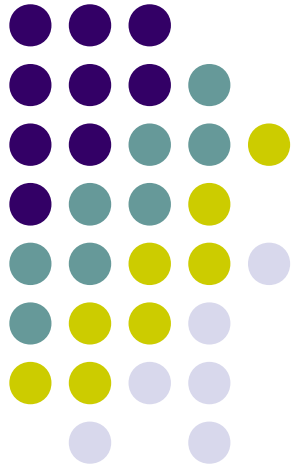
28 April 2011

Algorithmic aspects of phylogenetic network construction*

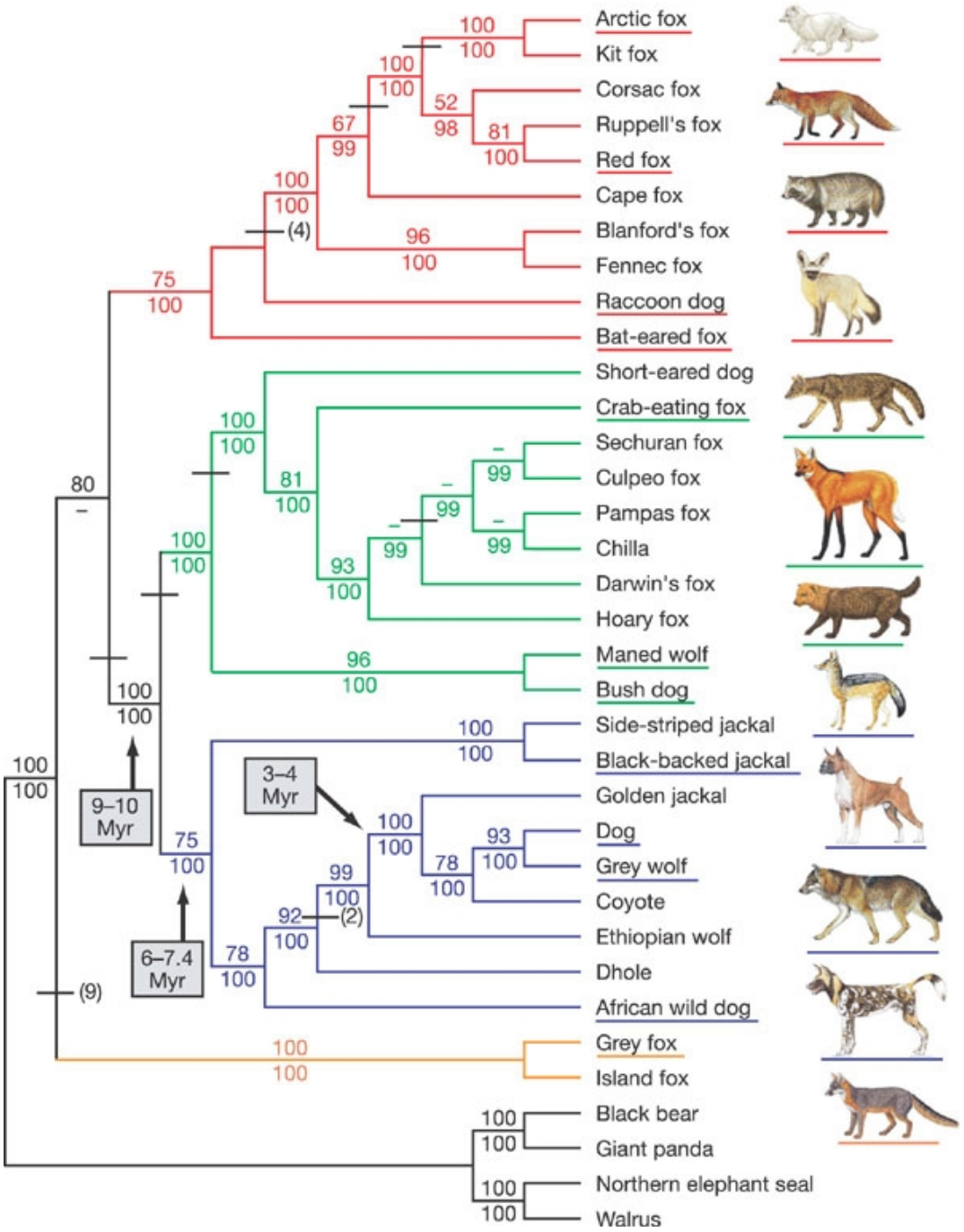
Steven Kelk

Networks and Strategic Optimization
Department of Knowledge Engineering

Universiteit Maastricht



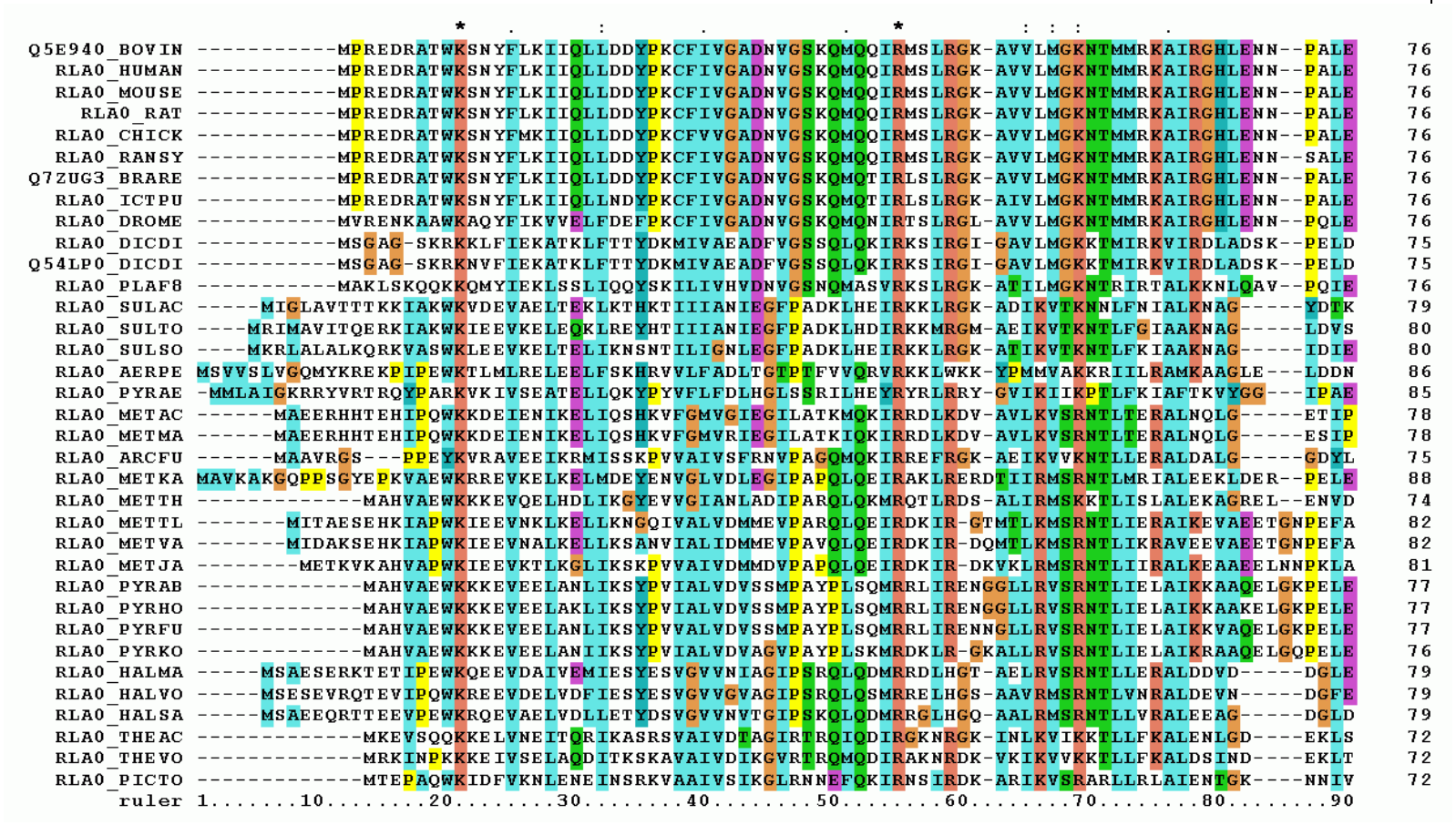
* On the elusiveness of optimal softwired phylogenetic networks



Genome sequence, comparative analysis and haplotype structure of the domestic dog

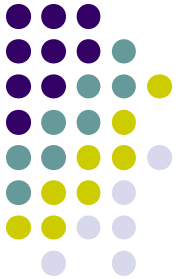
Lindblad-Toh et al, Nature 2005

(Almost) everything begins with Multiple Sequence Alignment



From: http://en.wikipedia.org/wiki/File:RPLP0_90_ClustalW_aln.gif

There is more to life than trees



· All these methods assume that a (single) tree is the best way to model the underlying evolution.

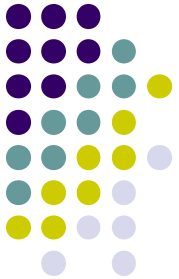
· If this is not true, then we have a problem, because there is a high risk that the output of tree-building algorithms will then be **meaningless**.

· Sometimes there are clues about this:
- Algorithms build very badly supported trees
- Extra knowledge about the underlying evolutionary mechanisms

· But in general it is **dangerously easy** to confuse non-treelike evolution with a **noisy tree signal**.

· Therefore **critical** to understand and model underlying mechanisms.

Why might we get weak support for a tree?



"Noisy tree"
Data *does* fit a single tree, weak support is only a consequence of "noise"

"Trees in trees"
Data consists of multiple different tree signals…but both gene and species evolution are still ultimately treelike (e.g. due to incomplete lineage sorting, gene loss, gene duplication)

"Trees in networks"
Data consists of multiple different tree signals…gene evolution is treelike, but species evolution is no longer treelike (e.g. hybridization, horizontal gene transfer)

Other phenomena
Such as recombination (Meiotic, Sexual)

Why might we get weak support for a tree?

"Noisy tree"
Data *does* fit a single tree, weak support is only a consequence of "noise"

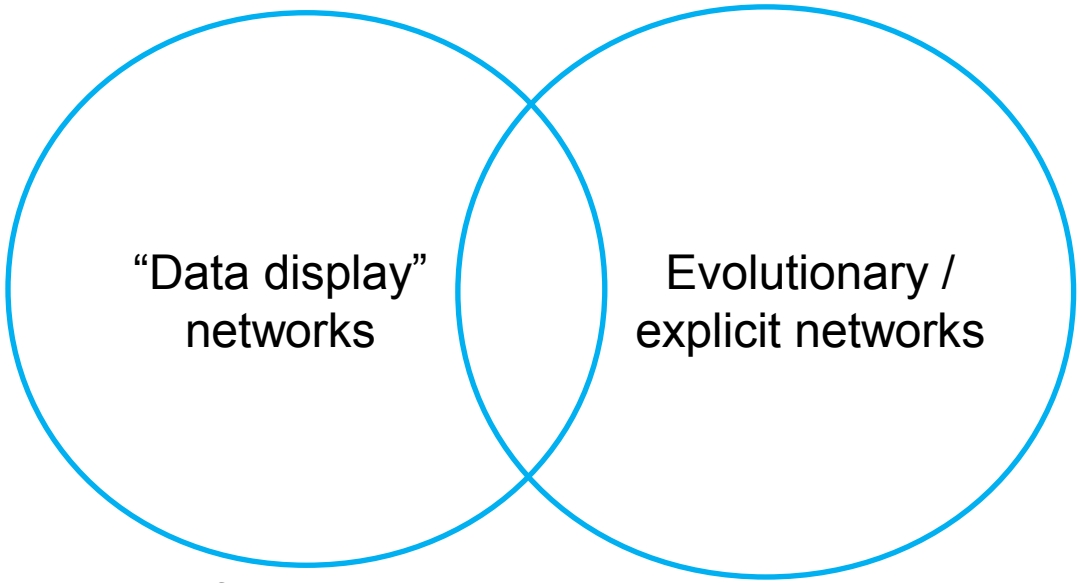
"Trees in trees"
Data consists of multiple different tree signals…but both gene and species evolution are still ultimately treelike (e.g. due to incomplete lineage sorting, gene loss, gene duplication)

Other phenomena
Such as recombination (Meiotic, Sexual)

"Trees in networks"
Data consists of multiple different tree signals…gene evolution is treelike, but species evolution is no longer treelike (e.g. hybridization, horizontal gene transfer)

"Reticulate" (i.e. non-treelike) evolutionary phenomena

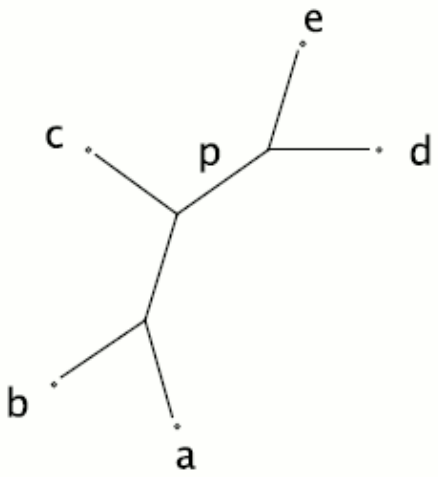
Phylogenetic networks



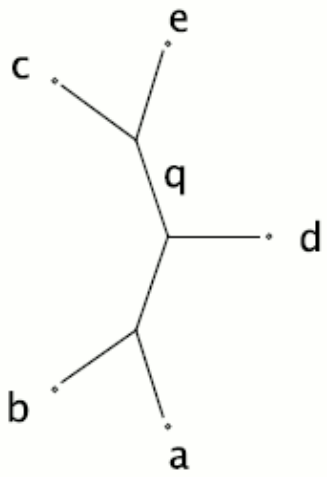
No explicit model of evolution: tries to graphically represent **where** the data is non-treelike

Tries to model the **events** that caused the data to be non-treelike

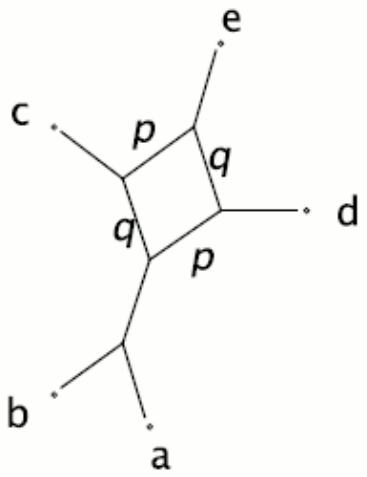
Data-display networks (1)



T_1



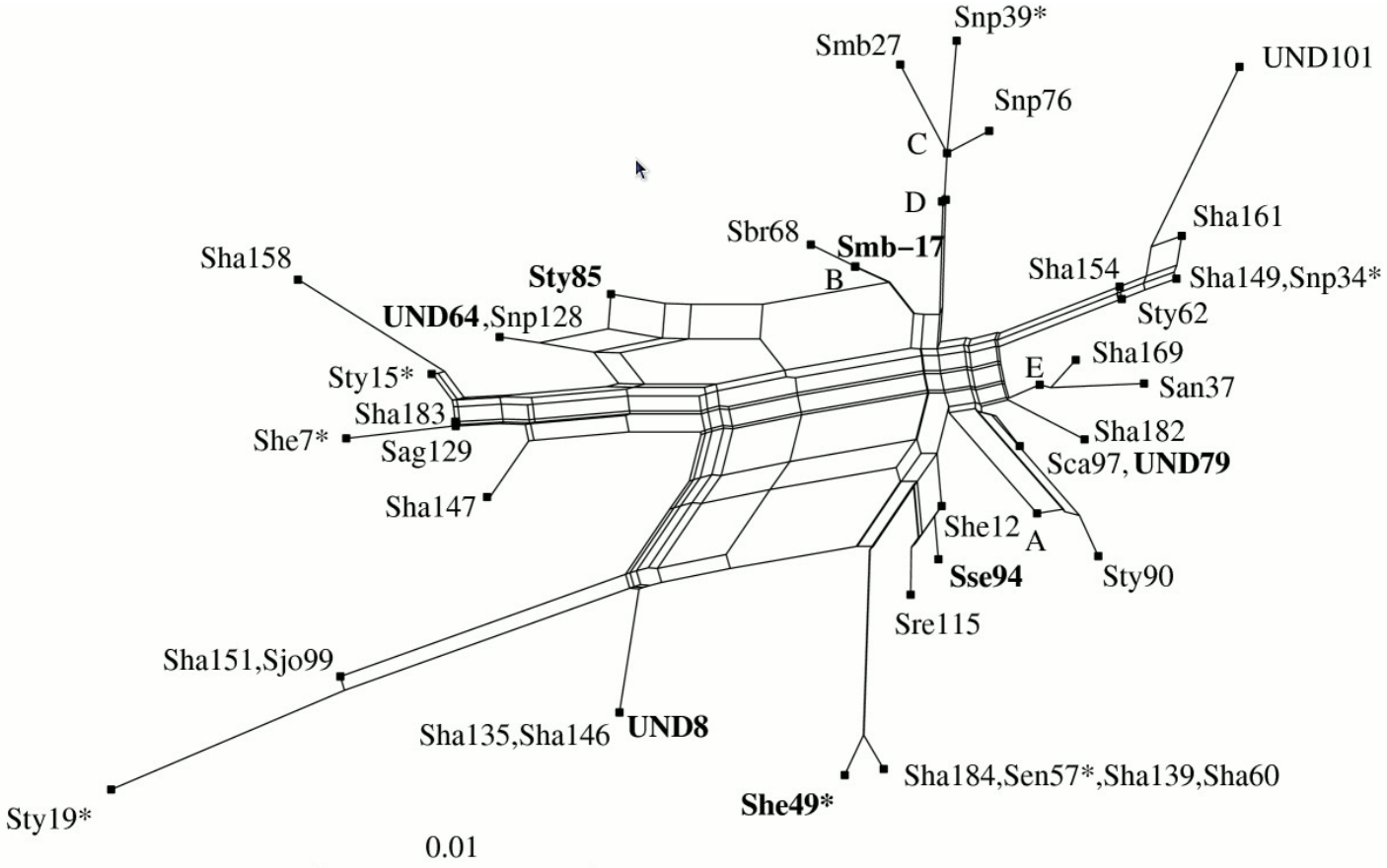
T_2



$SN(\Sigma)$

From: Daniel Huson, ISMB-Tutorial 2007: Introduction to Phylogenetic Networks

Data-display networks (2)



A phylogenetic network. The network was generated by Neighbor-Net for a sequence-based data set comprising of *Salmonella* isolates that originally appeared in [17]. A detailed network-based analysis of this data is presented in [2], where the strains indicated in bold-face are tested for the presence of recombination. Note that the network is planar (that is, it can be drawn in the plane without any crossing edges), and that parallel edges in the network represent bipartitions of the data.

Bryant *et al. Algorithms for Molecular Biology* 2007 **2**:8 doi:10.1186/1748-7188-2-8

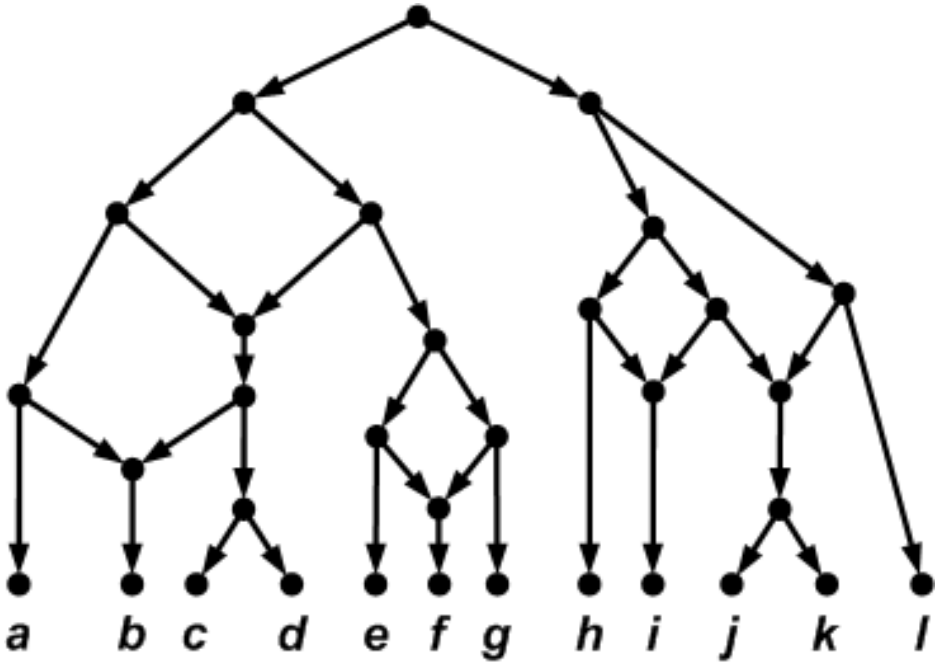
Evolutionary phylogenetic networks

- Can be used to explicitly model reticulate evolution:
  - Hybridization
  - Horizontal Gene Transfer (HGT)
  - Recombination

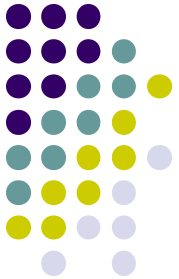
- Reticulation vertices often have an explicit biological interpretation

- Rooted, with an explicit "direction" of evolution

- Underlying mathematical abstractions are often similar, despite different scale levels of interpretation



Evolutionary phylogenetic networks

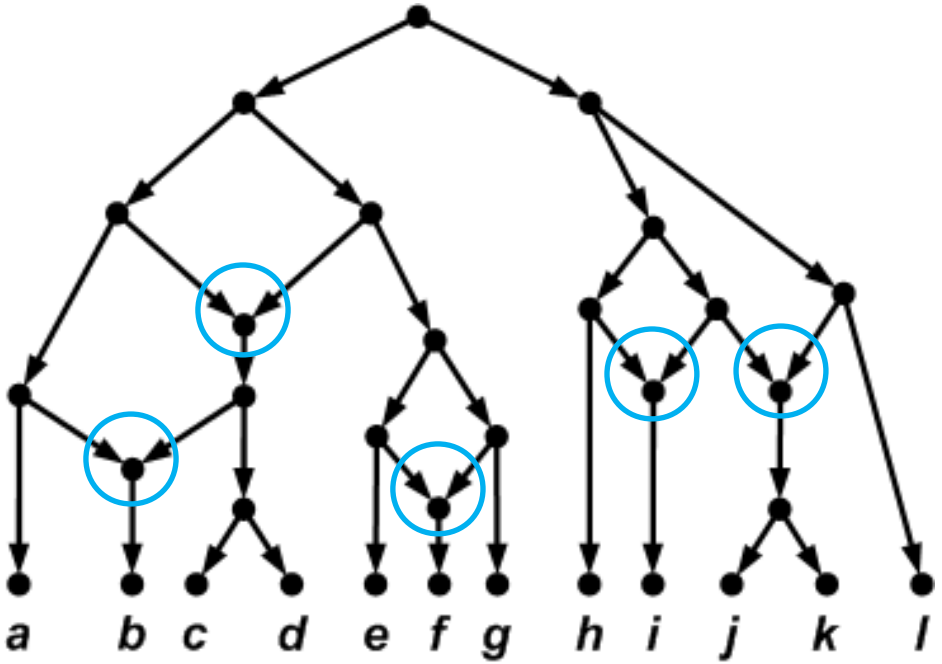


- Can be used to explicitly model reticulate evolution:
  - Hybridization
  - Horizontal Gene Transfer (HGT)
  - Recombination

- Reticulation vertices often have an explicit biological interpretation

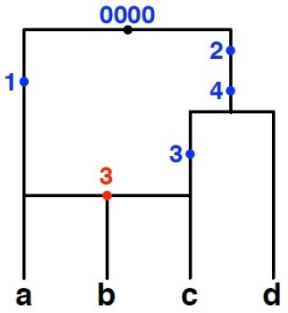
- Rooted, with an explicit "direction" of evolution

- Underlying mathematical abstractions are often similar, despite different scale levels of interpretation

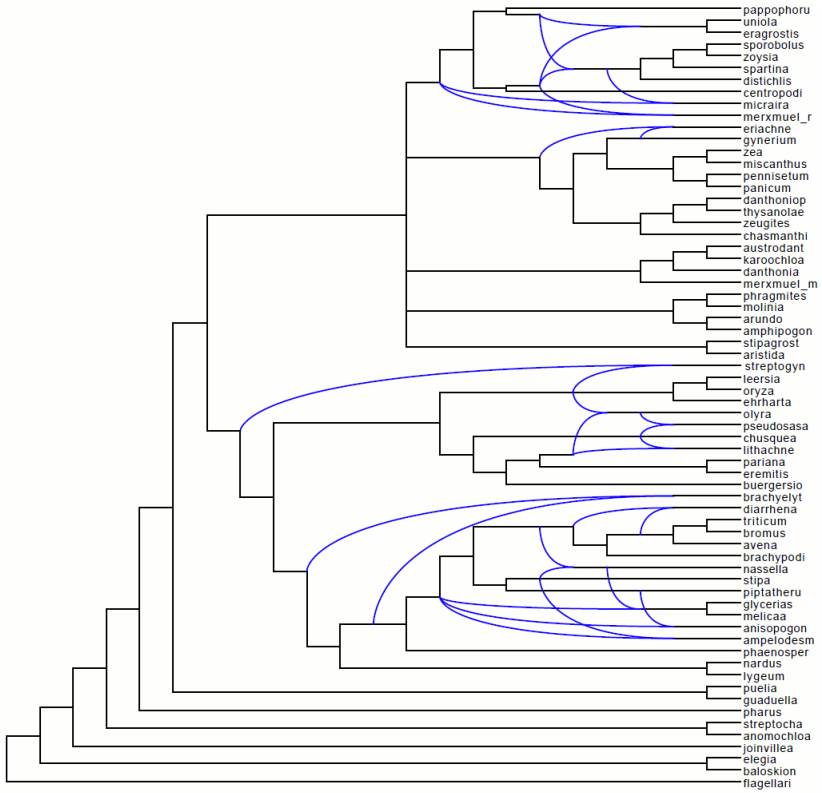


Different models and scales, always rooted, directed acyclic graphs (DAGs)

	c_1	c_2	c_3	c_4
a	1	0	0	0
b	1	0	1	1
c	0	1	1	1
d	0	1	0	1

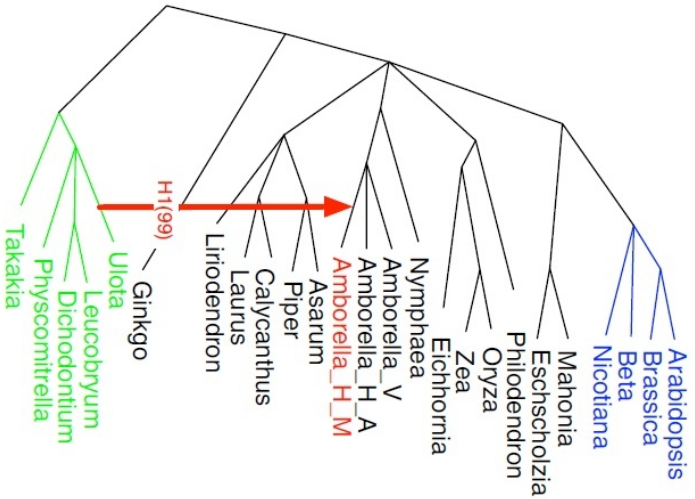


Recombination network



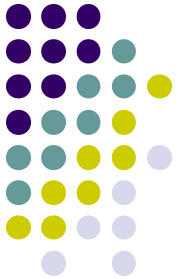
"Softwired cluster" network

(f) nad7



Horizontal Gene Transfer (HGT)

Constructing evolutionary phylogenetic networks



· It's important to ask ourselves several questions:

1. MODEL: What are we trying to **model** exactly? Is it biologically realistic?

2. OBJECTIVE: What do we consider to be an **optimal solution** within that model?

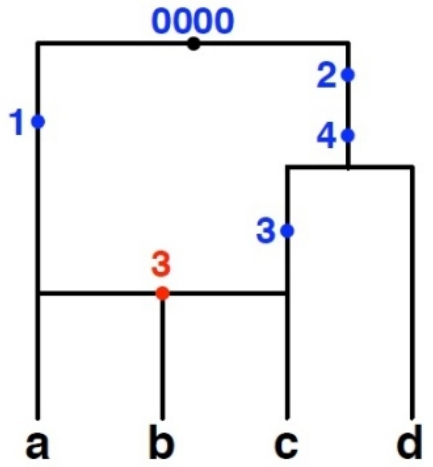
3. TRACTABILITY: Is there any hope of developing **efficient algorithms** to compute optimal solutions?

· Extremely challenging to simultaneously answer these questions well!

· In the meantime: many different models, algorithms, packages

Case study 1: constructing Recombination Networks

	c_1	c_2	c_3	c_4
a	1	0	0	0
b	1	0	1	1
c	0	1	1	1
d	0	1	0	1

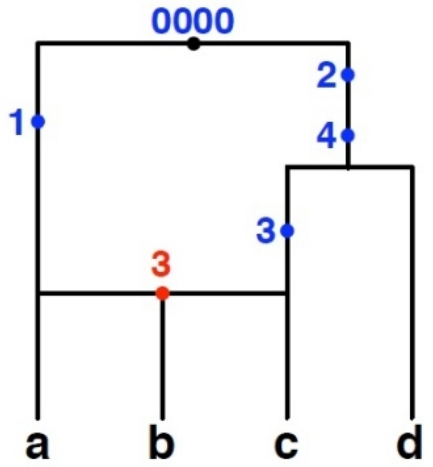


From: "Evolutionary phylogenetic networks: models and issues." By Luay Nakhleh

· Input is **binary character data** (i.e. strings of binary data)
· Reticulations represent **chromosomal crossover** (mostly single crossover, sometimes multiple crossover). Sometimes also gene conversion.
· Mutation model is the **"infinite sites"** model: at most one mutation per site (0 to 1, or 1 to 0).
· Goal is to construct a recombination network with a **minimum number** of reticulation events.

Case study 1: constructing Recombination Networks

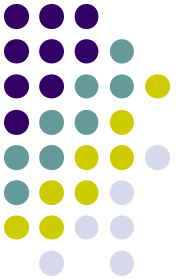
	c_1	c_2	c_3	c_4
a	1	0	0	0
b	1	0	1	1
c	0	1	1	1
d	0	1	0	1



From: "Evolutionary phylogenetic networks: models and issues." By Luay Nakhleh

· Input is **binary character data** (i.e. strings of binary data)
· Reticulations represent **chromosomal crossover** (mostly single crossover, sometimes multiple crossover). Sometimes also gene conversion.
· Mutation model is the **"infinite sites"** model: at most one mutation per site (0 to 1, or 1 to 0).
· Goal is to construct a recombination network with a **minimum number** of reticulation events.

Case study 1: constructing Recombination Networks



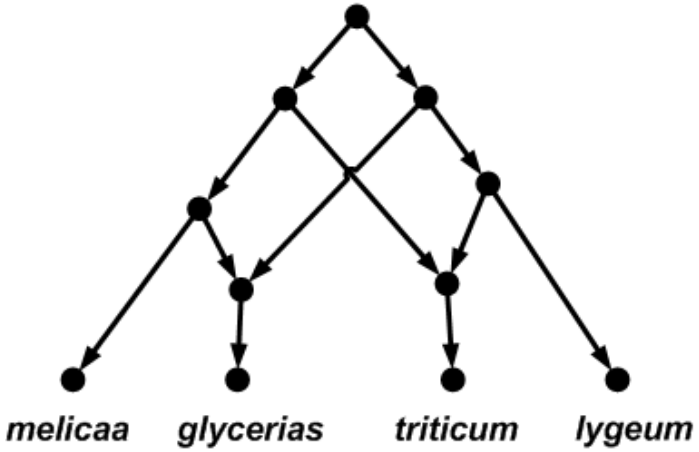
· Extensive interest and research from the theoretical computer science community: computing a network with a minimum number of recombinations **is NP-hard.**

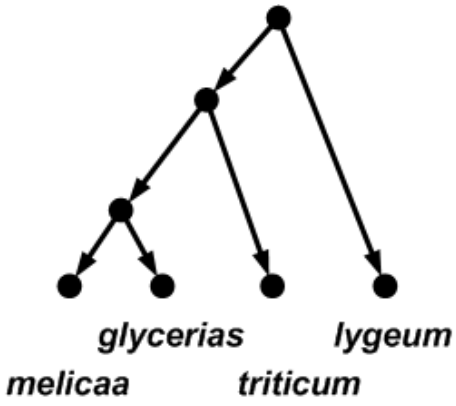
· The groups who worked on this problem (e.g. Dan Gusfield's group at UC Davis) mainly responded to this hardness by developing (computational) **lower and upper bounds** on the minimum number of reticulations required. Many of these bounds are also NP-hard to compute.

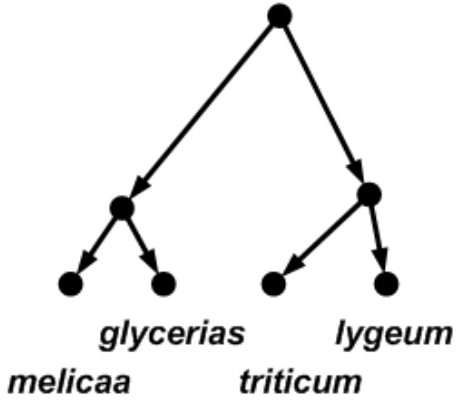
· Also: **branch and bound** techniques for computing optimal solutions for (very) small instances.

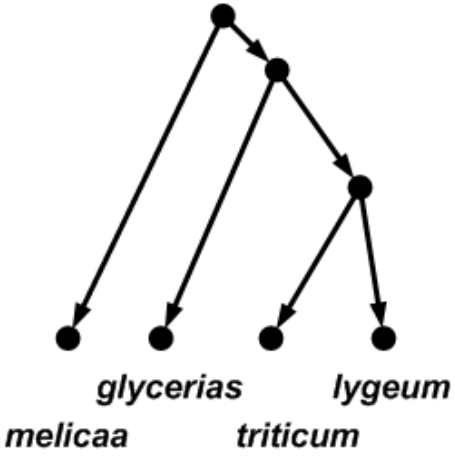
· Curiously there has been **very little work on approximation algorithms** i.e. fast algorithms that compute solutions that are within a certain multiplicative factor of optimality.

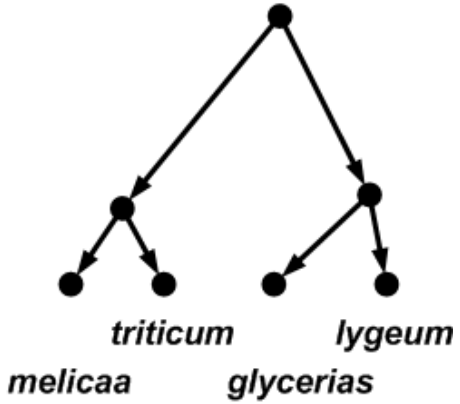
Case study 2: methods based on combining several trees into a single network



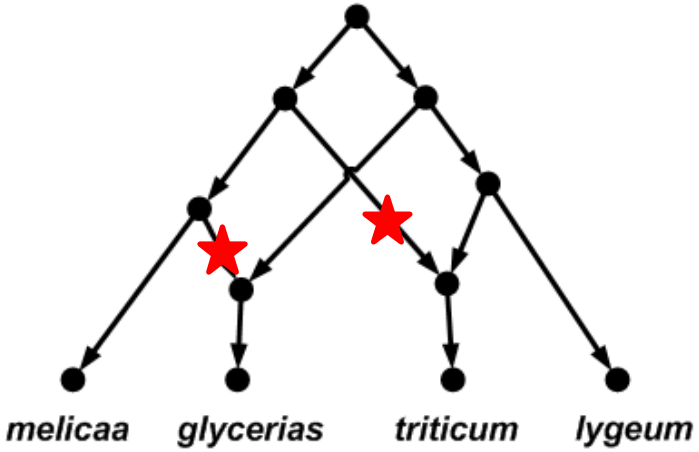


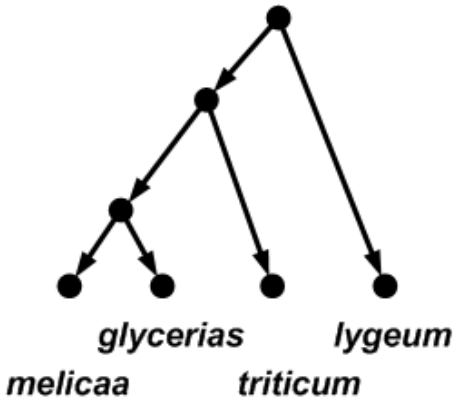


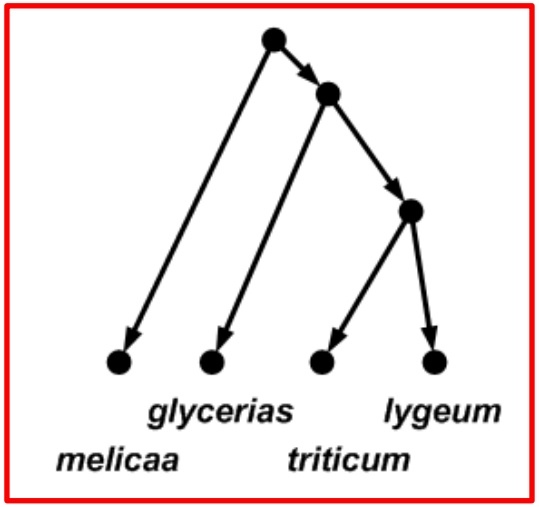


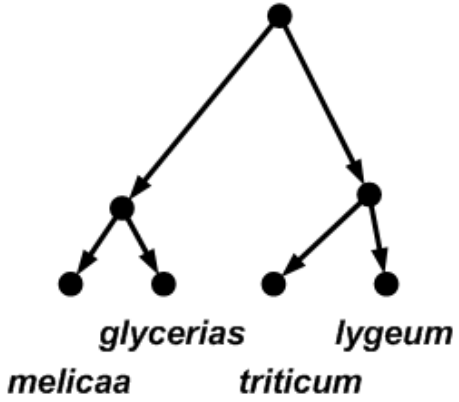


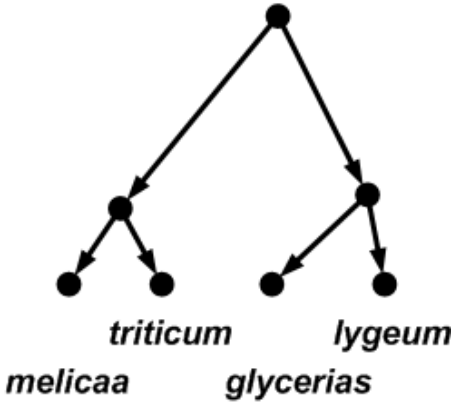
Case study 2: methods based on combining several trees into a single network





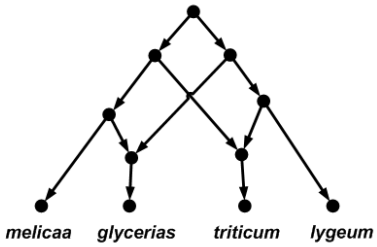




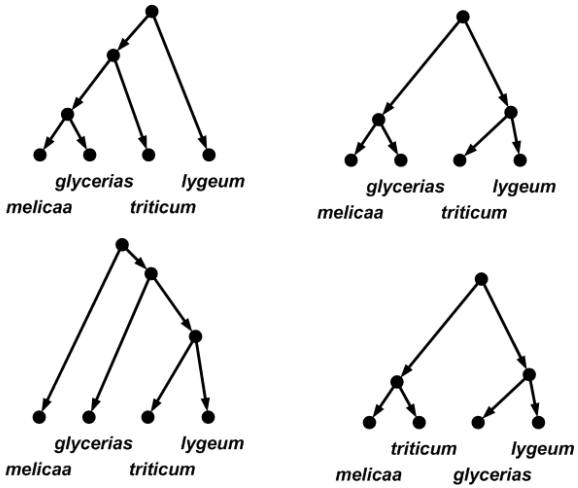


Case study 2: methods based on combining several trees into a single network

· Recall this example:



Species network



Four gene trees contained by the species network

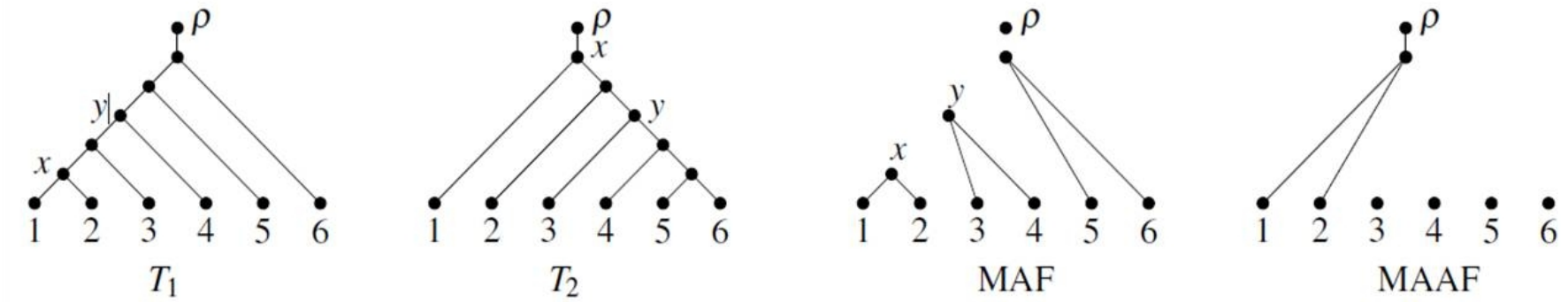
· Input: a set of gene trees

· Output: a species network that **contains all the input gene trees** and which has a **minimum number** of reticulations

Case study 2: methods based on combining several trees into a single network

· There has been a huge amount of research from (a different wing of) the theoretical computer science community for this problem, mainly focusing on case when the input consists of **exactly two binary gene trees.**

· Most research has focused on the very close link with a problem called the **Maximum Acyclic Agreement Forest** problem (MAAF).

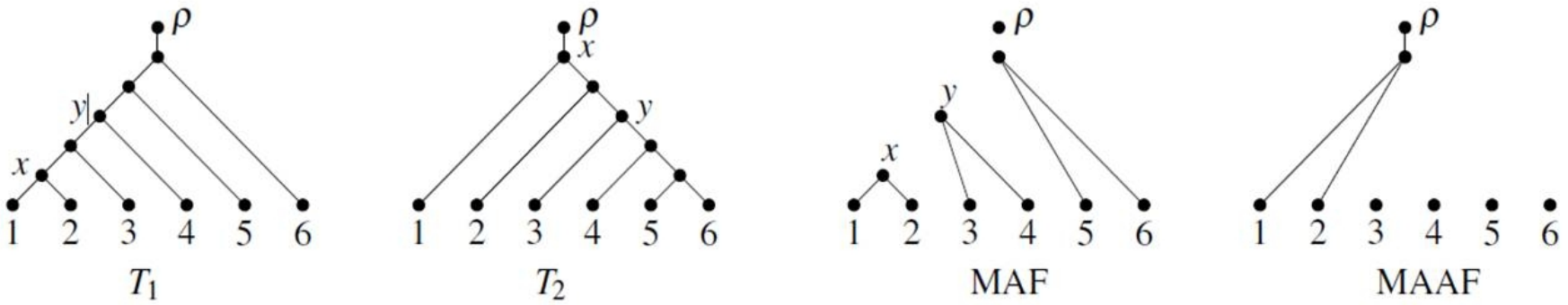


From: A UNIFYING VIEW ON APPROXIMATION AND FPT OF AGREEMENT FORESTS, Christopher Widden 2009 (Master's Thesis)

Case study 2: methods based on combining several trees into a single network

· The problem is **NP-hard** and **APX-hard** but despite these complexity-theoretic limitations algorithmic progress has been **considerable.**

- Reduction rules (correctness of divide and conquer)
- Fixed parameter tractability
- Integer linear programming solutions (exploiting the static nature of the MAAF problem)
- Algorithms to enumerate all optimal solutions
- Approximation algorithms (…?)



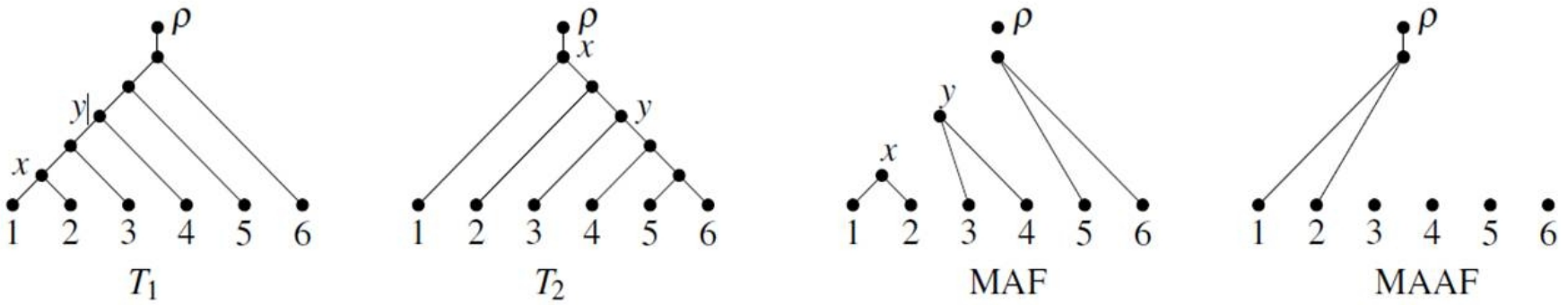
From: A UNIFYING VIEW ON APPROXIMATION AND FPT OF AGREEMENT FORESTS, Christopher Widden 2009 (Master's Thesis)

Case study 2: methods based on combining several trees into a single network

· However, people working on this problem have hit upon barrier.

· Much of the "MAAF theory" starts to break down when there are **more than two trees** in the input. In the absence of a rigorous theory for more than two trees, researchers are again seeking refuge in **lower/upper bound** computations. Approximation algorithms seem difficult to develop.

· Multiple research groups are moving towards a "beyond MAAFs" theory…who will get there first?



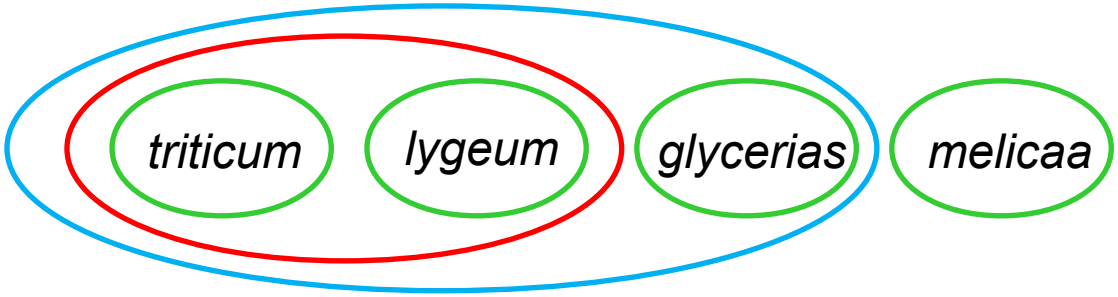
From: A UNIFYING VIEW ON APPROXIMATION AND FPT OF AGREEMENT FORESTS, Christopher Widden 2009 (Master's Thesis)

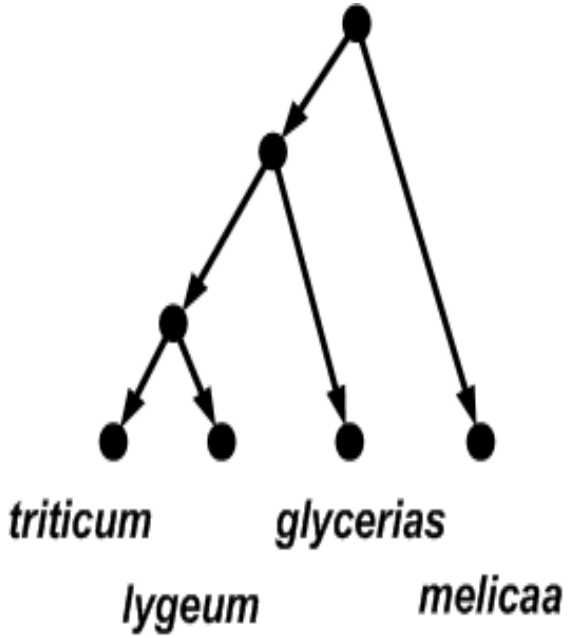
Case study 3: combining softwired clusters into a single network

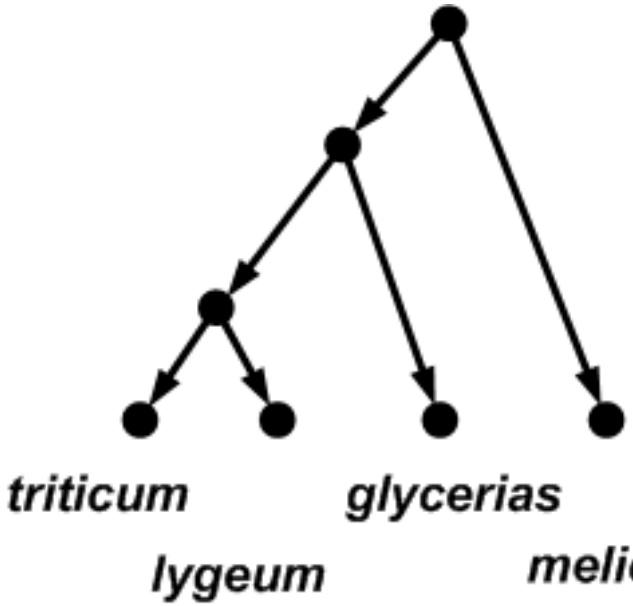
· Every edge *(u,v)* of a tree induces a **cluster**: the set of leaf descendants of v.

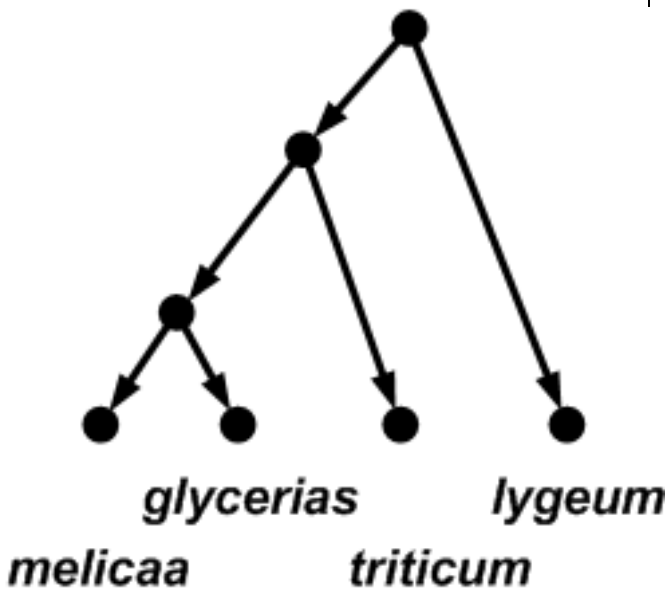
· The set of clusters induced by the edges of a tree, is a **laminar family** .

· A tree is completely characterised by its set of clusters.





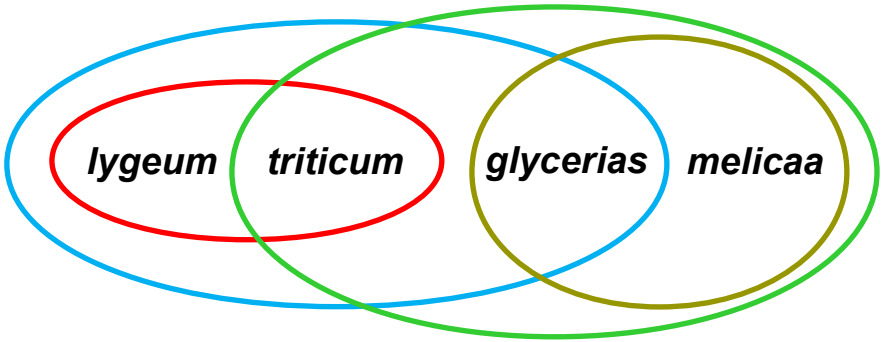


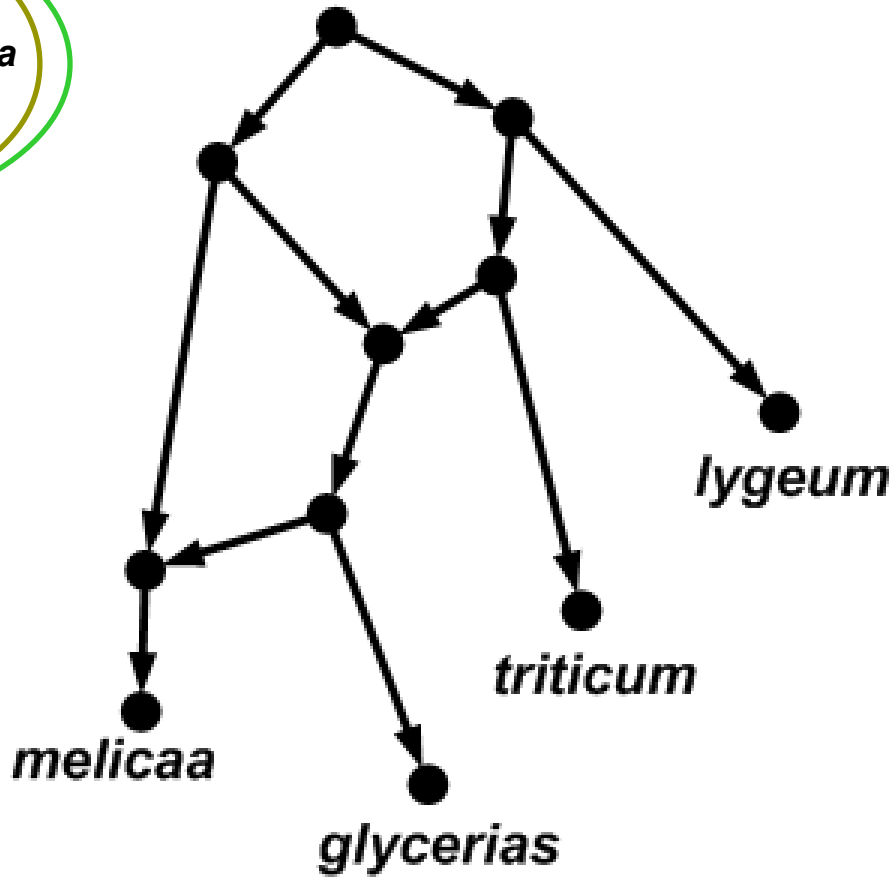


{triticum}, {lygeum}, {glycerias}, {melicaa},
{triticum, lygeum},
{triticum, lygeum, glycerias}.

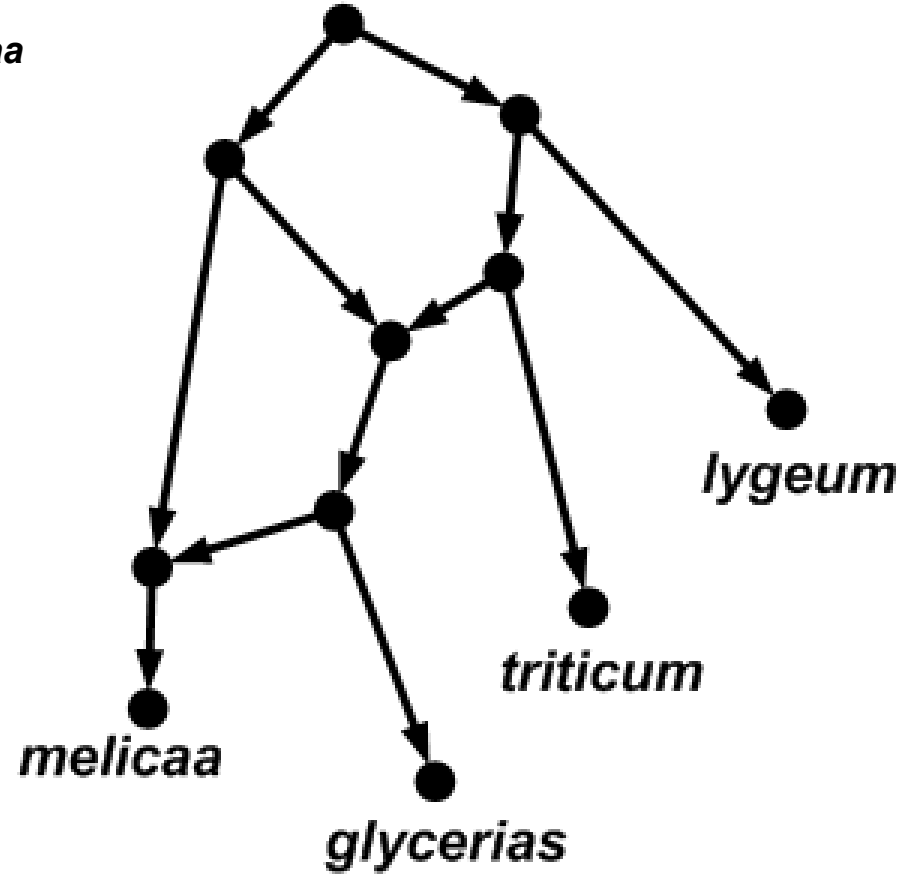
{triticum}, {lygeum}, {glycerias}, {melicaa},
{melicaa, glycerias},
{melicaa, glycerias, triticum}.

Union of clusters from both trees: {triticum}, {lygeum}, {glycerias}, {melicaa}, {triticum, lygeum}, {triticum, lygeum, glycerias}, {melicaa, glycerias, triticum}.

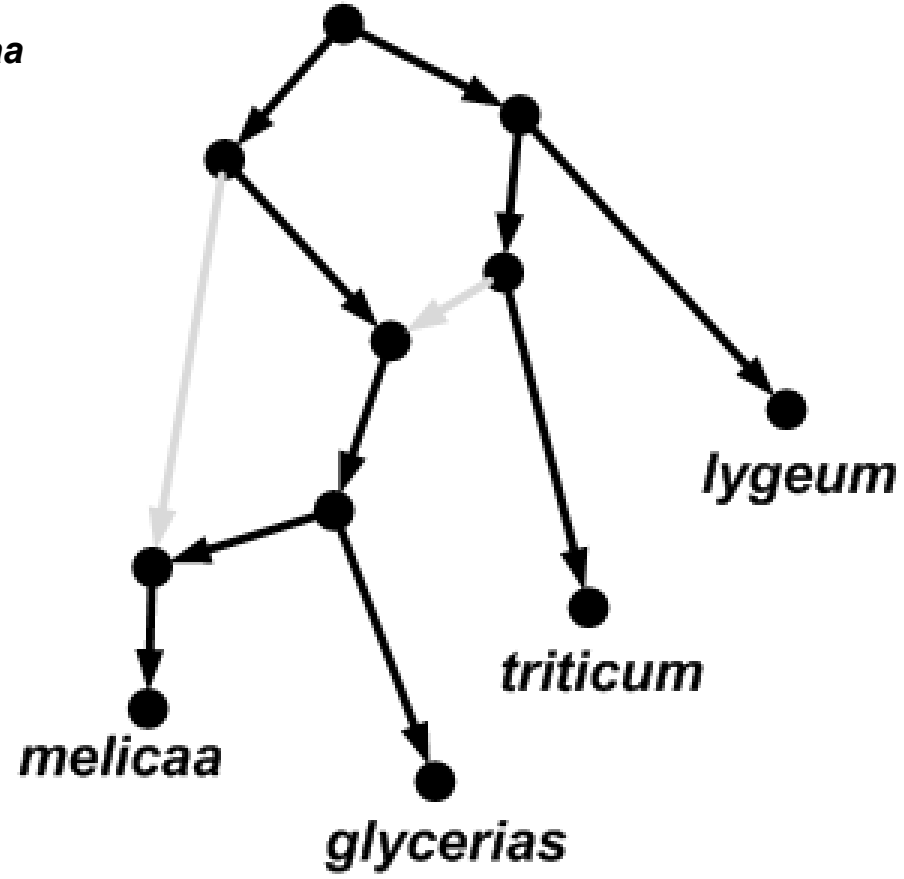

lygeum
triticum
glycerias
melicaa


lygeum
triticum
melicaa
glycerias

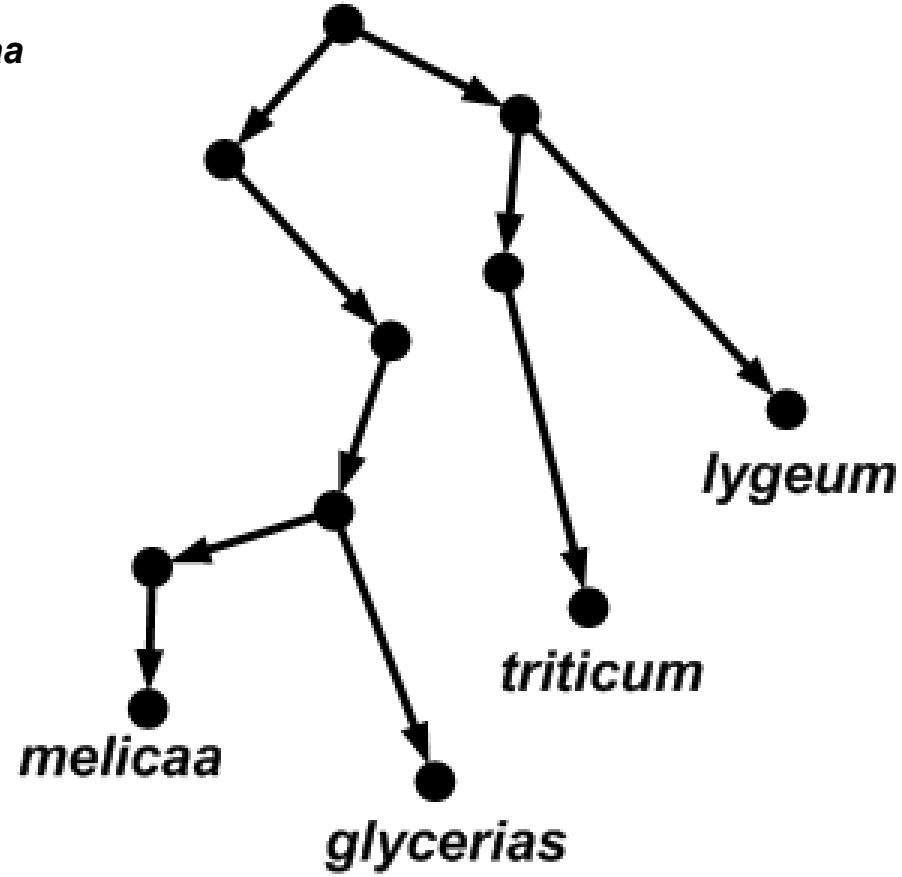
lygeum *triticum* *glycerias* *melicaa*



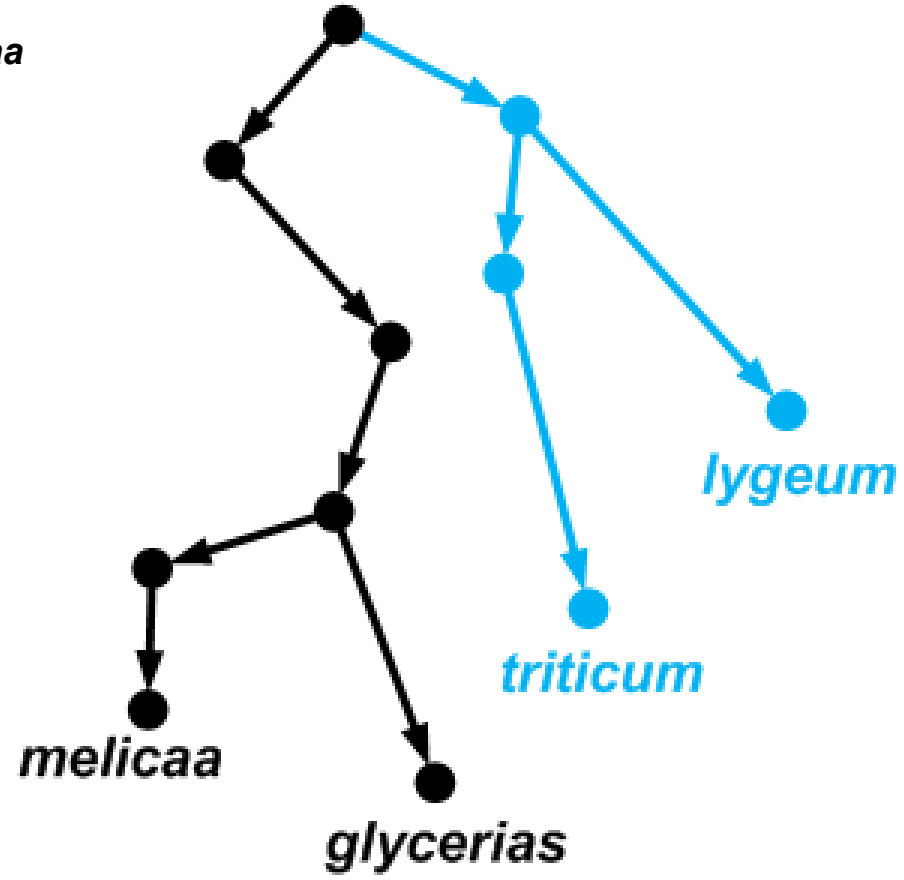
lygeum *triticum* *glycerias* *melicaa*

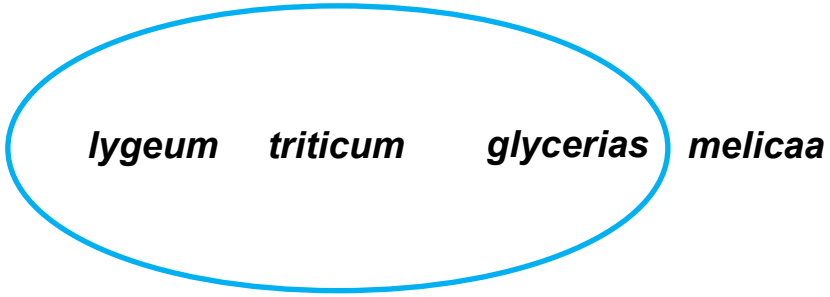


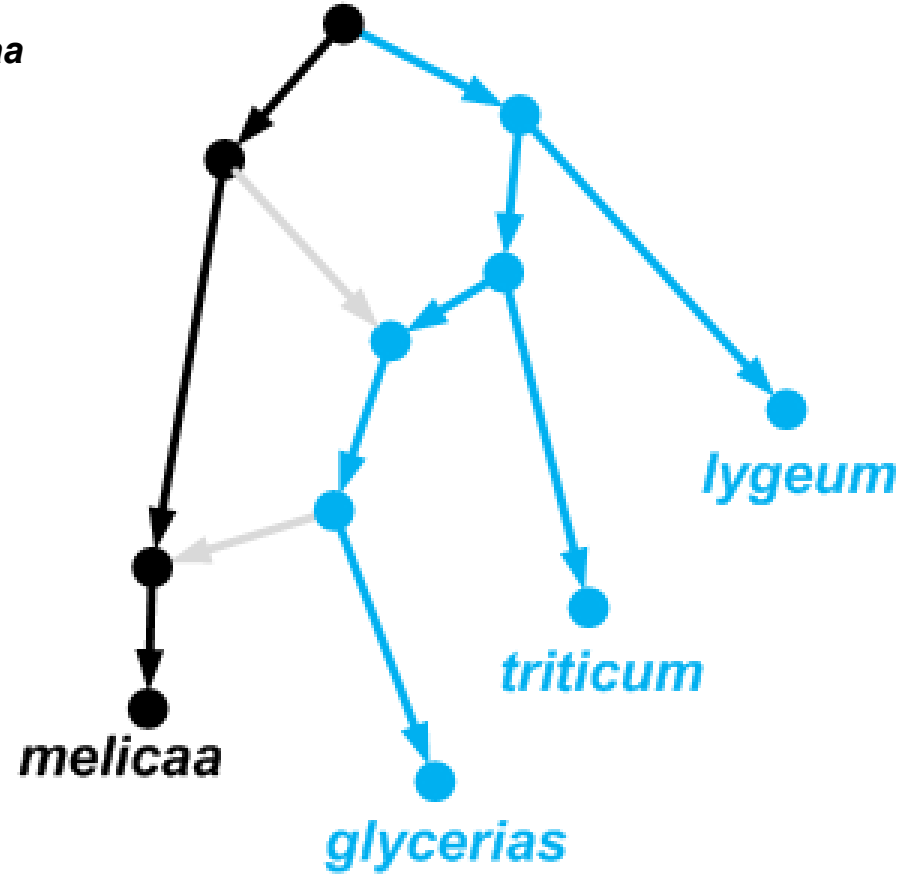
lygeum *triticum* *glycerias* *melicaa*



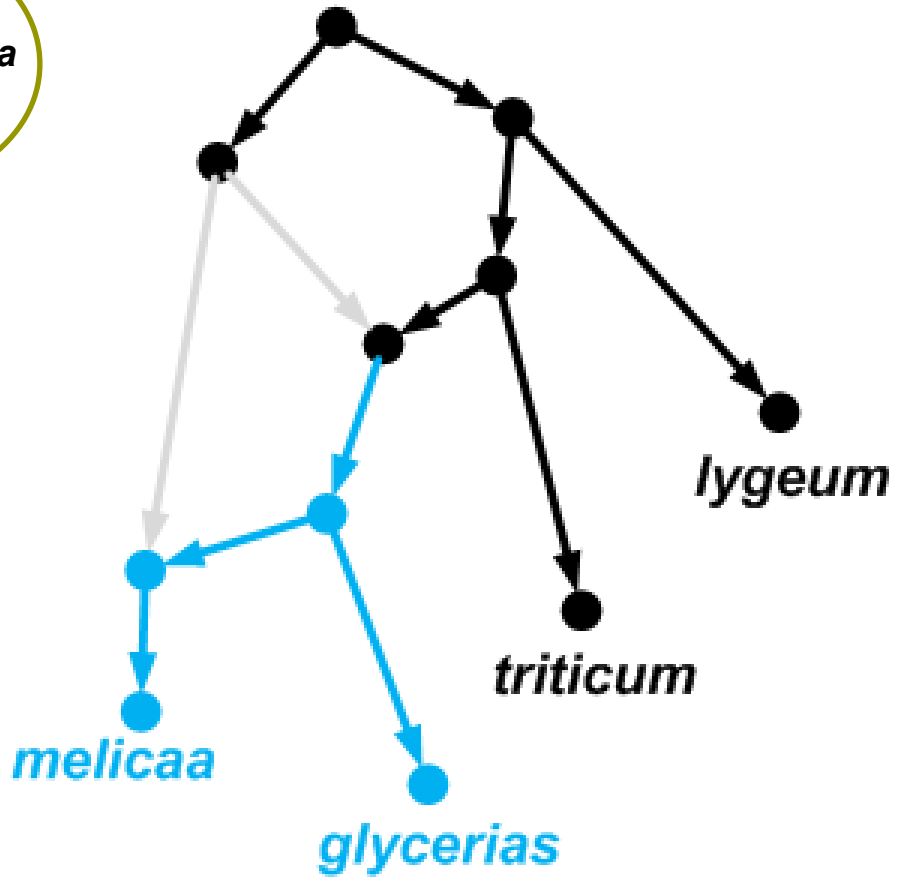
lygeum *triticum* *glycerias* *melicaa*

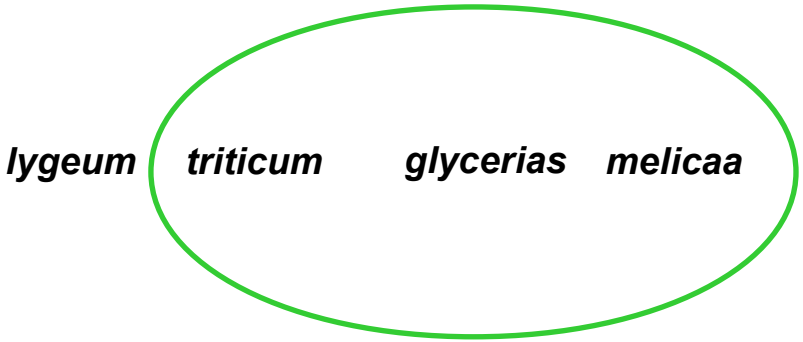


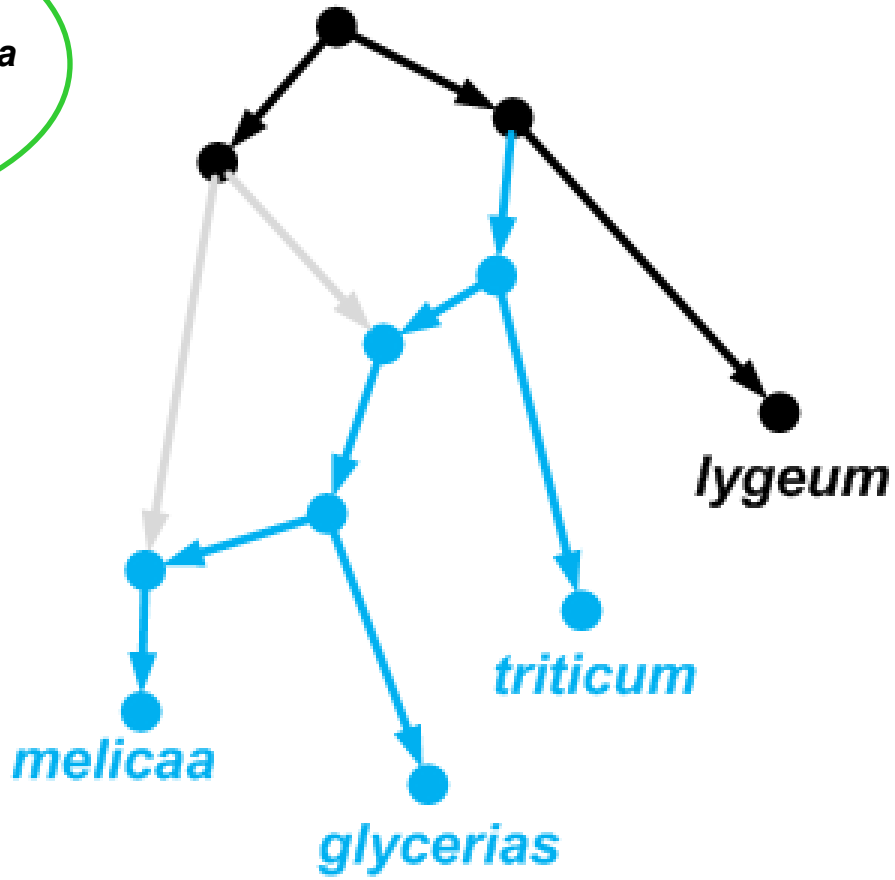

lygeum
triticum
glycerias
melicaa


lygeum
triticum
melicaa
glycerias

lygeum *triticum* *glycerias* *melicaa*



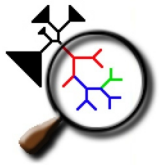

lygeum
triticum
glycerias
melicaa


lygeum
triticum
melicaa
glycerias

Case study 3: combining softwired clusters into a single network

· There are multiple algorithms and software packages for constructing networks with a small number of reticulations that display all the clusters contained in a set of input trees, e.g.

- CLUSTERNETWORK (2008)
- GALLEDNETWORK (2009)
- CASS (2010)
- CLUSTISTIC (2011).



Dendroscope

by Daniel H. Huson

with contributions from Tobias Dezulian, Markus Franz, Christian Rausch, Daniel C. Richter and Regula Rupp

www-ab.informatik.uni-tuebingen.de/software/dendroscope

· Producing solutions with a minimum number of reticulation is **still NP-hard and APX-hard, even for clusters obtained from two trees** but there has been positive algorithmic progress despite this. In particular: if we assume the minimum number of reticulations has been fixed as a constant.

· Advantage of using clusters, rather than the trees themselves, is that it allows a focus on **only well-supported "clades" in the input trees.** But…

Case study 3: combining softwired clusters into a single network

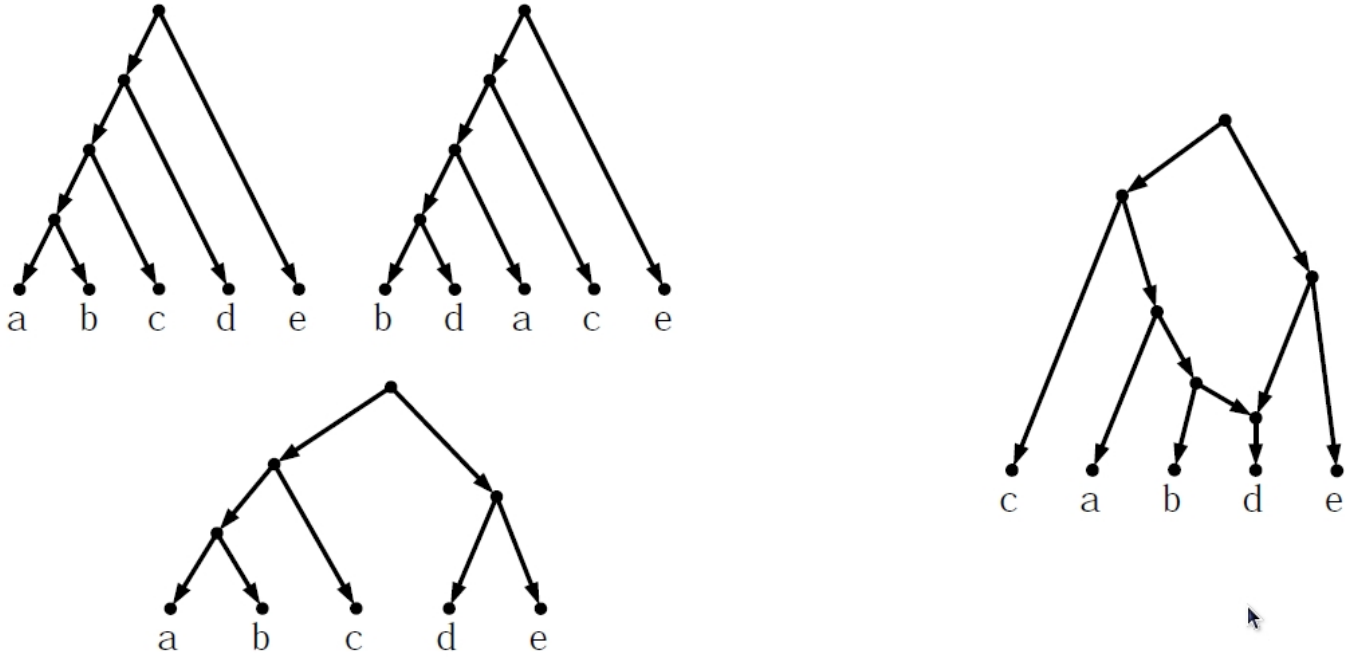
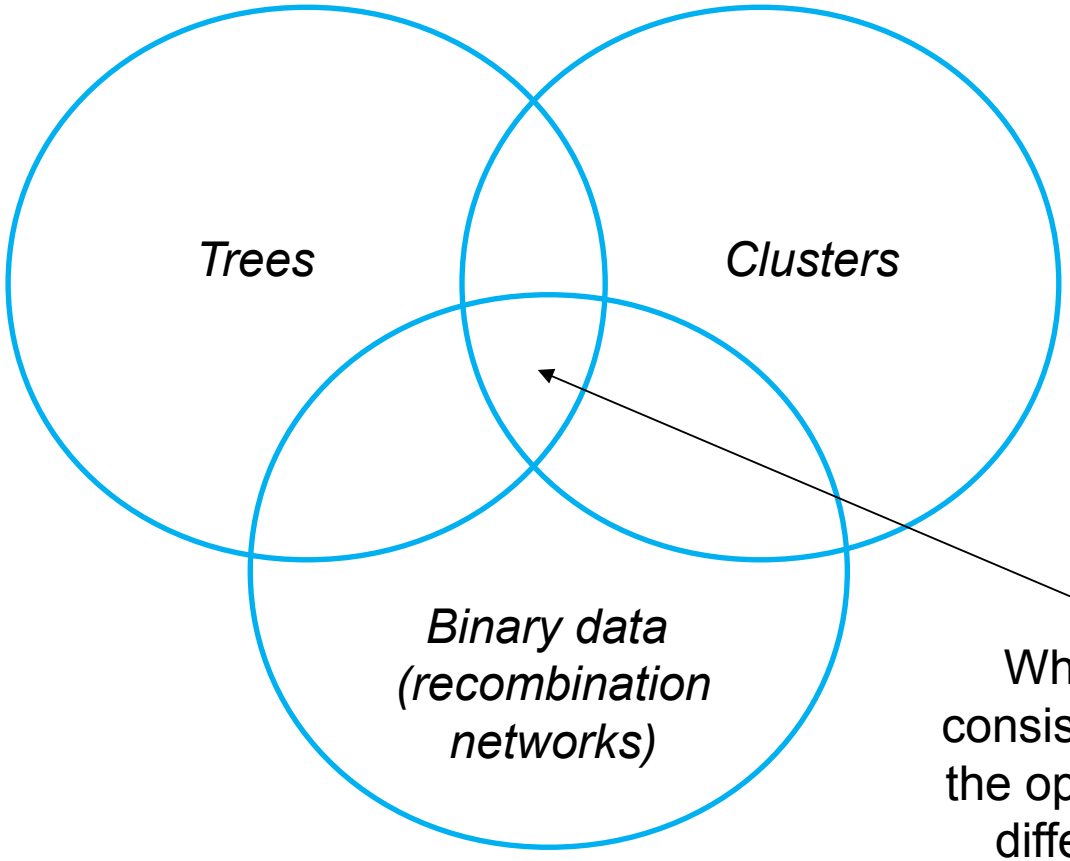


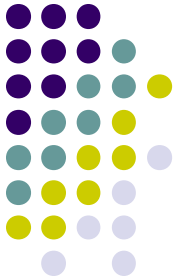
Figure 5. The level-1 network on the right with a single reticulation represents the union of the clusters (and triplets) obtained from the three trees on the left. However, any network that displays all three trees will have at least two reticulations and have level at least two.

Case study 3: combining softwired clusters into a single network



When the input consists of two trees, the optima under the different models become **equal**

Where does the future lie? 1: Unification



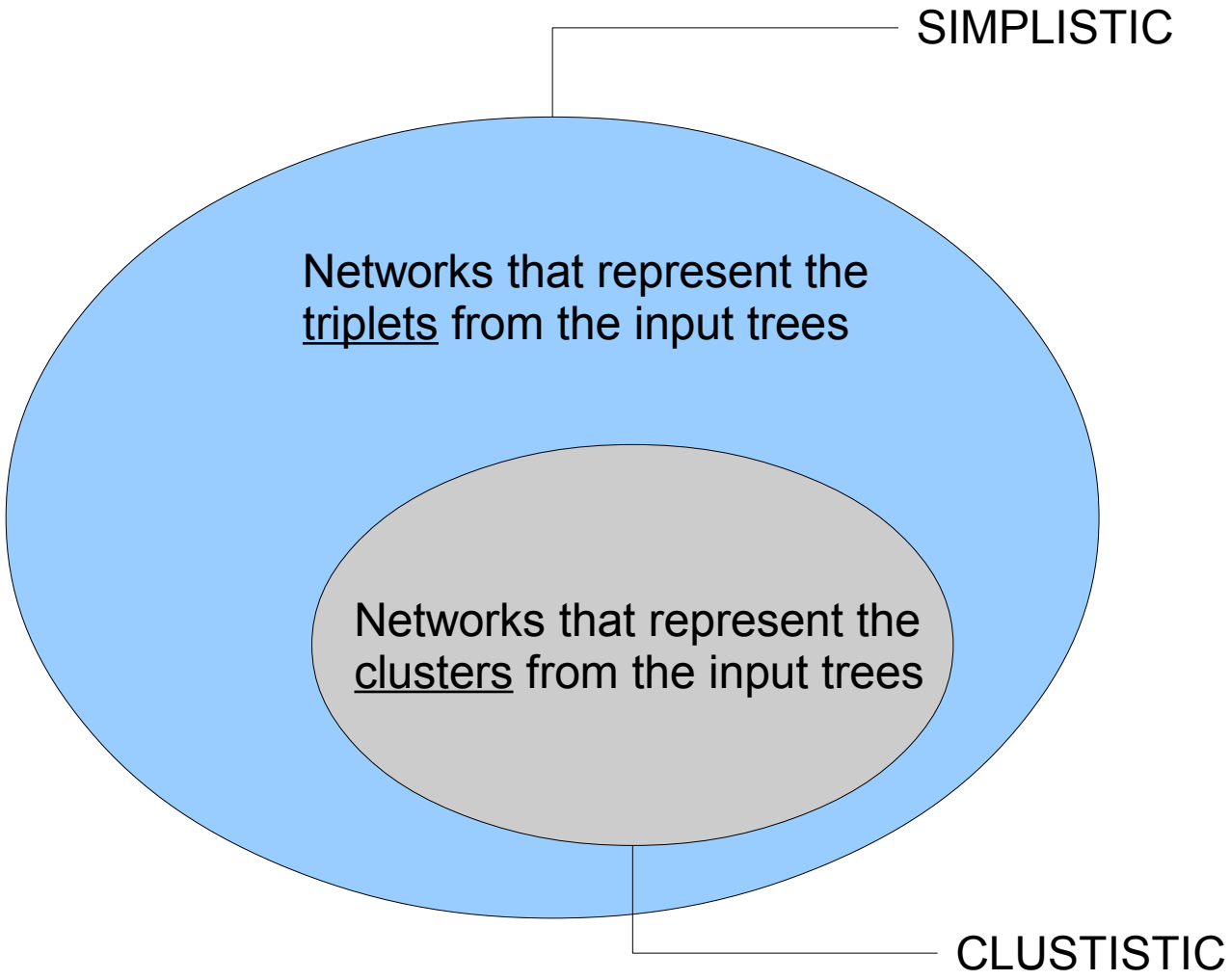
· We have seen three different techniques for constructing phylogenetic networks.

· All models suffer from hardness. Different groups tend to work on different models, and the groups have responded to the hardness in **different ways**.

· (Computational) lower and upper bounds	binary data
· Maximum Acyclic Agreement Forest	trees
· Fixed upper bound on the number of reticulations	clusters

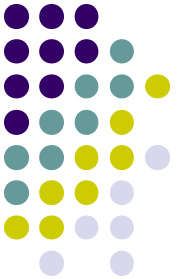
· Using insights gained from model can lead to **deeper structural insights** in another, with the case of two trees being an extreme example.

· "When two trees go to war" (Van Iersel and Kelk 2010)
· "On the elusiveness of clusters" (Kelk, Scornavacca and Van Iersel 2011)



CLUSTISTIC = SIMPLISTIC + filtering oracle

Where does the future lie? 2: Deepening

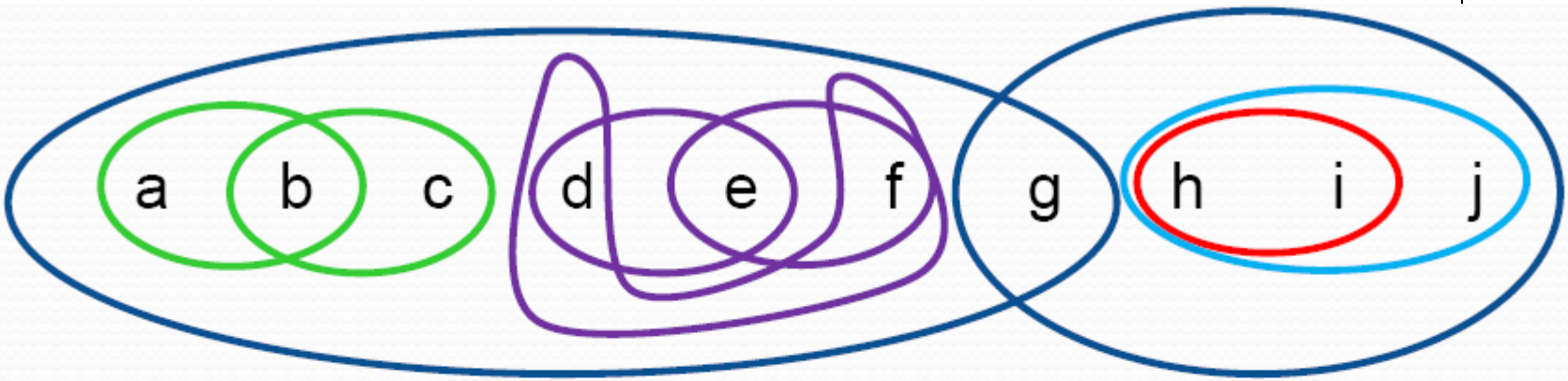


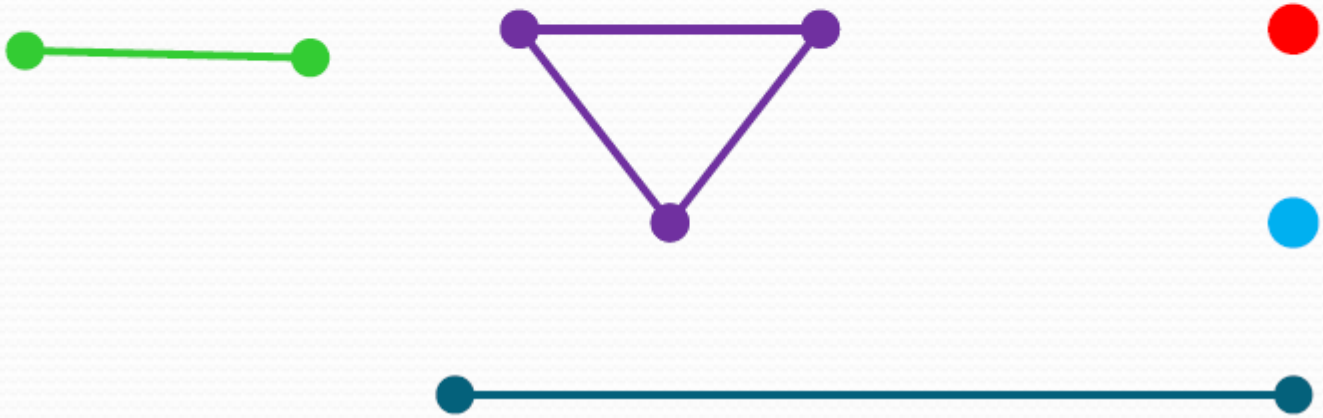
· All these different models produce **beautiful and novel combinatorial optimization problems**.

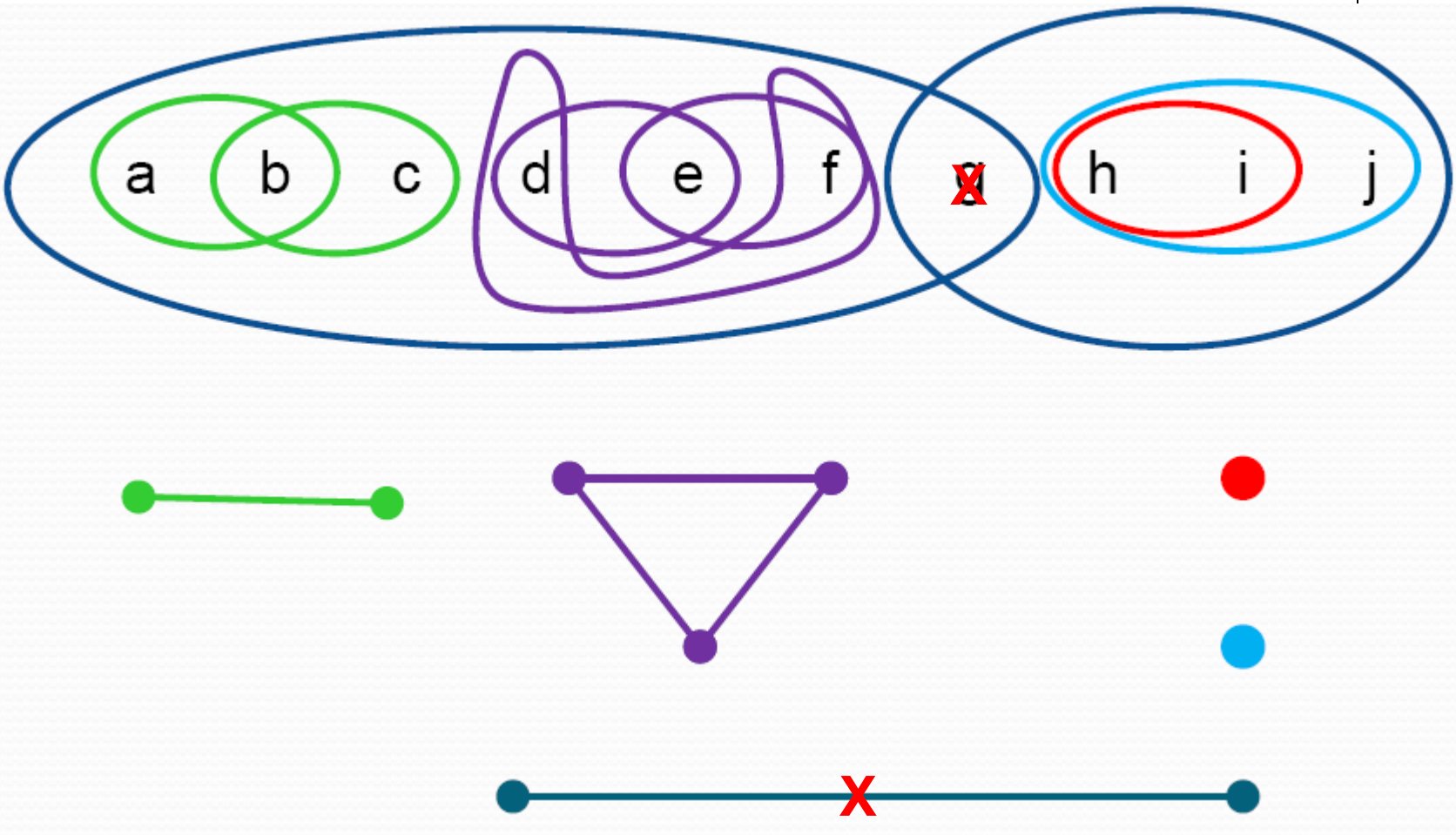
· For example, some of the softwired cluster problems can be thought of as a kind of “iterative, laminar” Hitting Set.

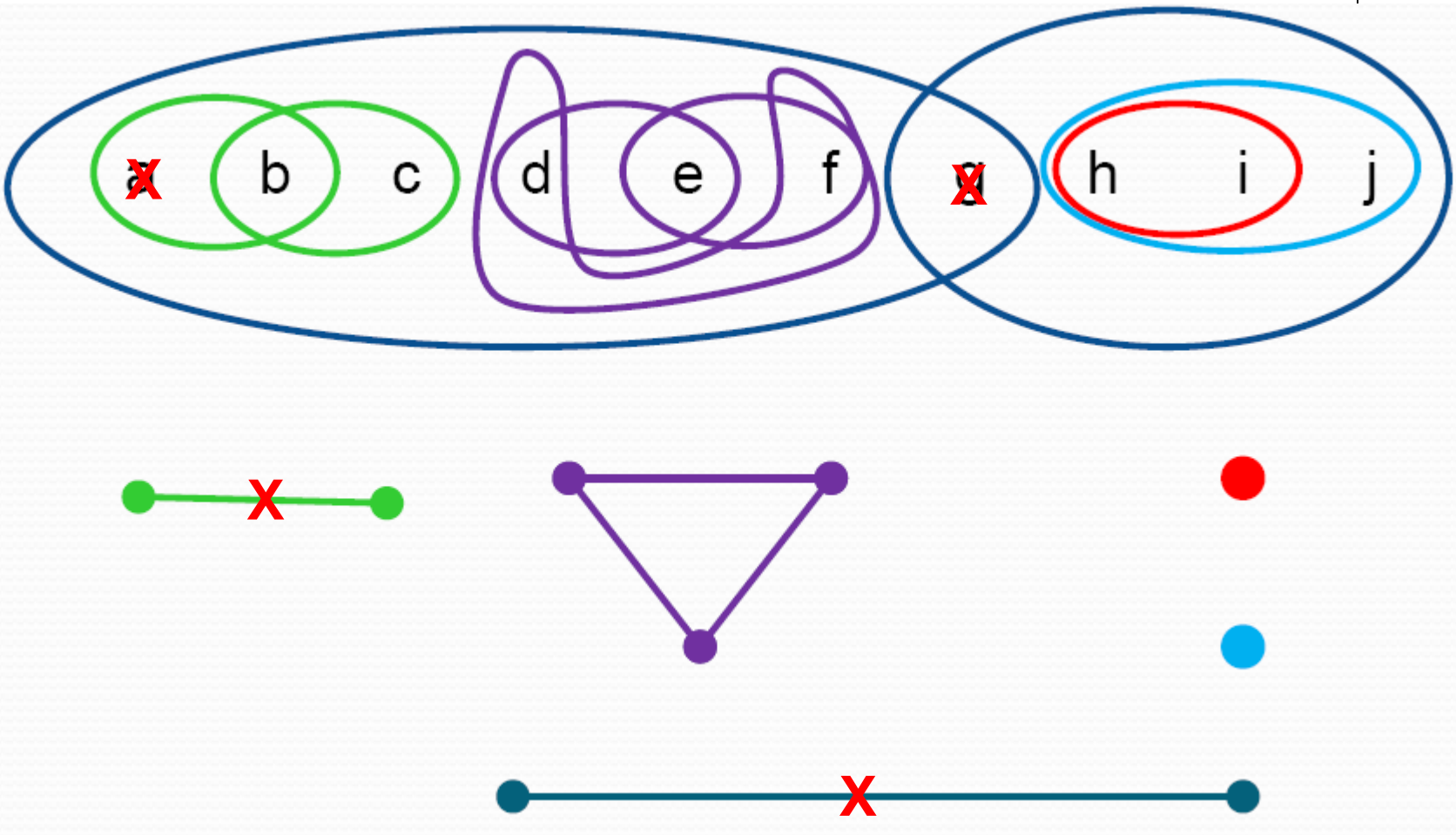
· However, these problems are not yet well-known to combinatorial optimization specialists, and advanced techniques from combinatorial optimization are not yet being used in the phylogenetic network literature.

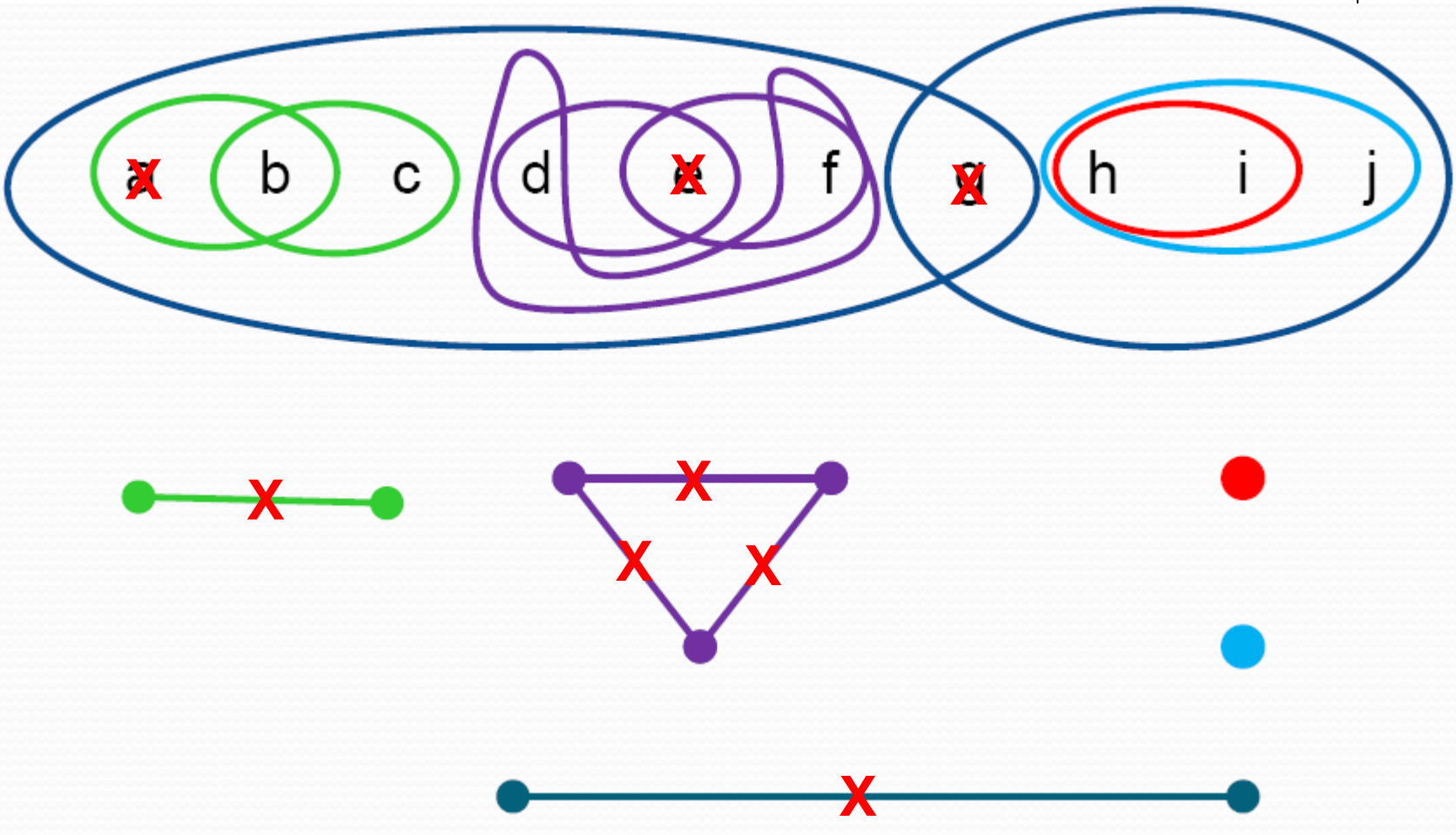
· There is an enormous amount to be gained by strengthening contact between these two groups.


a
b
c
d
e
f
g
h
i
j

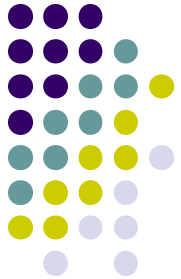



a
b
c
d
e
f
g
h
i
j


a
b
c
d
e
f
g
h
i
j


a
b
c
d
e
f
g
h
i
j

Where does the future lie? 3: Modeling



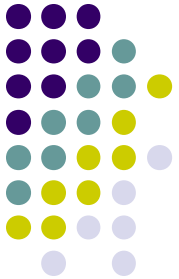
· Phylogenetic networks are a very good example of how attempting to model biological phenomena can lead to new mathematical problems.

· It is important to stay **close** to the biological problems that we are trying to solve. Biologists really do want to solve this problem so algorithms should be turned into easy-to-use **software**.

· There is still a **huge amount of uncertainty** regarding the best model for phylogenetic networks and what exactly we should be trying to “optimize”.

· Algorithmic specialists need to actively get involved in this modelling debate. A challenging balancing act!

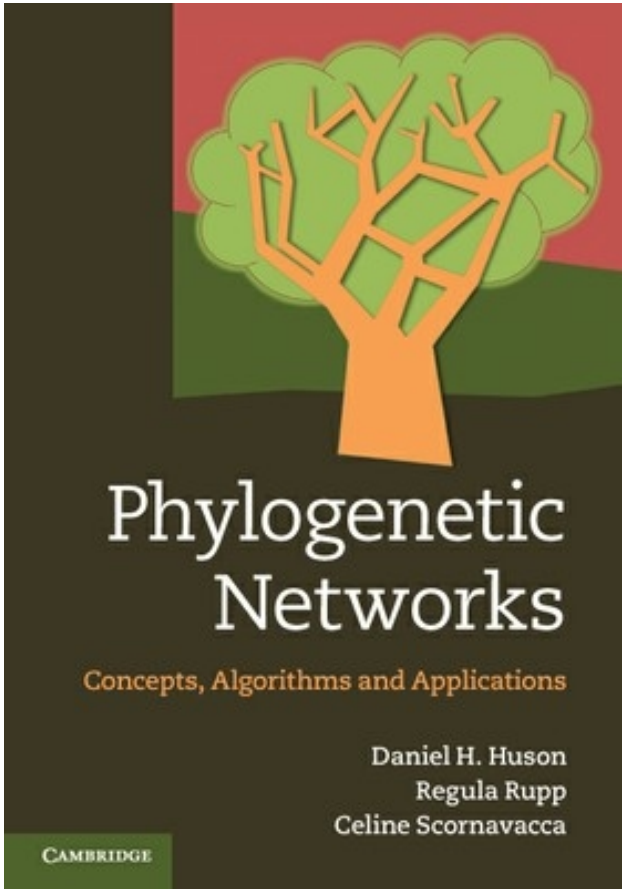
Finally…further reading



· Luay Nakhleh, *"Evolutionary phylogenetic networks: models and issues."* In: The Problem Solving Handbook for Computational Biology and Bioinformatics, L. Heath and N. Ramakrishnan (editors). Springer, 125-158, 2010.

· Daniel Huson, Regula Rupp and Celine Scornavacca, *"Phylogenetic Networks"*, Cambridge University Press.

· David Morrison, *"An introduction to phylogenetic networks",* Dystenium LLC, New York, to appear shortly.



Thanks for listening

PhD position available

· I currently have a PhD position available on this topic (algorithmic aspects of phylogenetic network construction). Focus is discrete maths, approximation algorithms, graph theory, fixed parameter tractability etc.

· Position is at the Department of Knowledge Engineering at the University of Maastricht. There will be collaboration with researchers based in Amsterdam (CWI/VU).

Het Vrijthof, Maastricht