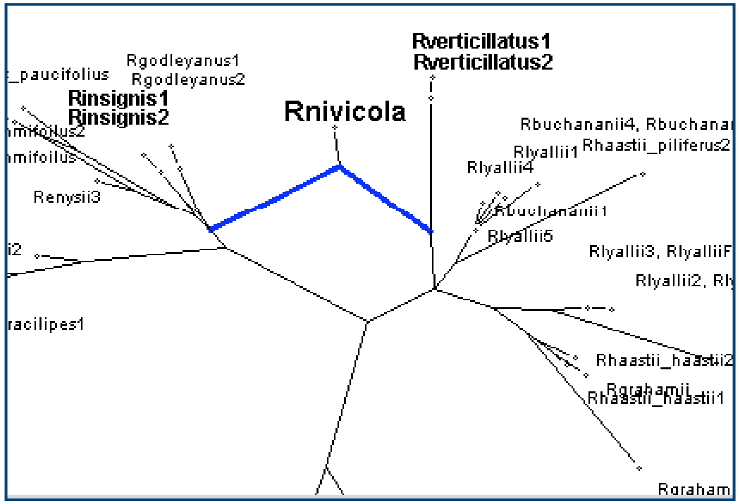
Introduction to Phylogenetic Networks



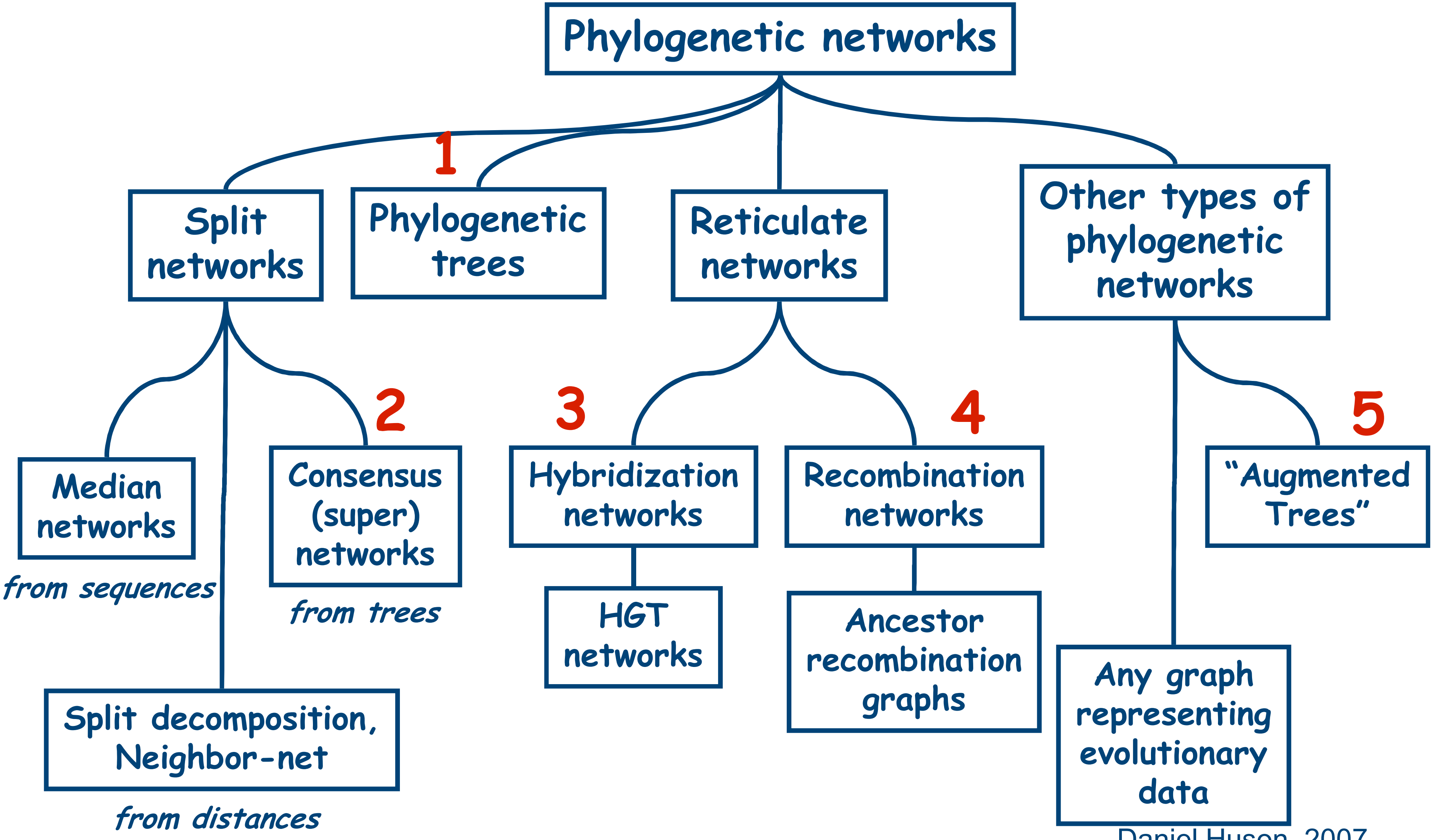
Daniel H. Huson

ISMB 2007, Vienna, July 21, 2007

Contents

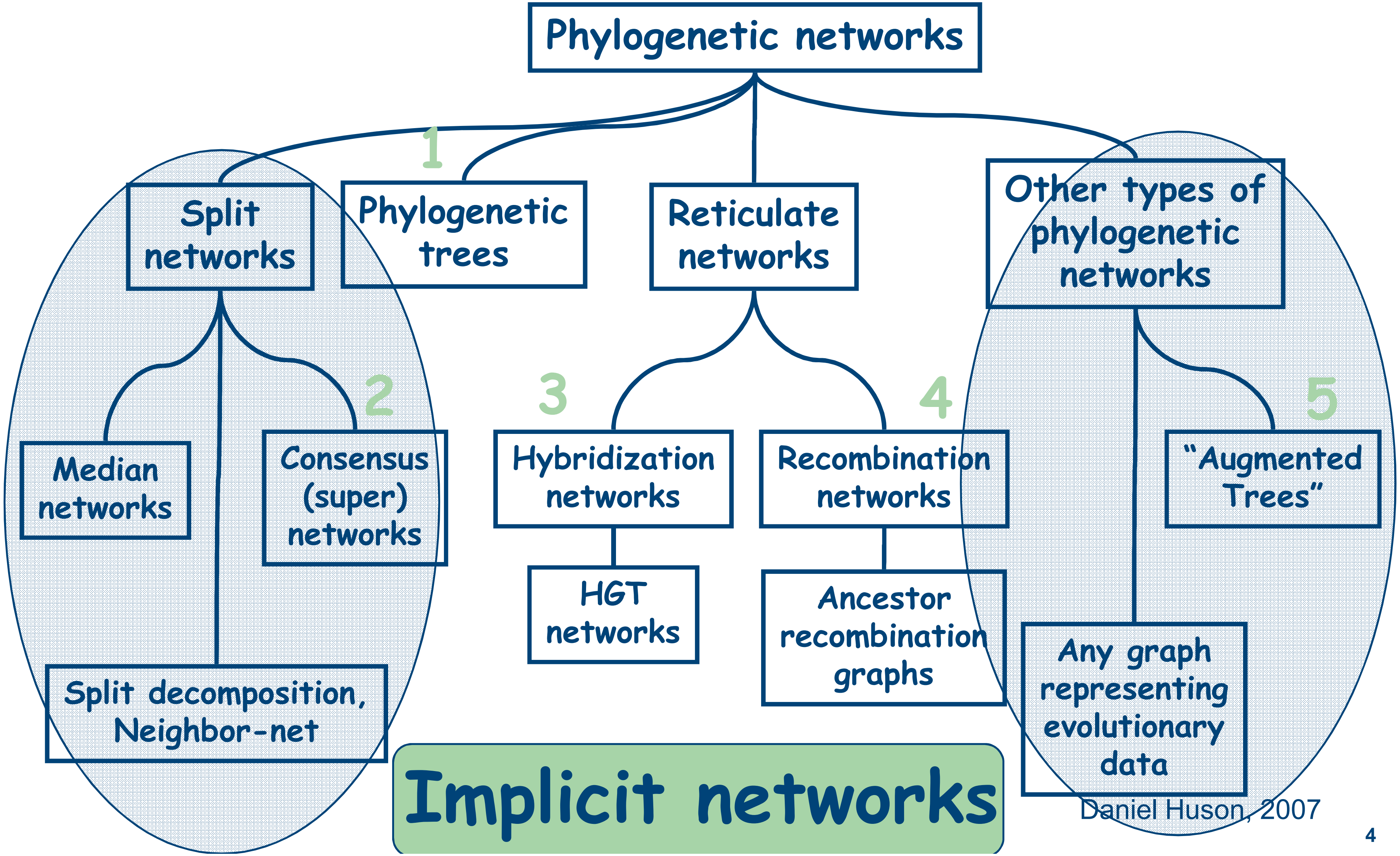
1. Phylogenetic trees
2. Consensus networks and super networks
3. Hybridization and reticulate networks
4. Recombination networks
5. Other

Overview of Existing Concepts



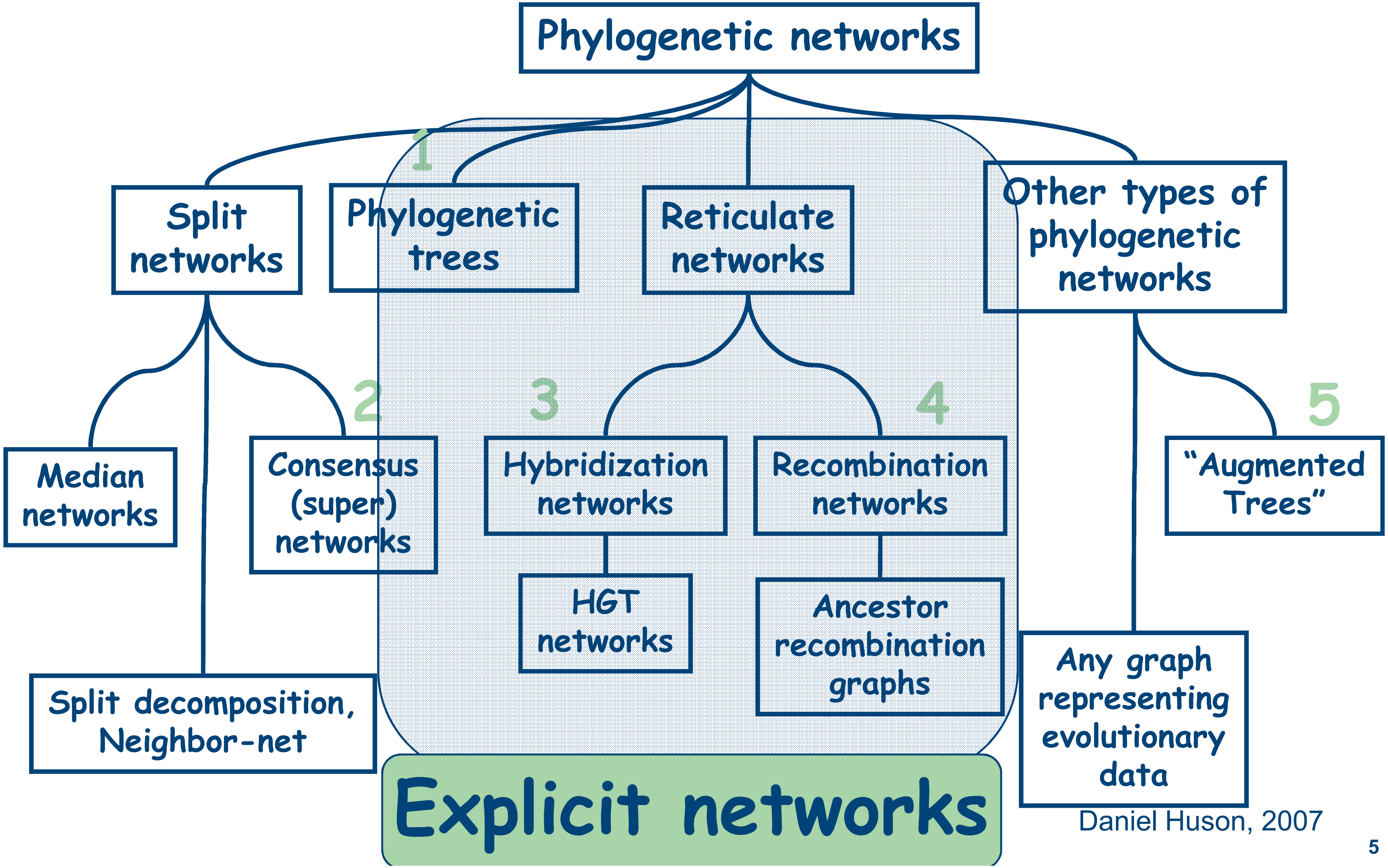
Daniel Huson, 2007

Two Different Kinds of Networks



Daniel Huson, 2007

Two Different Kinds of Networks



Phylogenetic networks

Split networks

Phylogenetic trees

Reticulate networks

Other types of phylogenetic networks

Median networks

Consensus (super) networks

Hybridization networks

Recombination networks

“Augmented Trees”

HGT networks

Ancestor recombination graphs

Split decomposition, Neighbor-net

Any graph representing evolutionary data

Explicit networks

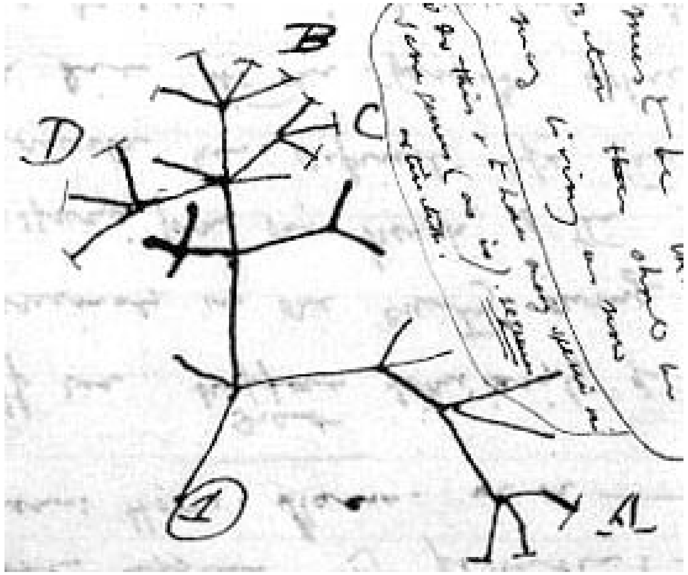
Daniel Huson, 2007

Part I

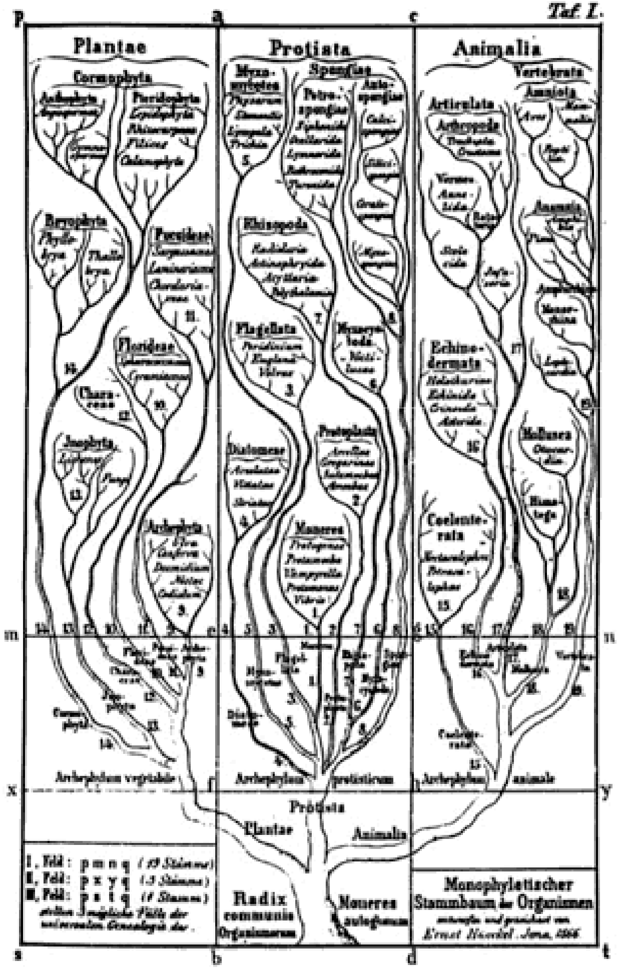
1. Phylogenetic trees
2. Consensus networks and super networks
3. Hybridization and reticulate networks
4. Recombination networks
5. Other

Phylogenetic Trees

- **The evolution of species is usually described by a phylogenetic tree**



Charles Darwin

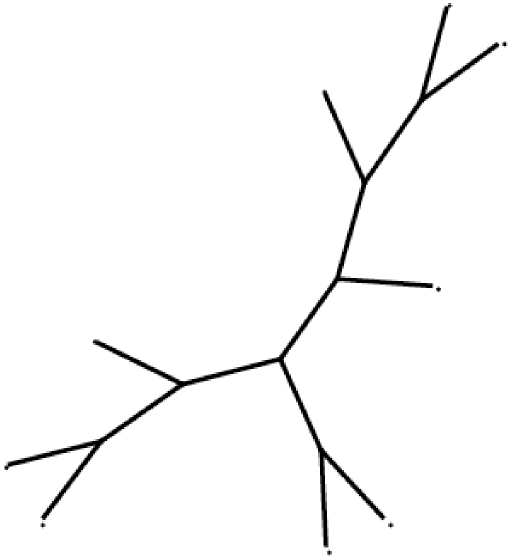

Taf. I.
Plantae
Protista
Animalia
Radix communis Organismorum
Moneres autogonum
Monophyletischer Stammbaum der Organismen
entworfen und gezeichnet von Ernst Haeckel. Jena, 1866

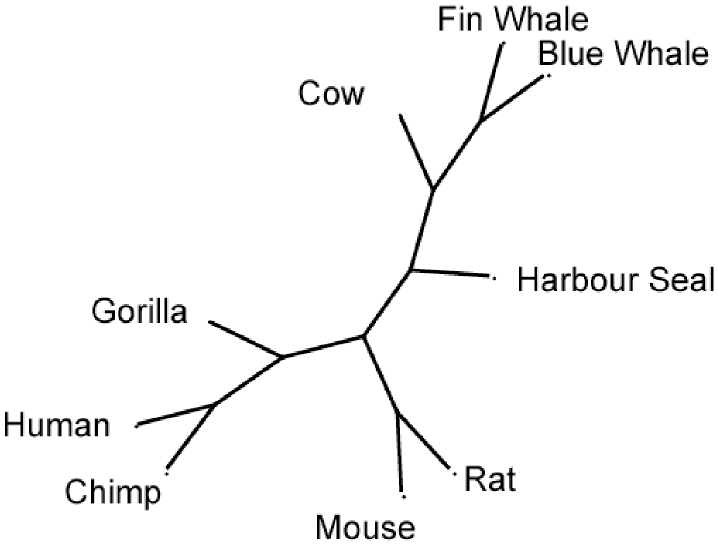
Ernst Haeckel,Tree of Life, 1866

Phylogenetic Trees

- Let X = $\{x_1,...,x_n\}$ denote a set of taxa.
- *A phylogenetic tree T* (or *X-tree*) is given by labeling the leaves of a tree by the set X:

Cow
Fin Whale
Blue Whale
Habor Seal
Rat
Mouse
Chimp
Human
Gorilla

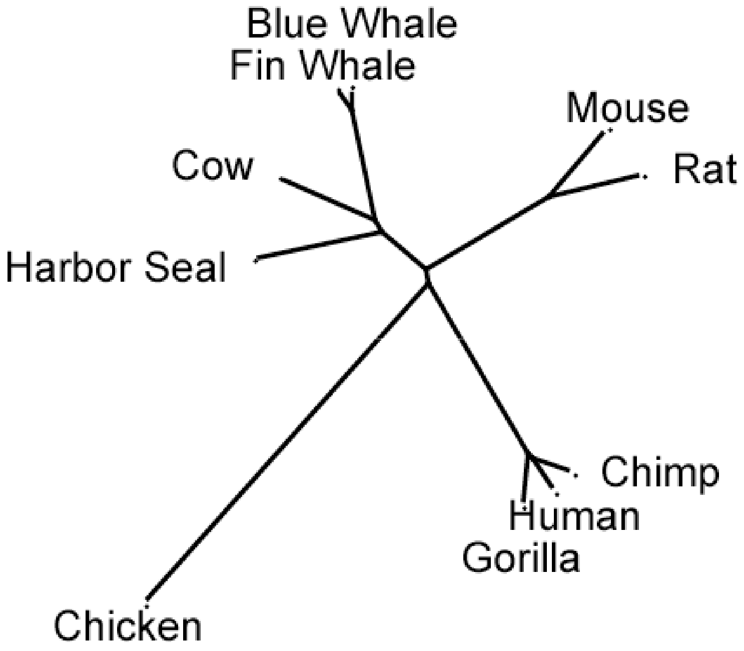




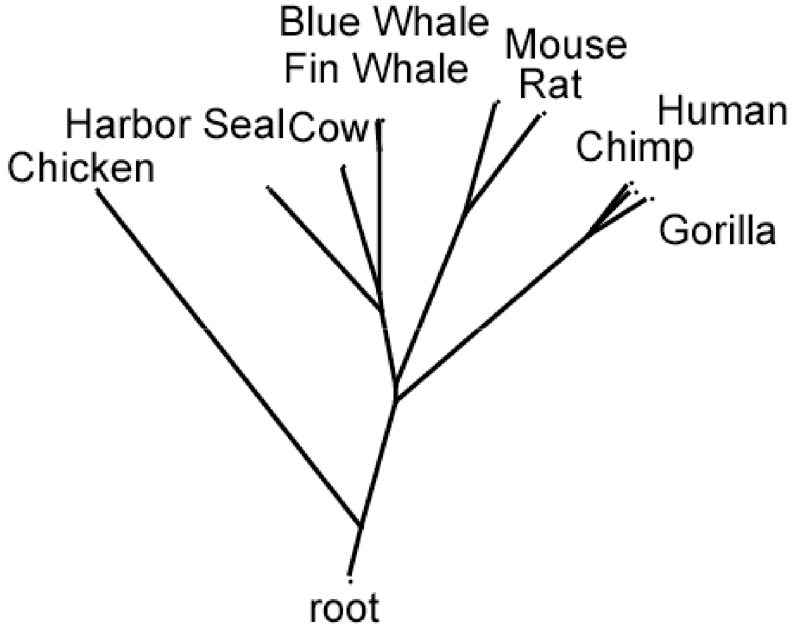
Taxa + tree ⇒ phylogenetic tree

Daniel Huson, 2007

Unrooted vs Rooted Trees



Unrooted tree



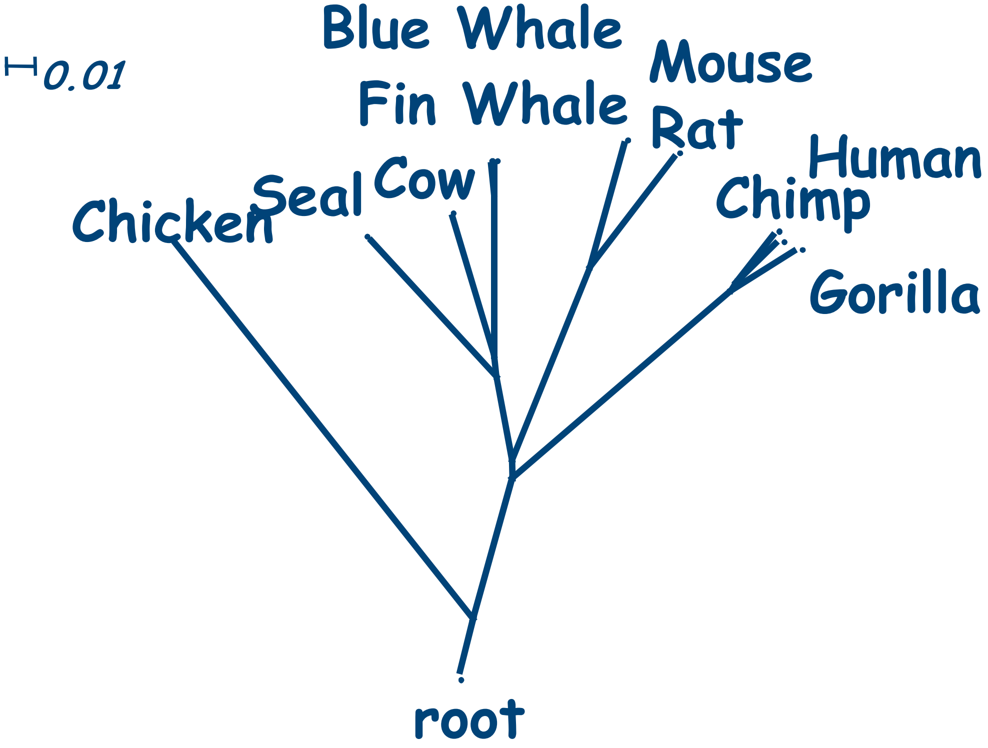
Rooted tree, rooted using Chicken as *outgroup*

Most popular methods produce unrooted trees

Biologically relevant, defines clades of related taxa

Branch Lengths

- Each branch *e* of a phylogenetic tree *T* may be scaled to represent *r* × *t*, the *rate of evolution r* × *time t* along *e*:

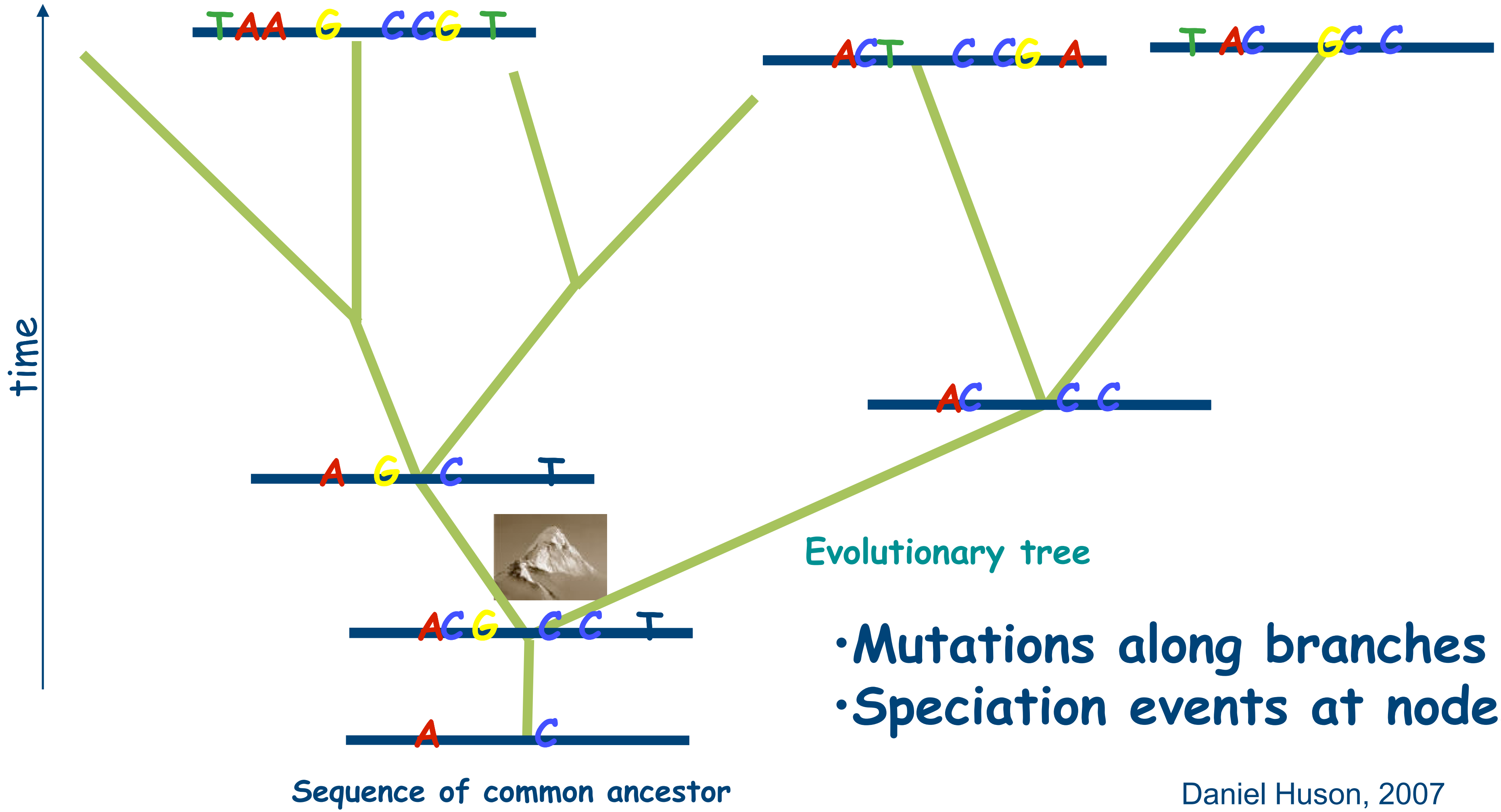


A Simple Model of Evolution

- Sequences evolve along a given tree *T*, called the *evolutionary* -, *model* - or the *true tree*

- Two types of events:
  - mutations and
  - speciation events

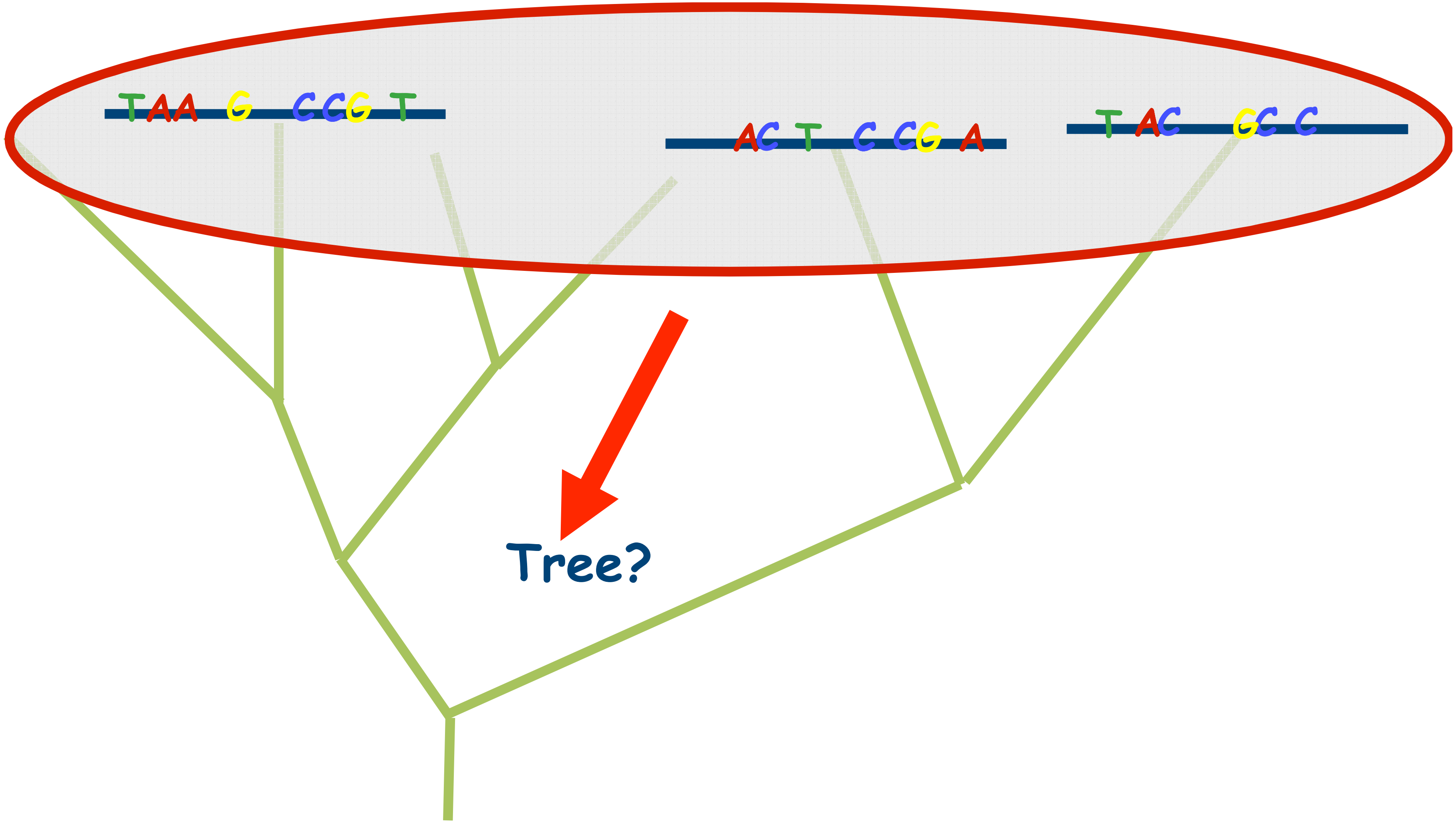
A Simple Model of Evolution



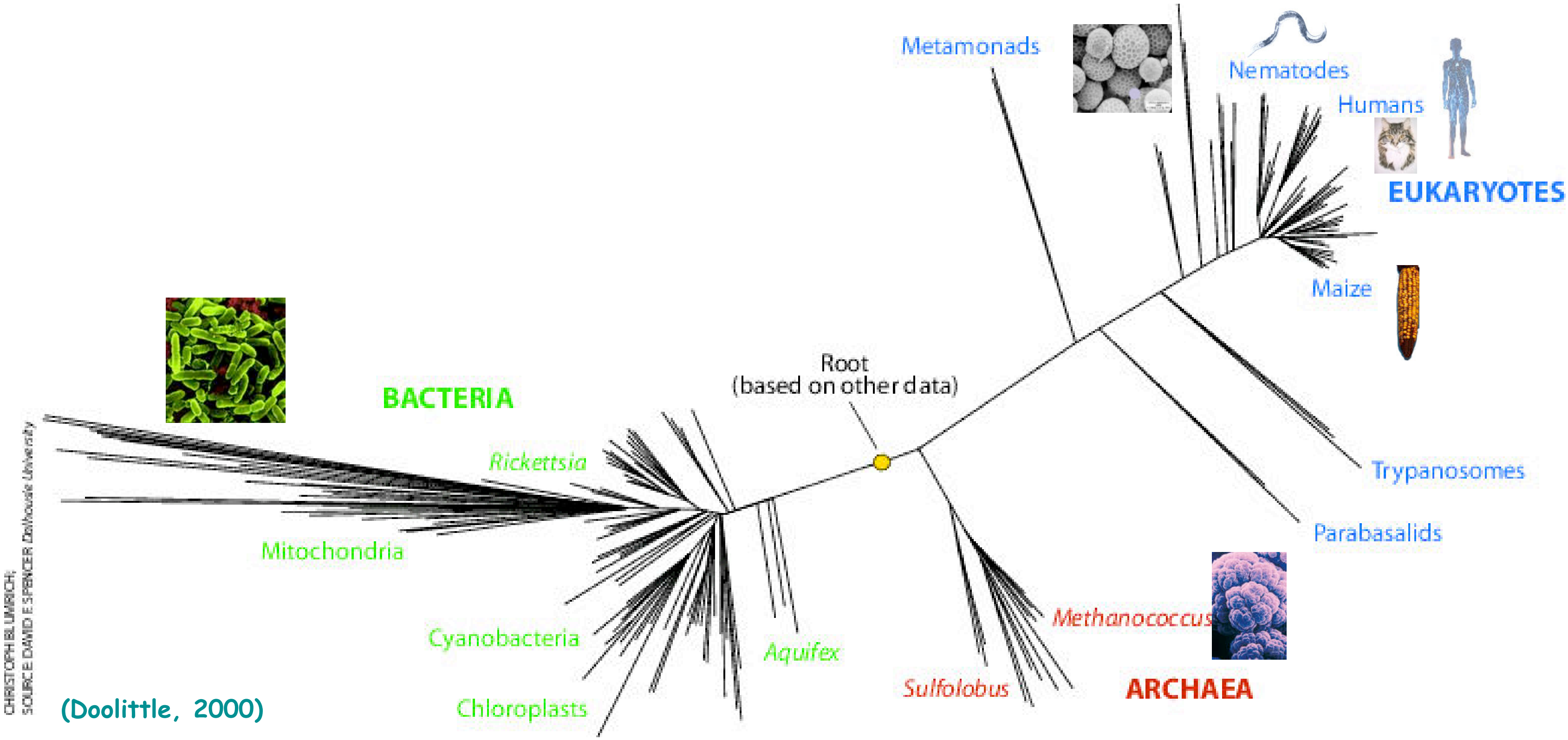
- Mutations along branches
- Speciation events at nodes

Daniel Huson, 2007

Tree Reconstruction Problem



Tree of Life Based on SSU rRNA



(Doolittle, 2000)

Jukes-Cantor Model of Evolution

Let T_0 be a rooted X-tree. In the Jukes-Cantor model:

1. The possible states for each site are A, C, G and T.
2. The 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.) along the branches from the root at a fixed rate u.
4. Each branch e has a duration $t = t(e)$ and the expected number of mutations per site is $u \times t(e)$. The probabilities of change to each of the 3 remaining states are equal.

Jukes-Cantor Model of Evolution

How does a sequence evolve along a branch *e* under the Jukes-Cantor model?

- *A* nucleotide change to one of the other three bases occurs at a fixed rate *u*.
- Thus, the probability of an *observable* change occurring at any given site in time t is:

$$\text{Prob(observable change} \mid t\,) = \tfrac{3}{4}(1-(e^{-4/3ut}))$$

Aligned Sequences

- A set of taxa $X = \{x_1, ..., x_n\}$ may be given as an alignment of molecular sequences, e.g.:

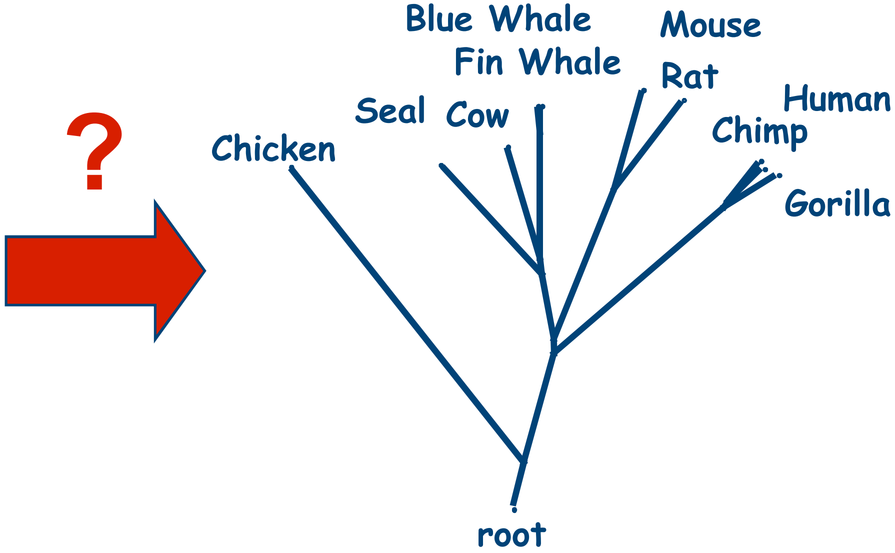
```
Human   fqtpmviilqaimgsatlamtliiftiiiiltvhdtnttvptmitpmllt
Chimp   fqtpmiiifqaimgsatlaltliiftiiviltvhdtntavpttitpmllt
Gorilla lqtpmviifqaimgsatlamtliiftvimiltvhetnttvptmiapmllt
H.Seal  fqlpmviifqaiiggatlalafitftiiifltvhdtdtstlimilsmilt
Cow     fqtpmviifqaiiggatlalalitftiiifmtvhdtdtstltmilsmflt
F.Whale lqtfmviifqaimgettlalafitftiaifltvhdtdtsmlltilsmllt
B.Whale lqtfmviifqaimgettlvlaiitftiaifltvhdtdtstlltilsmllt
Rat     fqismiiifqaimggatlvlatitfiilvfltvhdtdtstfitiissmat
Mouse   fqismiiifqaimggatlvlatitfiilifltvhdtdtstfitiissmit
```

- Usually obtained from some gene or locus that all taxa have in common.

Tree Reconstruction Problem

Given an alignment that evolved along some evolutionary tree *T*, can we reconstruct the tree?

Human	fqtpmviilqaimgsatlamtliift
Chimp	fqtpmiiifqaimgsatlaltliift
Gorilla	lqtpmviifqaimgsatlamtliift
Seal	fqlpmviifqaiiggatlalafitft
Cow	fqtpmviifqaiiggatlalalitft
Fin Whale	lqtfmviifqaimgettlalafitft
Blue Whale	lqtfmviifqaimgettlvlaiitft
Rat	fqismiiifqaimggatlvlatitfi
Mouse	fqismiiifqaimggatlvlatitfi
Chicken	pqismiaffqaimggatlfaatitfi



Challenges:

1. **Determine the unrooted topology of *T*.**
2. **Estimate the branch lengths of *T*.**
3. **Infer the position of the root in *T*.**

Tree Reconstruction Methods

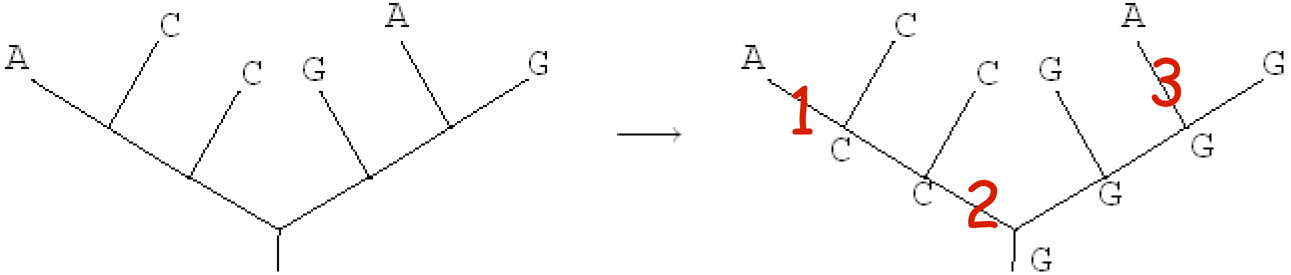
- Sequence-based methods search for a tree that optimally explains the given sequence data:
  - Maximum Parsimony,
  - Maximum Likelihood, and
  - Bayesian Inference.

- Distance-based methods infer a distance matrix and construct a tree from it:
  - UPGMA,
  - Neighbor-Joining, Bio-NJ and Weighbor.

- Tree-based methods infer a tree from a set of trees
  - Consensus tree (e.g. strict, majority, loose)
  - Super tree (if trees are defined on overlapping subsets of taxa)

Maximum Parsimony Tree Reconstruction

- These methods search for a tree *T* that explains an alignment *A* of sequences 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 be considered to find the optimal tree!

Maximum-Likelihood & Bayesian Methods

Maximum-Likelihood and Bayesian methods are based on an explicit model of evolution, such as the Jukes-Cantor model.

- In a Maximum-Likelihood approach, one computes the likelihood $P(A \mid T)$ that the true tree is T, given the alignment A. The method returns:

$$T_{ML} = \text{argmax}_T P(A \mid T)$$

- More desirable may be the most probable tree T, given the alignment A (computed using Bayes' Theorem):

$$T_{\text{Bayesian}} = \text{argmax}_T P(T \mid A).$$

  ‘MCMC’ methods are used to address this.

- Both approaches are computationally expensive.

Distance Estimation

- Compute a distance matrix *D* from *A*. One approach for DNA is to use the uncorrected-P (or Hamming) distance, the proportion of observed differences:

- Example:

a_1	C	A	A	C	C	C	C	C	A	A	A	A	A
a_2	T	A	A	T	T	T	-	C	A	A	A	A	A
a_3	C	G	G	T	T	T	-	-	A	A	A	A	A

- Distances:
  - Ham(a_1,a_2) =4/12 $\approx$ 0.33
  - Ham(a_1,a_3) =5/11 $\approx$ 0.45
  - Ham(a_2,a_3) =3/11 $\approx$ 0.27

Distance Corrections

- The uncorrected-P distance often *under-estimates* the true evolutionary distance, as back mutations and multiple hits are not counted.
- Only suitable for closely related sequences.
- In general, correction formula based on some model of evolution are used.
- For example, under the Jukes-Cantor model, the correction is:

$$JC(a,b) = -\tfrac{3}{4}\ln(1-4/3Ham(a,b))$$

(Inverse of expression discussed earlier)

Distance-Based Methods

- Let *D* be a distance matrix for taxa *X* obtained from an alignment *A*.
- The goal is a phylogenetic tree *T* such that the distances between taxa in *T* approximate the distances in *D*.
- The most popular methods are UPGMA and Neighbor-Joining (NJ).

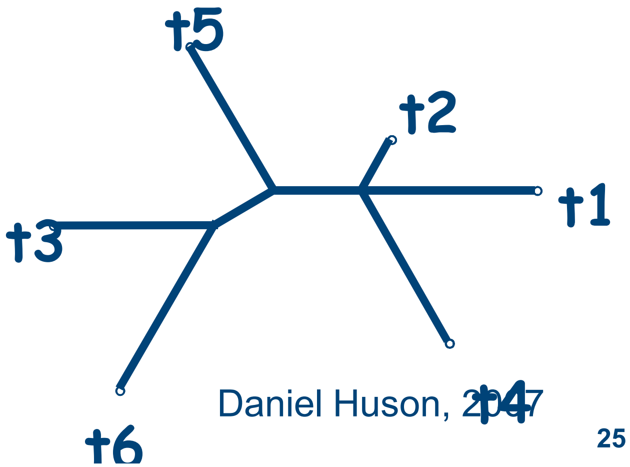
UPGMA and NJ

- UPGMA and NJ 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.

A → *D* → *T*

t_1	ACTTCTATCA
t_2	ACTCCTATCA
t_3	ACTCCTATCT
t_4	ACCCCTGTCA
t_5	ACTTCGGTCA
t_6	ACTTGTATGT

0.0	0.2	0.4	0.3	0.4	0.4
0.2	0.0	0.3	0.2	0.3	0.3
0.4	0.3	0.0	0.5	0.3	0.3
0.3	0.2	0.5	0.0	0.3	0.5
0.4	0.3	0.3	0.3	0.0	0.4
0.4	0.3	0.3	0.5	0.4	0.0



Software

Selection of programs that build phylogenetic trees:

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

Part II

1. Phylogenetic trees
2. **Consensus networks and super networks**
3. Hybridization and reticulate networks
4. Recombination networks
5. Other

Overview

- Will include additional evolutionary events that are not considered in simple tree models.
- Fundamental observation:

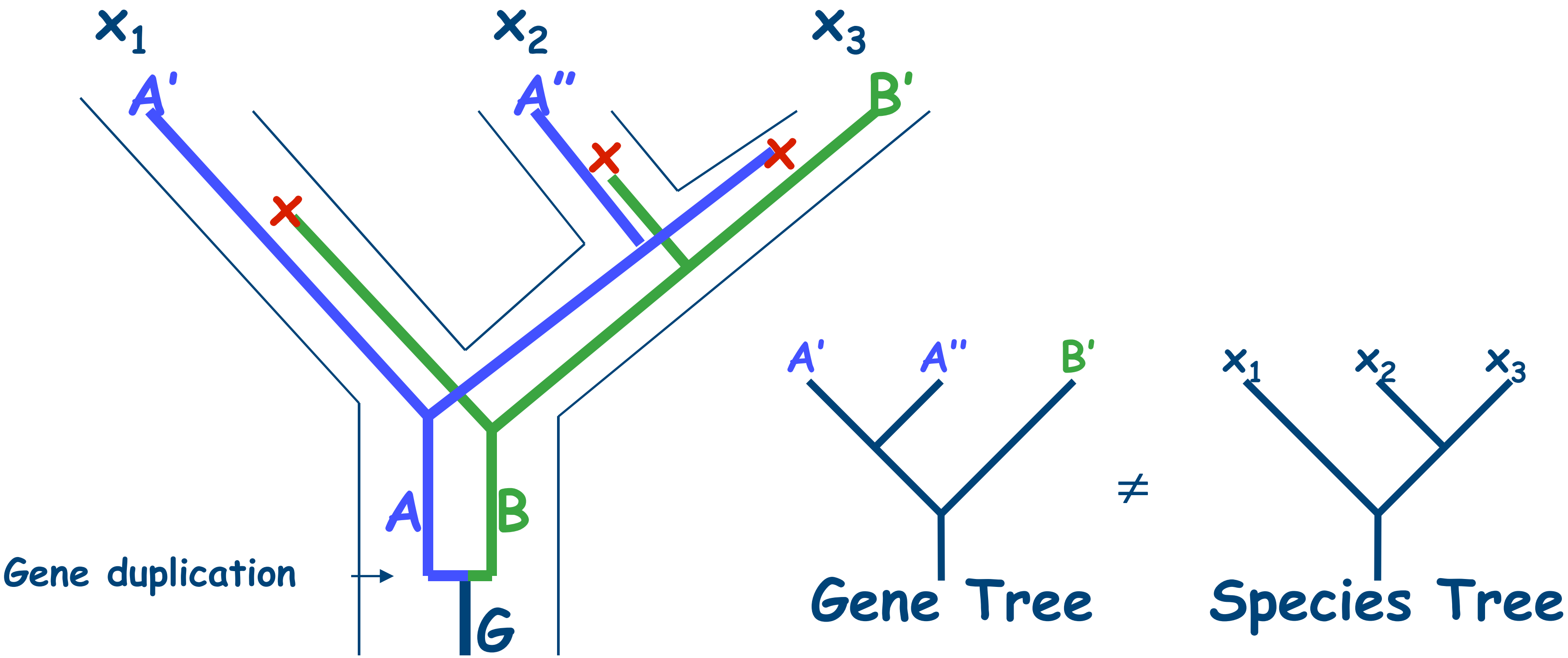
  gene trees differ.
- How to represent conflicting signals using a consensus network or super network.
- Some other methods that use a network to represent conflicting signals.

Additional Evolutionary Events

- Models as discussed above represent the evolution of a single gene.
- When studying more than one gene simultaneously, one should also consider that:
  - individual genes may be born, duplicated or lost.
- Moreover, biological mechanisms such as
  - recombination, hybridization, or horizontal gene transfer may be involved.
- But even when the data evolved on a tree, networks can help to understand problems due to sampling or model-specification error

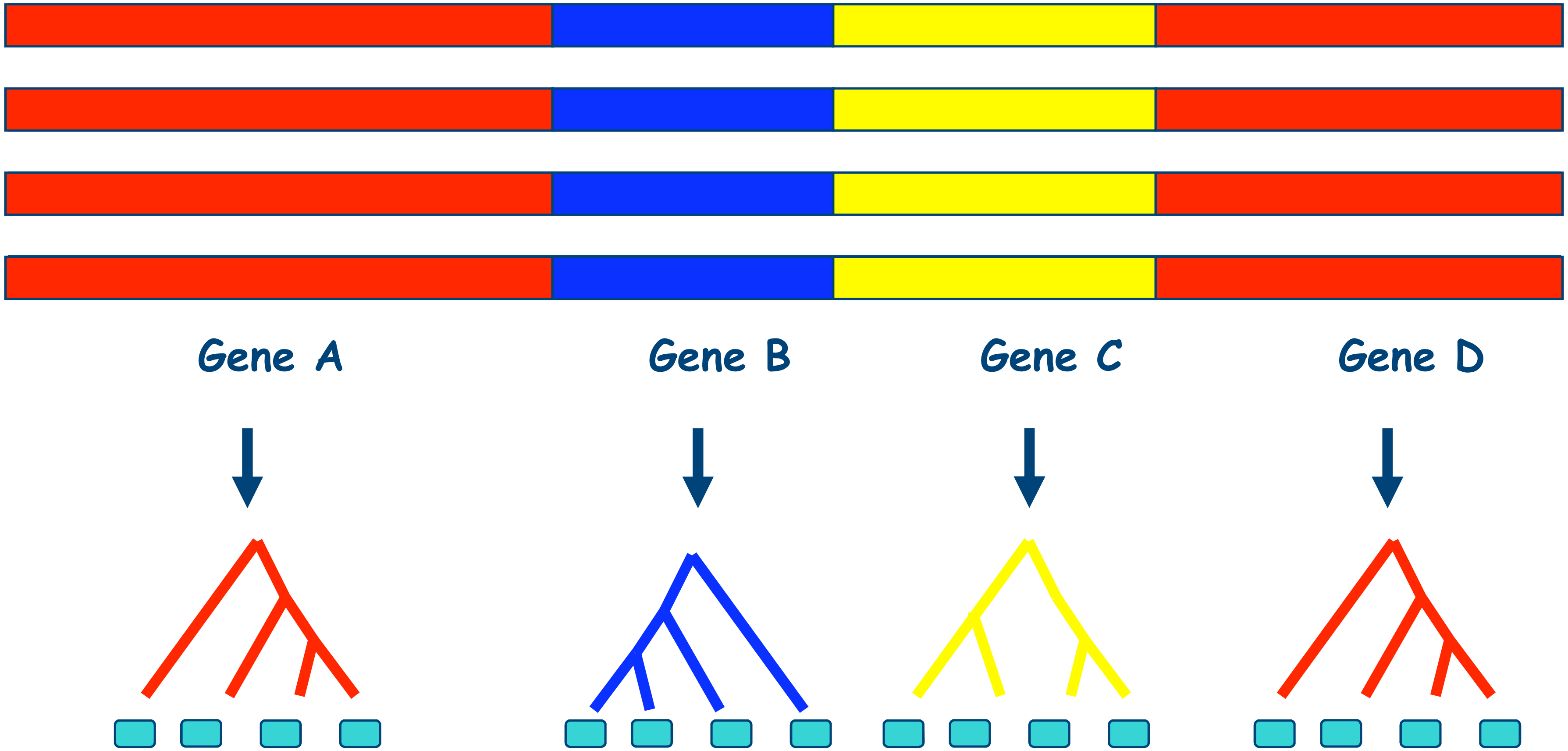
Gene Trees Can Differ

- Consider a model in which the sequence of a gene evolves via mutations, but we also allow gene duplication and loss:



Gene Trees vs Species Trees

Differing gene trees give rise to “mosaic sequences”



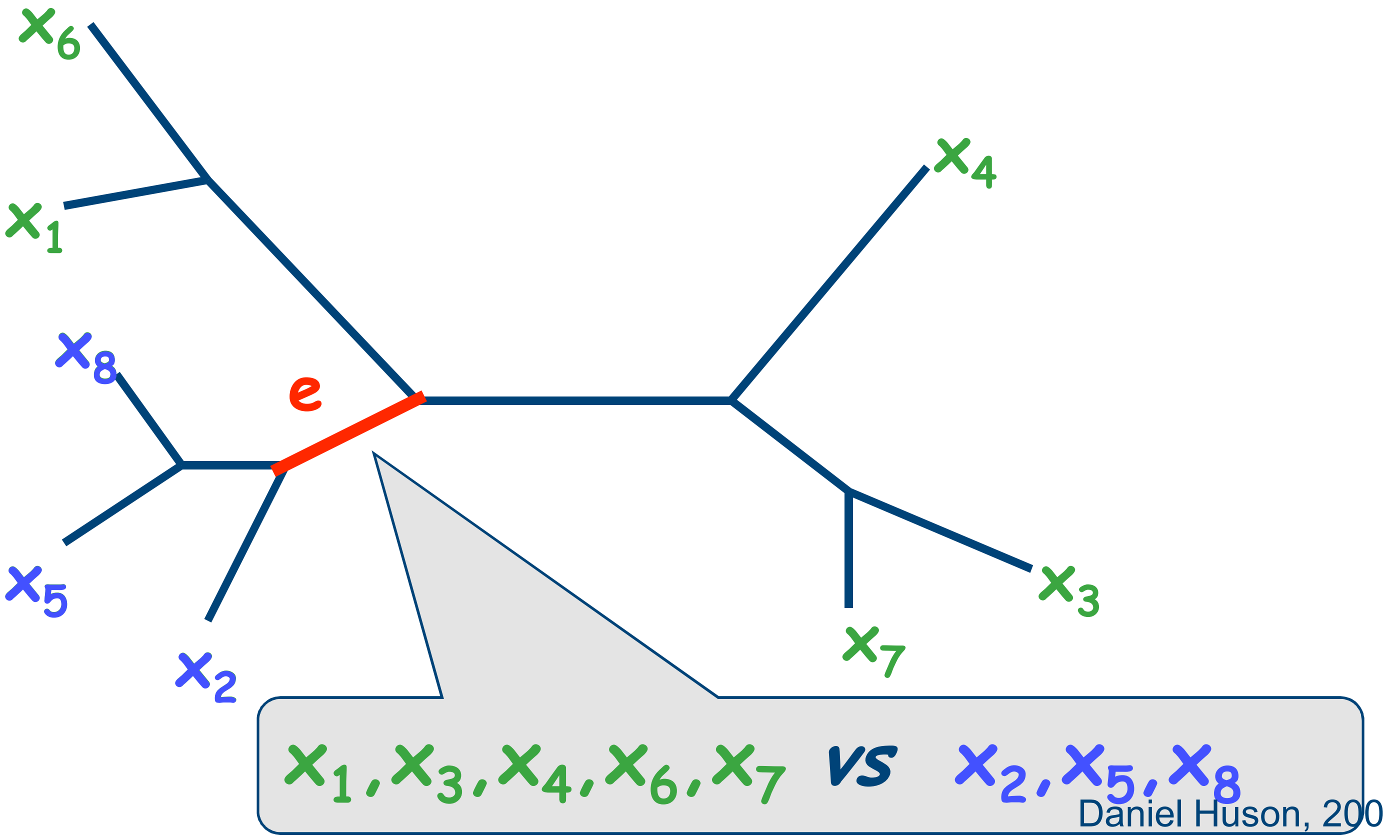
Daniel Huson, 2007

The Consensus of Different Gene Trees

- For a given set of species, we can build evolutionary trees based on different genes
- How to form a consensus of the trees?
  - Consensus trees
  - Consensus networks
  - Consensus super networks

The Splits of a Tree

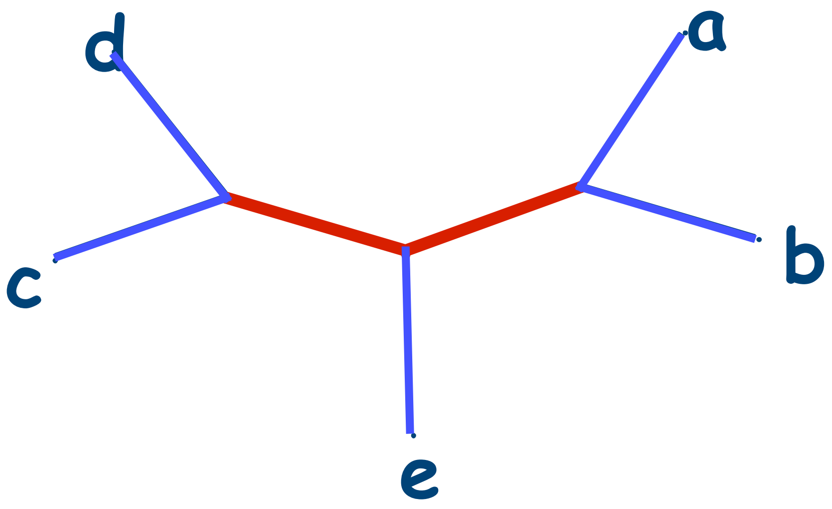
- Every edge of a tree defines a split of the taxon set *X*:



x_1,x_3,x_4,x_6,x_7 *vs* x_2,x_5,x_8

The Split Encoding of a Tree

Tree *T*:



Split encoding Σ(*T*):

5 trivial splits: $\frac{\{a\}}{\{b,c,d,e\}}, \frac{\{b\}}{\{a,c,d,e\}}, \frac{\{c\}}{\{a,b,d,e\}}, \frac{\{d\}}{\{a,b,c,e\}}$ and $\frac{\{e\}}{\{a,b,c,d\}}$,

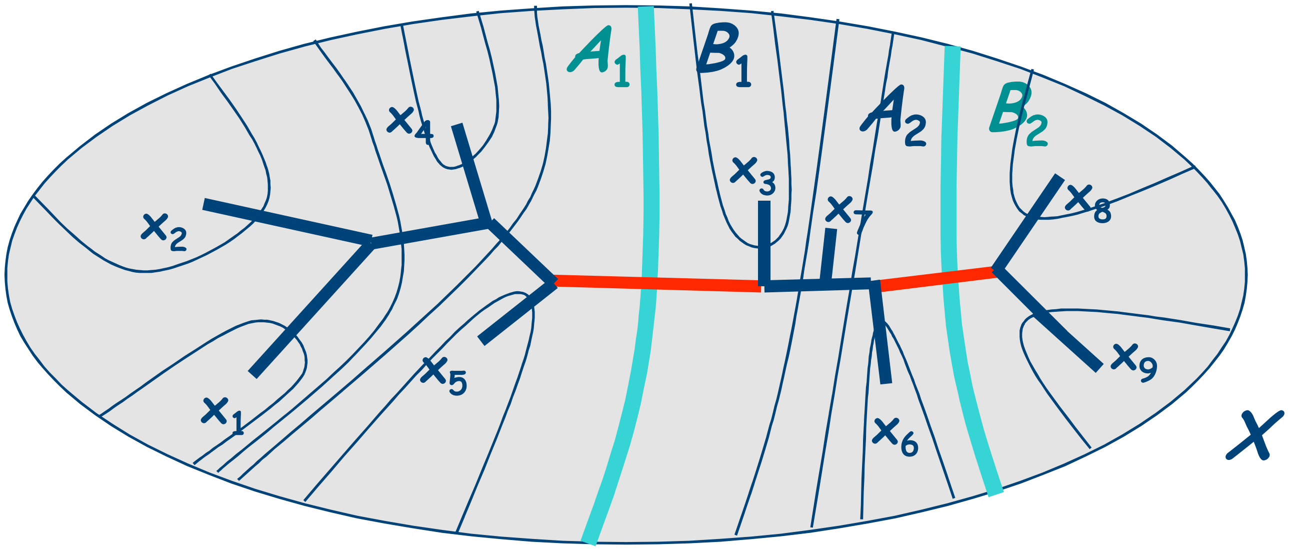
2 non-trivial splits: $\frac{\{a,b,e\}}{\{c,d\}}$ and $\frac{\{a,b\}}{\{c,d,e\}}$.

Compatibility

- Two splits $A_1|B_1$ and $A_2|B_2$ of X are *compatible*,

if $\emptyset \in \{A_1 \cap A_2, A_1 \cap B_2, B_1 \cap A_2, B_1 \cap A_2\}$

Two compatible splits:



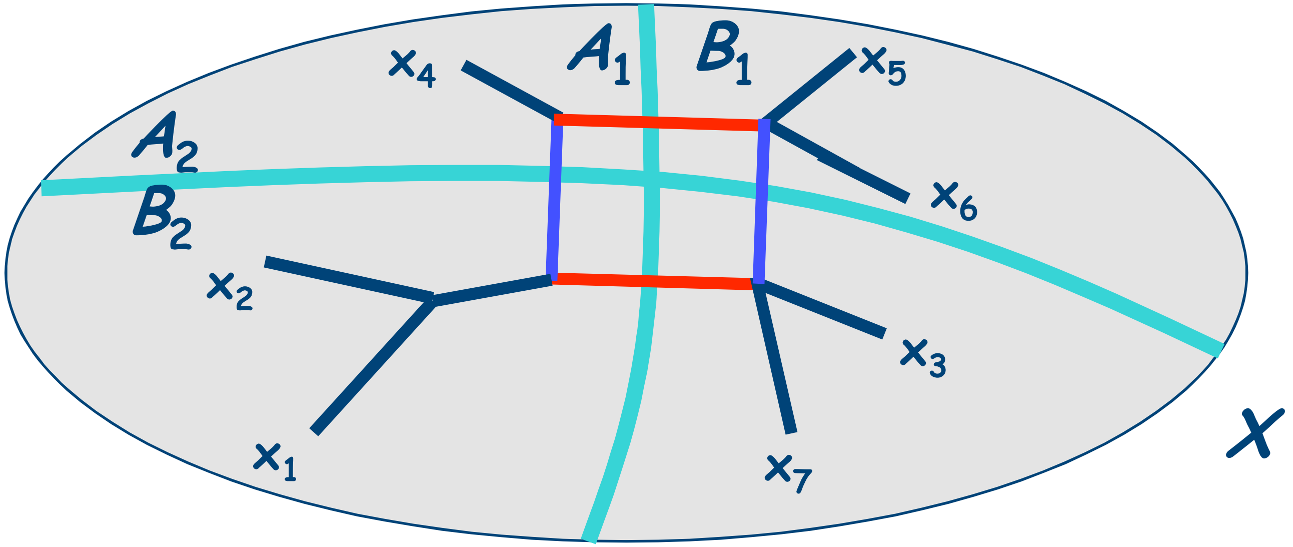
Daniel Huson, 2007

Compatibility

- Two splits $A_1|B_1$ and $A_2|B_2$ of X are *compatible*,

if $\emptyset \in \{A_1 \cap A_2, A_1 \cap B_2, B_1 \cap A_2, B_1 \cap A_2\}$

Two incompatible splits:



Daniel Huson, 2007

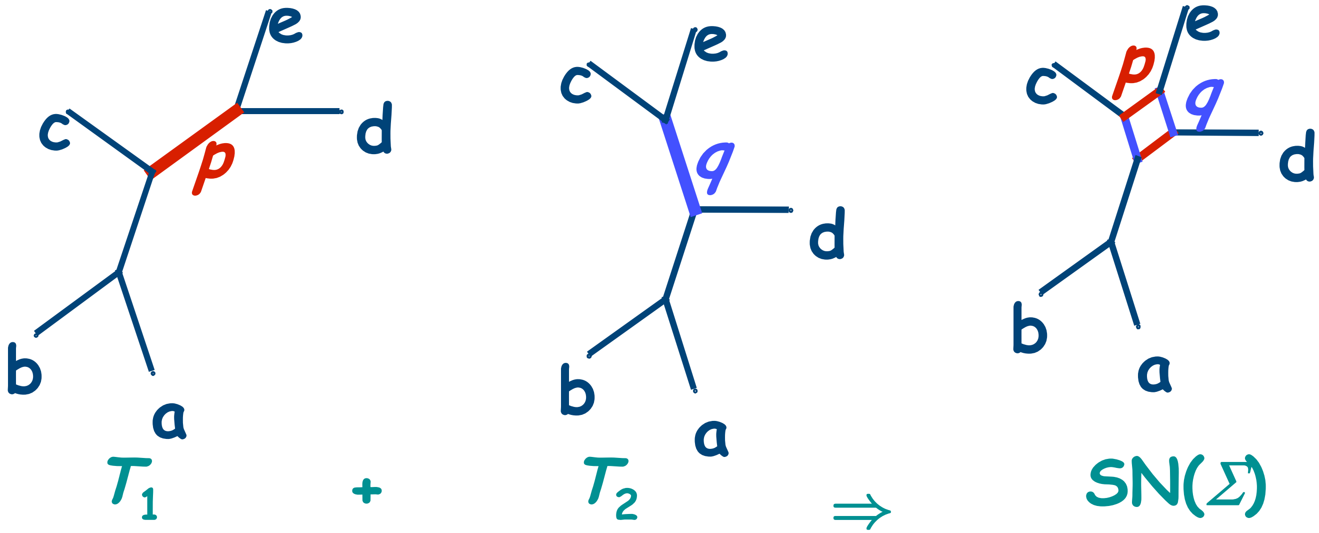
Compatibility Theorem

A set of splits Σ corresponds to a (unique) tree T, if and only if all pairs of splits in Σ are compatible.

Representing Incompatible Trees

- Consider the following two trees T_1 and T_2, for which the splits are incompatible:

$$S_p = \frac{\{a,b,c\}}{\{d,e\}} \in \Sigma(T_1) \text{ and } S_q = \frac{\{a,b,d\}}{\{c,e\}} \in \Sigma(T_2)$$



- The split network SN(Σ) represents the incompatible set of splits $\Sigma := \Sigma(T_1) \cup \Sigma(T_2)$, using bands of parallel edges for incompatible splits.

Daniel Huson, 2007

Consensus of Trees

For trees $T_1,\dots,T_k$ define

$$\Sigma(p) := \{\, S : |\{i: S \in \Sigma(T_i)\}| > pk \,\}$$

- Majority consensus:

$$\Sigma_{maj} = \Sigma(1/2)$$

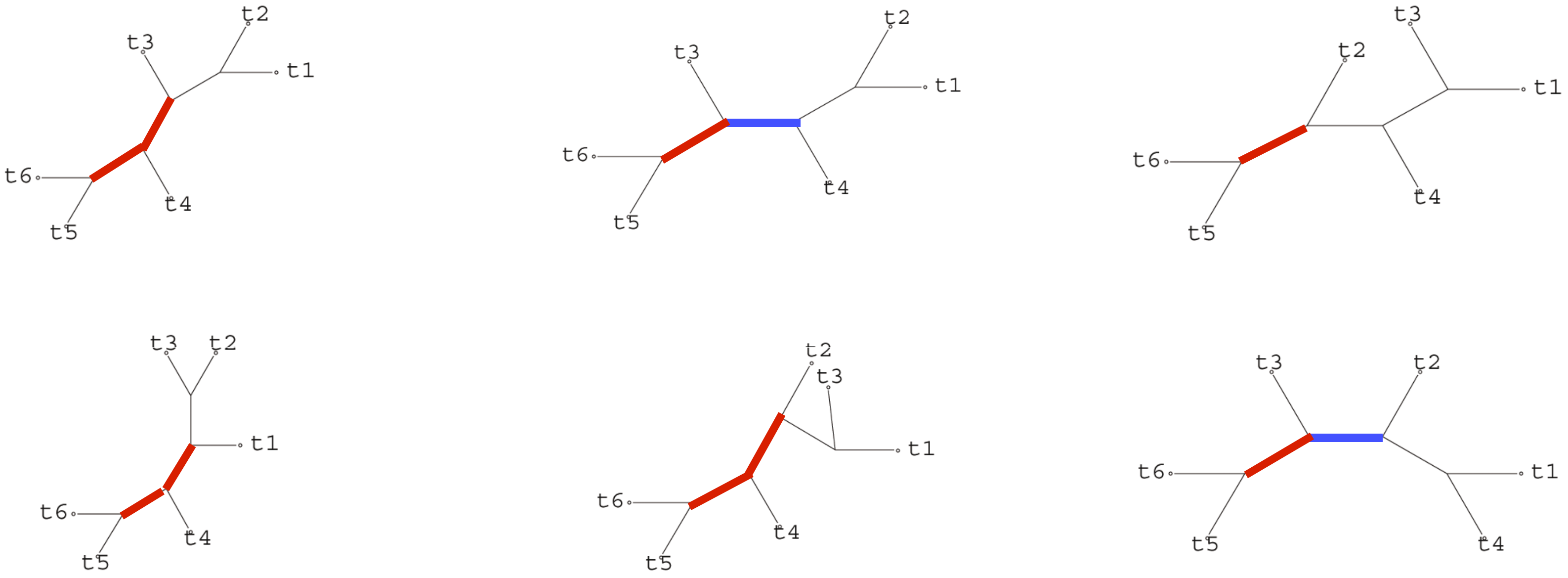
- Strict consensus:

$$\Sigma_{strict} = \Sigma^*(1/1)$$

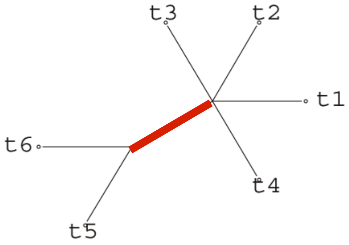
- In general, $\Sigma(1/(d+1))$ defines a set of consensus splits for $d \geq 1$

Consensus of Trees

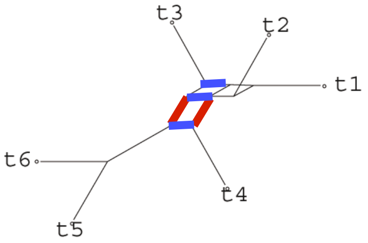
Six gene trees:



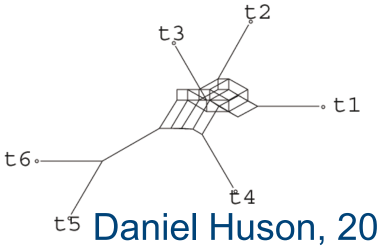
Σ(1/2): majority consensus: splits contained in more than 50% of trees



Σ(1/6): splits contained in more than one tree



Σ(0): splits contained in at least one tree



Consensus Networks

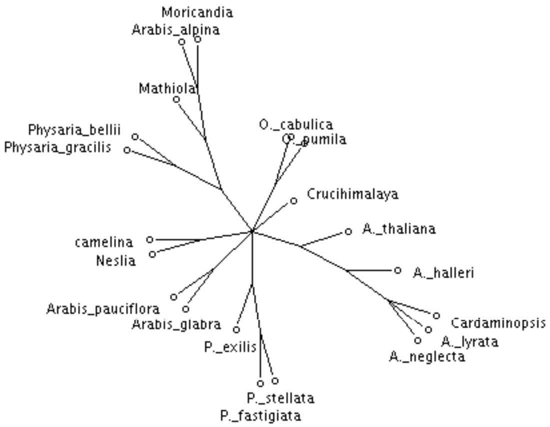
- A consensus network is obtained by computing the consensus splits $\Sigma(1/(d+1))$ for some value $d \geq 1$.
- The parameter d determines the maximum dimensionality of the corresponding network:
  - for $d = 1$ the network will be 1-dimensional, a tree,
  - for $d = 2$ the network may contain parallelograms, &
  - in general it may contain cubes of dimension $\leq d$.

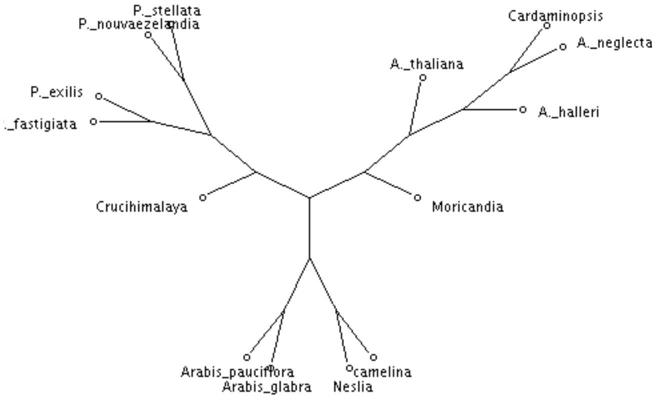
Consensus of Partial Gene Trees

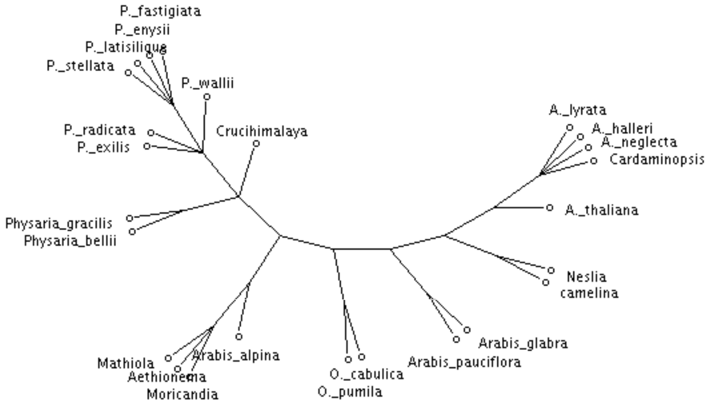
- For a given set of species, we may build evolutionary trees based on many different genes
- But: not every species has every gene, or some sequences may be unavailable
- How to deal with *partial trees*, i.e. trees that do not mention all species?

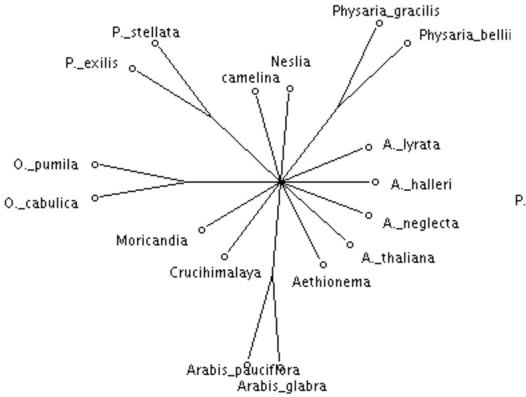
- Answer: Compute a *super-network*

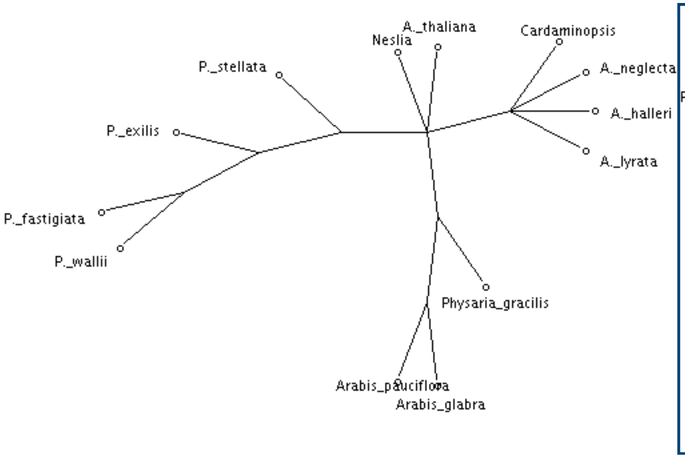
Example of A Super Network (Plants)

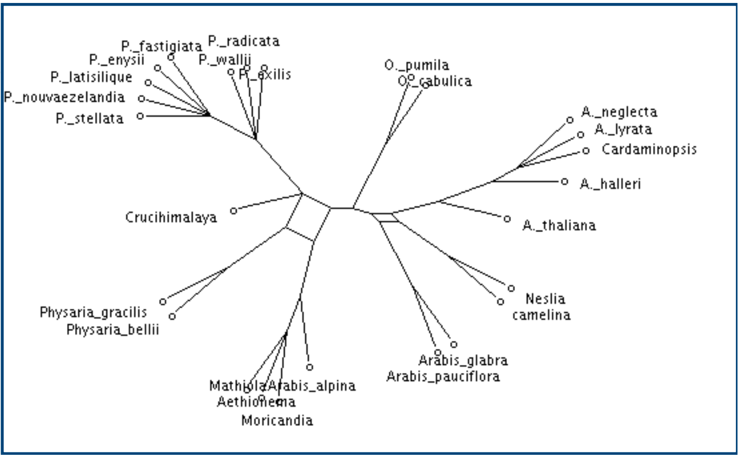












Partial trees for five plant genes **Super network**

Z-Closure Method

- Idea: Extend partial splits.
- Z-rule:

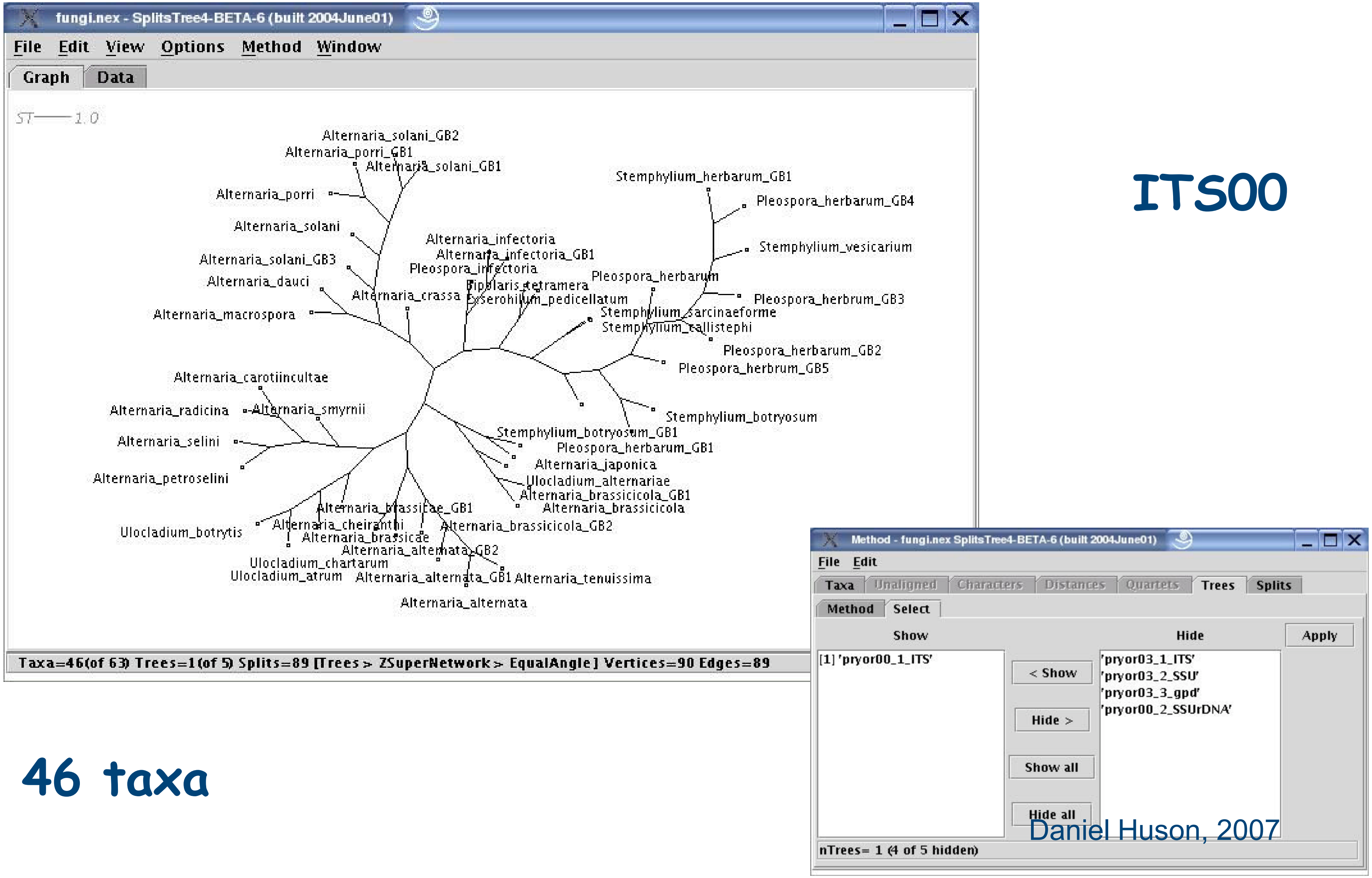
$$\frac{A_1}{B_1} \overset{\cap}{Z} \frac{A_2}{B_2} \longrightarrow \frac{A_1}{B_1 \cup B_2}, \frac{A_1 \cup A_2}{B_2}$$

- Repeatedly apply to completion.
- Return all full splits.

Example

- Five fungal trees from (Pryor 2000) and (Pryor 2003)
- Trees:
  - ITS (two trees)
  - SSU (two trees)
  - Gpd (one tree)
- Numbers of taxa differ: "partial trees"

Individual Gene Trees



Taxa=46(of 63) Trees=1(of 5) Splits=89 [Trees > ZSuperNetwork > EqualAngle] Vertices=90 Edges=89

Method - fungi.nex SplitsTree4-BETA-6 (built 2004June01)
File Edit
Taxa Unaligned Characters Distances Quartets Trees Splits
Method Select
Show
Hide
Apply
< Show
Hide >
Show all
Hide all

Show:
[1] 'pryor00_1_ITS'

Hide:
'pryor03_1_ITS'
'pryor03_2_SSU'
'pryor03_3_gpd'
'pryor00_2_SSUrDNA'

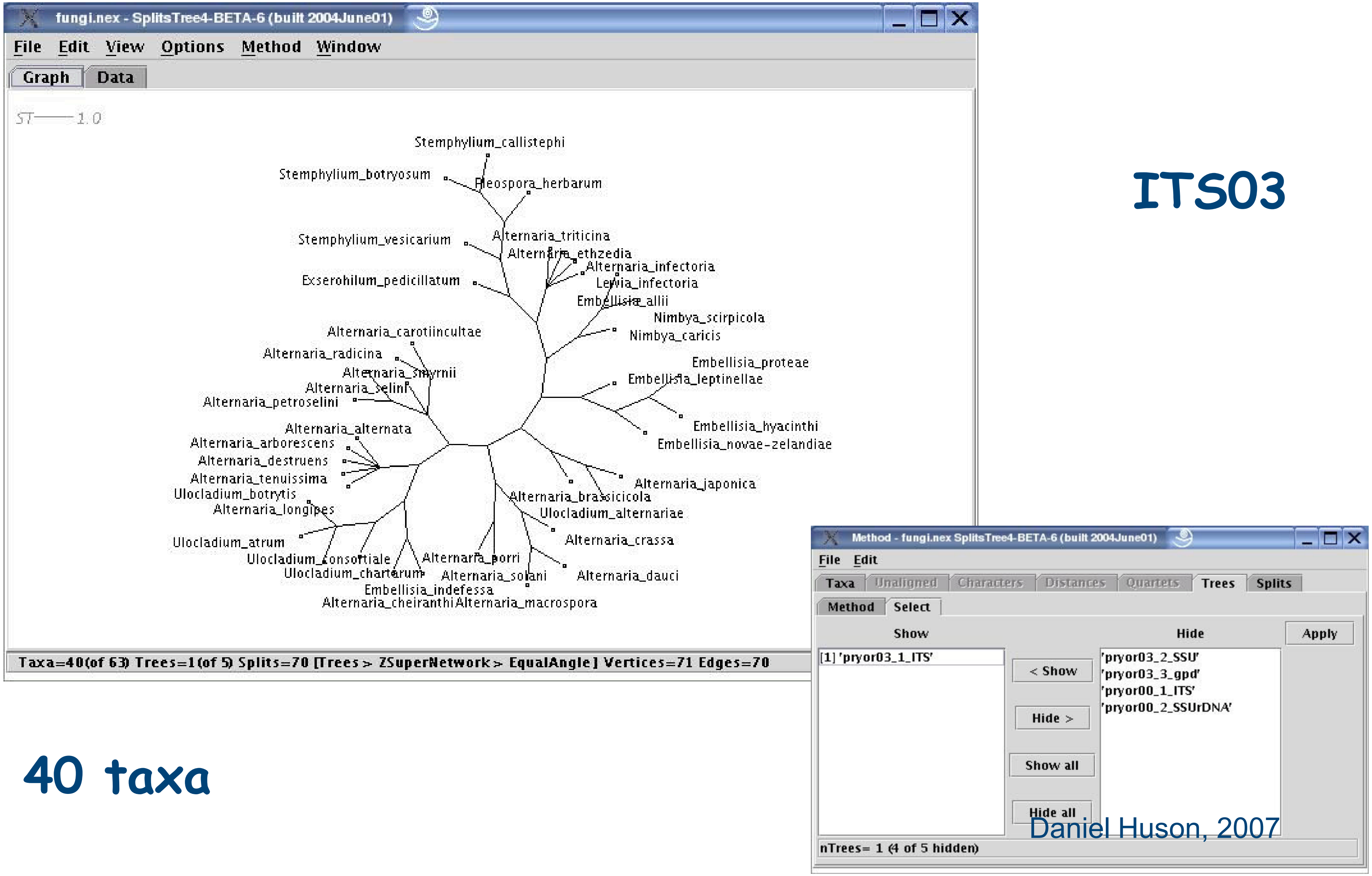
nTrees= 1 (4 of 5 hidden)

ITS00

46 taxa

Daniel Huson, 2007

Individual Gene Trees



Taxa=40(of 63) Trees=1(of 5) Splits=70 [Trees > ZSuperNetwork > EqualAngle] Vertices=71 Edges=70

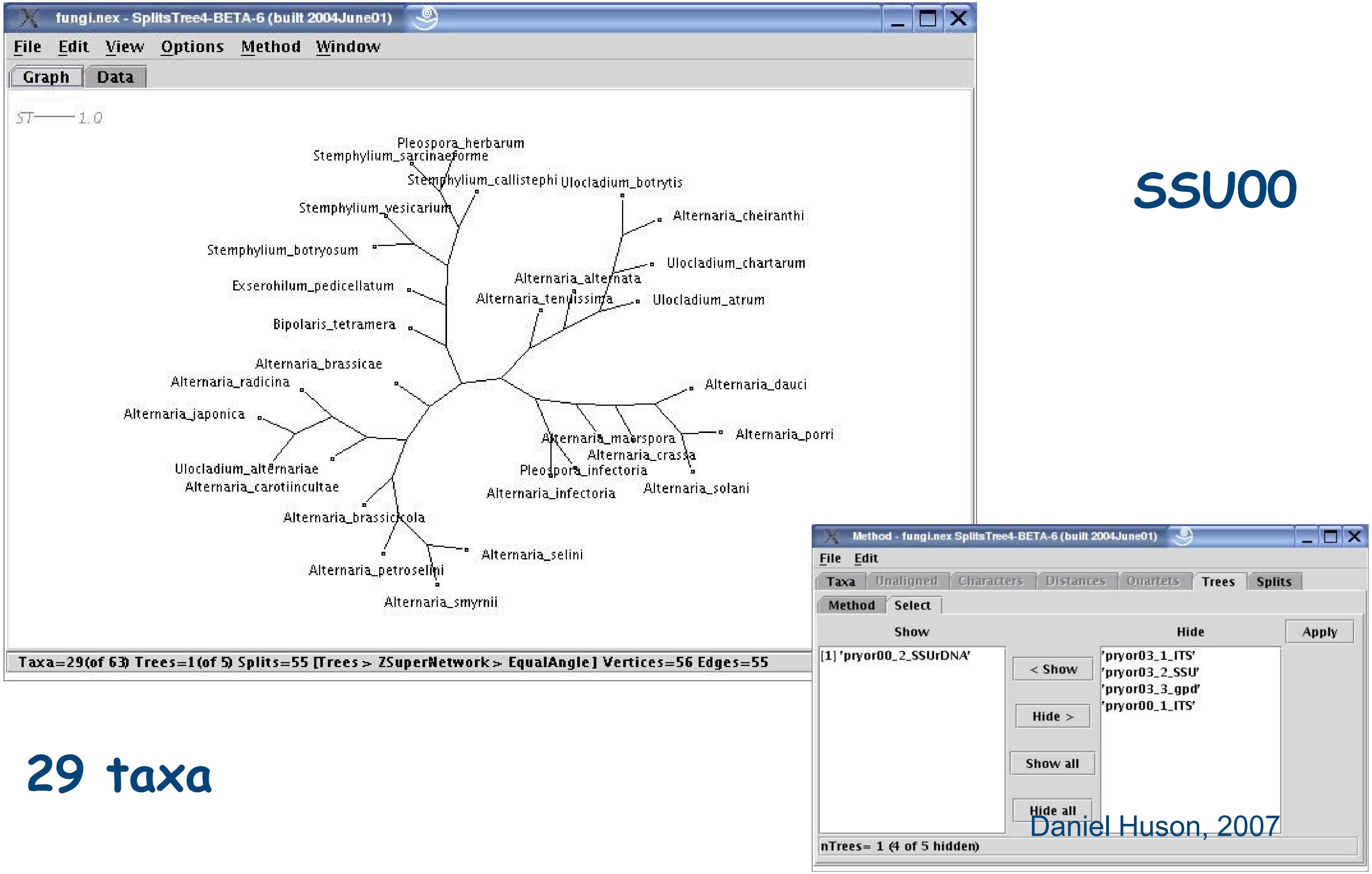
Show	Hide
[1] 'pryor03_1_ITS'	'pryor03_2_SSU' 'pryor03_3_gpd' 'pryor00_1_ITS' 'pryor00_2_SSUrDNA'

nTrees= 1 (4 of 5 hidden)

ITS03

40 taxa

Individual Gene Trees



Taxa=29(of 63) Trees=1(of 5) Splits=55 [Trees > ZSuperNetwork > EqualAngle] Vertices=56 Edges=55

Show:
[1] 'pryor00_2_SSUrDNA'

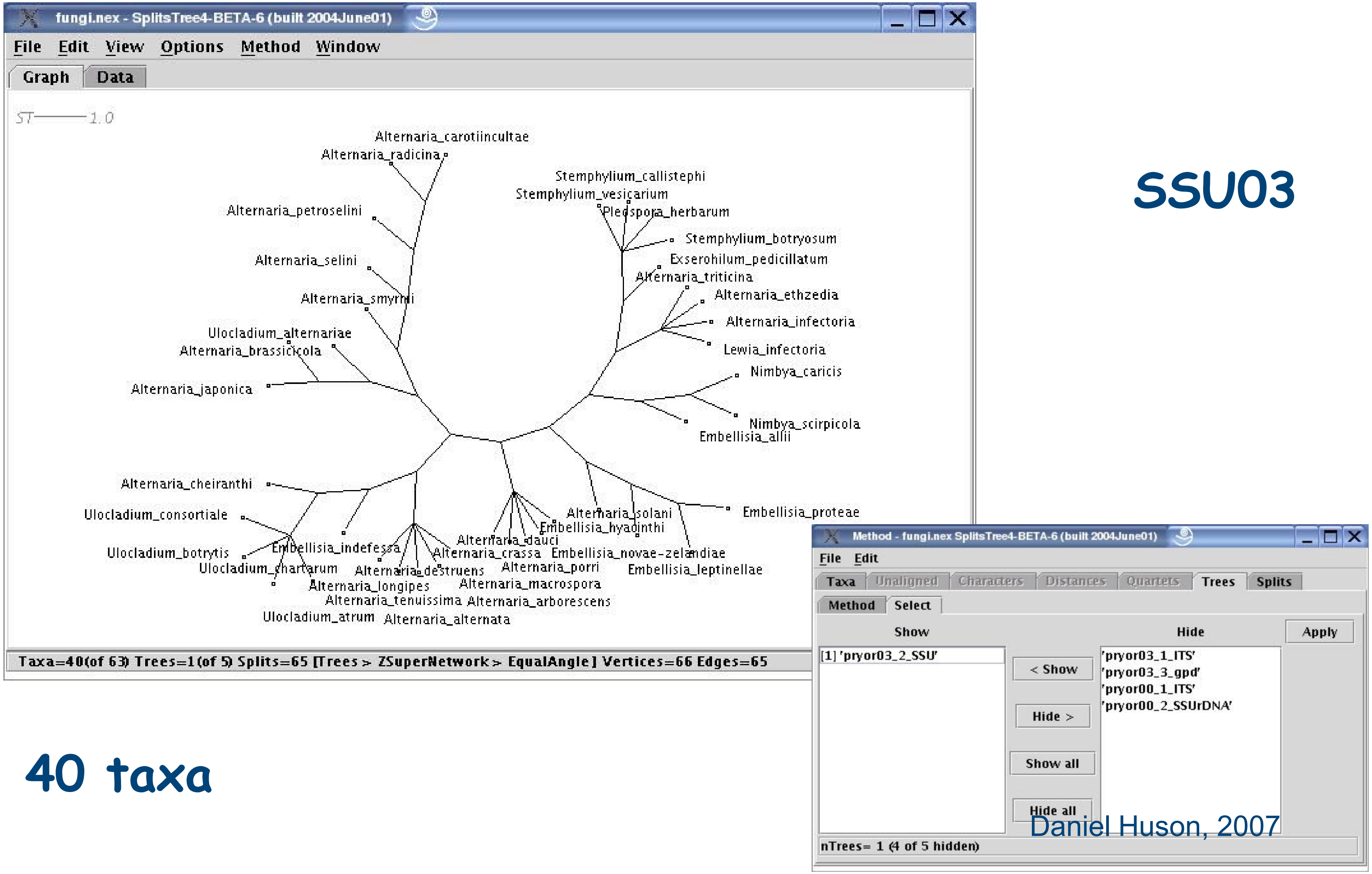
Hide:
'pryor03_1_ITS'
'pryor03_2_SSU'
'pryor03_3_gpd'
'pryor00_1_ITS'

nTrees= 1 (4 of 5 hidden)

SSU00

29 taxa

Individual Gene Trees



Taxa=40(of 63) Trees=1(of 5) Splits=65 [Trees > ZSuperNetwork > EqualAngle] Vertices=66 Edges=65

Show: [1] 'pryor03_2_SSU'

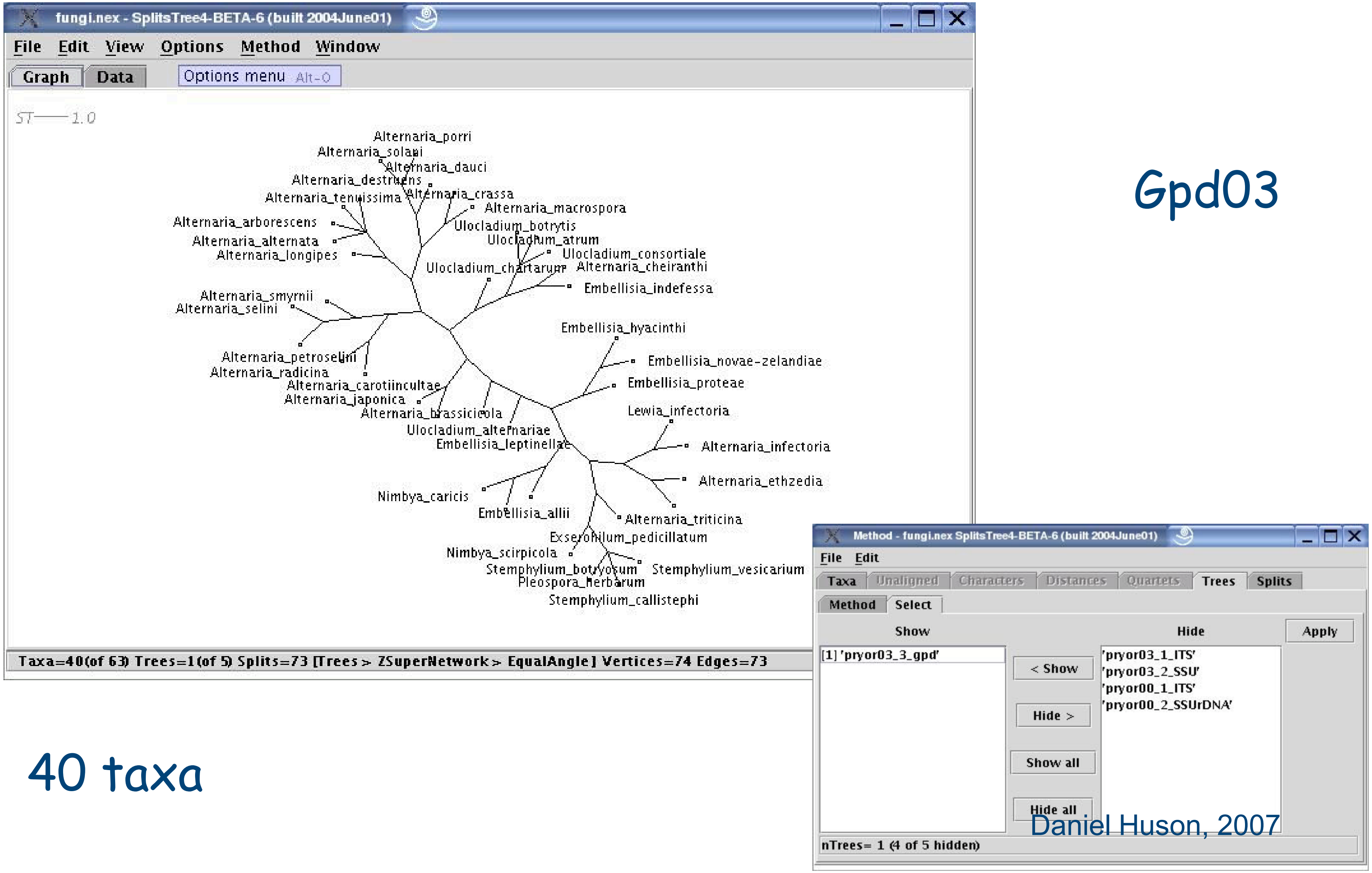
Hide: 'pryor03_1_ITS' 'pryor03_3_gpd' 'pryor00_1_ITS' 'pryor00_2_SSUrDNA'

nTrees= 1 (4 of 5 hidden)

SSU03

40 taxa

Individual Gene Trees



Taxa=40(of 63) Trees=1(of 5) Splits=73 [Trees > ZSuperNetwork > EqualAngle] Vertices=74 Edges=73

Show: [1] 'pryor03_3_gpd'

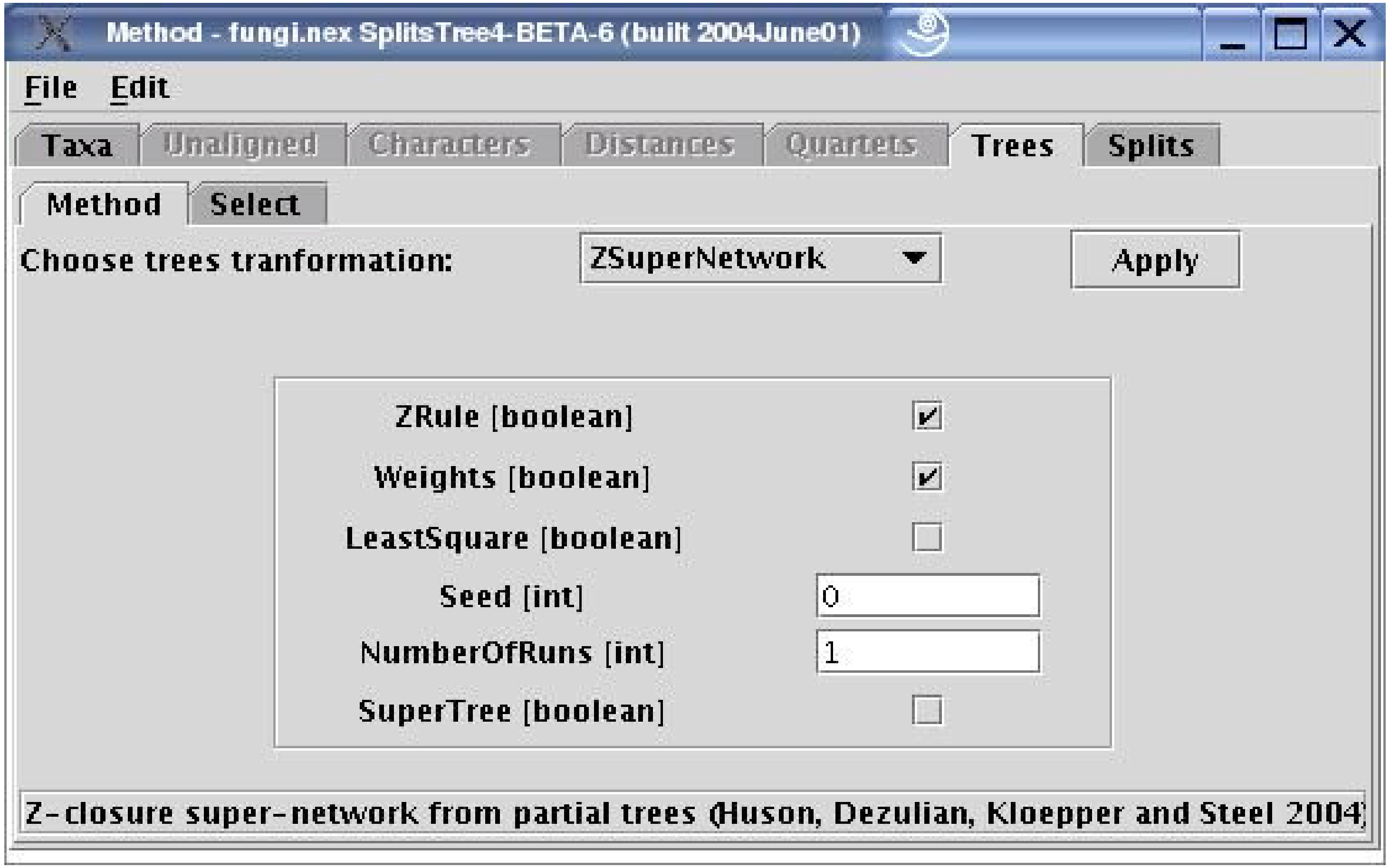
Hide: 'pryor03_1_ITS' 'pryor03_2_SSU' 'pryor00_1_ITS' 'pryor00_2_SSUrDNA'

nTrees= 1 (4 of 5 hidden)

Gpd03

40 taxa

Gene Trees as Super Network



Choose trees tranformation: ZSuperNetwork

ZRule [boolean] ☑

Weights [boolean] ☑

LeastSquare [boolean] □

Seed [int] 0

NumberOfRuns [int] 1

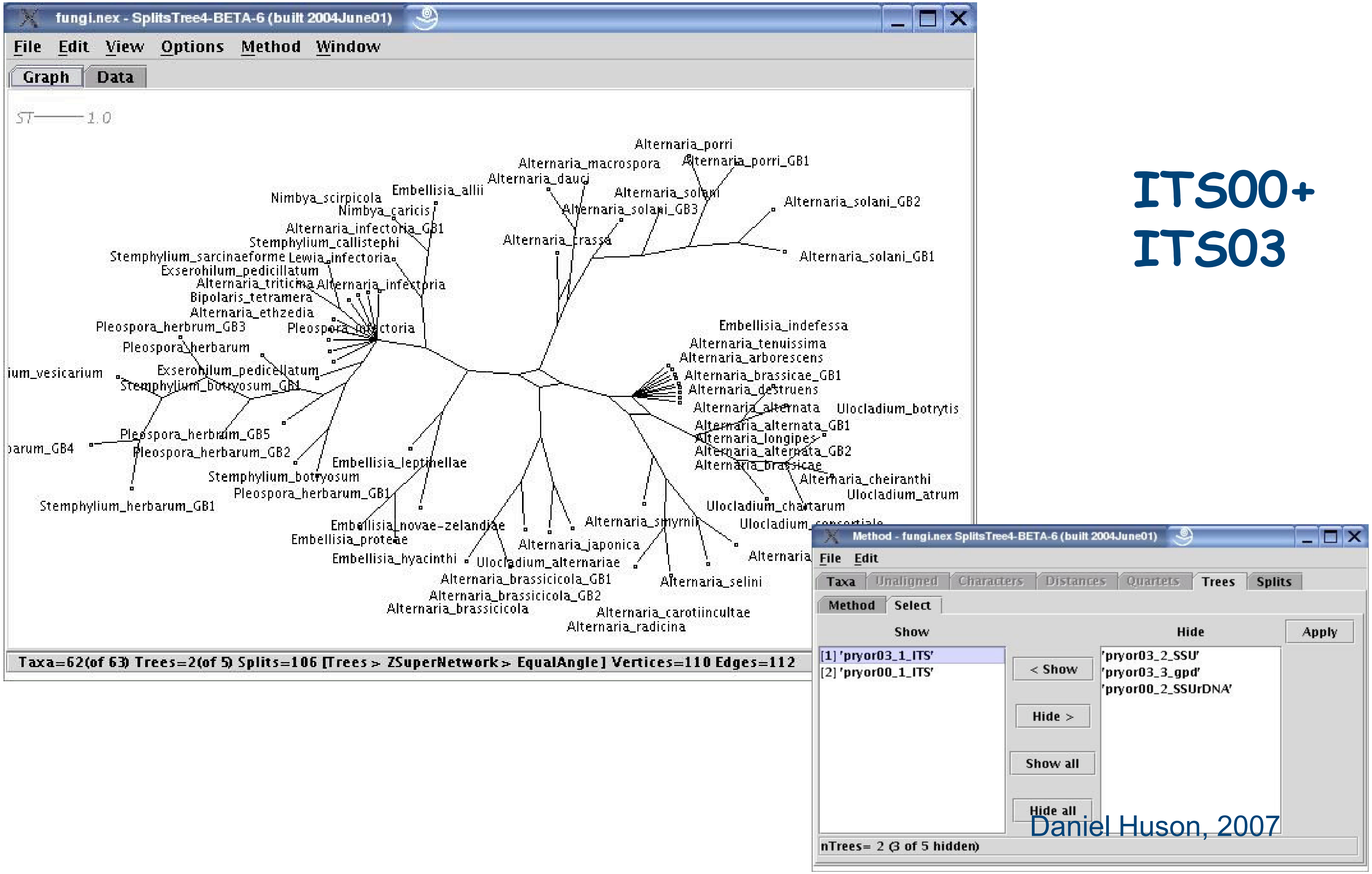
SuperTree [boolean] □

Z-closure super-network from partial trees (Huson, Dezulian, Kloepper and Steel 2004)

Z-closure: a fast super-network method

Daniel Huson, 2007

Gene Trees as Super Network



Taxa=62(of 63) Trees=2(of 5) Splits=106 [Trees > ZSuperNetwork > EqualAngle] Vertices=110 Edges=112

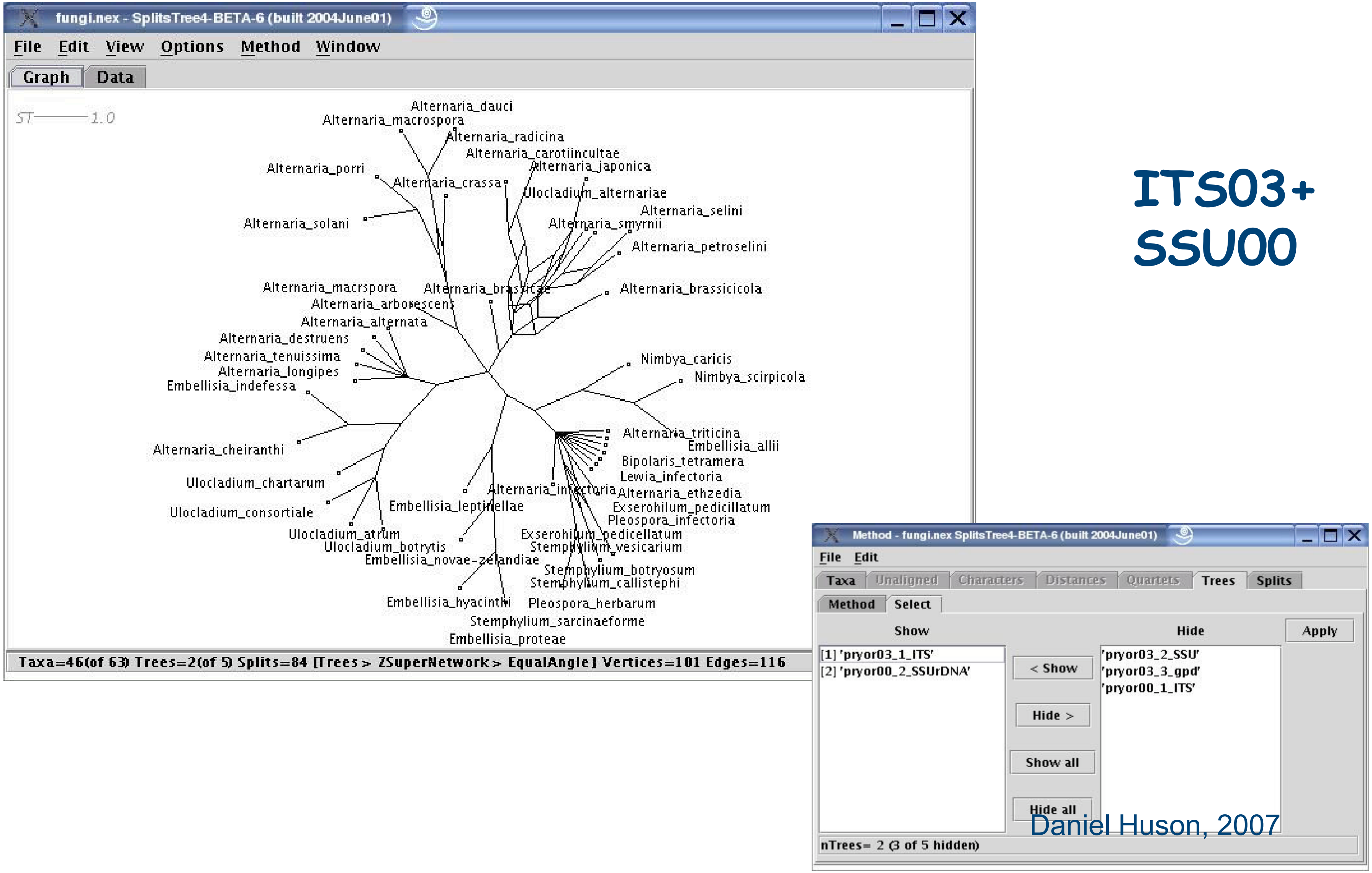
Show:
[1] 'pryor03_1_ITS'
[2] 'pryor00_1_ITS'

Hide:
'pryor03_2_SSU'
'pryor03_3_gpd'
'pryor00_2_SSUrDNA'

nTrees= 2 (3 of 5 hidden)

ITS00+
ITS03

Gene Trees as Super Network



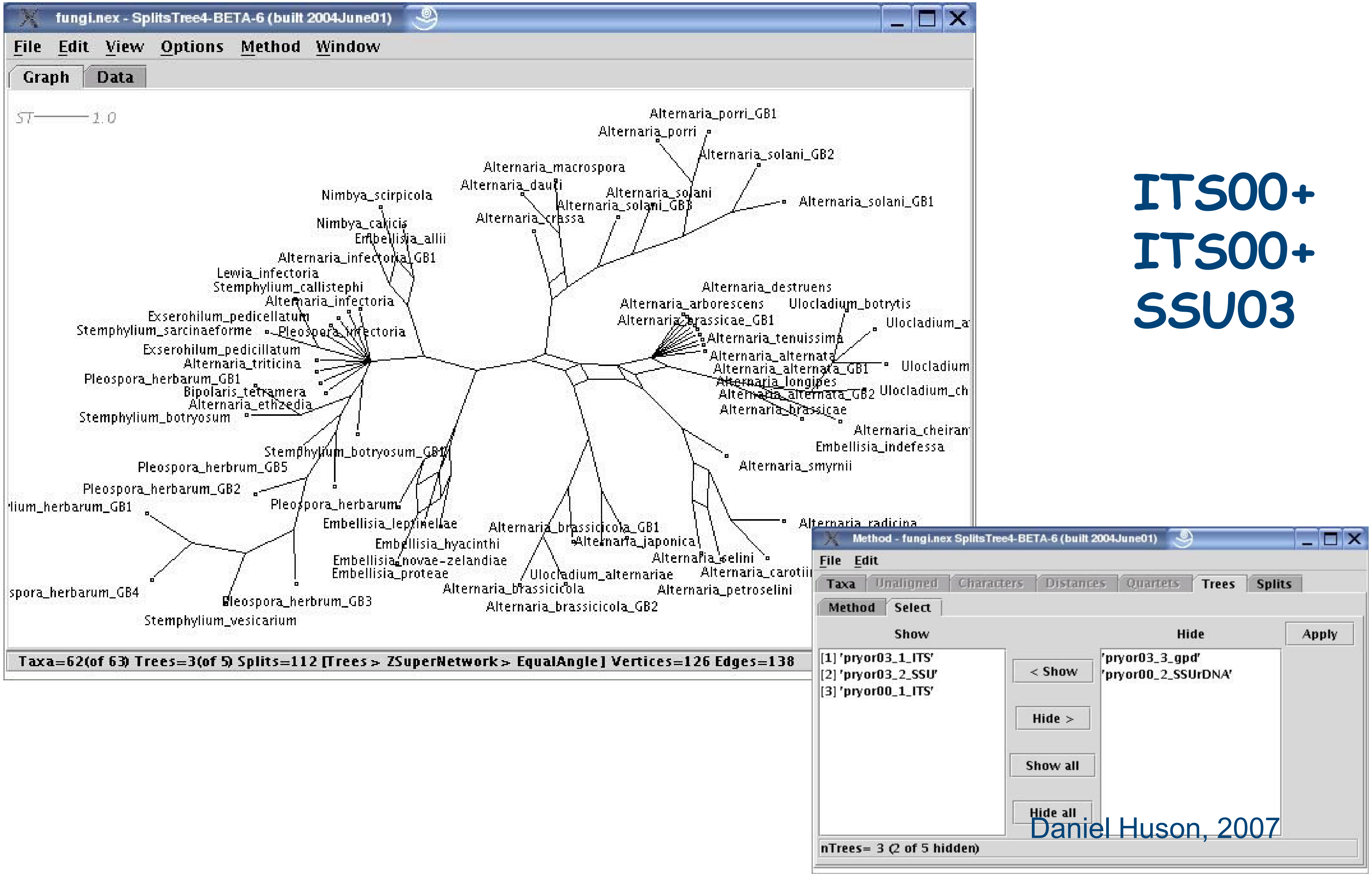
Taxa=46(of 63) Trees=2(of 5) Splits=84 [Trees > ZSuperNetwork > EqualAngle] Vertices=101 Edges=116

Show	Hide
[1] 'pryor03_1_ITS'	'pryor03_2_SSU'
[2] 'pryor00_2_SSUrDNA'	'pryor03_3_gpd'
	'pryor00_1_ITS'

nTrees= 2 (3 of 5 hidden)

ITS03+ SSU00

Gene Trees as Super Network



Taxa=62(of 63) Trees=3(of 5) Splits=112 [Trees > ZSuperNetwork > EqualAngle] Vertices=126 Edges=138

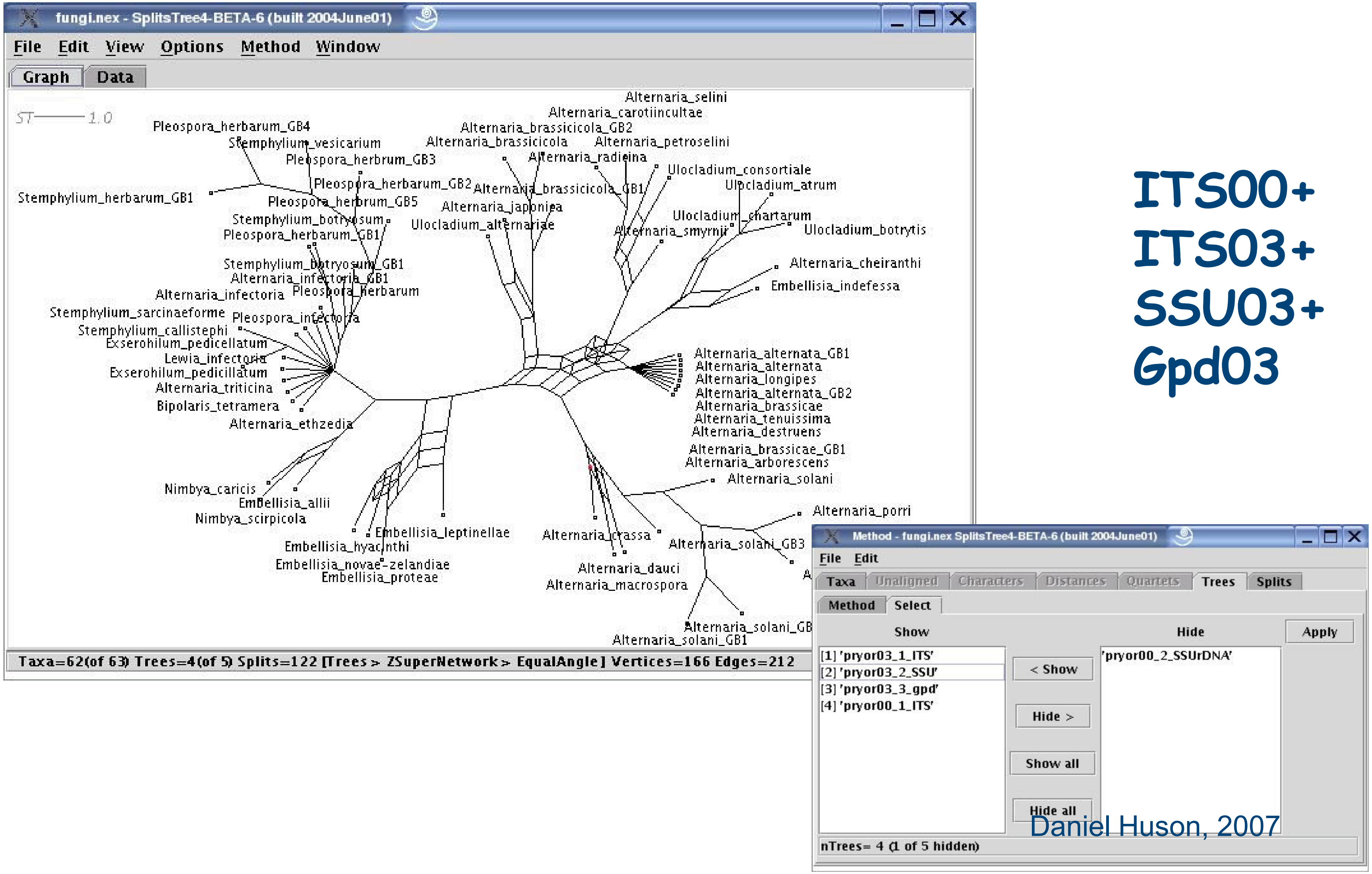
Show:
[1] 'pryor03_1_ITS'
[2] 'pryor03_2_SSU'
[3] 'pryor00_1_ITS'

Hide:
'pryor03_3_gpd'
'pryor00_2_SSUrDNA'

nTrees= 3 (2 of 5 hidden)

ITS00+
ITS00+
SSU03

Gene Trees as Super Network



Taxa=62(of 63) Trees=4(of 5) Splits=122 [Trees > ZSuperNetwork > EqualAngle] Vertices=166 Edges=212

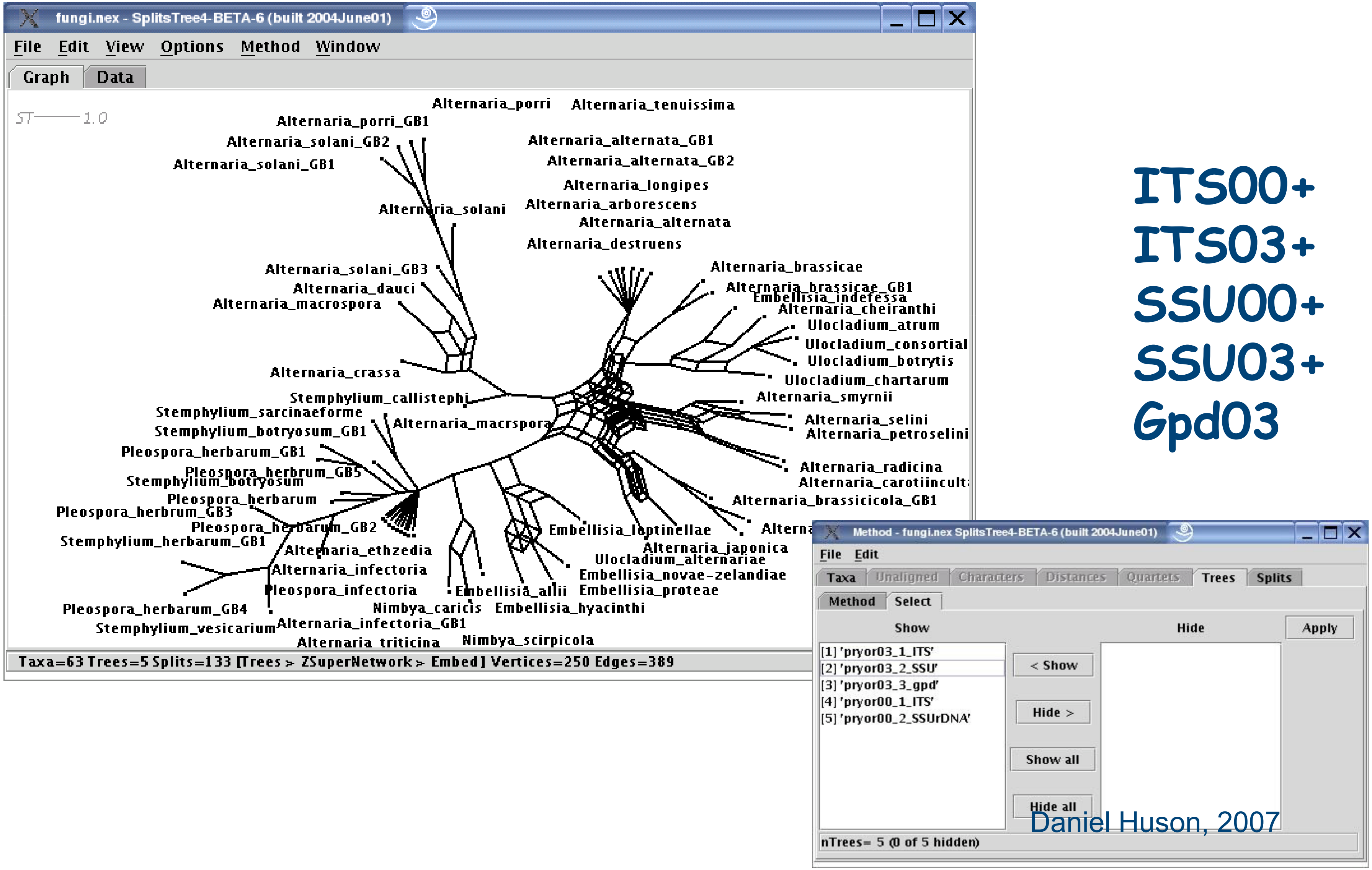
Show:
[1] 'pryor03_1_ITS'
[2] 'pryor03_2_SSU'
[3] 'pryor03_3_gpd'
[4] 'pryor00_1_ITS'

Hide:
'pryor00_2_SSUrDNA'

nTrees= 4 (1 of 5 hidden)

**ITS00+
ITS03+
SSU03+
Gpd03**

Gene Trees as Super Network



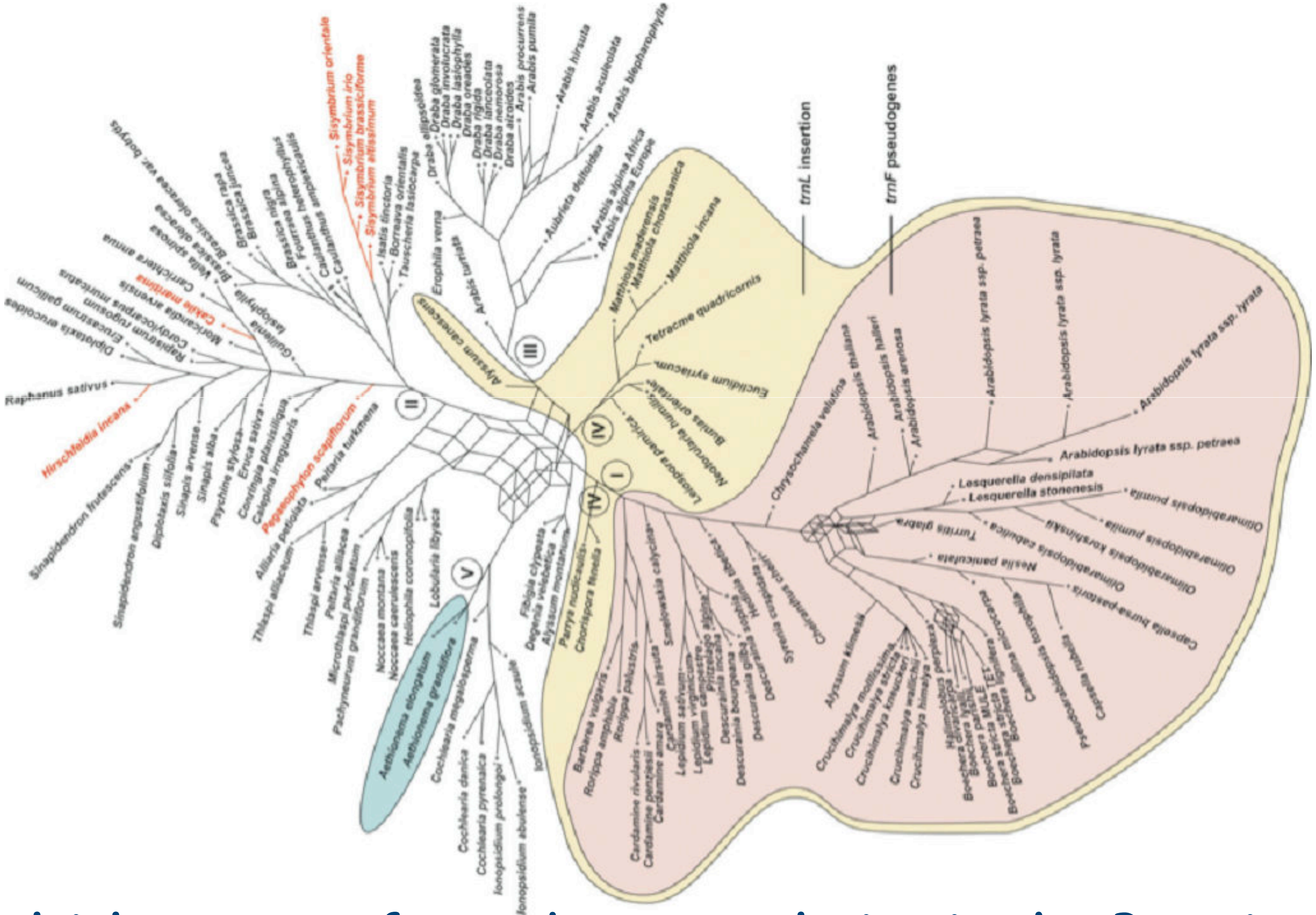
Taxa=63 Trees=5 Splits=133 [Trees > ZSuperNetwork > Embed] Vertices=250 Edges=389

[1] 'pryor03_1_ITS'
[2] 'pryor03_2_SSU'
[3] 'pryor03_3_gpd'
[4] 'pryor00_1_ITS'
[5] 'pryor00_2_SSUrDNA'

nTrees= 5 (0 of 5 hidden)

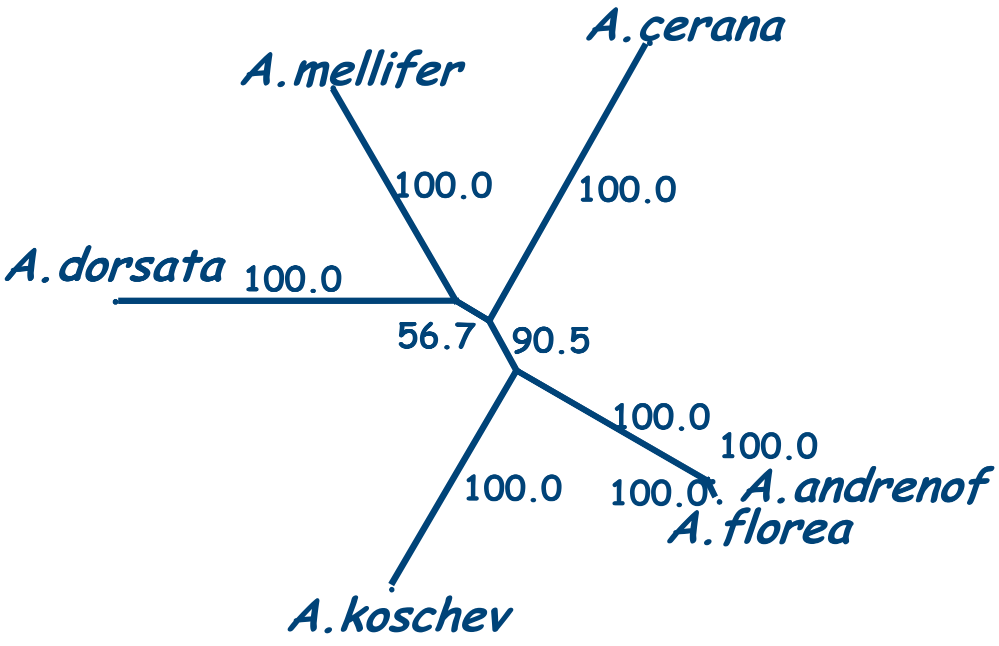
ITS00+
ITS03+
SSU00+
SSU03+
Gpd03

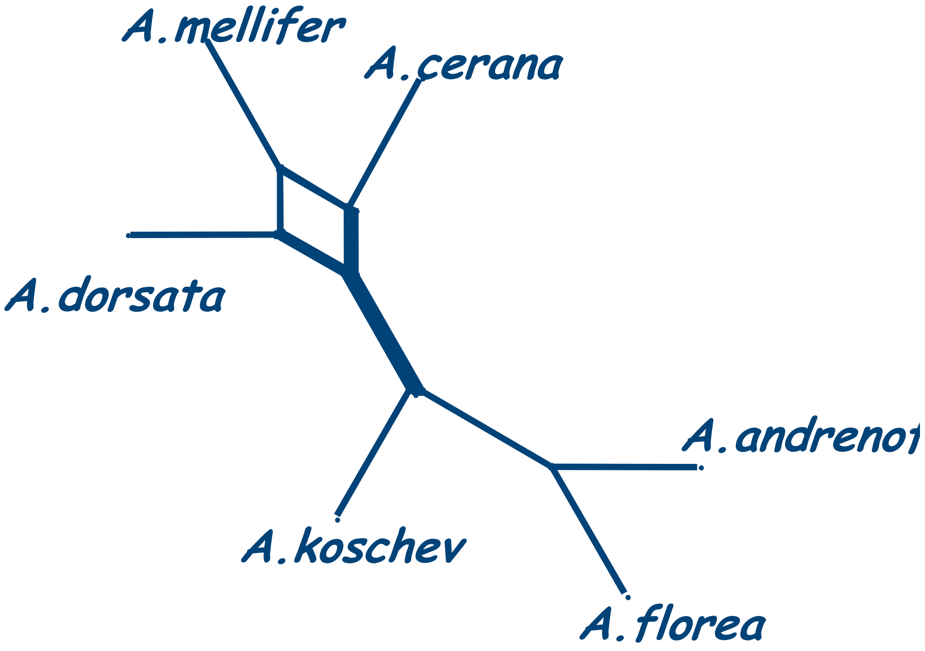
Super Network



Multiple events of pseudogene evolution in the Brassicaceae, obtained from 4 gene trees (Koch *et al.* 2007)

Related: Bootstrap Network





Bootstrapping tests robustness of tree with respect to sampling

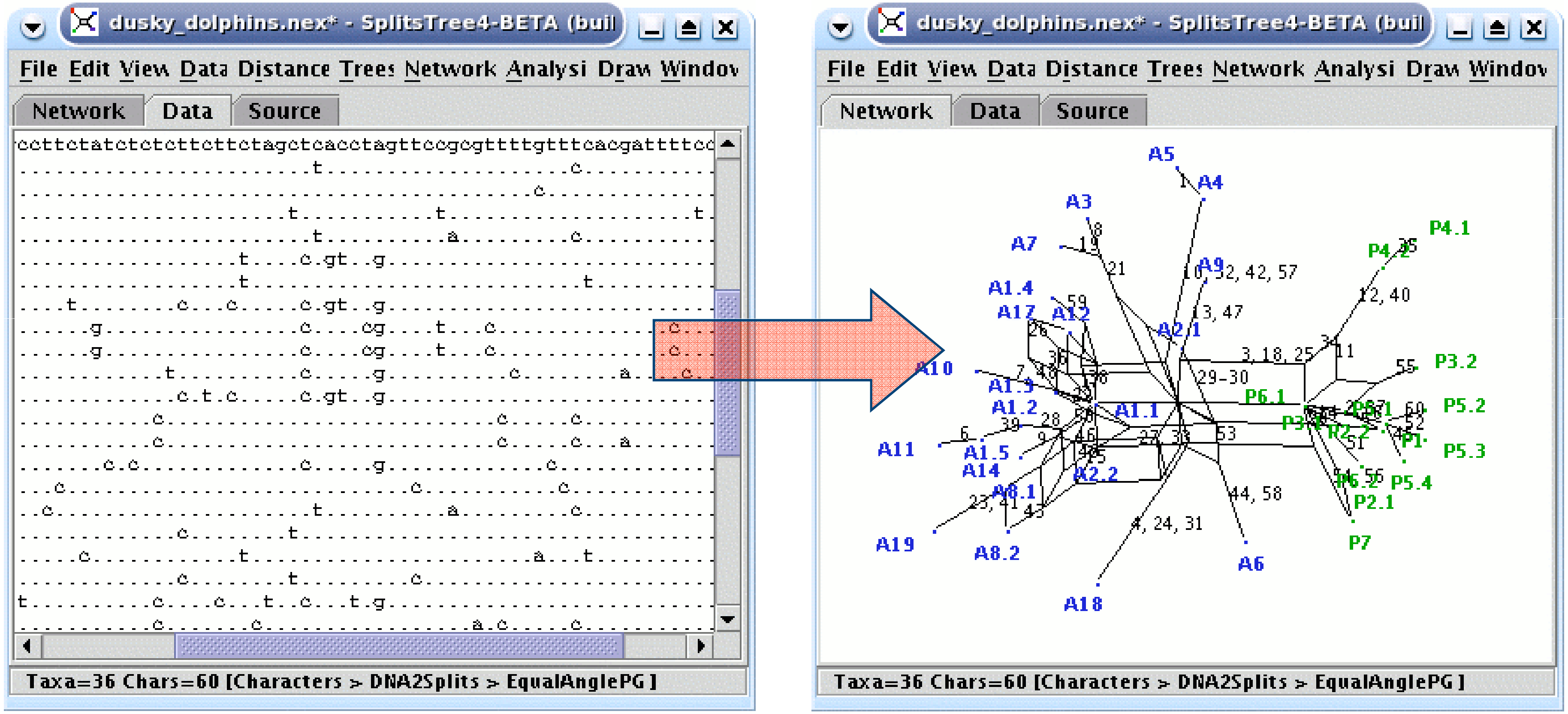
Bootstrap network displays competing signals

Data: DNA sequences of length 677 for six honey bee species.

Sequence & Distance-Based Split Networks

- So, incompatible splits arise naturally in the context of multiple trees.
- There also exist a number of methods that generate incompatible splits directly from
  - characters (a multiple sequence alignment), or
  - a distance matrix.

Sequences to Split Network



```
ccttctatctctcttcttctagctcacctagttccgcgttttgtttcacgatttcc
........................t....................c...........
..........................................c..............
......................t...........t....................t.
........................t..........a.........c...........
..................t....c.gt..g...........................
..................t...........................t..........
....t........c...c.....c.gt..g...........................
......g................c....cg....t...c..............c...
......g................c....cg....t...c..............c...
............t..........c.....g..........c........a....c..
.............c.t.c.....c.gt..g...........................
...........c...........................c.....c...........
...........c...........................c.....c...a.......
.......c.c.............c.....g.............c.............
...c............................c...........c............
..c.....................t..........a.........c...........
.............c........t..................................
.....c............t.......................a...t..........
.............c........t.........c........................
t..........c....c...t..c...t.g...........................
...........c.......c.................a.c.....c...........
```

Taxa=36 Chars=60 [Characters > DNA2Splits > EqualAnglePG]

Taxa=36 Chars=60 [Characters > DNA2Splits > EqualAnglePG]

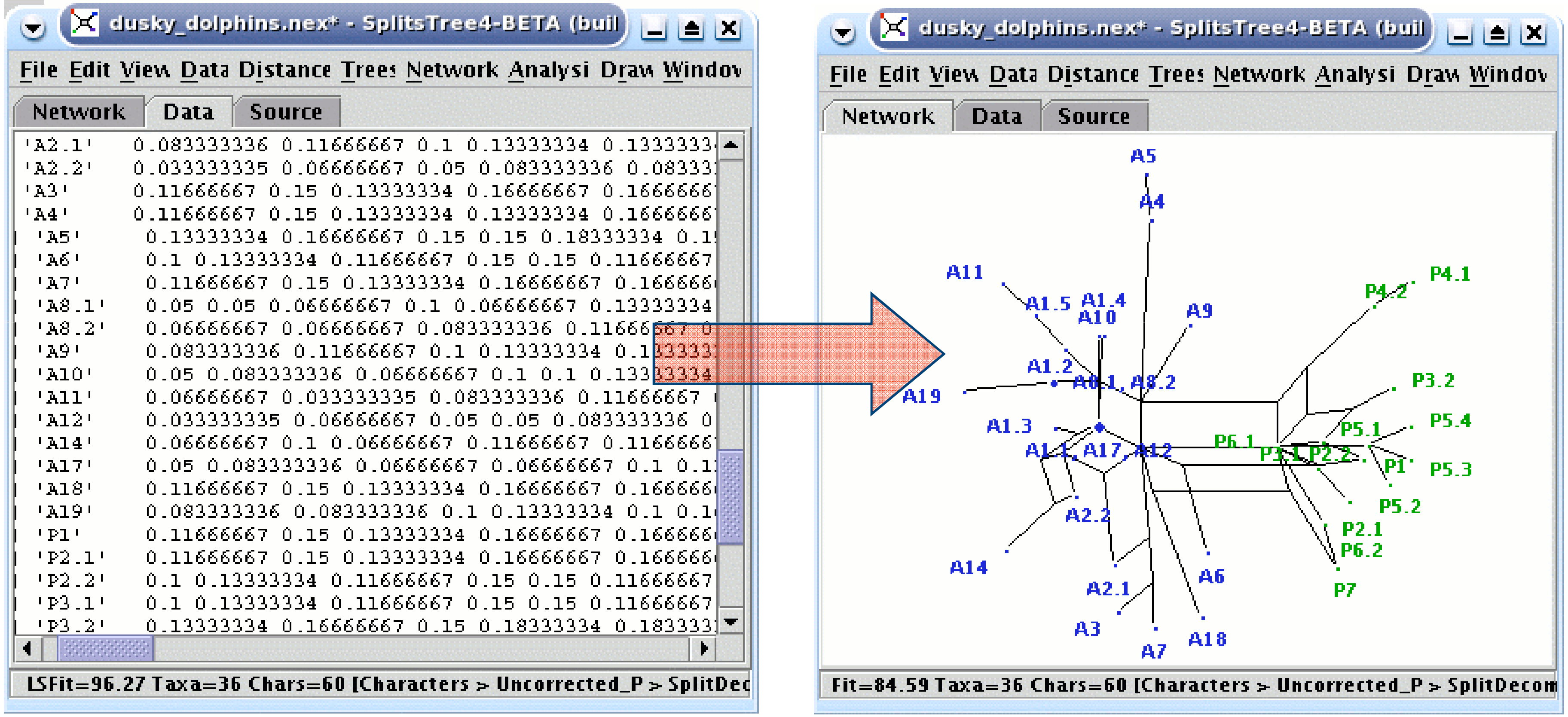
Data: Dusky dolphins (Cassens *et al.*, 2003)

If characters have only 2 states and not too conflicting: interpret columns as splits and draw full split network (median network)

Split Decomposition

- The Split Decomposition method computes a set of weighted X-splits Σ_{decomp} such that the sum of weights of all splits that separate two taxa $x, y \in X$ approximates the distance $D(x,y)$.
- It produces a *tree*, whenever the distance matrix fits a tree, and else produces a *network* that displays different and incompatible signals.

Distances to Split Network


dusky_dolphins.nex* - SplitsTree4-BETA (buil
File Edit View Data Distance Trees Network Analysi Draw Windov
Network
Data
Source

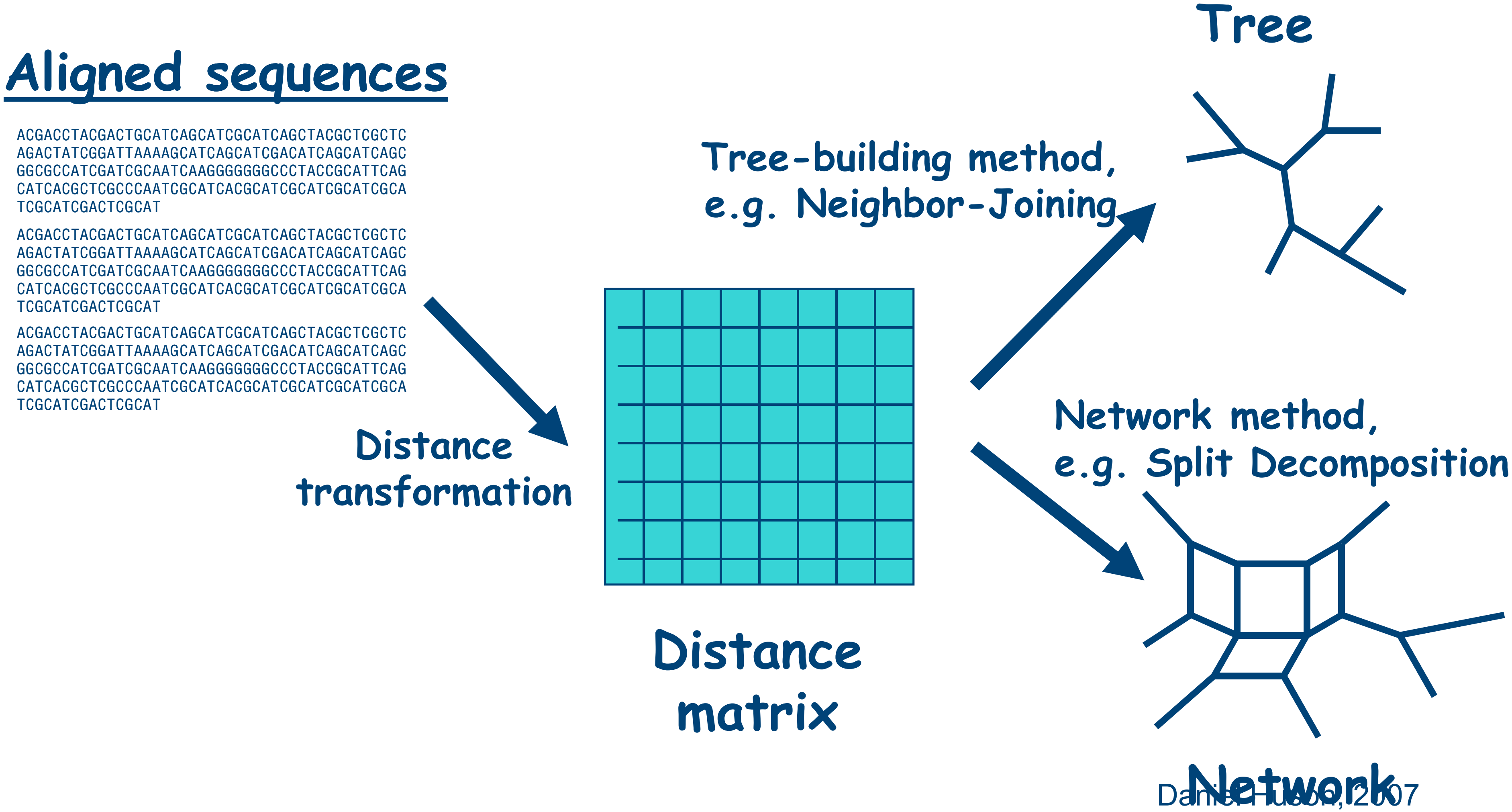
```
'A2.1'    0.083333336 0.11666667 0.1 0.13333334 0.1333333
'A2.2'    0.033333335 0.06666667 0.05 0.083333336 0.08333
'A3'      0.11666667 0.15 0.13333334 0.16666667 0.1666666
'A4'      0.11666667 0.15 0.13333334 0.13333334 0.1666666
| 'A5'     0.13333334 0.16666667 0.15 0.15 0.18333334 0.1
| 'A6'     0.1 0.13333334 0.11666667 0.15 0.15 0.11666667
| 'A7'     0.11666667 0.15 0.13333334 0.16666667 0.166666
| 'A8.1'   0.05 0.05 0.06666667 0.1 0.06666667 0.13333334
| 'A8.2'   0.06666667 0.06666667 0.083333336 0.11666667 0
| 'A9'     0.083333336 0.11666667 0.1 0.13333334 0.1333333
| 'A10'    0.05 0.083333336 0.06666667 0.1 0.1 0.13333334
| 'A11'    0.06666667 0.033333335 0.083333336 0.11666667
| 'A12'    0.033333335 0.06666667 0.05 0.05 0.083333336 0
| 'A14'    0.06666667 0.1 0.06666667 0.11666667 0.1166666
| 'A17'    0.05 0.083333336 0.06666667 0.06666667 0.1 0.1
| 'A18'    0.11666667 0.15 0.13333334 0.16666667 0.166666
| 'A19'    0.083333336 0.083333336 0.1 0.13333334 0.1 0.1
| 'P1'     0.11666667 0.15 0.13333334 0.16666667 0.166666
| 'P2.1'   0.11666667 0.15 0.13333334 0.16666667 0.166666
| 'P2.2'   0.1 0.13333334 0.11666667 0.15 0.15 0.11666667
| 'P3.1'   0.1 0.13333334 0.11666667 0.15 0.15 0.11666667
| 'P3.2'   0.13333334 0.16666667 0.15 0.18333334 0.183333
```

LSFit=96.27 Taxa=36 Chars=60 [Characters > Uncorrected_P > SplitDec

Fit=84.59 Taxa=36 Chars=60 [Characters > Uncorrected_P > SplitDecom

Split Decomposition or Neighbor-Net produces network from distances

Distance Methods



Aligned sequences

```
ACGACCTACGACTGCATCAGCATCGCATCAGCTACGCTCGCTC
AGACTATCGGATTAAAAGCATCAGCATCGACATCAGCATCAGC
GGCGCCATCGATCGCAATCAAGGGGGGGCCCTACCGCATTCAG
CATCACGCTCGCCCAATCGCATCACGCATCGCATCGCATCGCA
TCGCATCGACTCGCAT

ACGACCTACGACTGCATCAGCATCGCATCAGCTACGCTCGCTC
AGACTATCGGATTAAAAGCATCAGCATCGACATCAGCATCAGC
GGCGCCATCGATCGCAATCAAGGGGGGGCCCTACCGCATTCAG
CATCACGCTCGCCCAATCGCATCACGCATCGCATCGCATCGCA
TCGCATCGACTCGCAT

ACGACCTACGACTGCATCAGCATCGCATCAGCTACGCTCGCTC
AGACTATCGGATTAAAAGCATCAGCATCGACATCAGCATCAGC
GGCGCCATCGATCGCAATCAAGGGGGGGCCCTACCGCATTCAG
CATCACGCTCGCCCAATCGCATCACGCATCGCATCGCATCGCA
TCGCATCGACTCGCAT
```

Distance transformation

Distance matrix

Tree-building method,
e.g. Neighbor-Joining

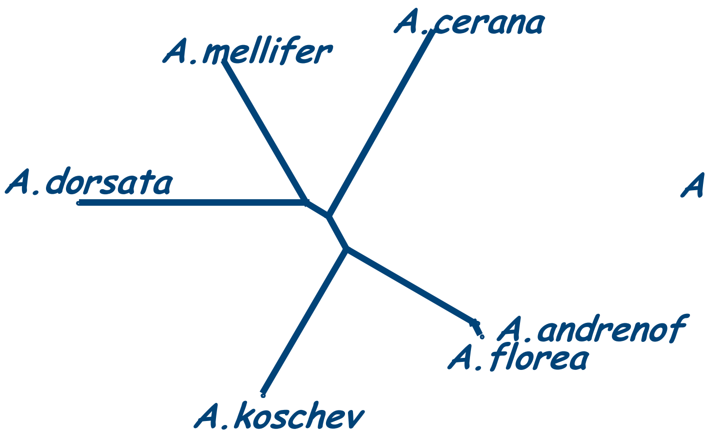
Tree

Network method,
e.g. Split Decomposition

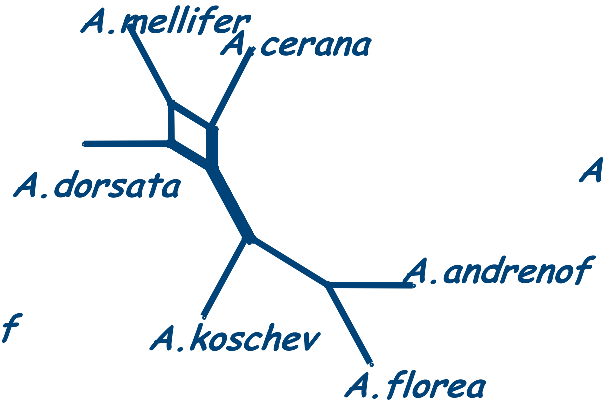
Network

Split Decomposition

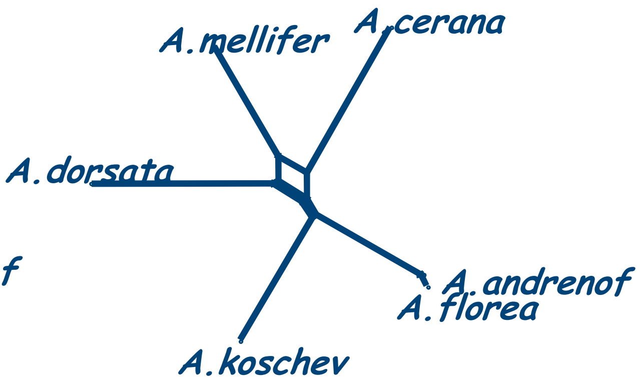
- Compare the result of Split Decomposition with an NJ tree and bootstrap network:



Bio-NJ tree



Bootstrap network



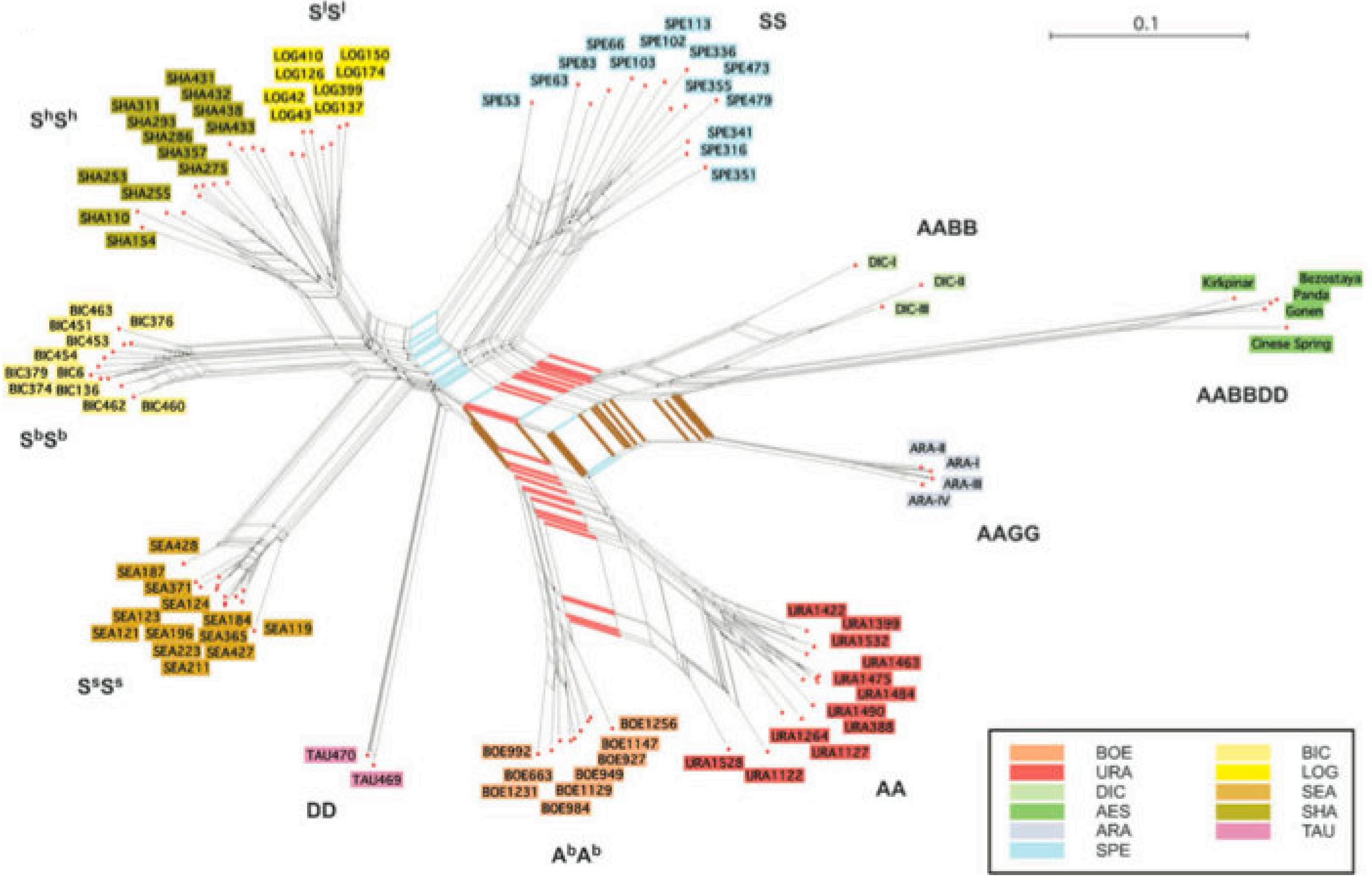
Split network obtained via the Split Decomposition

Neighbor-Net

- Split Decomposition is useful for visualizing conflicting signals in a data set. However, it is sensitive to noise and only has good resolution for small or clean data sets.

- The Neighbor-Net method is a hybrid of Neighbor-Joining and Split Decomposition. It is applicable to data sets containing hundreds of taxa. However, it tends to produce spider-webs.

Neighbor-Net



Split network computed via Neighbor-net from distances from AFLP markers (Kilian *et al* 2007)

Daniel Huson, 2007

Software

- SplitsTree4 provides implementations of *all* methods described in this chapter, including a number of different algorithms for constructing networks from splits.

- SpectroNet provides an algorithm for constructing a split network (a special case, namely the median network) and some related methods

Implicit vs Explicit Networks

Two fundamentally different types of phylogenetic networks:

- Implicit networks aim at displaying incompatible signals
  - Example: split networks
- Explicit networks aim at providing an explicit model of “reticulate evolution“
  - Example: hybridization and recombination networks

Part III

1. Phylogenetic trees
2. Consensus networks and super networks
3. **Hybridization and reticulate networks**
4. Recombination networks
5. Other

Overview

- Hybrid speciation.
- A simple model of evolution that incorporates gene trees and reticulation events.
- Reticulate networks and some approaches for inferring them from gene trees.
- Software.

Hybridization

- Occurs when two organisms from different species interbreed and combine their chromosomes

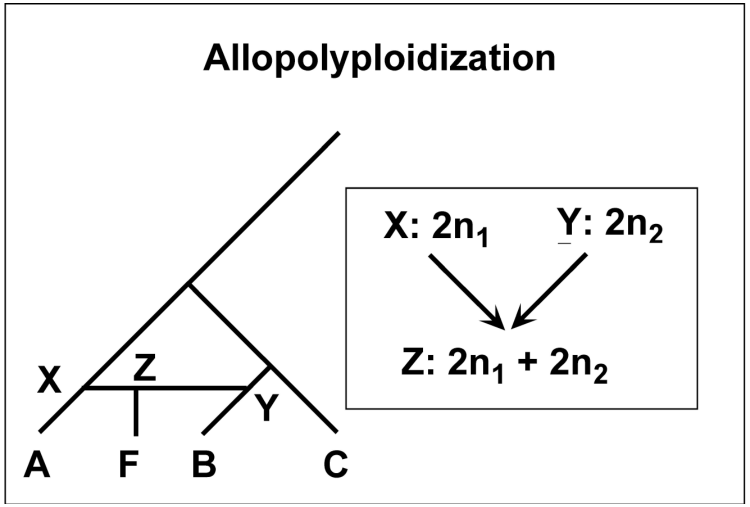
Water hemp

Hybrid

Pigs weed

Speciation by Hybridization 1

- In allopolyploidization, two different lineages produce a new species that has the complete nuclear genomes of both parental species:



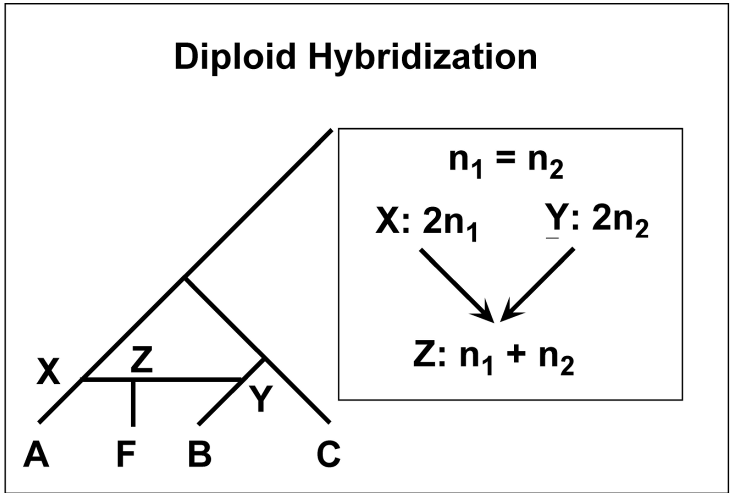
C.R. Linder, B.M.E. Moret, L. Nakhleh, and T. Warnow (2004)

Speciation by Hybridization 1

- Two parents *X* and *Y* each pass on their whole diploid genomes, with $2n_1$ and $2n_2$ chromosomes, respectively, to produce a polyploid 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.

Speciation by Hybridization 2

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



C.R. Linder, B.M.E. Moret, L. Nakhleh, and T. Warnow (2004)

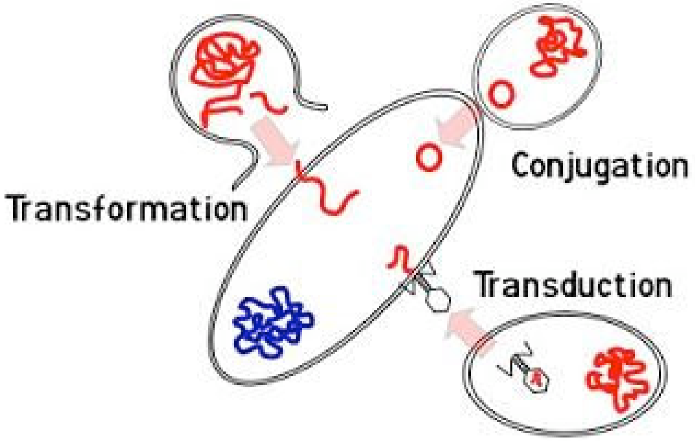
Speciation by Hybridization 2

- 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.

Horizontal Gene Transfer

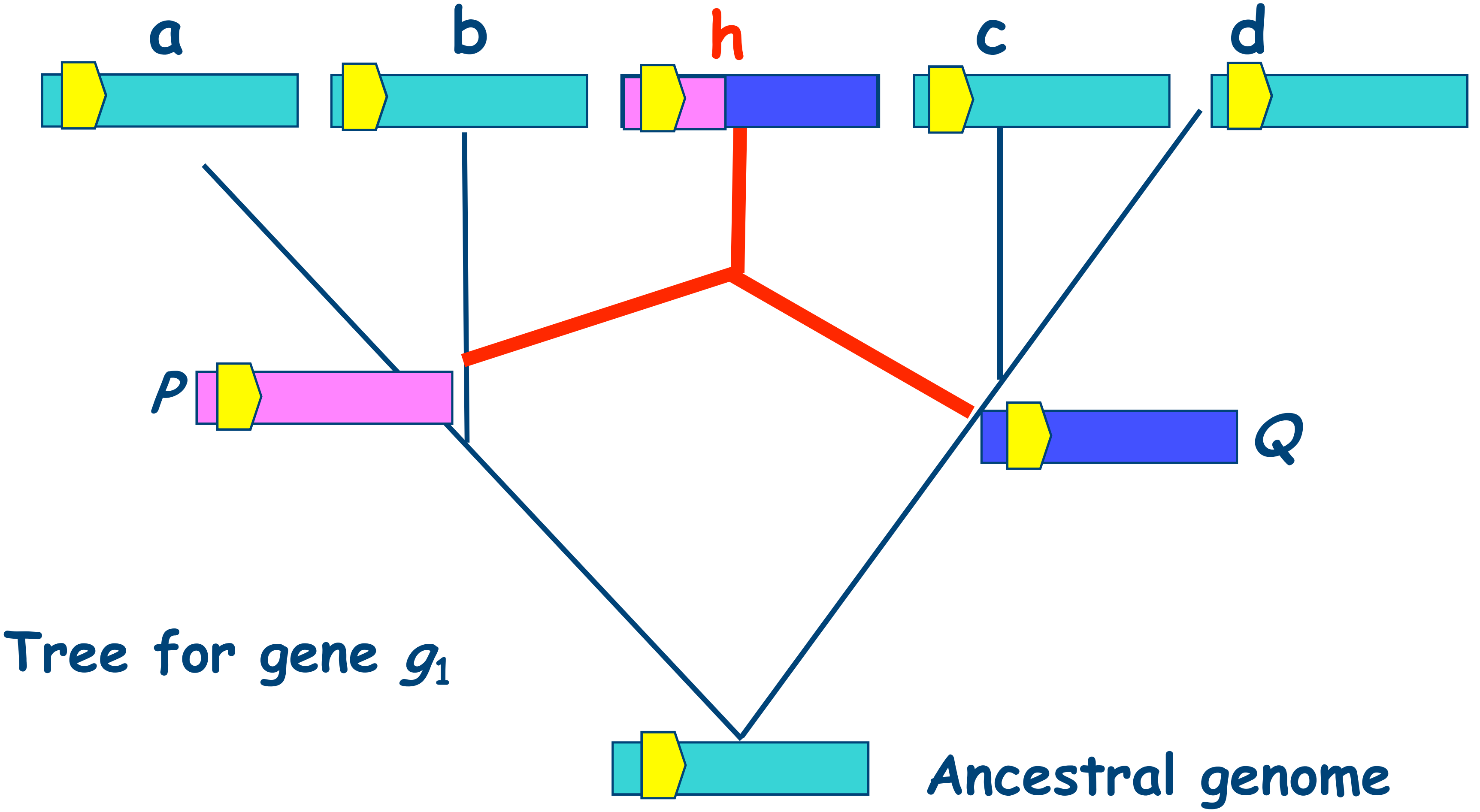
- There are a number of known mechanisms by which bacteria can exchange genes
  - Transformation
  - Conjugation
  - transduction

Mechanisms of Gene Exchange



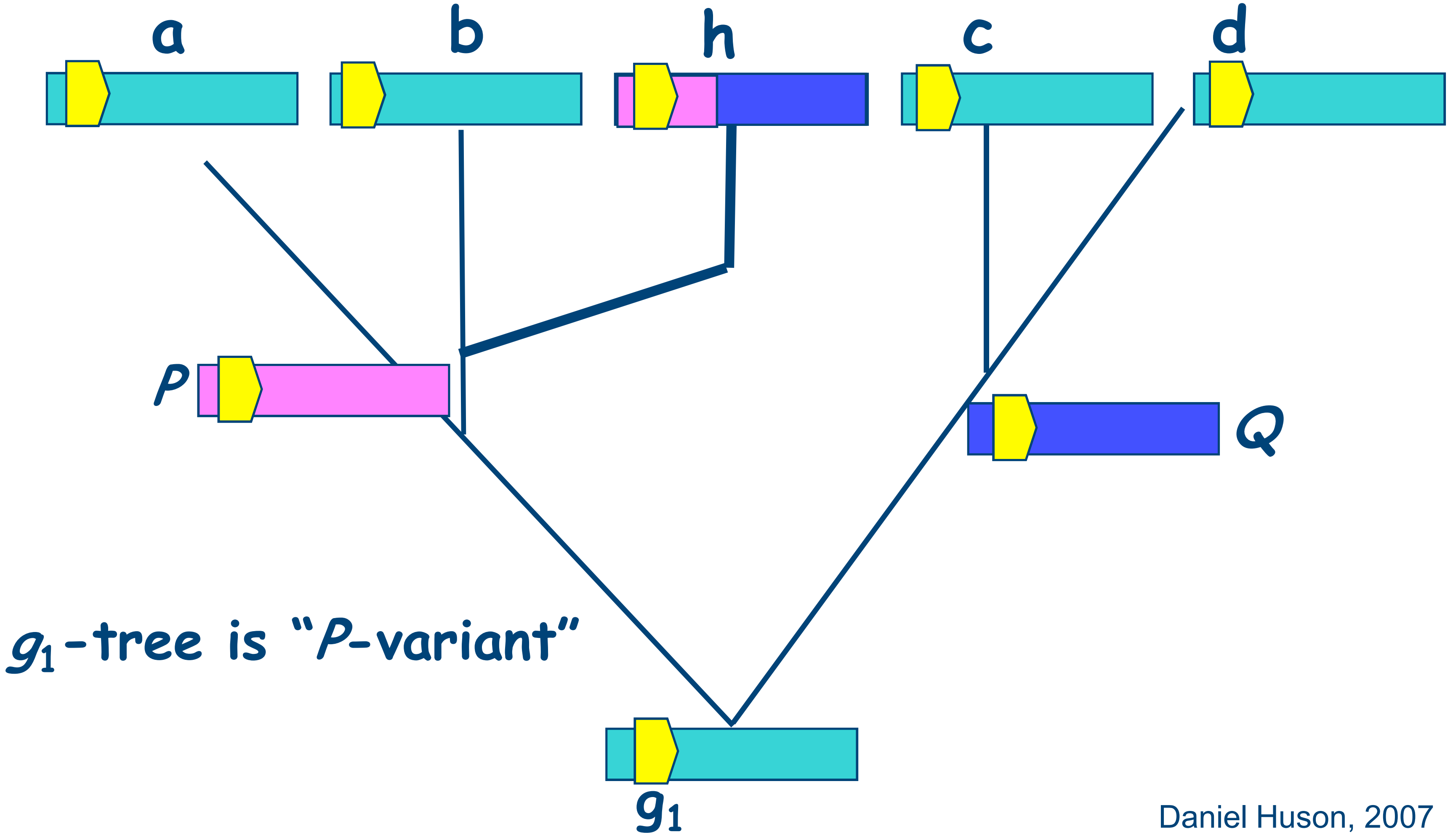
http://www.pitt.edu/~heh1/research.html

A Simple Model Of Reticulate Evolution



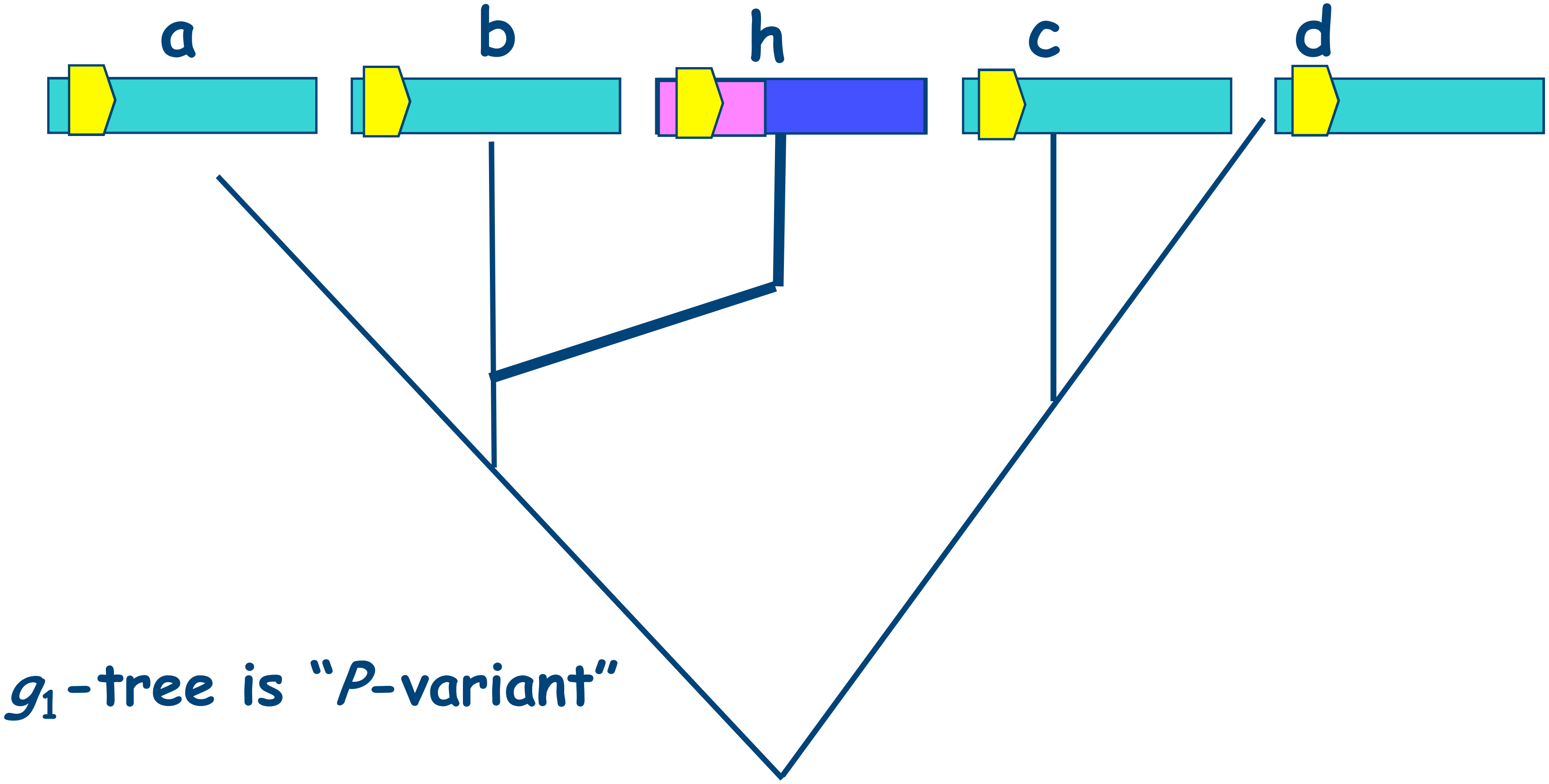
Daniel Huson, 2007

A Simple Model Of Reticulate Evolution



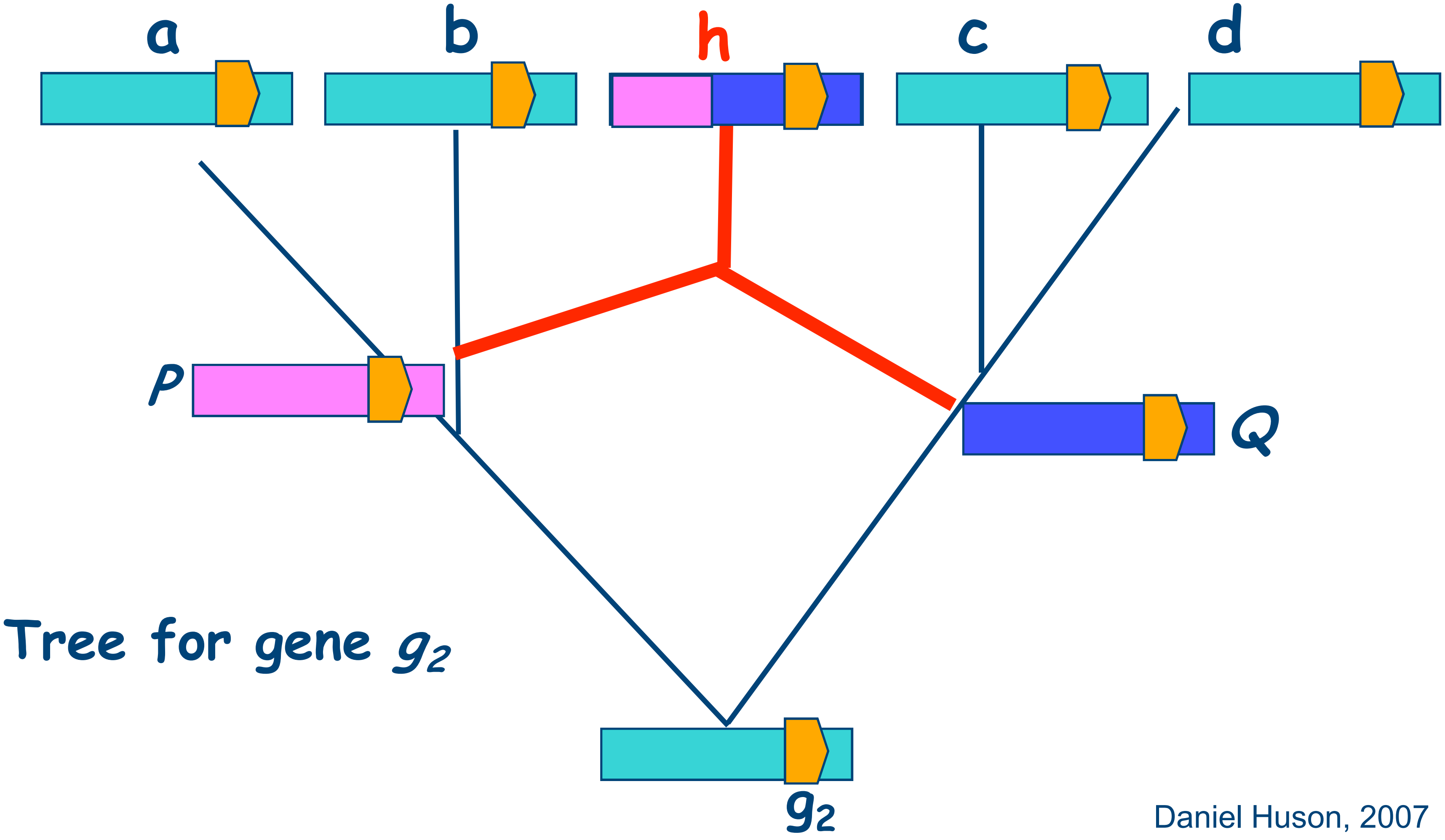
g_1-tree is "P-variant"

A Simple Model Of Reticulate Evolution



g_1-tree is "*P*-variant"

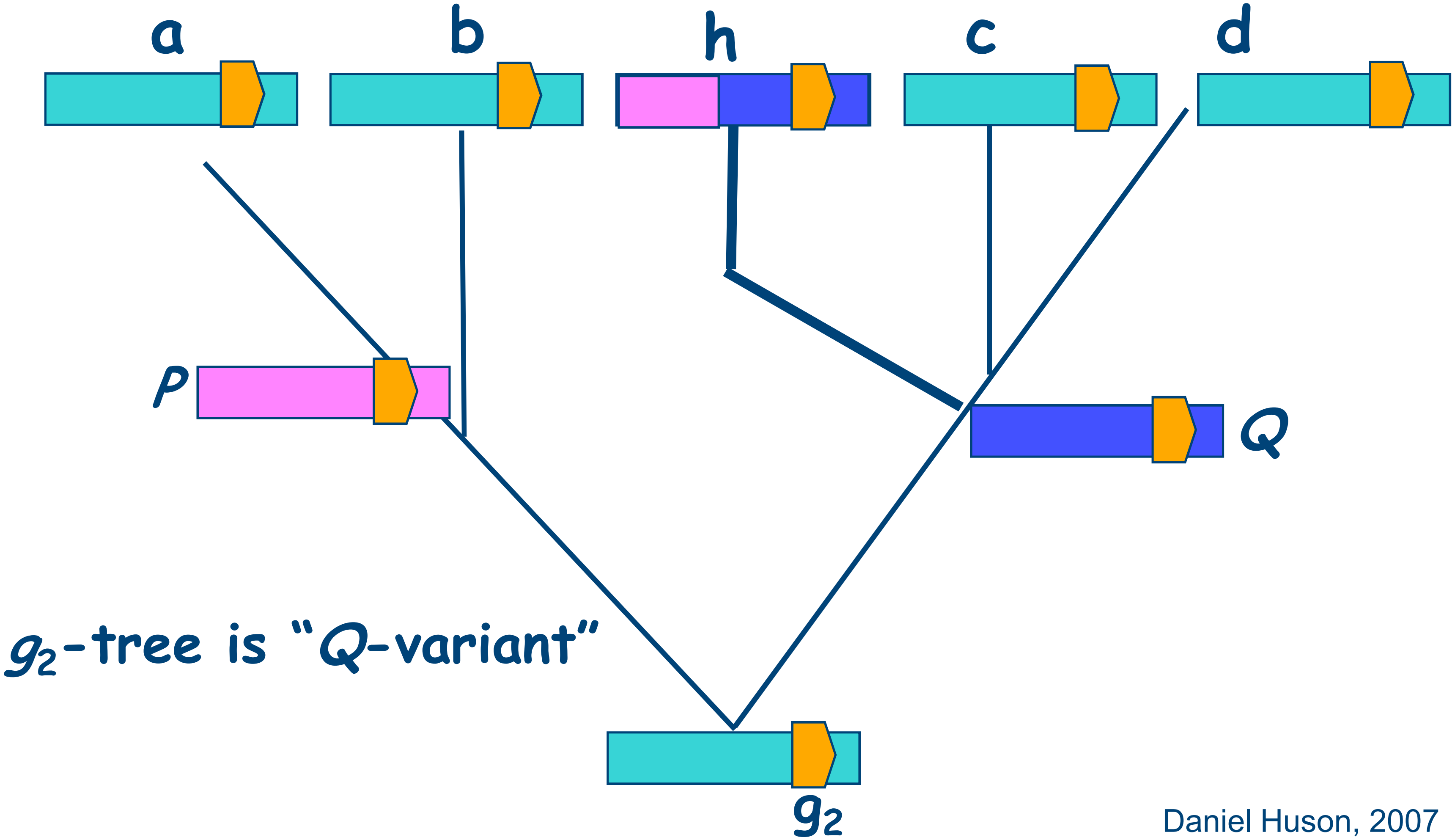
A Simple Model Of Reticulate Evolution



Tree for gene g_2

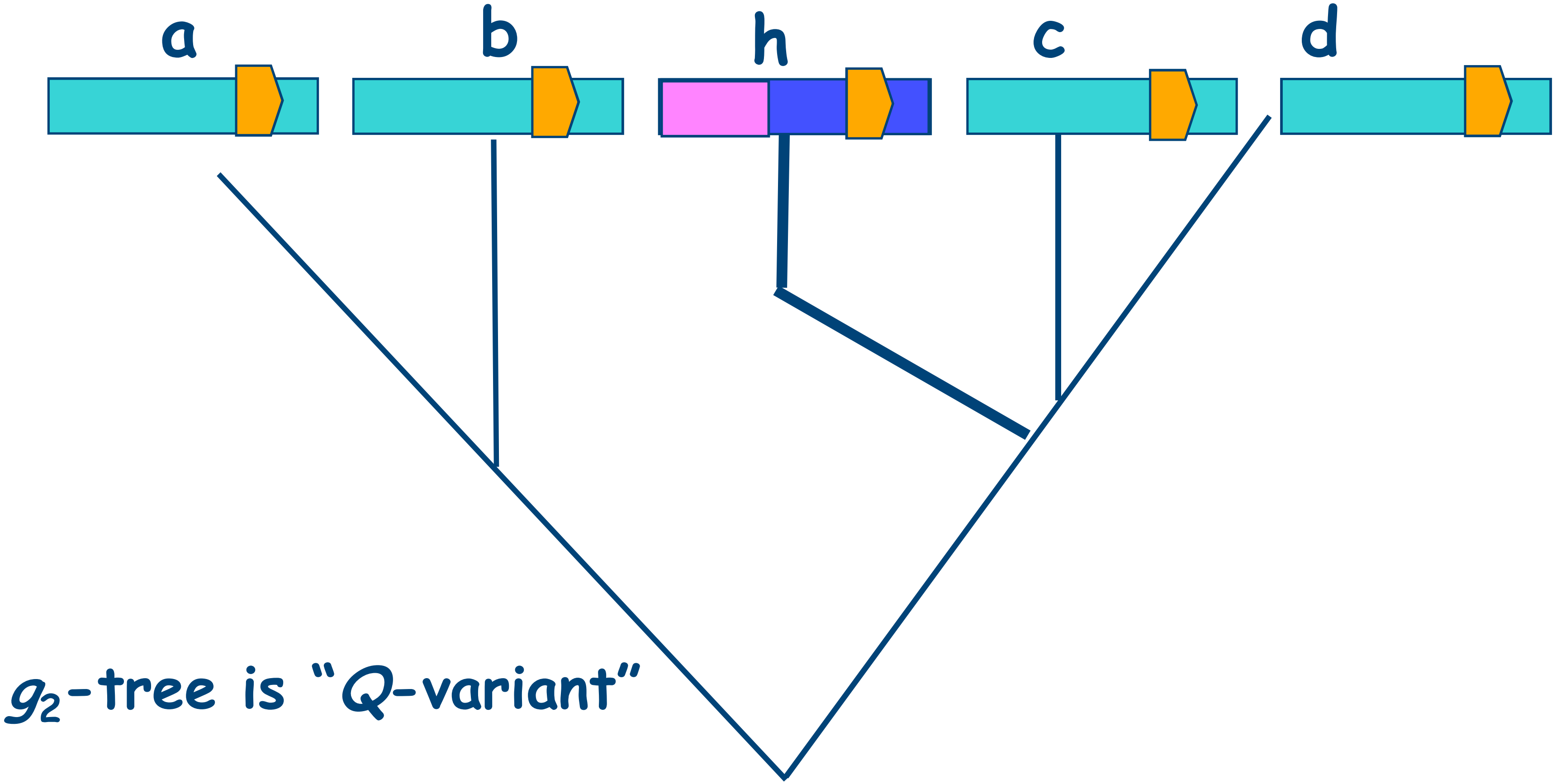
Daniel Huson, 2007

A Simple Model Of Reticulate Evolution



g_2-tree is "Q-variant"

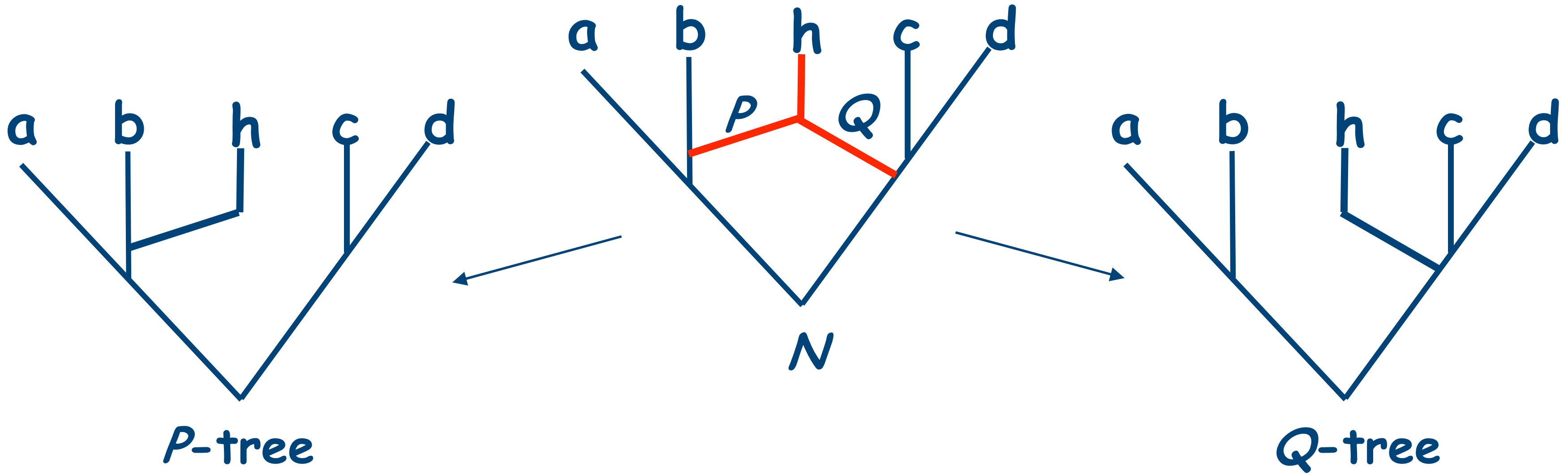
A Simple Model Of Reticulate Evolution



g_2-tree is "*Q-variant*"

Reticulate Networks and Trees

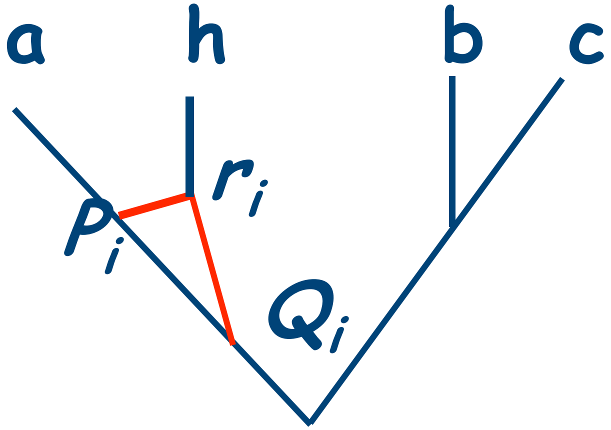
- The evolutionary history associated with any given gene is a tree
- A network N with k reticulations gives rise to 2^k different gene trees



Daniel Huson, 2007

Reticulate Networks and Trees

- Note, however that the two choices P_i and Q_i can lead to the same tree *topology*:



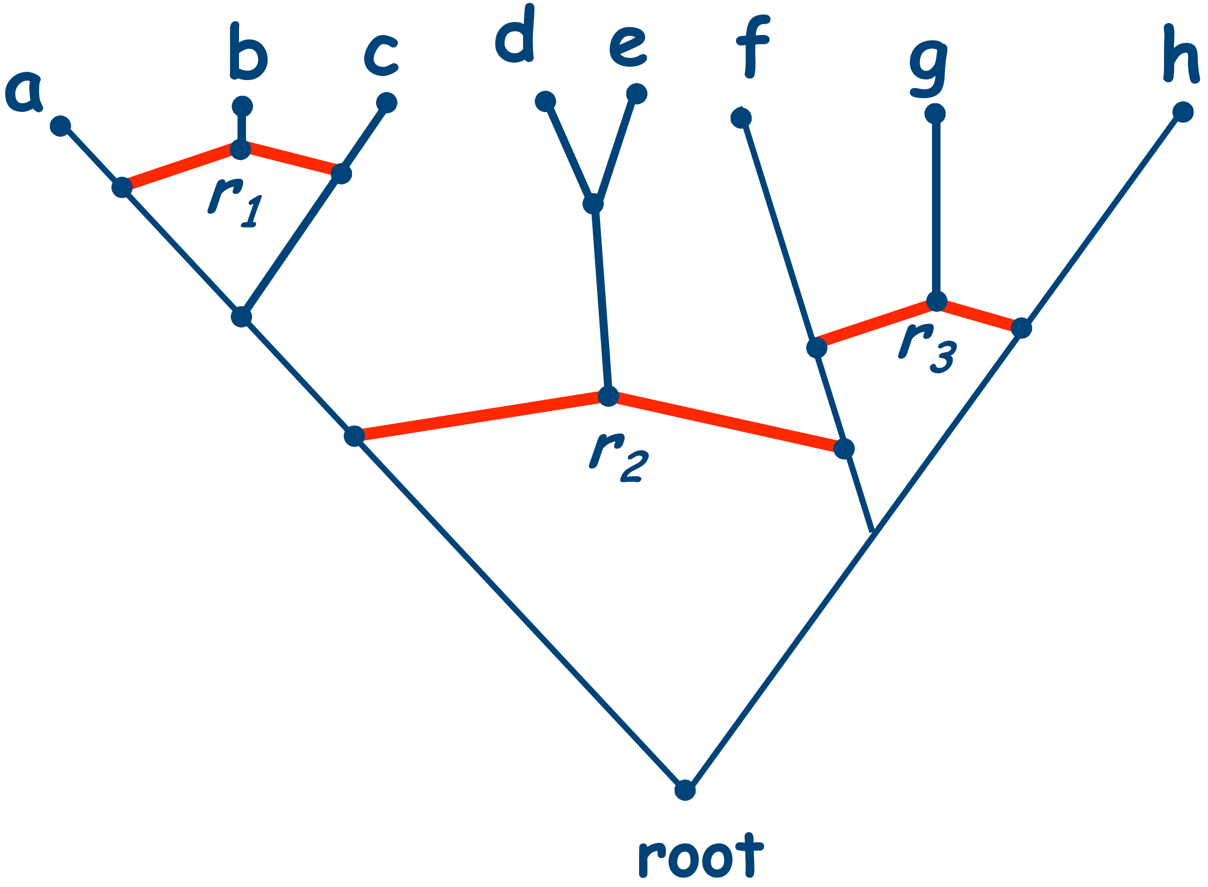
- Here, both induced trees are of the form: ((a,h),(b,c)).

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:

- precisely one node of indegree 0, the *root*,
- all other nodes are *tree* nodes of indegree 1, or *reticulation* nodes of indegree 2,
- every edge is a *tree* edge joining two tree nodes, or a *reticulation* edge from a tree node to a reticulation node, and
- the set of leaves consists of tree nodes and is labeled by *X*.

Rooted Reticulate Network



Reconstruction of Reticulate Networks

- Given a set of trees $\mathcal{T}=\{T_1,\ldots,T_m\}$, want to determine the reticulate network N from which the trees were sampled with $\mathcal{T}$ = T(N).
- 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:

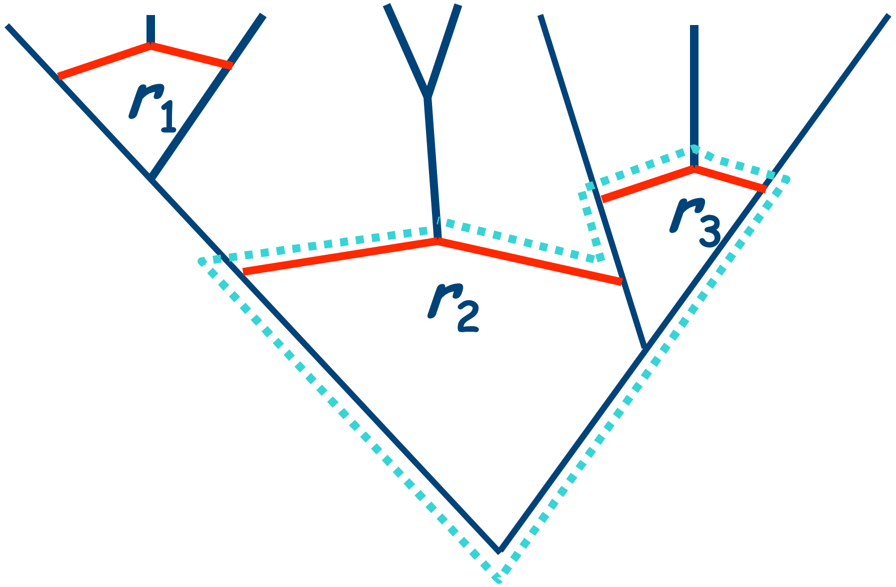
Reconstruction of Reticulate Networks

Most Parsimonious Network Problem: **Given a set of trees $\mathcal{T}$, determine a reticulate network N such that $\mathcal{T} \subseteq$ T(N) and N contains a minimum number of reticulation nodes.**

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

Independent Reticulations

- Two reticulation nodes r_1, r_2 in N are *independent* of each other, if they are not contained in any common simple cycle.



- 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.

Galled Trees

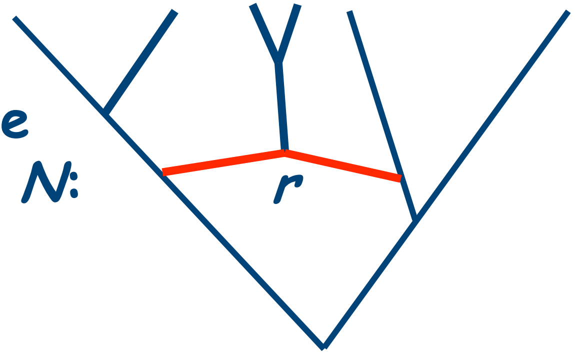
- A reticulation that is independent of all others is also called a *gall*.

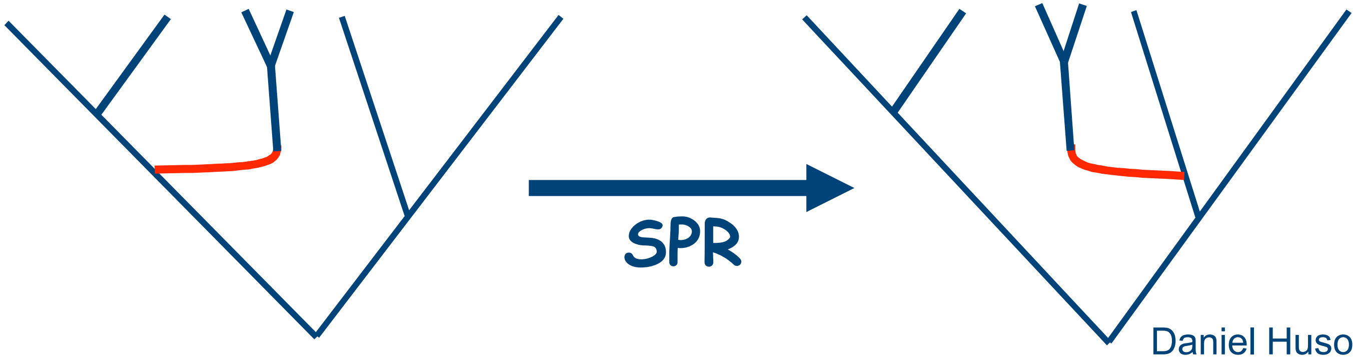
- A network *N* in which all reticulations are galls is also called a *galled tree*.
(Gusfield, 2003-2005)

SPR's & Independent Reticulations

Observation: If N contains only a single reticulation r, then it corresponds to a "sub-tree prune and regraft" operation:

Reticulate network N:





Daniel Huson, 2007

SPR-Based Algorithm

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

- This approach has been generalized to networks with multiple independent reticulations

Challenge

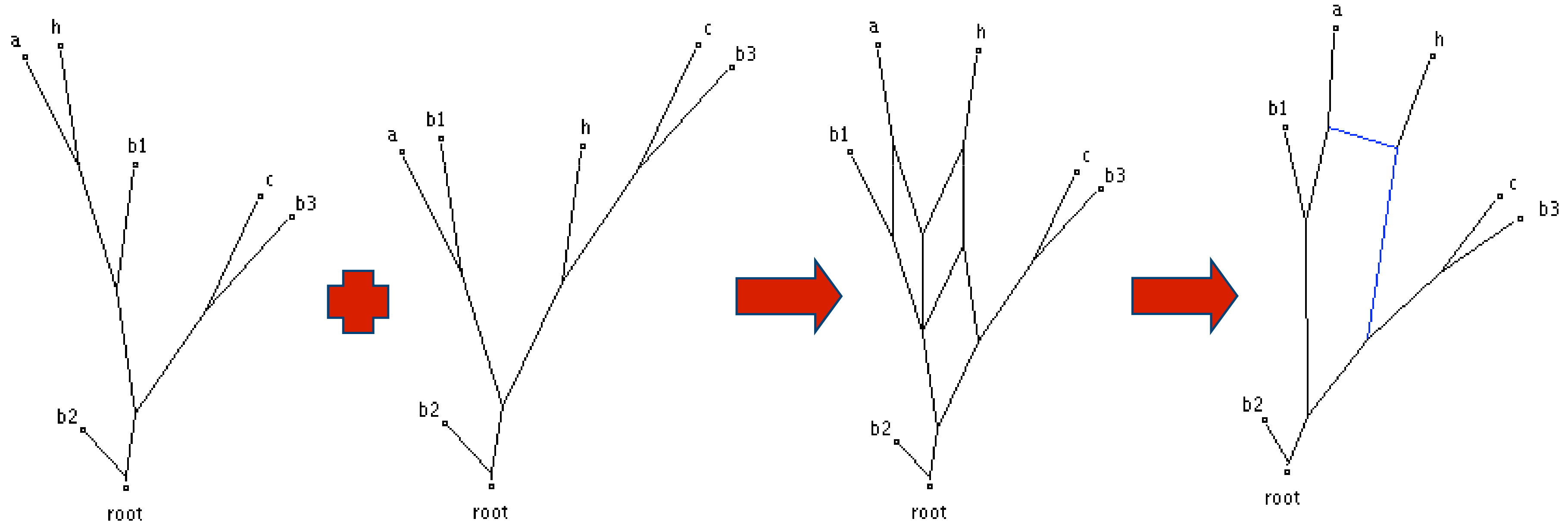
- Unfortunately, on real data, such algorithms will often return „fail".
- Please note: All current approaches aim at solving a combinatorial puzzle:

  Does there exist a network that induces the given set of trees?

  (Below: … that induces the given alignment of binary sequences?)
- A main challenge is to produce useful output in the case of imperfect data.

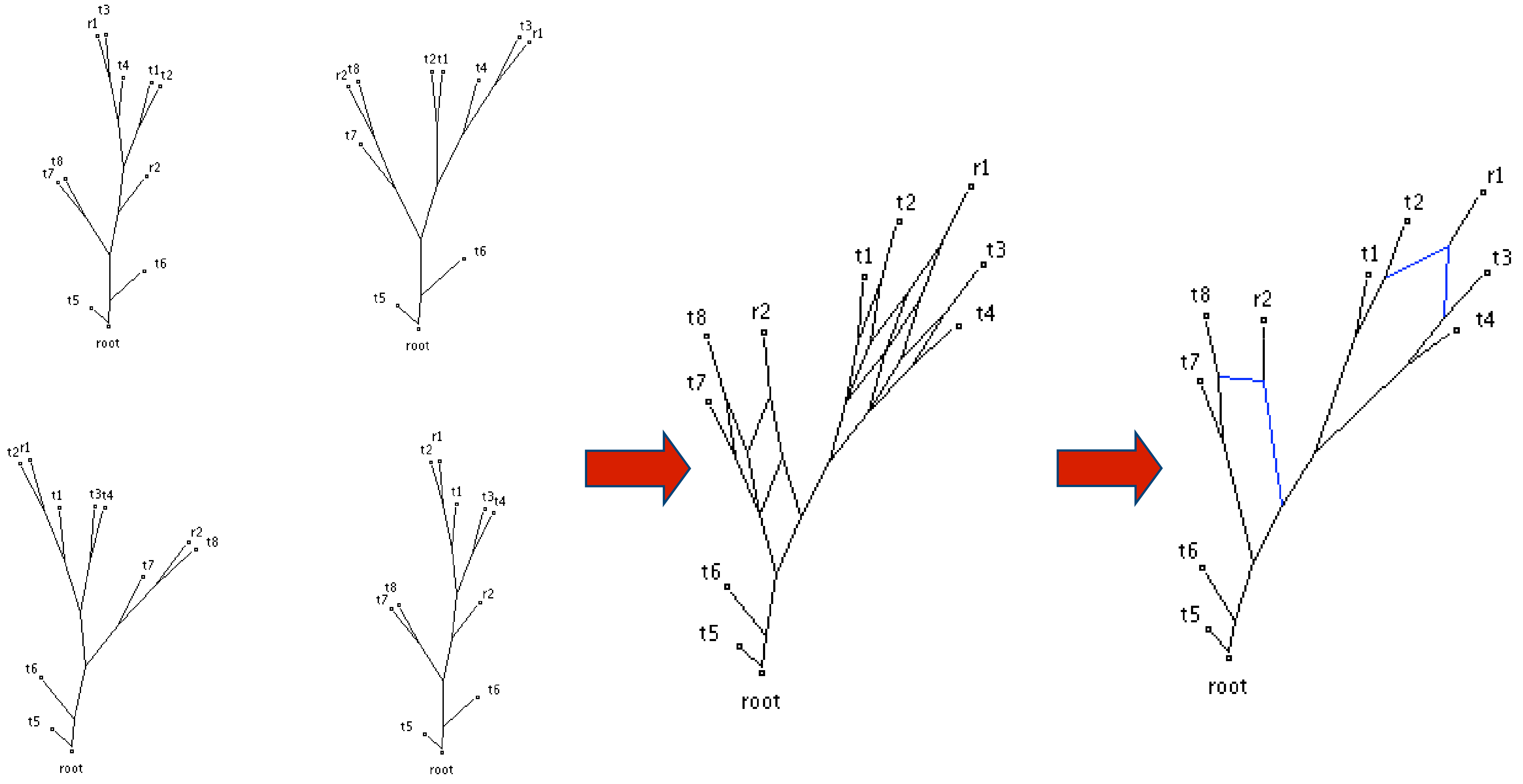
Splits-Based Approach

The splits-based approach:



gene tree1 gene tree2 split network of all splits reticulate network

Multiple Independent Reticulations



Two reticulations ⇒ four different gene trees

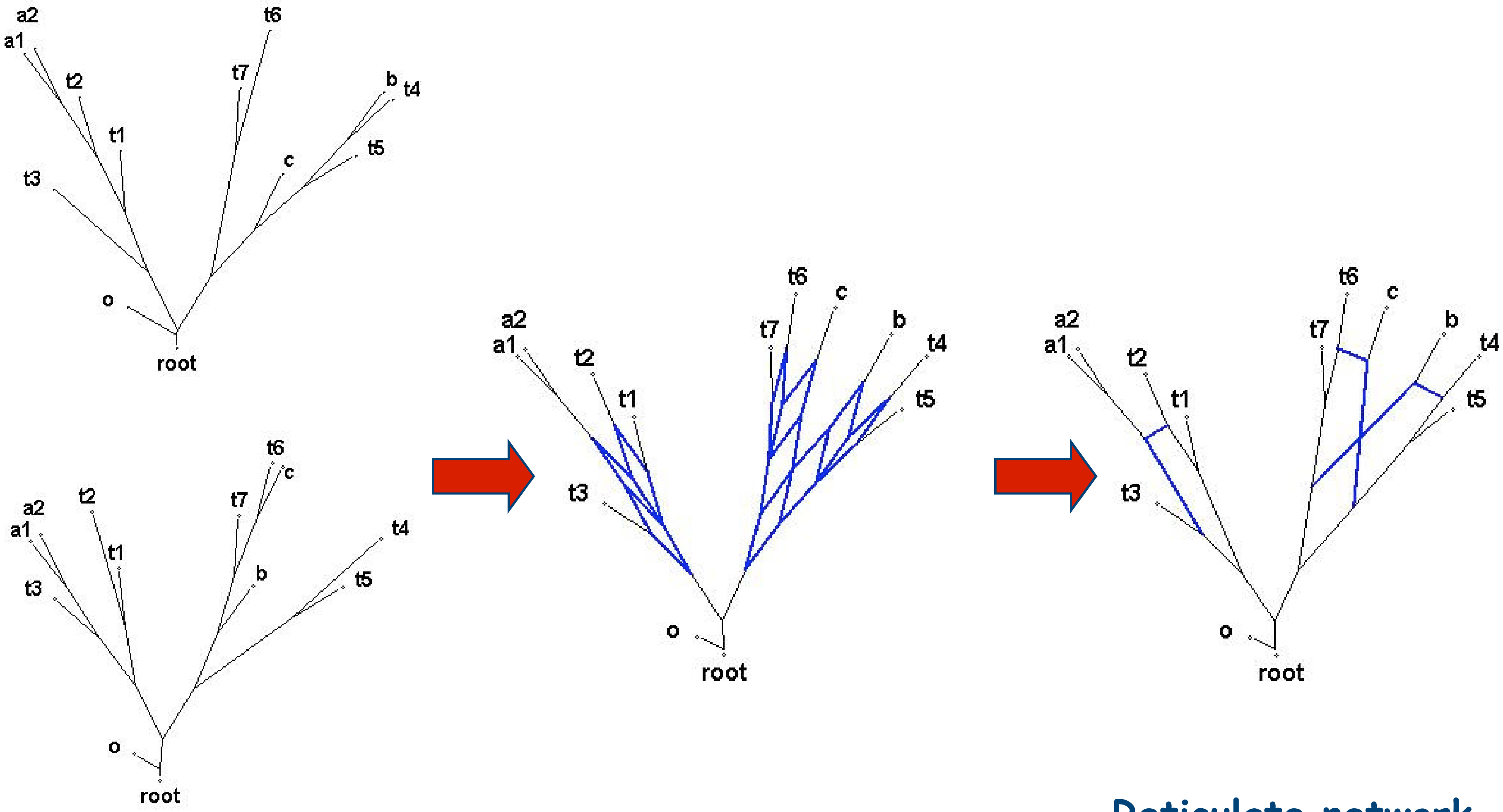
all splits

Reticulate network that induces all input trees

Overlapping Reticulations

- Current splits-based methods can resolve components in which the reticulation cycles overlap along a common tree:

Multiple and Overlapping Reticulations



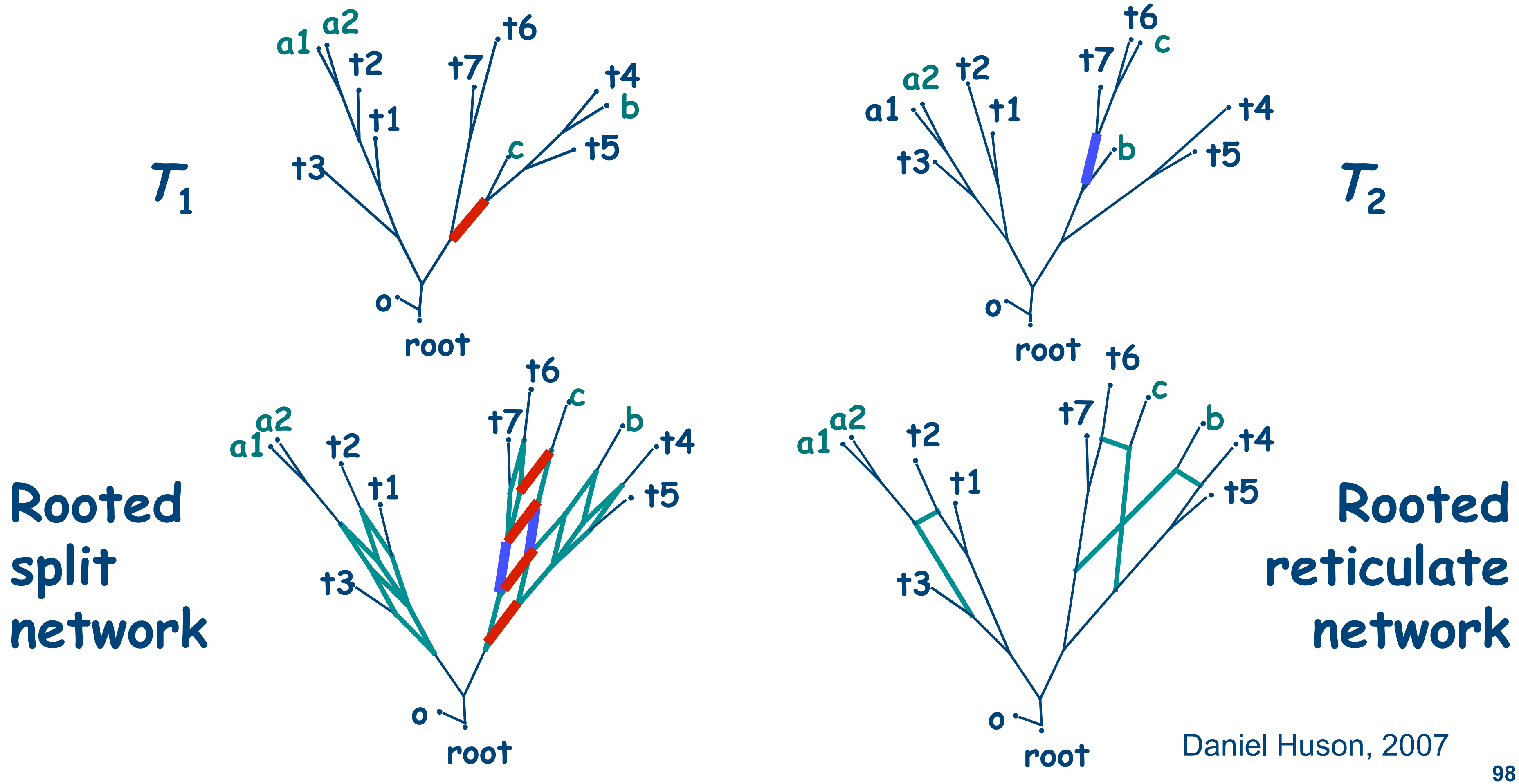
Input trees

all splits

Reticulate network that induces all input trees

Reticulate Networks & Split Networks

- There is a nice relationship between a reticulate network *N* and the network of all splits of trees sampled from *N*:



Decomposition Conjecture

- There exists a one-to-one correspondence between:
    - the connected components of the "incompatibility graph",
    - the "netted regions" of the split network and
    - the "tangles" of dependent reticulations of a "minimal" reticulate network
- Not true in general (Yun Song, unpublished)

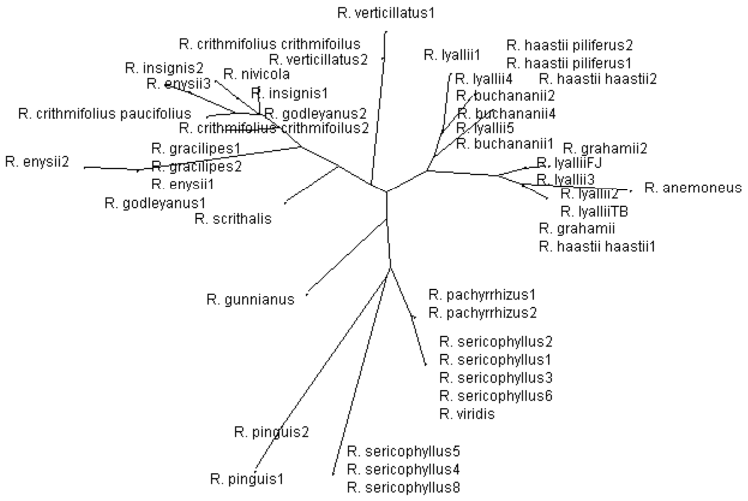
Splits-Based Algorithm

This leads to the following approach:

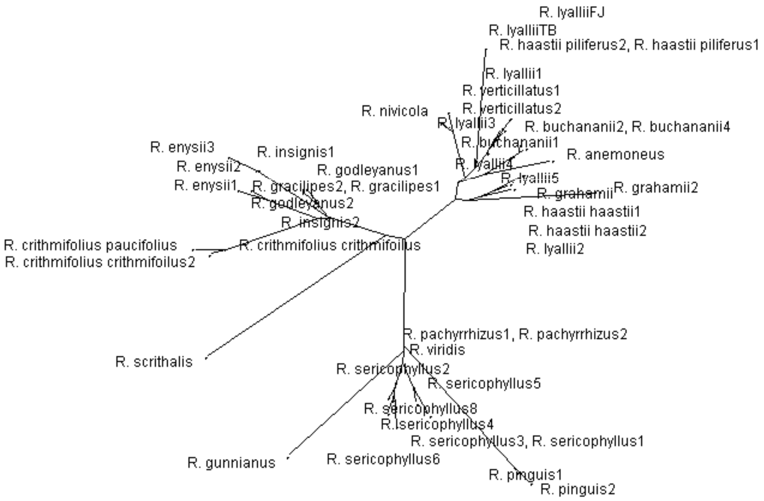
- Determine the set of all input splits
- Determine the connected components of the incompatibility graph or split network
- Analyze each component $\mathcal{C}$ separately:
  - If $\mathcal{C}$ can be explained by a reticulate network N($\mathcal{C}$), then locally replace $\mathcal{C}$ by N($\mathcal{C}$)

Application to Real Data

New Zealand *Ranunculus* (buttercup) species

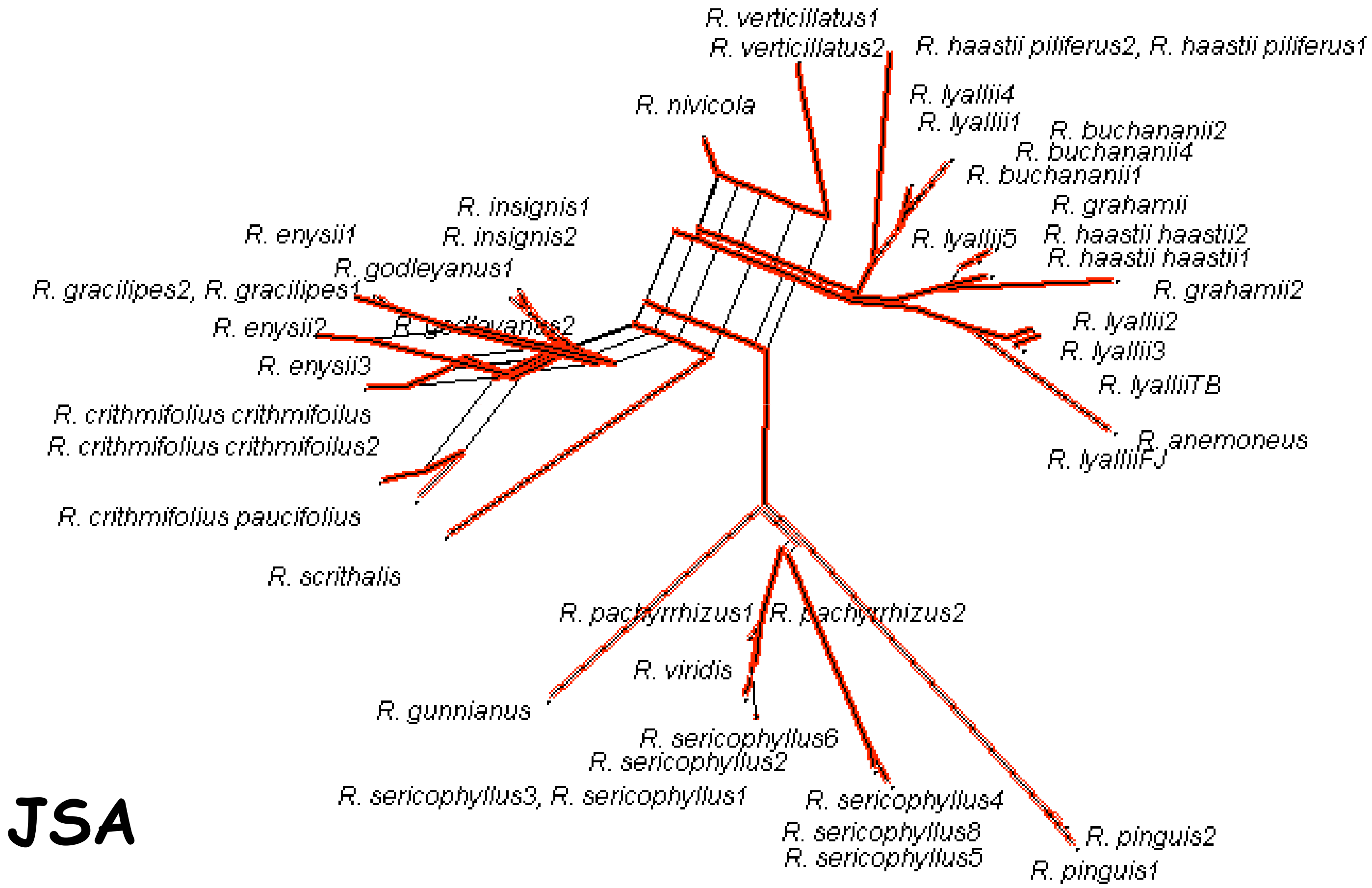


JSA region in chloroplast



ITS region in nuclear genome

Application to Real Data

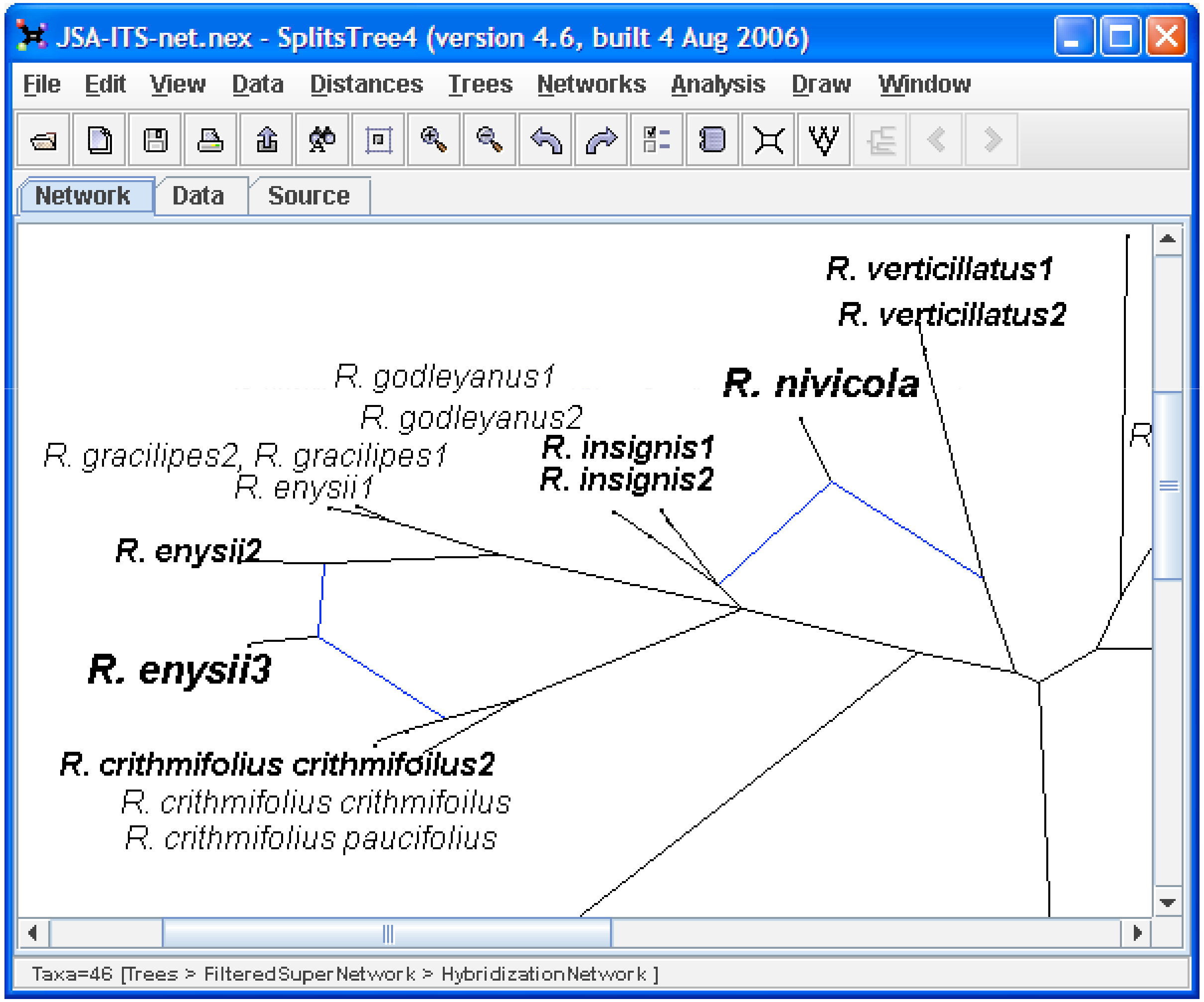


Split network representing both trees simultaneously

Not “explicit”

Application to Real Data

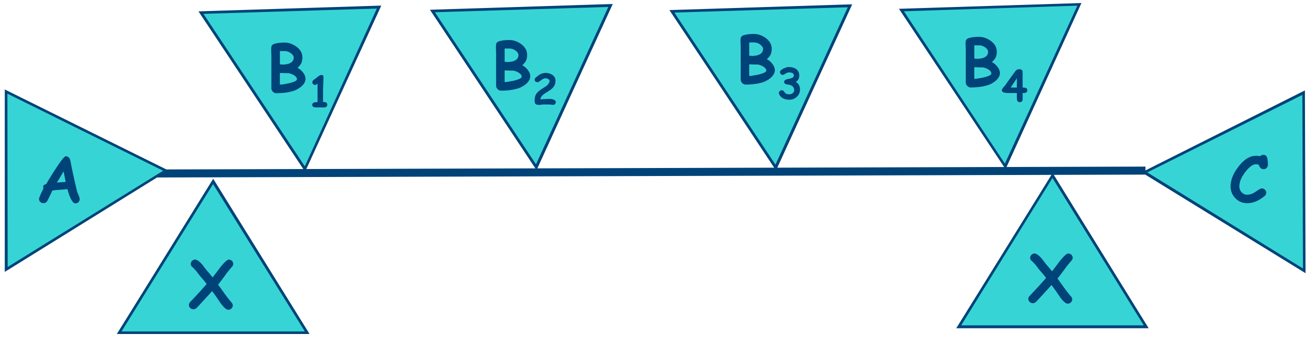
- Split network for ITS & JSA trees
- „Filter splits“
- Hybridization network
- Two cases of hybridization


JSA-ITS-net.nex - SplitsTree4 (version 4.6, built 4 Aug 2006)
File Edit View Data Distances Trees Networks Analysis Draw Window
Network Data Source
R. verticillatus1
R. verticillatus2
R. godleyanus1
R. nivicola
R. godleyanus2
R. gracilipes2, R. gracilipes1
R. insignis1
R. insignis2
R. enysii1
R. enysii2
R. enysii3
R. crithmifolius crithmifoilus2
R. crithmifolius crithmifoilus
R. crithmifolius paucifolius
R
Taxa=46 [Trees > FilteredSuperNetwork > HybridizationNetwork]

„explicit“

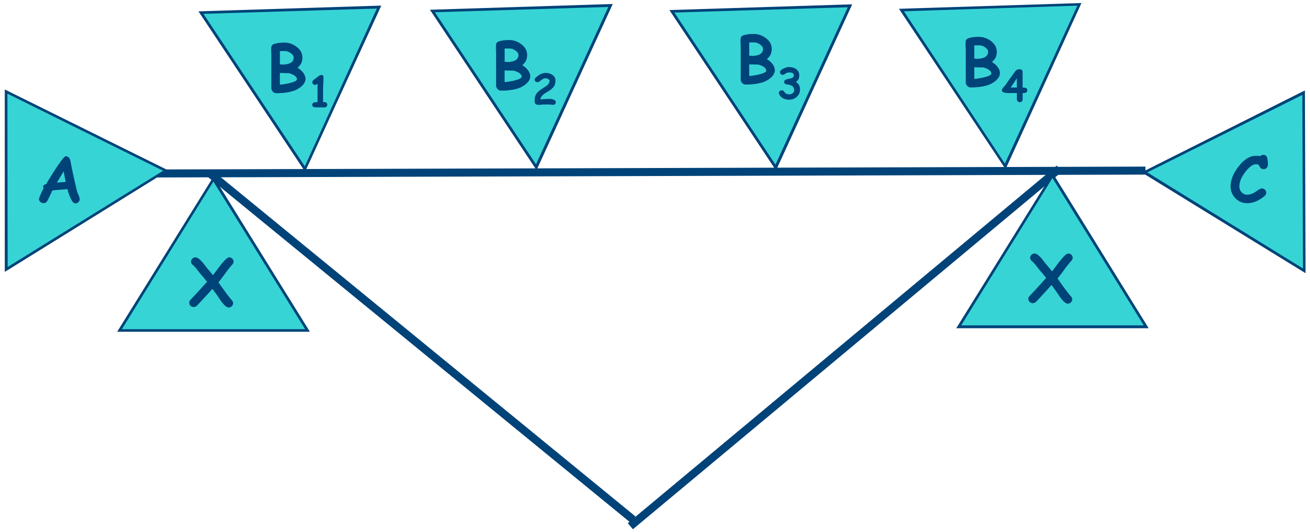
Details of Algorithm for “Galls”

- A reticulation corresponds to a subtree that attaches at two places:

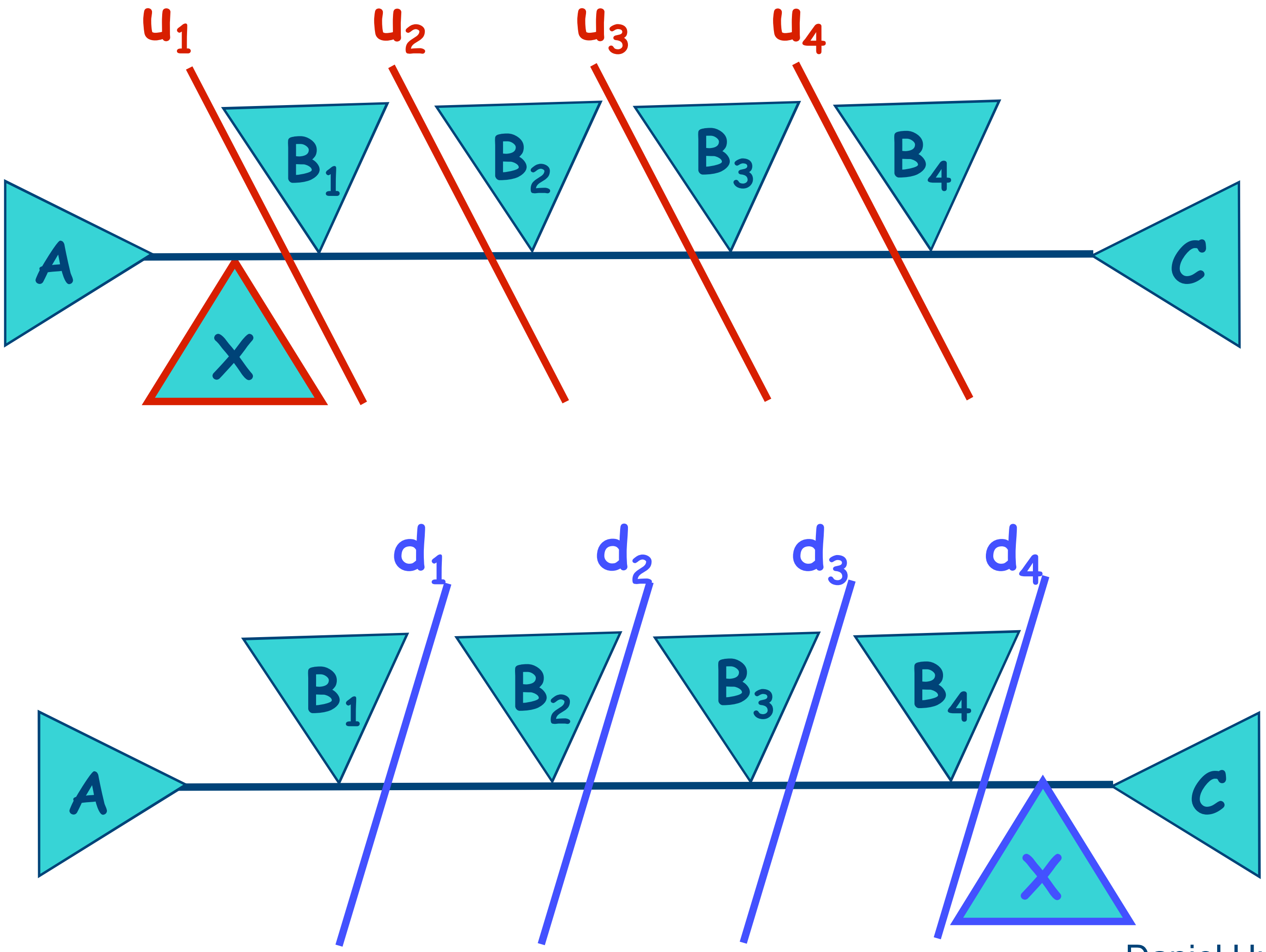


Details of Algorithm for "Galls"

- A reticulation corresponds to a subtree that attaches at two places:



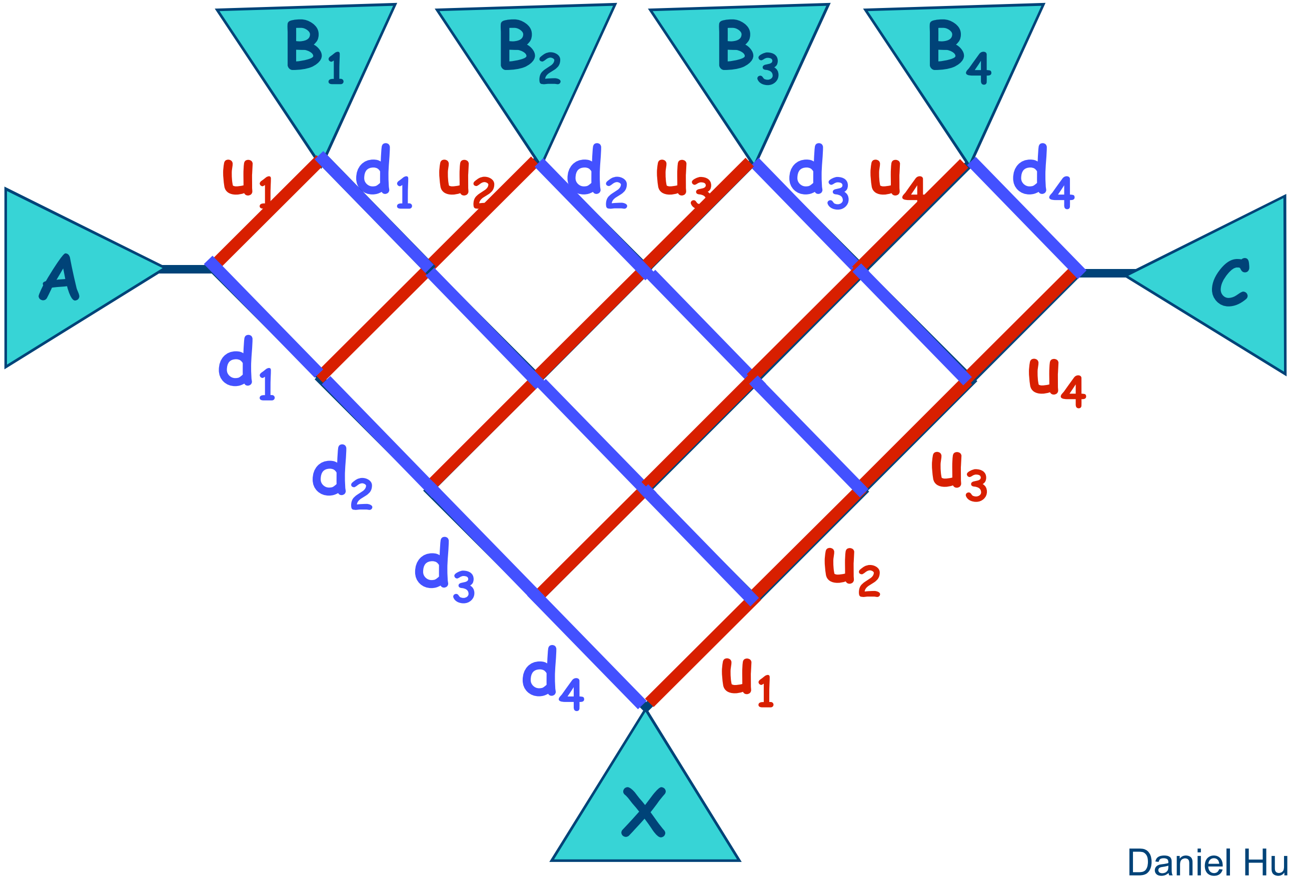
Detecting a Reticulation



Daniel Huson, 2007

Detecting a Reticulation

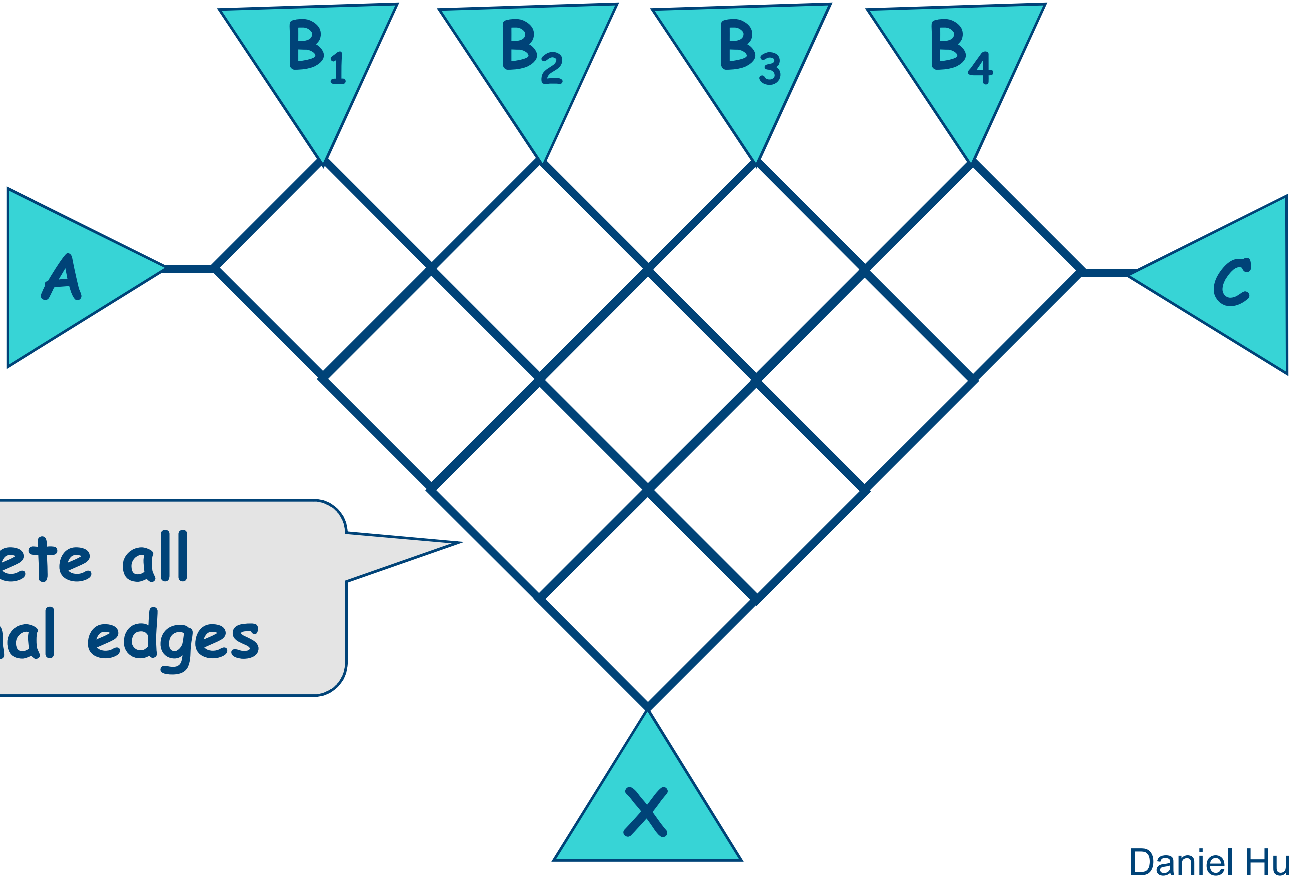
The associated split network…



Daniel Huson, 2007

Split Network to Reticulate Network

The associated split network...

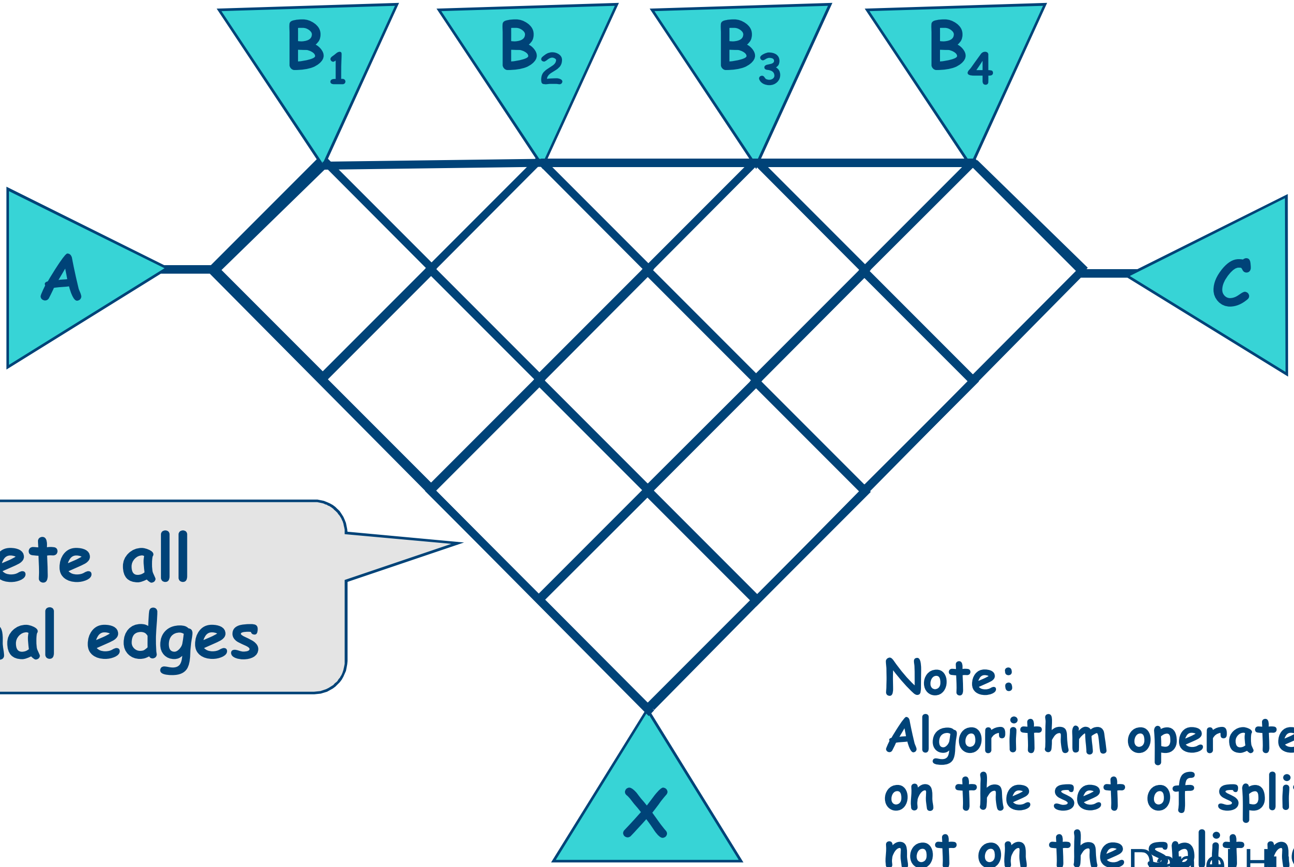


Delete all internal edges

Daniel Huson, 2007

Split Network to Reticulate Network

The associated split network & the reticulate network



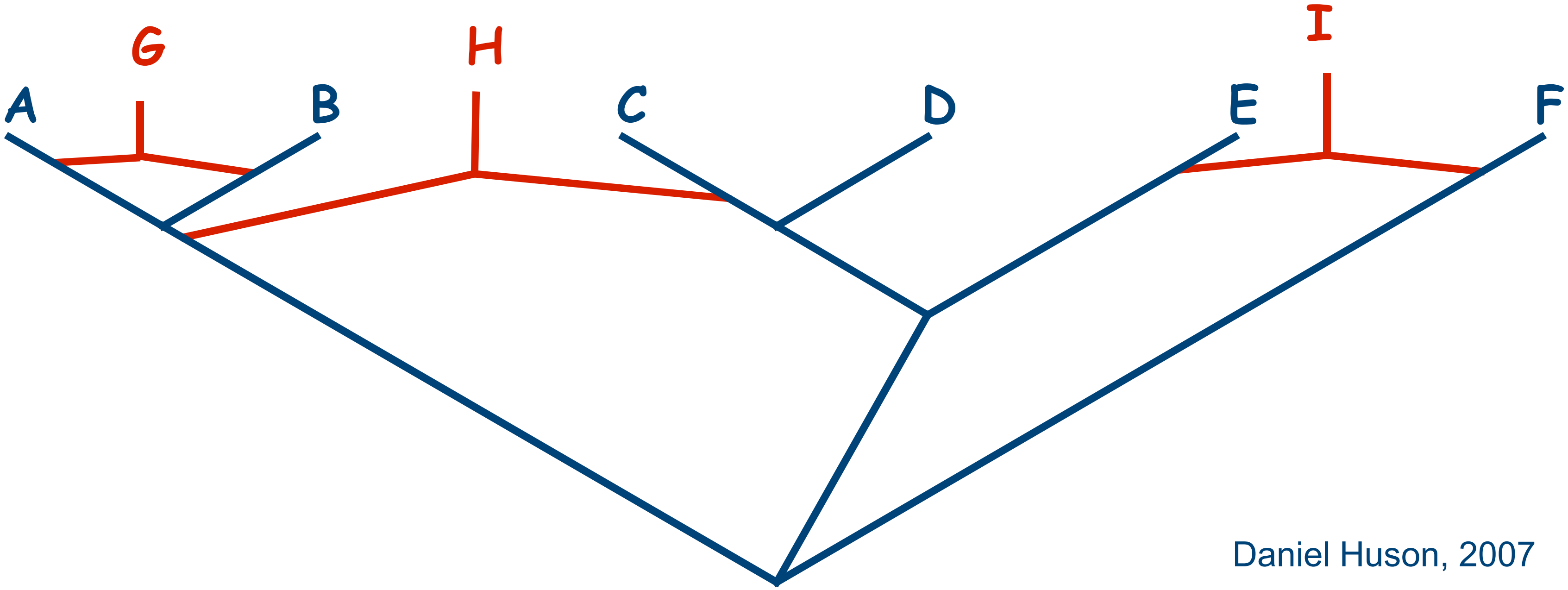
Delete all internal edges

Note:
Algorithm operates directly on the set of splits, not on the split network

Algorithm for Overlapping Reticulations

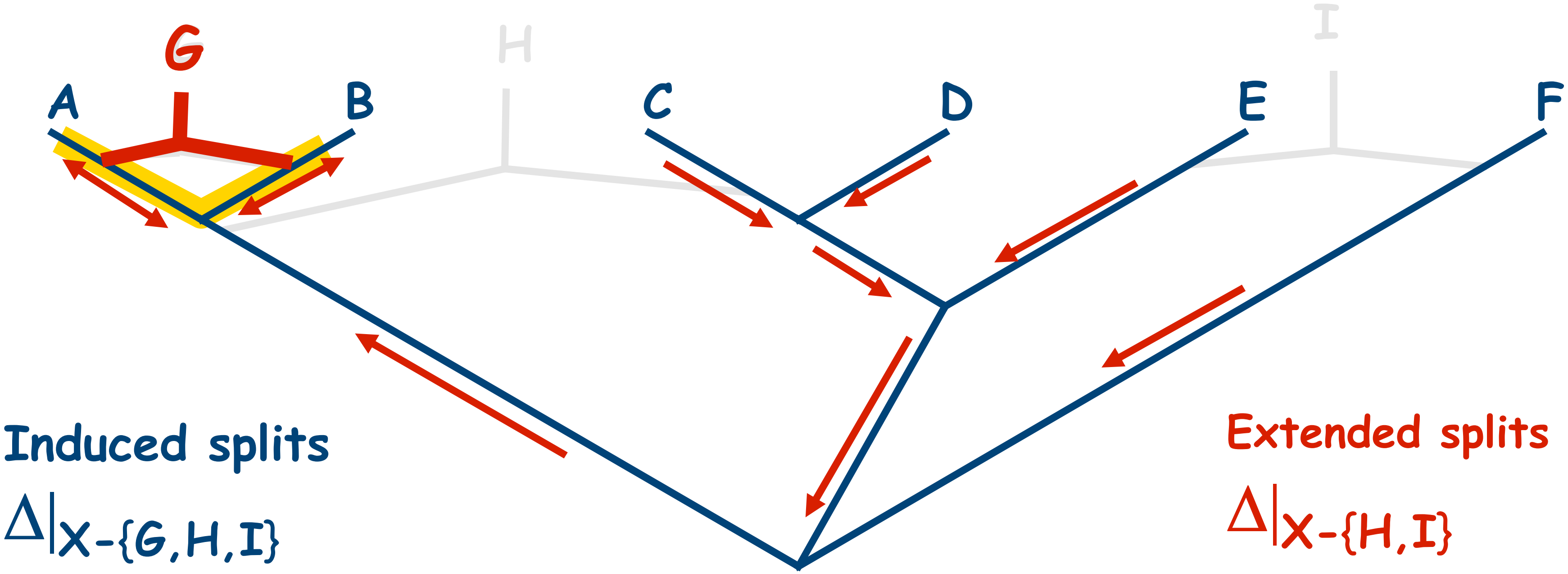
Input:

- Set of splits Δ on X={A,B,…,I} that comes from a network, either via trees or binary sequences, e.g.:



Computing A Reticulate Network

- Assume we know G,H,I are reticulate taxa
- Where to attach G, H, I?



Induced splits

$\Delta|_{X-\{G,H,I\}}$

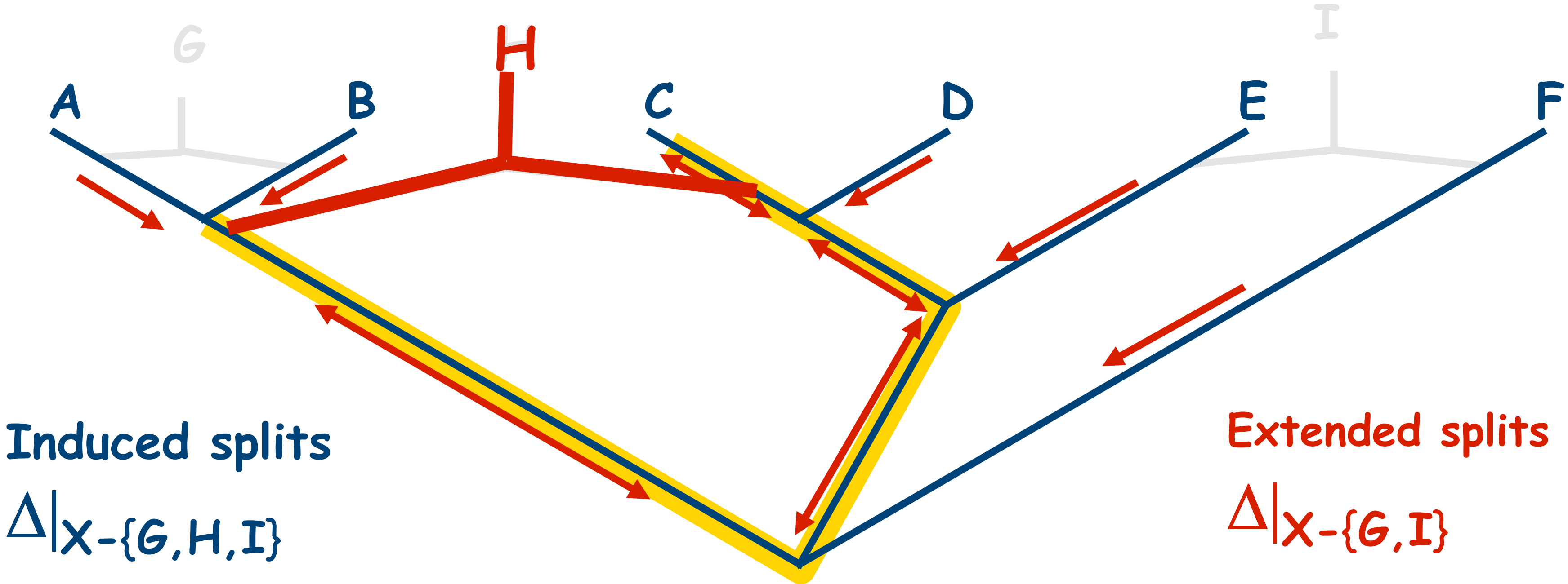
Extended splits

$\Delta|_{X-\{H,I\}}$

- Orient edges to show where splits place G
- Attach G to ends of "target path"

Computing A Reticulate Network

- Assume we know G,H,I are reticulate taxa
- Where to attach G, H, I?



Induced splits

$\Delta|_{X-\{G,H,I\}}$

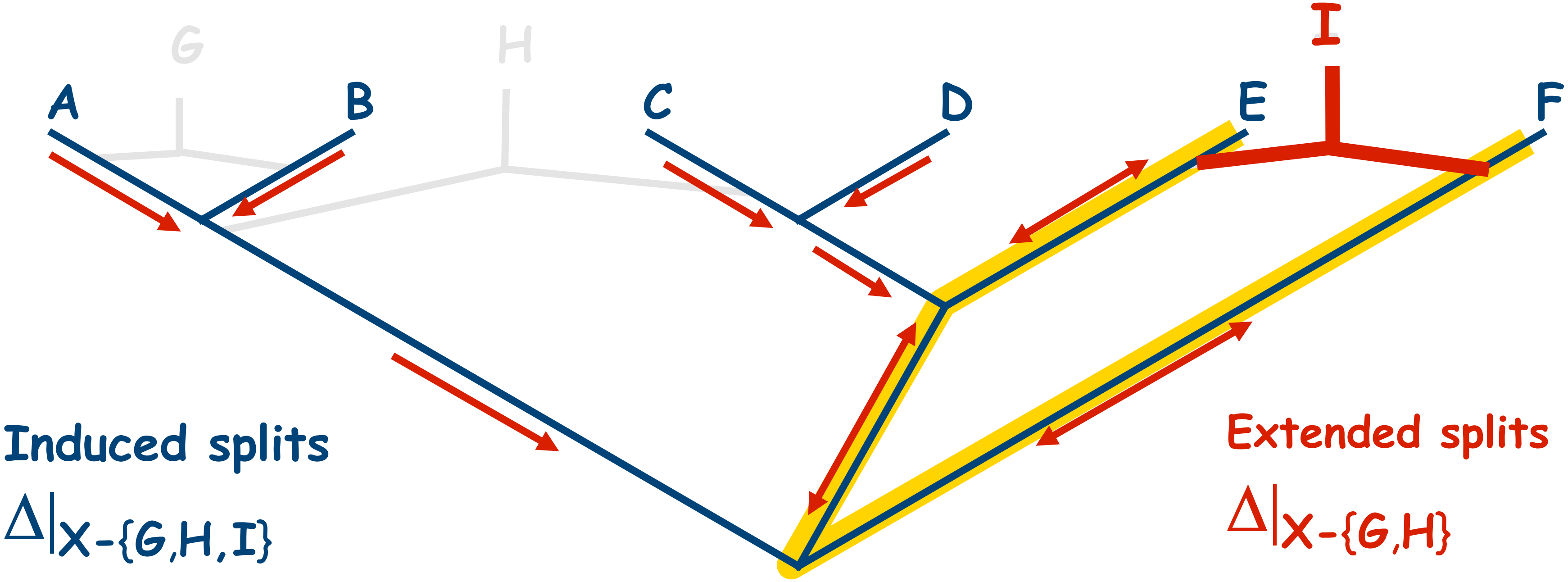
Extended splits

$\Delta|_{X-\{G,I\}}$

- Orient edges to show where splits place H
- Attach H to ends of “target path”

Computing A Reticulate Network

- Assume we know G,H,I are reticulate taxa
- Where to attach G, H, I?



Induced splits

$\Delta|_{X-\{G,H,I\}}$

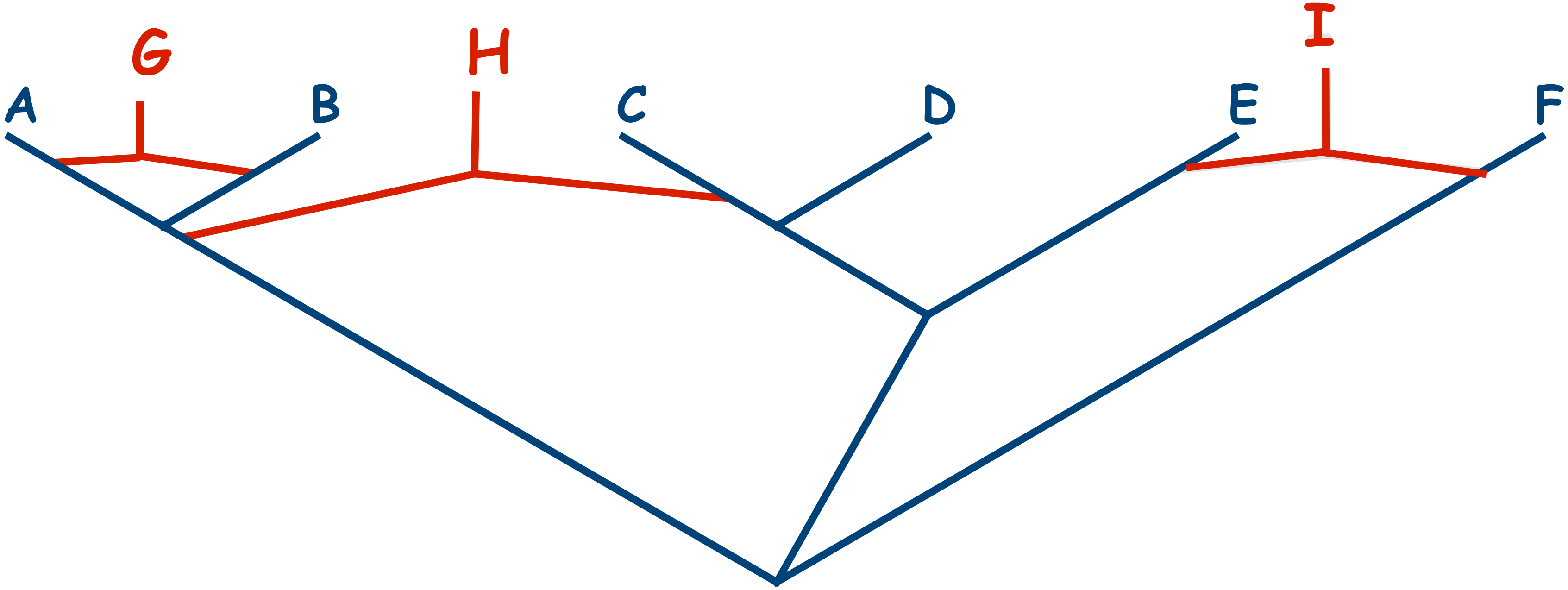
Extended splits

$\Delta|_{X-\{G,H\}}$

- Orient edges to show where splits place I
- Attach I to ends of "target path"

Computing A Reticulate Network

- Assume we know G,H,I are reticulate taxa
- Where to attach G, H, I?



- If $\Delta \subseteq \Sigma(N)$, then return N

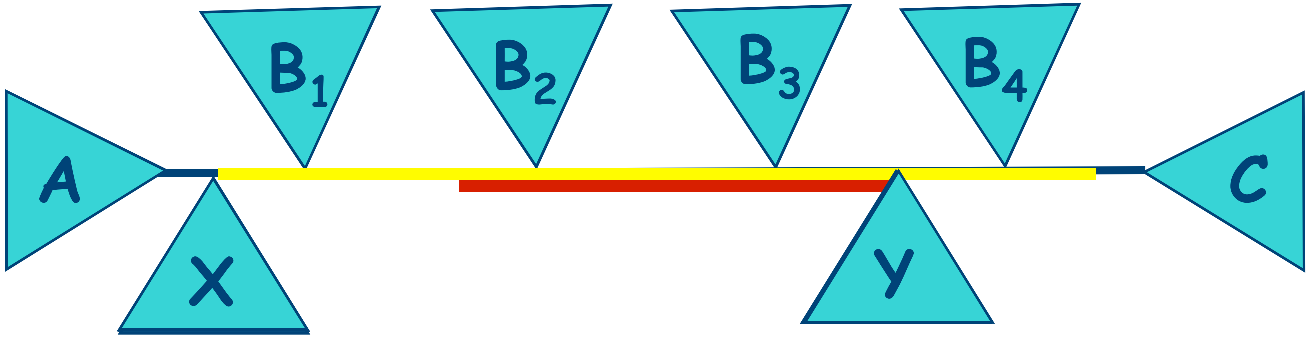
Algorithm For Overlapping Reticulations

Input: Set of splits Δ on X, parameter k

- In increasing order of size $\leq$ k:
- Consider a set of taxa $R \subset X$
- If $\Delta|_{X-R}$ is compatible:
- Attempt to attach each $r \in R$ to $T(\Delta|_{X-R})$
- If successful, construct network N
- If $\Delta \subseteq \Sigma(N)$, return N
- Return fail

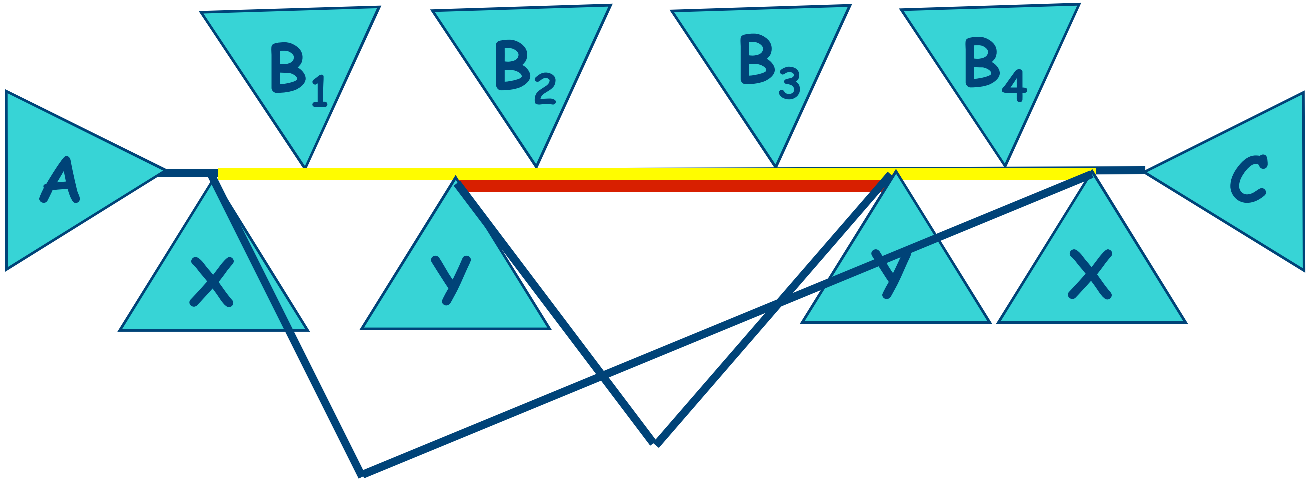
Details of Splits-Based Approach

- Multiple reticulations can overlap along a path:



Details of Splits-Based Approach

- Multiple reticulations can overlap along a path:



Software

- SplitsTree4 contains an implementation of the splits-based algorithm that can handle overlapping reticulations.
- In a program SPNet is described for galled trees, but it is not available for download.

Part IV

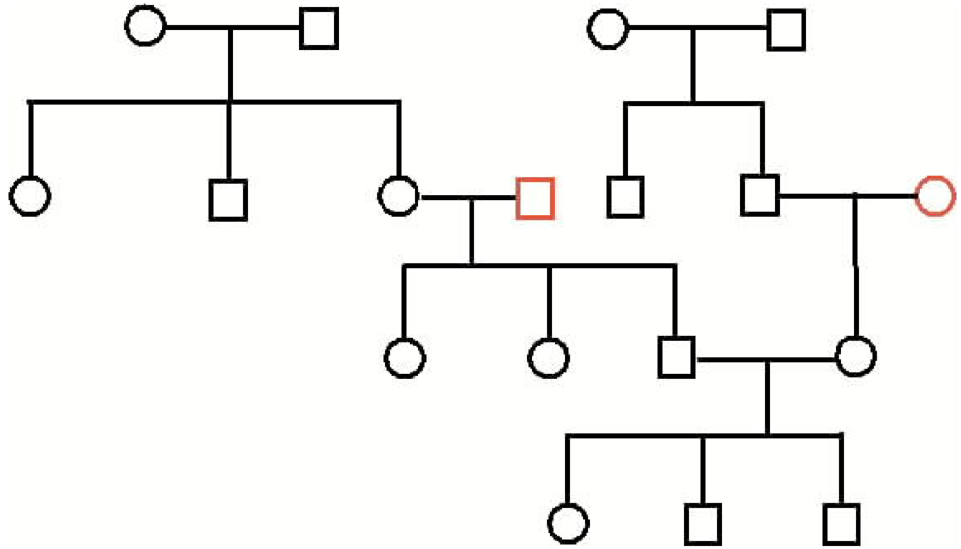
1. Phylogenetic trees
2. Consensus networks and super networks
3. Hybridization and reticulate networks
4. **Recombination networks**
5. Other

Overview

- Consider an alignment of binary sequences that have evolved under a model of mutation-, speciation- and recombination events
- We will look at the problem of reconstructing the underlying reticulate network
- Software

Recombination

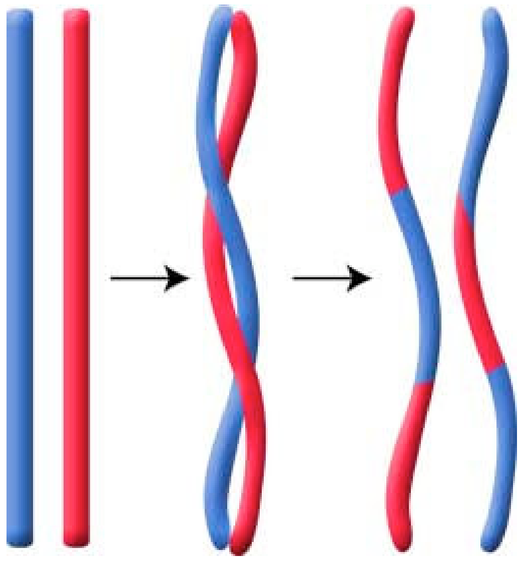
- (Sexual) recombination is studied in population genetics and there *ancestor recombination graphs* (ARGs) are used for statistical purposes.



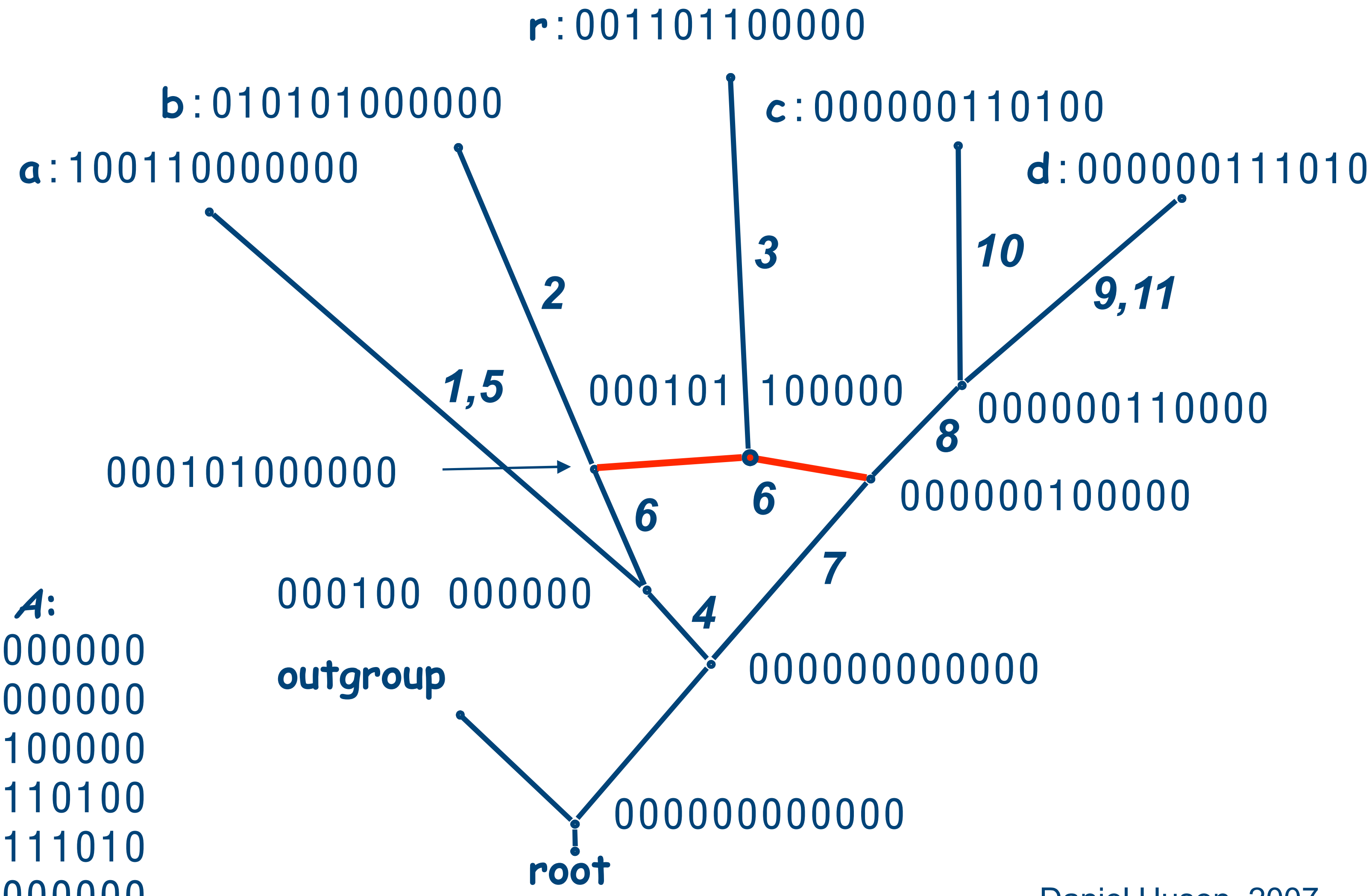
Chromosomal Recombination

- We will study the *combinatorial* aspects of chromosomal (meiotic) recombination and thus consider *recombination networks* rather than ARGs.

- Simplifying assumptions:
  - all sequences have a common ancestor, and
  - any position can mutate at most once.



Example of a Recombination Network



Alignment A:
a:100110000000
b:010101000000
r:001101100000
c:000000110100
d:000000111010
o:000000000000

Daniel Huson, 2007

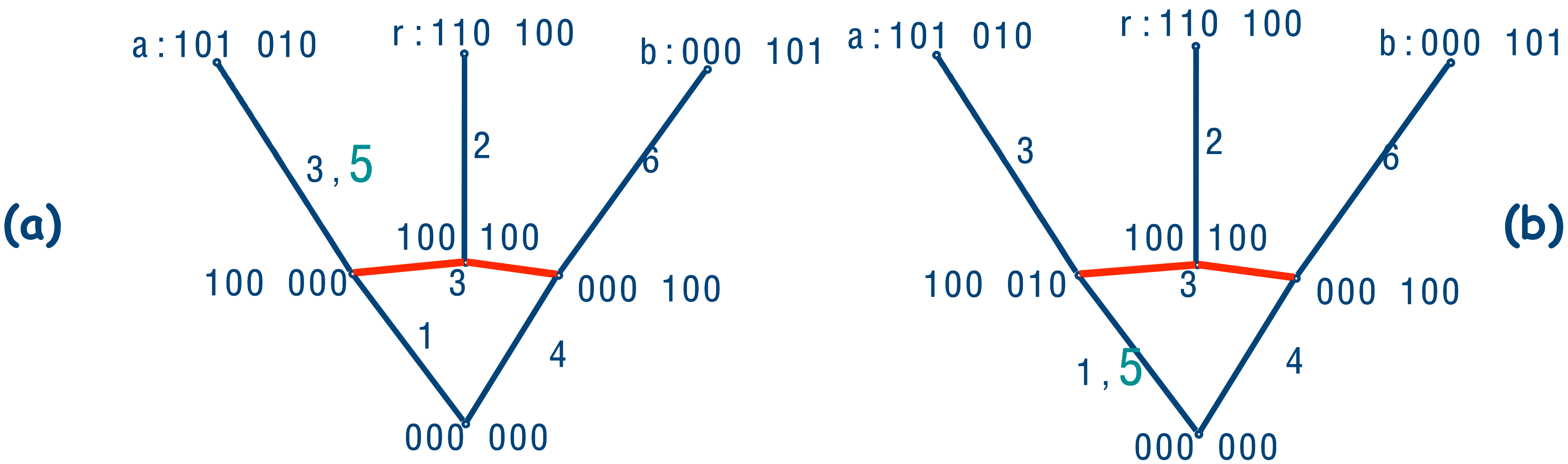
Recombination Network

For an alignment *A* of binary sequences of length *n*, a *recombination network R* is a reticulate network *N*, together with:

- a labeling of all nodes by binary sequences of length *n*, such that the leaves of *R* are labeled by *A*,
- a labeling of each tree edge *e* by the positions that mutate along *e*, and
- a labeling of each reticulation node *r* determining the recombination at *r*.

Non-Uniqueness of Mutations

- The placement of mutations on edges is not uniquely defined. Here, the mutation at position 5 can happen along two different edges:



- Current algorithms place such ambiguous mutations *outside* of the reticulation cycle, as in (a).

Recombination Network

Tree-based approach for computing galled trees:

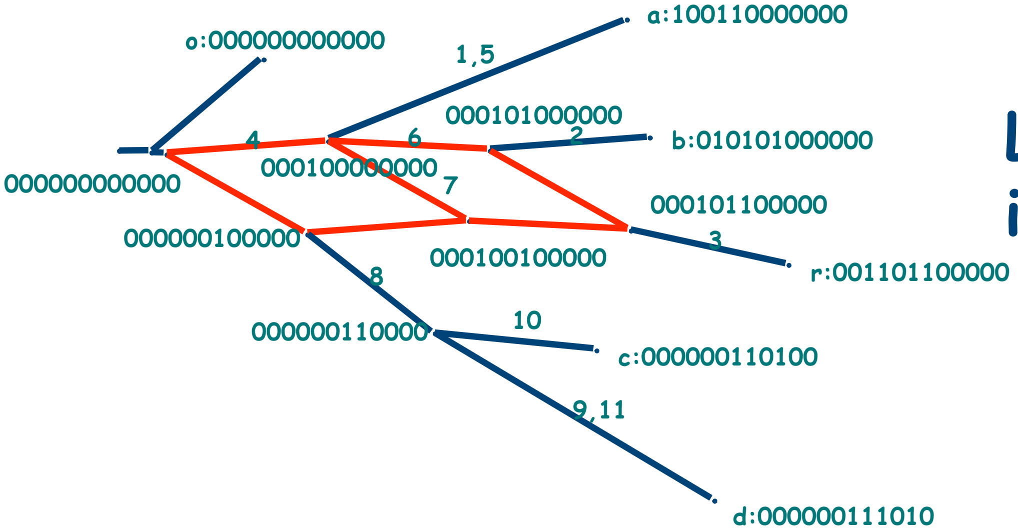
- Determine the components of the incompatibility graph
- For each component:
  - Determine restricted dataset
  - Determine whether removing one taxon produces a perfect phylogeny
  - If so, arrange taxa in gall
  - Return description of network

Recombination Network

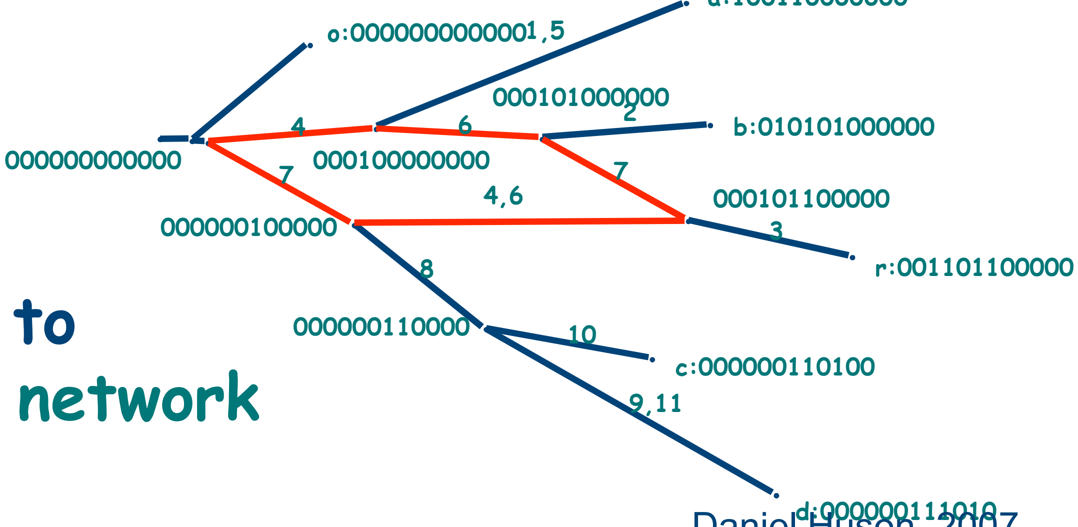
Splits-based approach for computing overlapping networks:

- Determine a reticulate network as described earlier.
- Compute the labeling of nodes and edges.

Computing a Labelling



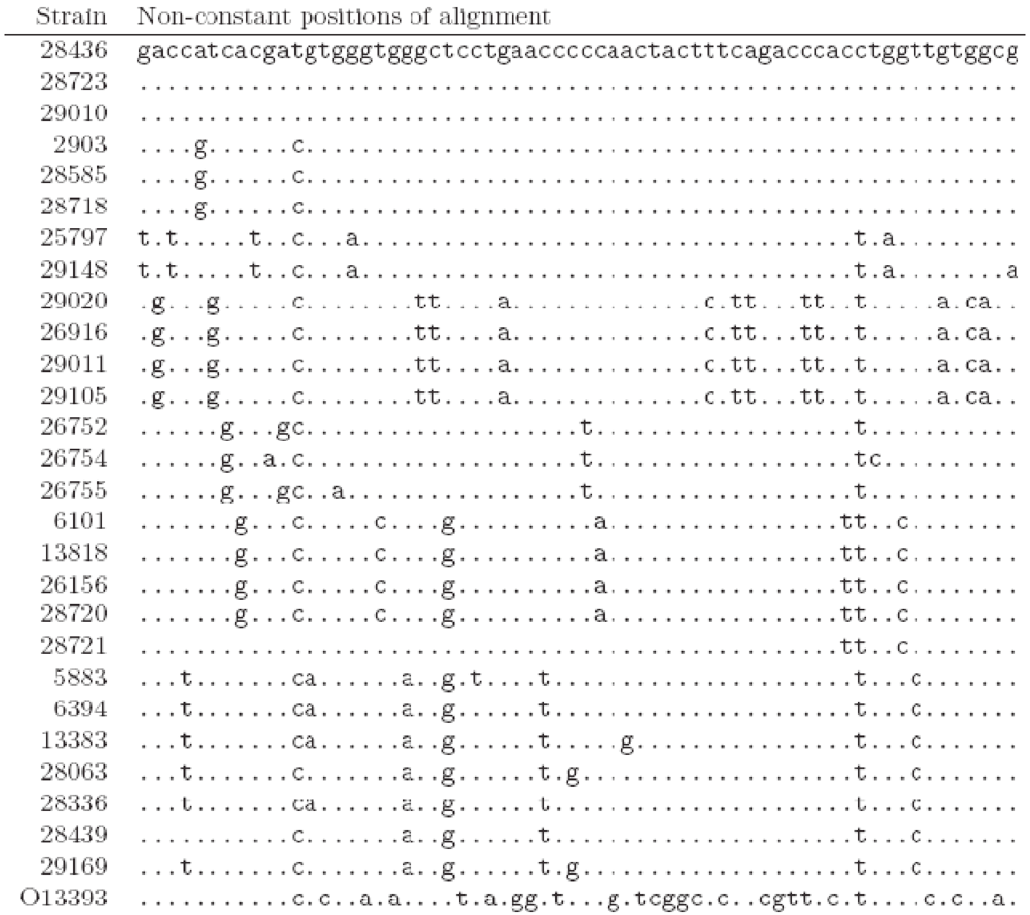
Labelling of split network is easy to compute



Copy labelling to recombination network

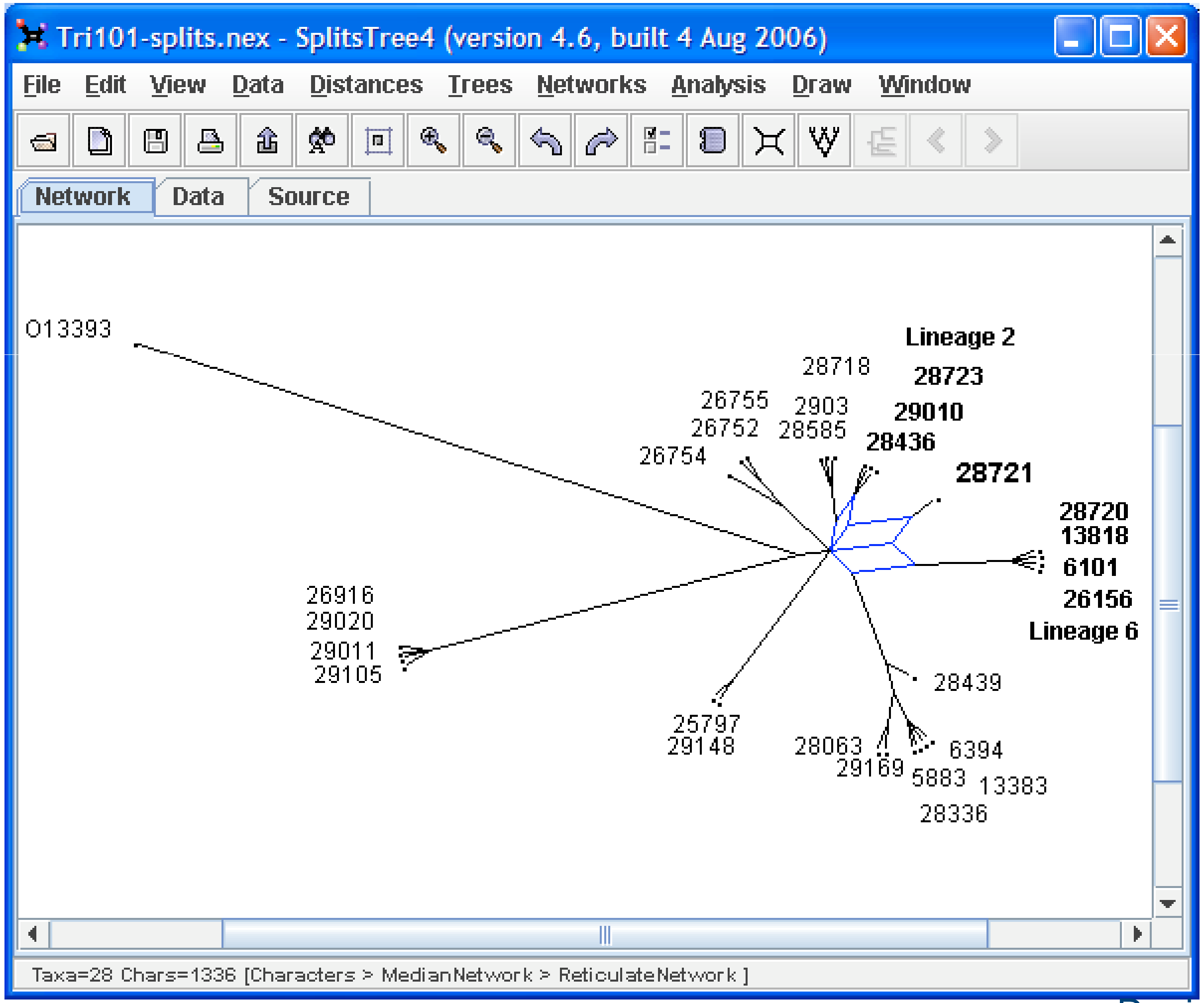
Example 1, Data

- *Fungus fusarium*, 37 strains reported in (K.O'Donnell *et al.*, 2000)
- Locus TRI101 known to undergone intragenic recombination



Strain	Non-constant positions of alignment
28436	gaccatcacgatgtgggtgggctcctgaacccccaactactttcagacccacctggttgtggcg
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..
26752	......g...gc....................t...................t...........
26754	......g..a.c....................t...................tc..........
26755	......g...gc..a.................t...................t...........
6101	.......g...c.....c....g..........a.................tt..c........
13818	.......g...c.....c....g..........a.................tt..c........
26156	.......g...c.....c....g..........a.................tt..c........
28720	.......g...c.....c....g..........a.................tt..c........
28721	....tt..c........
5883	...t.......ca......a..g.t....t......................t...c.......
6394	...t.......ca......a..g......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.......
28439	...........c.......a..g......t......................t...c.......
29169	...t.......c.......a..g......t.g....................t...c.......
O13393	...........c.c..a.a....t.a.gg.t...g.tcggc.c..cgtt.c.t....c.c..a.

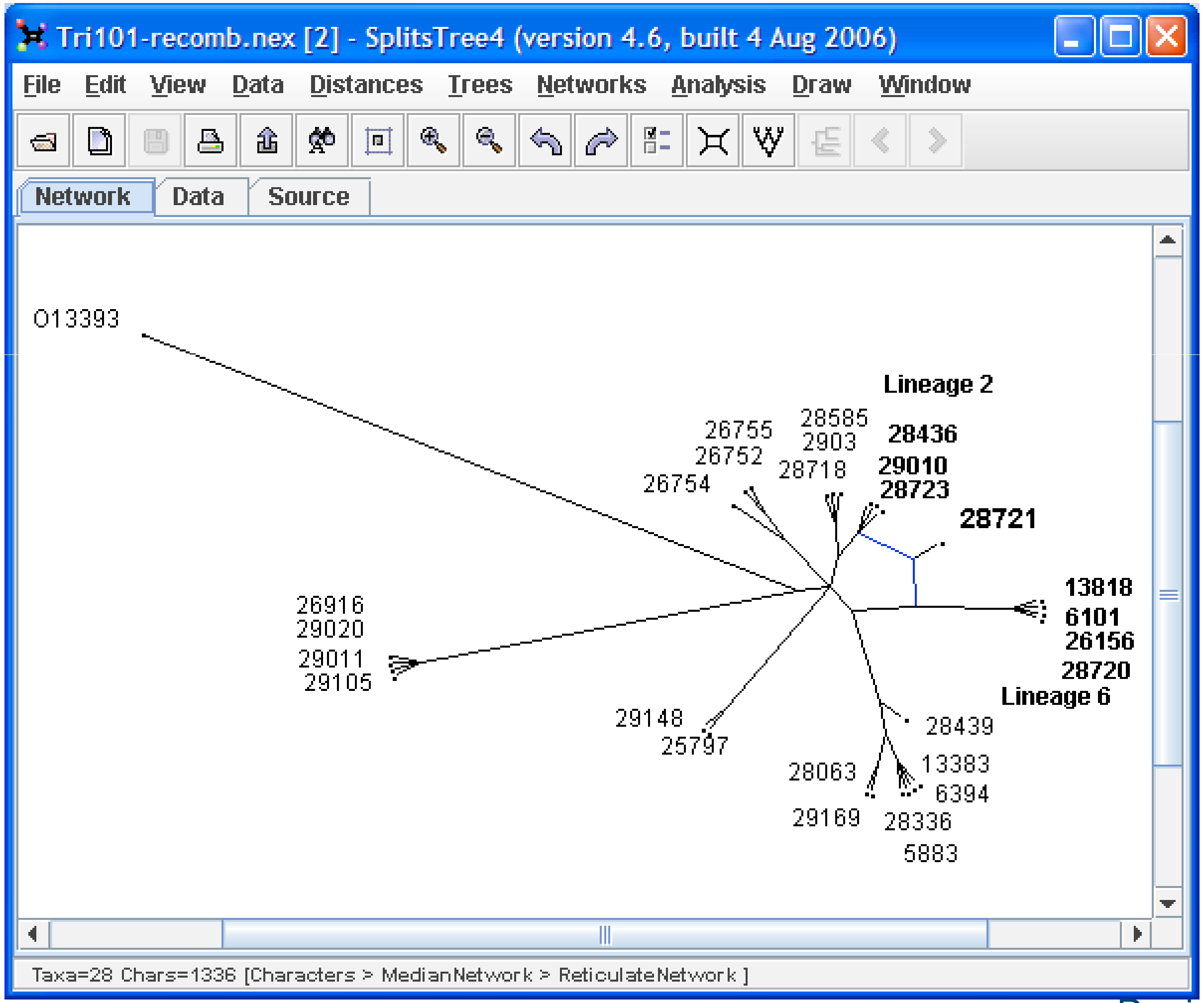
Example 1, Split Network



Taxa=28 Chars=1336 [Characters > MedianNetwork > ReticulateNetwork]

Implicit network

Example 1, Recombination Network



Taxa=28 Chars=1336 [Characters > MedianNetwork > ReticulateNetwork]

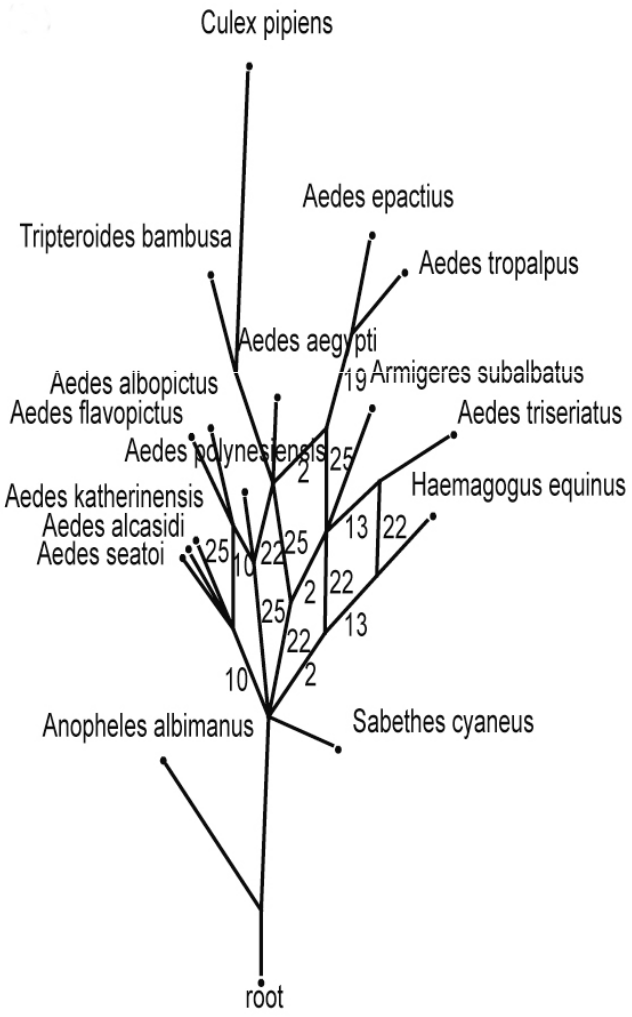
Explicit network

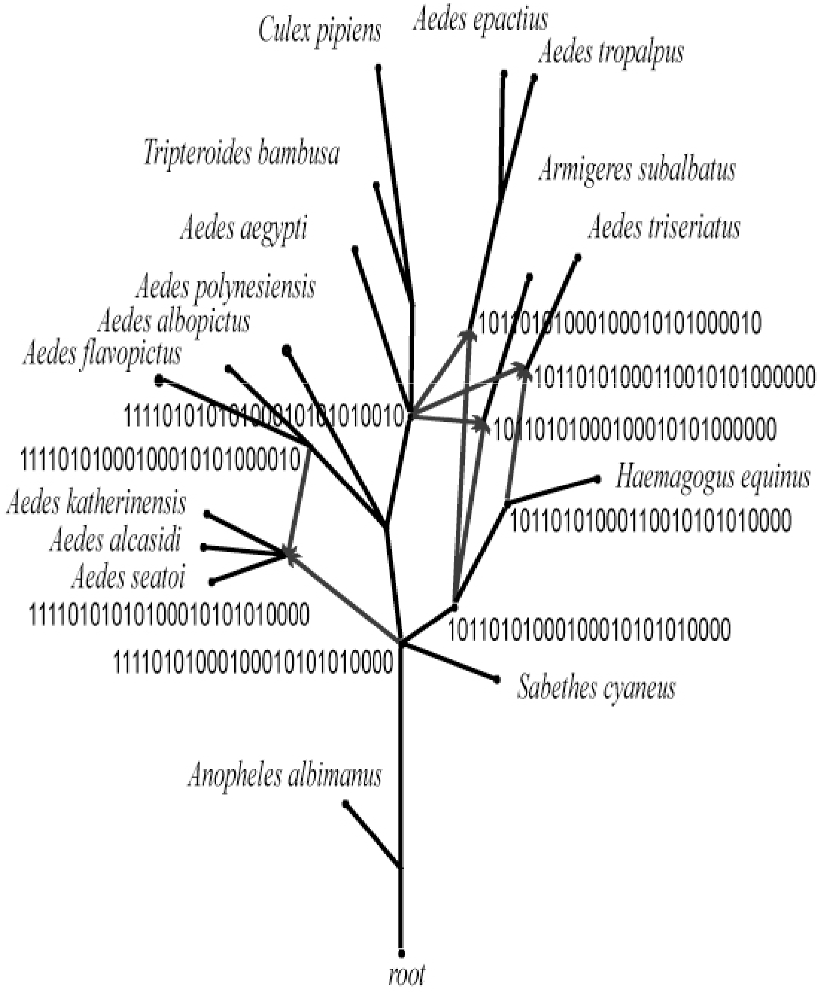
Example 2, Data

- **Input: Restriction maps of the rDNA cistron (length ≈ 10kb) of twelve species of mosquitoes using eight 6bp restriction enzymes:**

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

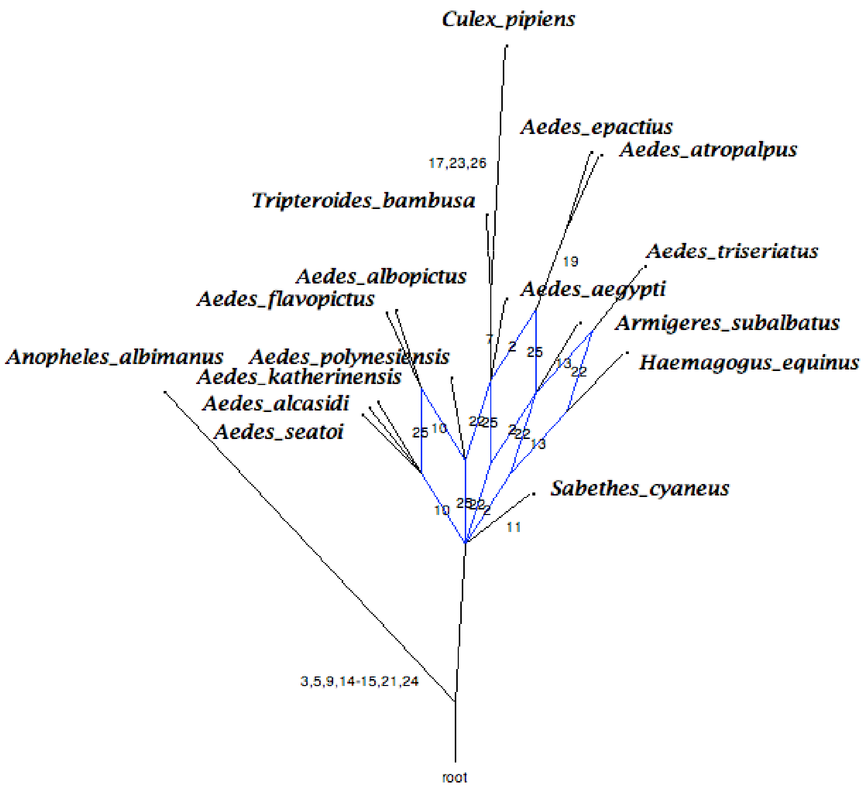
Split Network & Recombination Network

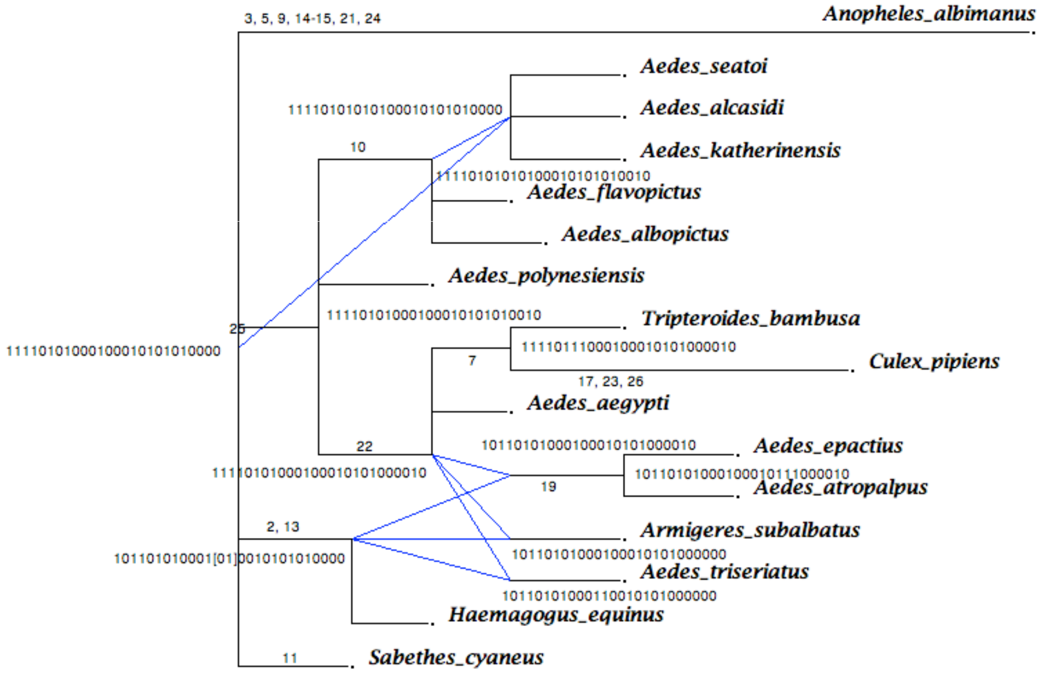




Daniel Huson, 2007

Split Network & Recombination Network





Recombination Network

Branch-and-bound approach

(Lyngso, Song and Hein, WABI 2005)

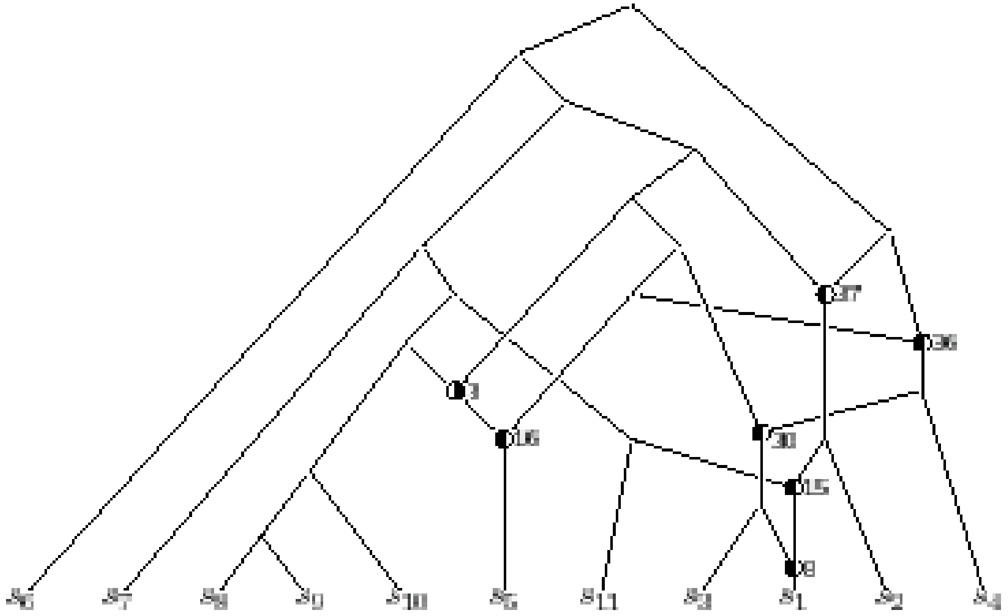
Input: data and limit number of recombinations

Branch: Starting from original data, consider all possible steps backward in time

Bound: If *recombinations used* plus a lower bound on *recombinations still needed* exceeds the prescribed limit, do not pursue current configuration

Example 3

a	0001000100000000
b	0100000100000000
c	0000000000000010
d	0000001000000010
e	0011111000000001
f	0100010001010111
g	0100010011111101
h	1111110011111101
i	1111010011111101



16 haplotyped sites of the alcohol dehydrogenase locus from 11 chromosomes of *D.melanogaster* (Kreitman 1985)

Recombination network with 7 events found using the branch-and-bound method

Daniel Huson, 2007

Software

Software for computing a recombination network from binary sequences:

- Software implementing the approach of Dan Gusfield *et al.*for constructing galled trees is available from: wwwcsif.cs.ucdavis.edu/~gusfield
- SplitsTree4 contains a method RecombinationNetwork for constructing galled trees and more general recombination networks [31, 30]. www.splitstree.org
- Beagle uses branch-and-bound to compute network. www.stats.ox.ac.uk/~lyngsoe/beagle

Part V

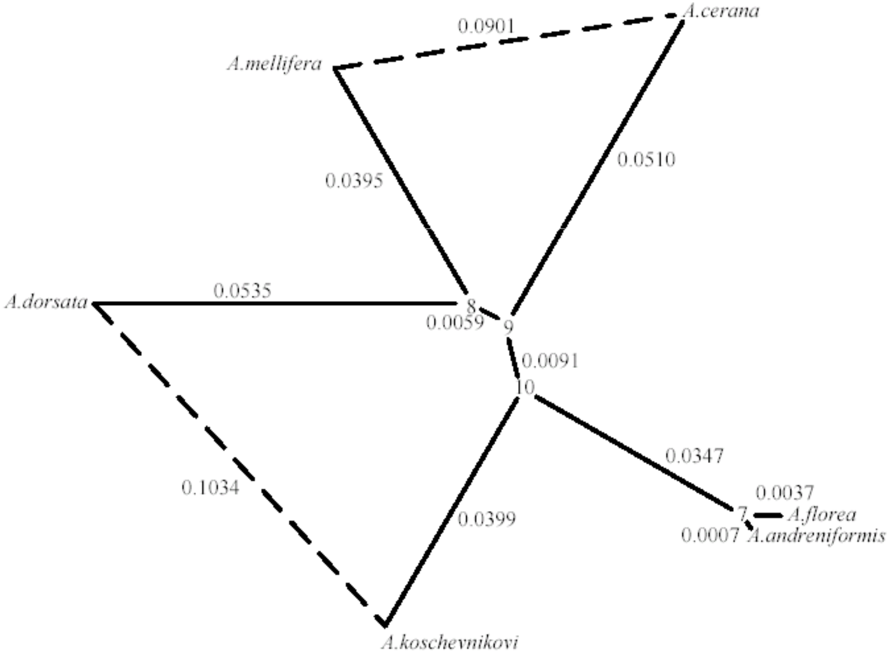
1. Phylogenetic trees
2. Consensus networks and super networks
3. Hybridization and reticulate networks
4. Recombination networks
5. Other

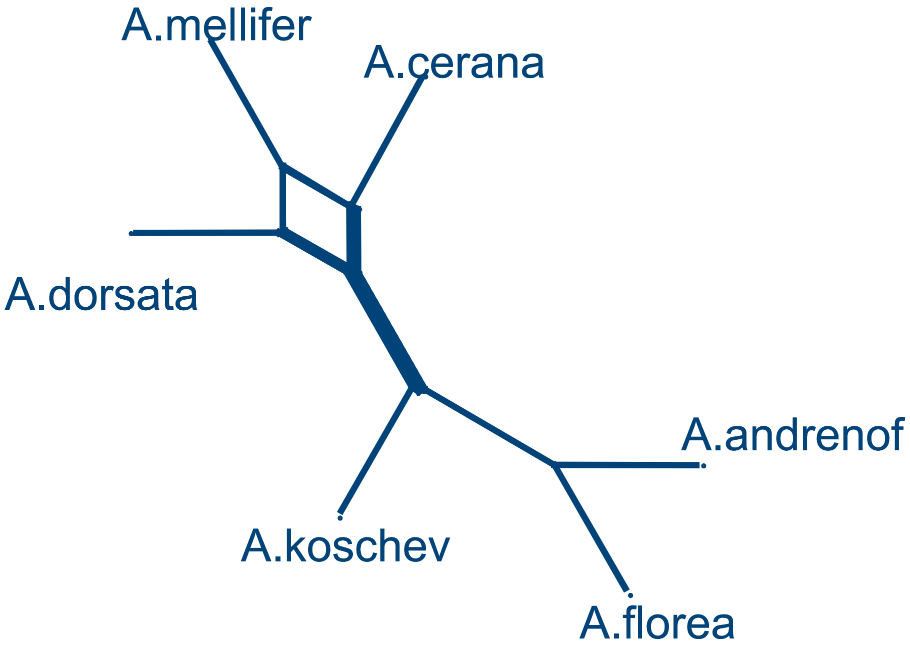
Augmented Trees: Reticulograms

- A *reticulogram* is a tree with additional short-cut edges.
- It is obtained from a distance matrix by first building a tree and then repeatedly adding new edges so as to optimize the least square fit of the graph distances to the matrix.
- Implemented in the program `T-Rex`

Augmented Trees: Reticulograms

Data: DNA sequences of 677 for honey bees.





Reticulogram produced using T-Rex

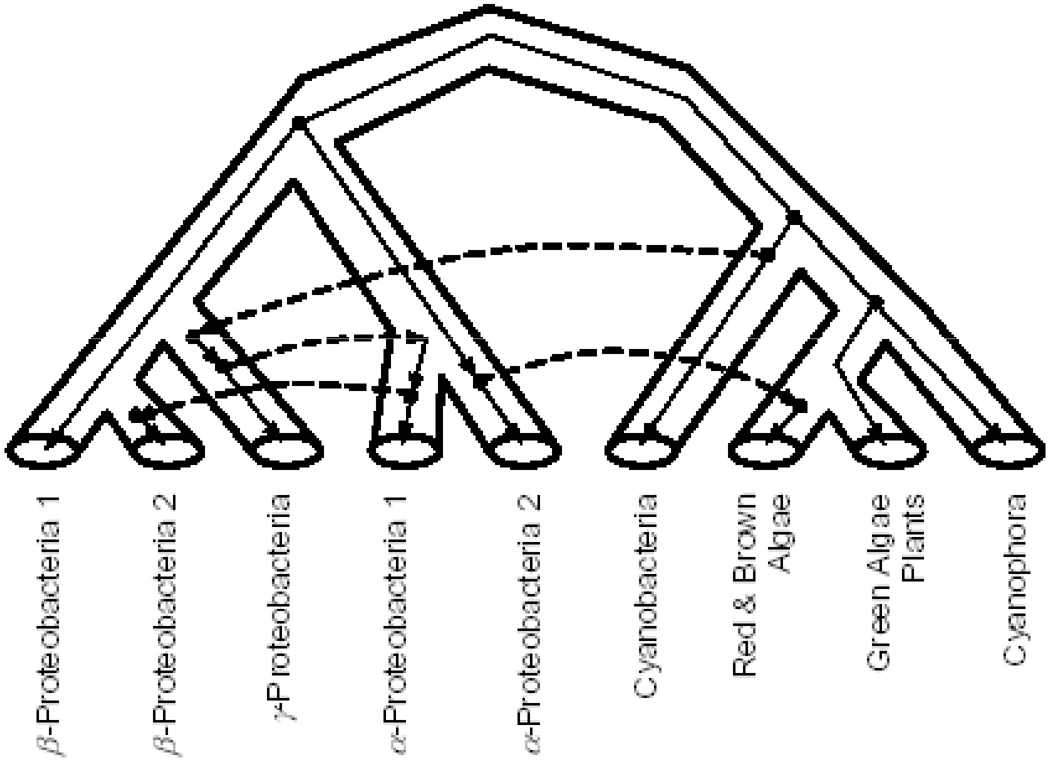
Bootstrap network displays competing signals

Daniel Huson, 2007

Augmenting Species Trees by Gene Trees

- The goal here is to map a set of gene trees on to a given species tree, thus postulating a set of horizontal gene transfer events
- Implemented in the program lattrans

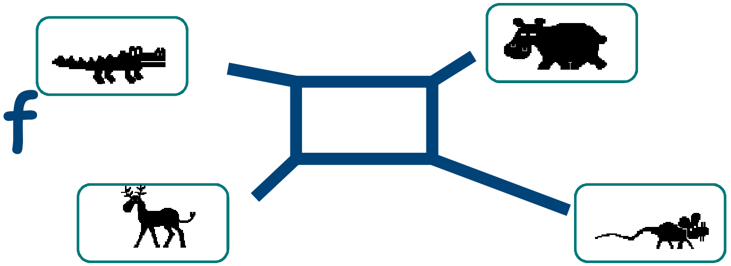
Augmenting Species Trees by Gene Trees



A horizontal gene transfer scenario for the rbcL gene presented in (Hallet and Lagergren, 2001)

Summary

- Implicit phylogenetic networks such as split networks robustly represent incompatible phylogenetic signals…
- …while reticultate networks such as hybridization networks and recombination networks provide explicit models of reticulate evolution



- A wide range of tree and network construction methods are implemented in SplitsTree4