *outgroup*, of length 12:



Interestingly, the placement of mutations on edges is not uniquely defined. In this graph the mutation at position 5 can happen along two different edges:



Mutation on leaf edge Mutation inside reticulation cycle

Faced with this choice, current algorithms [17, 33] place such ambiguous mutations outside of the *reticulation cycle*.

In the case of independent reticulations, Dan Gusfield and colleagues have developed an algorithm for computing a galled tree from binary sequences [17, 16].

This tree-based approach computes a galled tree as follows:

- Determine the components of the “incompatibility graph”
- For each component C do:
 - Determine the restriction of the dataset with respect to C
 - Determine whether removing on taxon produces a perfect phylogeny
 - If so, arrange the taxa in a gall
 - Return a description of the network

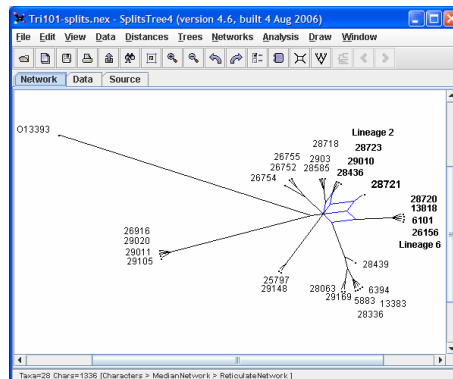
An alternative splits-based approach is to first construct the underlying reticulate network using the approach briefly outlined in the context of hybridization networks [32, 33] and then computing the appropriate labeling of nodes and edges.

4.2 Example 1

The following data set consists of the 64 non-constant sites in the alignment of TRI101 sequences for different strains of the fungus *F. graminearum* and one outgroup sequence O13393 representing *F. lunulosporum* (from [38]):

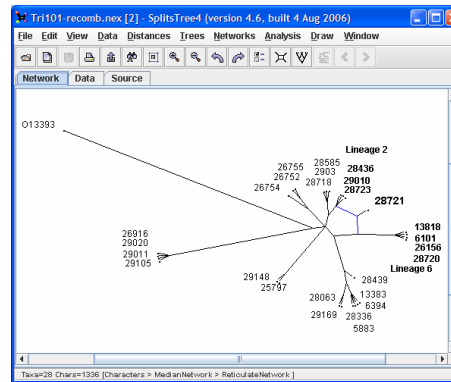
Strain	Non-constant positions of alignment
28436	gaccatcacgatgtgggtgggctcctgaaccccaactactttcagaccacctggttgtggcg
28723
29010
2903	...g.....c.....
28585	...g.....c.....
28718	...g.....c.....
25797	t.t....t.c...a.....t.a.....
29148	t.t....t.c...a.....t.a.....a
29020	.g...g....c.....tt...a.....c.tt...tt.t...a.ca..
26916	.g...g....c.....tt...a.....c.tt...tt.t...a.ca..
29011	.g...g....c.....tt...a.....c.tt...tt.t...a.ca..
29105	.g...g....c.....tt...a.....c.tt...tt.t...a.ca..
26752g..gc.....t.....t.....
26754g..a.c.....t.....tc.....
26755g..gc..a.....t.....t.....
6101g..c....c...g.....a.....tt..c.....
13818g..c....c...g.....a.....tt..c.....
26156g..c....c...g.....a.....tt..c.....
28720g..c....c...g.....a.....tt..c.....
28721g..c....c...g.....a.....tt..c.....
5883	..t.....ca....a.g.t...t.....t.....c.....
6394	..t.....ca....a.g.t...t.....t.....c.....
13383	..t.....ca....a.g.t...g.....t.....c.....
28063	..t.....c.....a.g.....t.g.....t.....c.....
28336	..t.....ca....a.g.....t.....t.....c.....
28439c.....a.g.....t.....t.....c.....
29169	..t.....c.....a.g.....t.g.....t.....c.....
O13393c.c..a.a...t.a.gg.t...g.tcggc.c..cgtt.c.t...c.c..a.

Loading this data in to the SplitsTree program and selecting the *MedianNetwork* method we obtain the following graph:



This is a simple dataset that can be explained using only one single-crossover

recombination:



Application of the recombination network algorithm implemented in [30], computes a recombination network that correctly displays strain 28721 as resulting from a hybrid of the lineages 2 and 6. As the computed network contains a single isolated reticulation, it is a “galled tree” and is therefore also obtainable by Gusfield’s algorithm [16].

4.3 Example 2

A second example dataset is taken from restriction maps of the rDNA cistron (length $\approx 10kb$) of twelve species of mosquitoes using eight 6bp recognition restriction enzymes [41]. Of 26 scored sites, 18 were polymorphic among the ingroup taxa:

<i>Aedes albopictus</i>	11110101010100010101010010
<i>Aedes aegypti</i>	11110101000100010101000010
<i>Aedes seatoi</i>	11110101010100010101010000
<i>Aedes flavopictus</i>	11110101010100010101010010
<i>Aedes alcasidi</i>	11110101010100010101010000
<i>Aedes katherinensis</i>	11110101010100010101010000
<i>Aedes polynesiensis</i>	11110101000100010101010010
<i>Aedes triseriatus</i>	10110101000110010101000000
<i>Aedes atropalpus</i>	10110101000100010111000010
<i>Aedes epactius</i>	10110101000100010111000010
<i>Haemagogus equinus</i>	10110101000110010101010000
<i>Armigeres subalbatus</i>	10110101000100010101000000
<i>Culex pipiens</i>	11110111000100011101001011
<i>Tripteroides bambusa</i>	11110111000100010101000010
<i>Sabethes cyaneus</i>	11110101001100010101010000
<i>Anopheles albimanus</i>	11011101100101110101110100

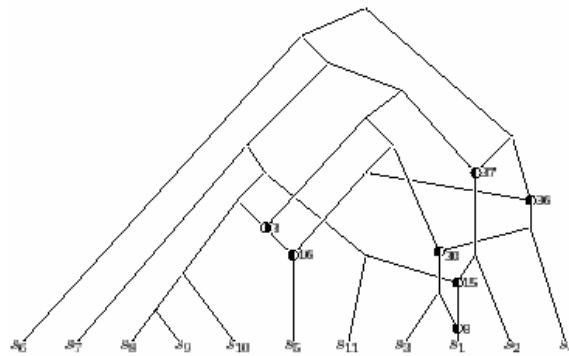
Here is the split network that represents this dataset, in which the edges have been labeled by the “mutations” that define the split represented by the edges:


```

a 0001000100000000
b 0100000100000000
c 0000000000000010
d 0000001000000010
e 00111111000000001
f 0100010001010111
g 0100010011111101
h 1111110011111101
i 1111010011111101

```

This is known to be a very difficult data set that cannot be explained using a small number of simple crossover events. A complicated scenario involving 7 events can be found using the branch-and-bound approach:



4.6 Software

Here is an overview over software for computing a recombination network from binary sequences:

- Software implementing the approach of Dan Gusfield and colleagues [17, 16] for constructing galled trees is available from: www.csif.cs.ucdavis.edu/~gusfield.
- SplitsTree4 [30] contains a method *RecombinationNetwork* for constructing galled trees and more general between reticulations [32, 33] recombination network, from binary sequences. Available from: www.splitstree.org
- Beagle [45] uses branch-and-bound to compute a recombination network, available from: www.stats.ox.ac.uk/~lyngsoe/beagle

Chapter 5

Other types of networks

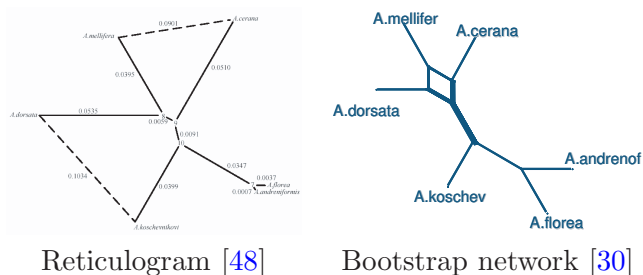
In the previous chapters, we have looked at the main types of phylogenetic networks, trees, split networks and reticulate networks.

Other types of networks exist. Here we look at two examples of what one might call *augmented trees*.

5.1 Reticulograms

A *reticulogram* is a phylogenetic tree with additional short-cut edges. It is computed from a distance matrix in two steps. First, a phylogenetic tree is constructed using a method such as neighbor-joining. Second, additional edges are added to the tree so as to optimize the least square fit of the path distances to the ones in the distance matrix. The method is implemented in the program *T-Rex* [48].

A comparison of the reticulogram and a bootstrap network for 677 bases of DNA from six honey bee species [58]:

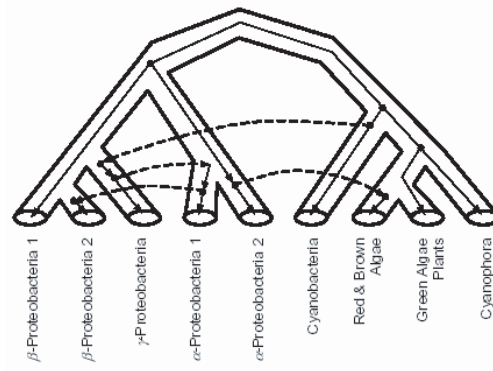


5.2 Augmenting species trees by gene trees

Given a set of gene trees and a fixed species tree, the goal is to map the gene trees on to the species tree, using a minimum set of horizontal gene transfer events to account for incongruencies between the gene trees and the species tree.

The program *latrans* implements this approach [18].

Here is a horizontal gene transfer scenario for the *rbcl* gene presented in [18]:



Bibliography

- [1] V. Bafna and V. Bansal. The number of recombination events in a sample history: conflict graph and lower bounds. *IEEE/ACM Transactions in Computational Biology and Bioinformatics*, 1(2):78–90, 2004.
- [2] H.-J. Bandelt and A. W. M. Dress. A canonical decomposition theory for metrics on a finite set. *Advances in Mathematics*, 92:47–105, 1992.
- [3] W. J. Bruno, N. D. Socci, and A. L. Halpern. Weighted Neighbor Joining: A likelihood-based approach to distance-based phylogeny reconstruction. *Molecular Biology and Evolution*, 17(1):189–197, 2000.
- [4] D. Bryant and V. Moulton. NeighborNet: An agglomerative method for the construction of planar phylogenetic networks. In R. Guigó and D. Gusfield, editors, *Algorithms in Bioinformatics, WABI 2002*, volume LNCS 2452, pages 375–391, 2002.
- [5] P. Buneman. The recovery of trees from measures of dissimilarity. In F. R. Hodson, D. G. Kendall, and P. Tautu, editors, *Mathematics in the Archaeological and Historical Sciences*, pages 387–395. Edinburgh University Press, 1971.
- [6] S. Eddhu D. Gusfield and C. Langley. The fine structure of galls in phylogenetic networks. *INFORMS J. of Computing Special Issue on Computational Biology*, 16(4):459–469, 2004.
- [7] A. W. M. Dress and D. H. Huson. Constructing splits graphs. *IEEE/ACM Transactions in Computational Biology and Bioinformatics*, 1(3):109–115, 2004.
- [8] A. Drummond and K. Strimmer. PAL: An object-oriented programming library for molecular evolution and phylogenetics. *Bioinformatics*, 17:662–663, 2001.
- [9] A.W.F. Edwards and L.L. Cavalli-Sfroza. The reconstruction of evolution. *Annals of Human Genetics*, 27:105–106, 1963.
- [10] A.W.F. Edwards and L.L. Cavalli-Sfroza. Reconstruction of evolutionary trees. In V.H. Heywood and J. McNeill, editors, *Phenetic and Phylogenetic Classification*, volume 6, pages 67–76. Systematics Association, London, 1964.

- [11] J. Felsenstein. PHYLIP – phylogeny inference package (version 3.2). *Cladistics*, 5:164–166, 1989.
- [12] J. Felsenstein. *Inferring Phylogenies*. Sinauer Associates, Inc., 2004.
- [13] K. Forslund, D.H. Huson, and V. Moulton. VisRD - visual recombination detection. *Bioinformatics*, 20(18):3654–3655, 2004.
- [14] O. Gascuel. BIONJ: An improved version of the NJ algorithm based on a simple model of sequence data. *Mol. Biol. Evol.*, 14:685–695, 1997.
- [15] R. C. Griffiths and P. Marjoram. Ancestral inference from samples of DNA sequences with recombination. *J. Computational Biology*, 3:479–502, 1996.
- [16] D. Gusfield and V. Bansal. A fundamental decomposition theory for phylogenetic networks and incompatible characters. In *Proceedings of the Ninth International Conference on Research in Computational Molecular Biology (RECOMB)*, pages 217–232, 2005.
- [17] D. Gusfield, S. Eddhu, and C. Langley. Efficient reconstruction of phylogenetic networks with constrained recombination. In *Proceedings of the IEEE Computer Society Conference on Bioinformatics*, page 363, 2003.
- [18] M. Hallett and J. Lagergren. Efficient algorithms for lateral gene transfer problems. In *5th Annual RECOMB Montreal, April 22-25*, pages 149–156, 2001.
- [19] M. Hallett, J. Lagergren, and A. Tofgh. Simultaneous identification of duplications and lateral transfers. In *Proceedings of the Eight International Conference on Research in Computational Molecular Biology (RECOMB)*, pages 347–356, 2004.
- [20] J. Hein. A heuristic method to reconstruct the history of sequences subject to recombination. *J. Mol. Evol.*, 36:396–405, 1993.
- [21] B. Holland, K. Huber, V. Moulton, and P. J. Lockhart. Using consensus networks to visualize contradictory evidence for species phylogeny. *Molecular Biology and Evolution*, 21:1459–1461, 2004.
- [22] B. Holland and V. Moulton. Consensus networks: A method for visualizing incompatibilities in collections of trees. In G. Benson and R. Page, editors, *Proceedings of “Workshop on Algorithms in Bioinformatics”*, volume 2812 of *LNBI*, pages 165–176. Springer, 2003.
- [23] K. T. Huber, M. Langton, D. Penny, V. Moulton, and M. Hendy. Spectronet: A package for computing spectra and median networks. *Applied Bioinformatics*, 1:159–161, 2002.
- [24] K.T. Huber and V. Moulton. Phylogenetic networks from multi-labelled trees. In press, 2006.

- [25] R. R. Hudson. Properties of the neutral allele model with intergenic recombination. *Theoretical Population Biology*, 23:183–201, 1983.
- [26] J.P. Huelsenbeck and F. Ronquist. MRBAYES: Bayesian inference of phylogenetic trees. *Bioinformatics*, 17(8):754–755, 2001.
- [27] J.P. Huelsenbeck, F. Ronquist, R. Nielsen, and J.P. Bollback. Bayesian inference of phylogeny and its impact on evolutionary biology. *Science*, 294:2310–2314, 2001.
- [28] D. H. Huson. SplitsTree: A program for analyzing and visualizing evolutionary data. *Bioinformatics*, 14(10):68–73, 1998.
- [29] D. H. Huson. Split networks and reticulate networks. In O. Gascuel and M.A. Steel, editors, *Reconstructing evolution: New mathematical and computational advances*. Oxford University Press, 2007. In press.
- [30] D. H. Huson and D. Bryant. Application of phylogenetic networks in evolutionary studies. *Molecular Biology and Evolution*, 23:254–267, 2006. Software available from www.splitstree.org.
- [31] D. H. Huson, T. DeZulian, T. Klopper, and M. A. Steel. Phylogenetic super-networks from partial trees. *IEEE/ACM Transactions in Computational Biology and Bioinformatics*, 1(4):151–158, 2004.
- [32] D.H. Huson, T. Klopper, P.J. Lockhart, and M.A. Steel. Reconstruction of reticulate networks from gene trees. In *Proceedings of the Ninth International Conference on Research in Computational Molecular Biology (RECOMB)*, volume 3500 of *LNCS*, pages 233–249. Springer Verlag, 2005.
- [33] D.H. Huson and T.H. Klopper. Computing recombination networks from binary sequences. *Bioinformatics*, 21(suppl. 2):ii159–ii165, 2005. ECCB.
- [34] D.H. Huson, M.A. Steel, and J. Whitfield. Reducing distortion in phylogenetic networks. In P. Bücher and B.M.E. Moret, editors, *Algorithms in Bioinformatics*, LNBI 4175, pages 150–161, 2006.
- [35] T.H. Jukes and C.R. Cantor. Evolution of protein molecules. In H. N. Munro, editor, *Mammalian Protein Metabolism*, pages 21–132. Academic Press, New York, NY, 1969.
- [36] B. Kilian, H. Özkan, O. Deusch, S. Effgen, A. Brandolini, J. Kohl, W. Martin, and F. Salamini. Independent wheat B and G genome origins in outcrossing *aegilops* progenitor haplotypes. *Molecular Biology and Evolution*, 24(1):217–227, 2007.
- [37] Marcus A. Koch, Christoph Dobes, Christiane Kiefer, Roswitha Schmickl, Leos Klimes, and Martin A. Lysak. Supernetwork identifies multiple events of plastid *trnf(gaa)* pseudogene evolution in the Brassicaceae. *Molecular Biology and Evolution*, 21(1):63–73, 2007.

- [38] K.O'Donnell, H. C. Kistler, B. K. Tacke, and H. H. Casper. Gene genealogies reveal global phylogeographic structure and reproductive isolation among lineages of *fusarium graminearum*, the fungus causing wheat scab. *PNAS*, 97(14):7905–7910, 2000.
- [39] M. Kreitman. Nucleotide polymorphism at the alcohol dehydrogenase locus of *Drosophila melanogaster*. *Genetics*, 11:147–164, 1985.
- [40] K. Kryukov and N. Saitou. Netview: Application software for constructing and visually exploring phylogenetic networks. *Genome Informatics*, 14:280–281, 2003.
- [41] A. Kumar, W.C. Black, and K.S. Rai. An estimate of phylogenetic relationships among culicine mosquitoes using a restriction map of the rDNA cistron. *Insect Molecular Biology*, 7(4):367–373, 1998.
- [42] C. R. Linder and L. H. Rieseberg. Reconstructing patterns of reticulate evolution in plants. *Am. J. Bot.*, 91(10):1700–1708, 2004.
- [43] C.R. Linder, B.M.E. Moret, L. Nakhleh, and T. Warnow. Network (reticulate) evolution: Biology, models, and algorithms. A tutorial presented at the Ninth Pacific Symposium on Biocomputing, 2004.
- [44] P. J. Lockhart, P. A. McLenachan, D. Havell, D. Glenny, D. H. Huson, and U. Jensen. Phylogeny, dispersal and radiation of New Zealand alpine buttercups: molecular evidence under split decomposition. *Ann Missouri Bot Gard*, 88:458–477, 2001.
- [45] Rune B. Lyngsø, Yun S. Song, and Jotun Hein. Minimum recombination histories by branch and bound. In *WABI*, pages 239–250, 2005.
- [46] W. Maddison and D. Maddison. Mesquite- a modular system for evolutionary analysis. version 1.05. <http://mesquiteproject.org>, 2005.
- [47] W. P. Maddison. Gene trees in species trees. *Syst. Biol.*, 46(3):523–536, 1997.
- [48] V. Makarenkov. T-REX: Reconstructing and visualizing phylogenetic trees and reticulation networks. *Bioinformatics*, 17(7):664–668, 2001.
- [49] L. Nakhleh, T. Warnow, and C. R. Linder. Reconstructing reticulate evolution in species - theory and practice. In *Proceedings of the Eighth International Conference on Research in Computational Molecular Biology (RECOMB)*, pages 337–346, 2004.
- [50] N. Saitou and M. Nei. The Neighbor-Joining method: a new method for reconstructing phylogenetic trees. *Molecular Biology and Evolution*, 4:406–425, 1987.
- [51] M. Salminen, J.K. Carr, D.S. Burke, and F.E. McCutchan. Identification of breakpoints in intergenotypic recombinants of HIV type 1 by bootscanning. *AIDS Res. Hum. Retroviruses*, 11:1423–1425, 2001.

- [52] R. R. Sokal and C. D. Michener. A statistical method for evaluating systematic relationships. *University of Kansas Scientific Bulletin*, 28:1409–1438, 1958.
- [53] Y.S. Song and J. Hein. Parsimonious reconstruction of sequence evolution and haplotype blocks: Finding the minimum number of recombination events. In *Proceedings of the Workshop on Algorithms in Bioinformatics*, pages 287–302, 2003.
- [54] Y.S. Song and J. Hein. On the minimum number of recombination events in the evolutionary history of DNA sequences. *J. Math. Biol.*, 48:160–186, 2004.
- [55] Y.S. Song and J. Hein. Constructing minimal ancestral recombination graphs. *J. Comp. Biol.*, 12:147–169, 2005.
- [56] D. L. Swofford. PAUP*: Phylogenetic analysis using parsimony (*: and other methods), version 4.2, 2000.
- [57] L. Wang, K. Zhang, and L. Zhang. Perfect phylogenetic networks with recombination. *Journal of Computational Biology*, 8(1):69–78, 2001.
- [58] L.G. Willis, M.L. Winston, and B.M. Honda. Phylogenetic relationships in the honeybee (genus *Apis*) as determined by the sequence of the cytochrome oxidase ii region of mitochondrial DNA. *Mol. Phylogenet. Evol.*, 1:169–178, 1992.
- [59] M. Worobey. A novel approach to detecting and measuring recombination: new insights into evolution in viruses, bacteria and mitochondria. *Mol. Biol. Evol.*, 18:1425–1434, 2001.