Quartet-Based Methods to Construct Phylogenetic Networks

Stefan Grünewald

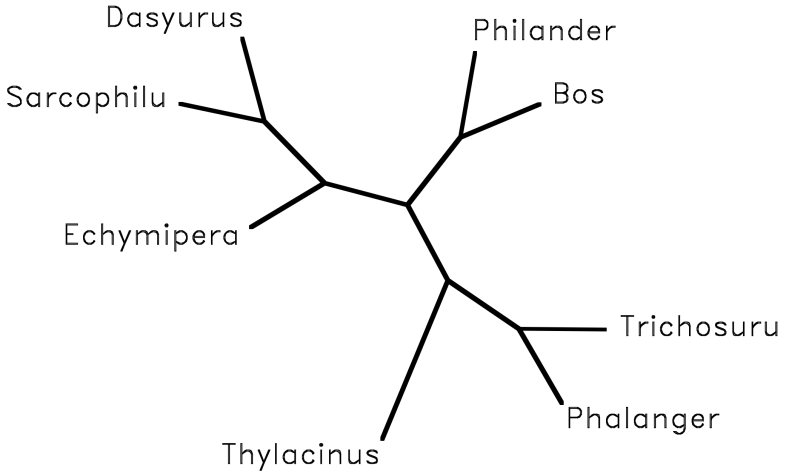
CAS-MPG Partner Institute for Computational Biology, Shanghai

MPI for Mathematics in the Sciences, Leipzig

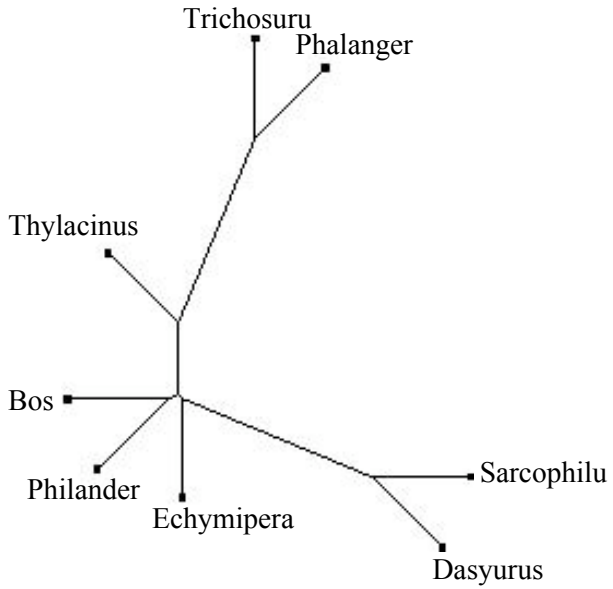
Splits in Trees

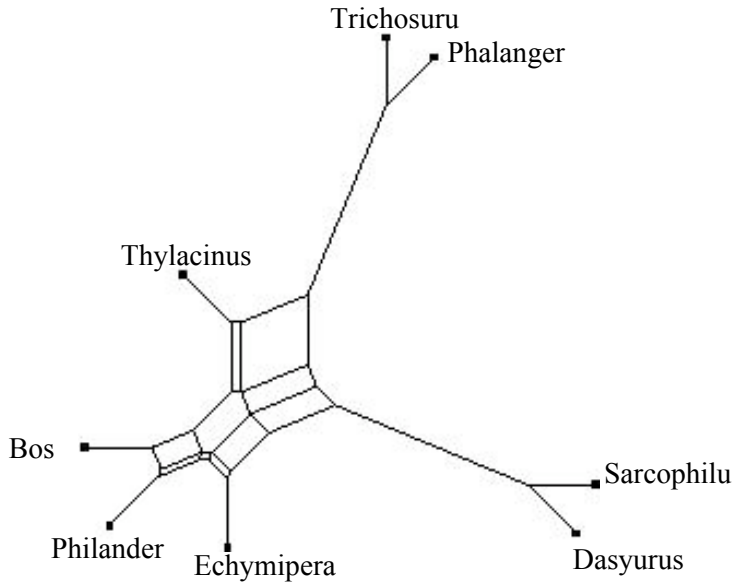
Every edge of a phylogenetic tree defines a split of the taxa set into two sets.

A list of all splits contains the full information of the tree.



Phylogenetic Trees and Networks



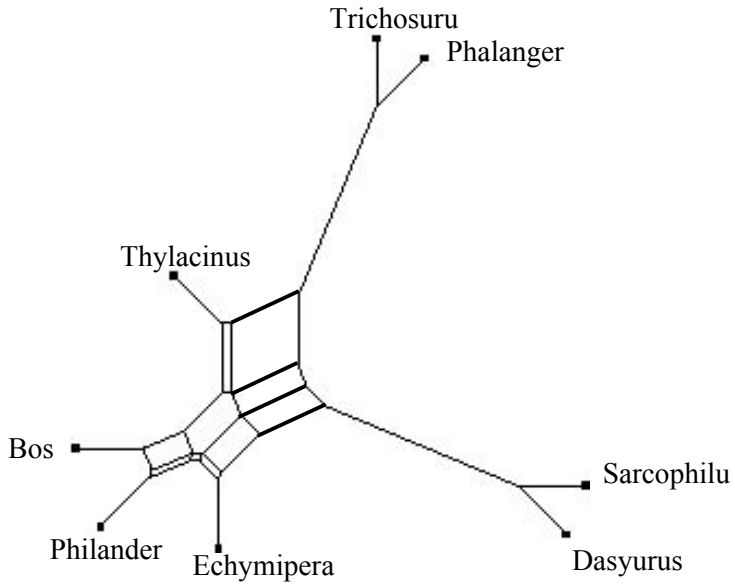


Why Networks?

Phylogenetic networks are used to visualize

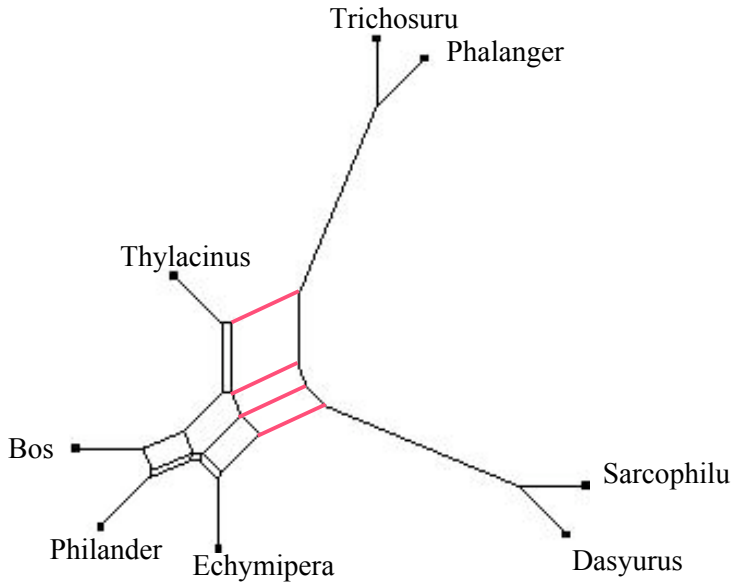
- reticulate evolution, and
- uncertainty in the data (if there is a true tree but there is support for conflicting splits).

Splits in Phylogenetic Networks



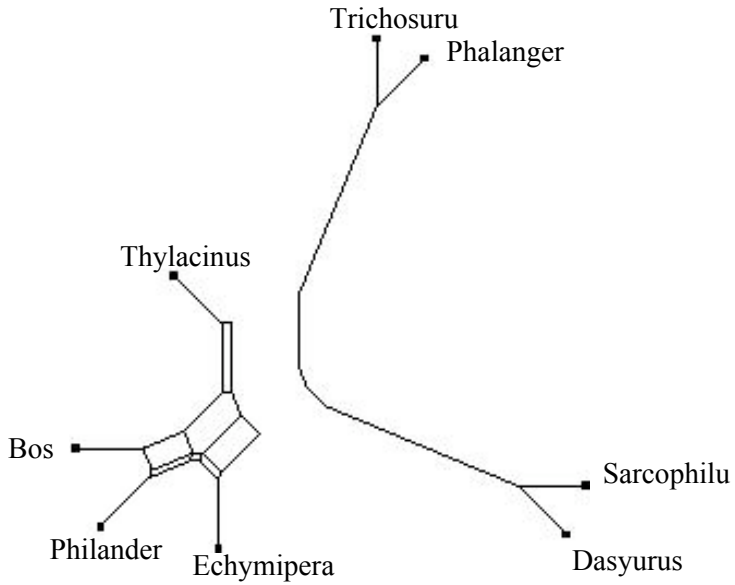
Splits in Phylogenetic Networks

Every edge is contained in a set of parallel edges of equal length. Removing them cuts the network into two components.



Splits in Phylogenetic Networks

Every edge is contained in a set of parallel edges of equal length. Removing them cuts the network into two components.

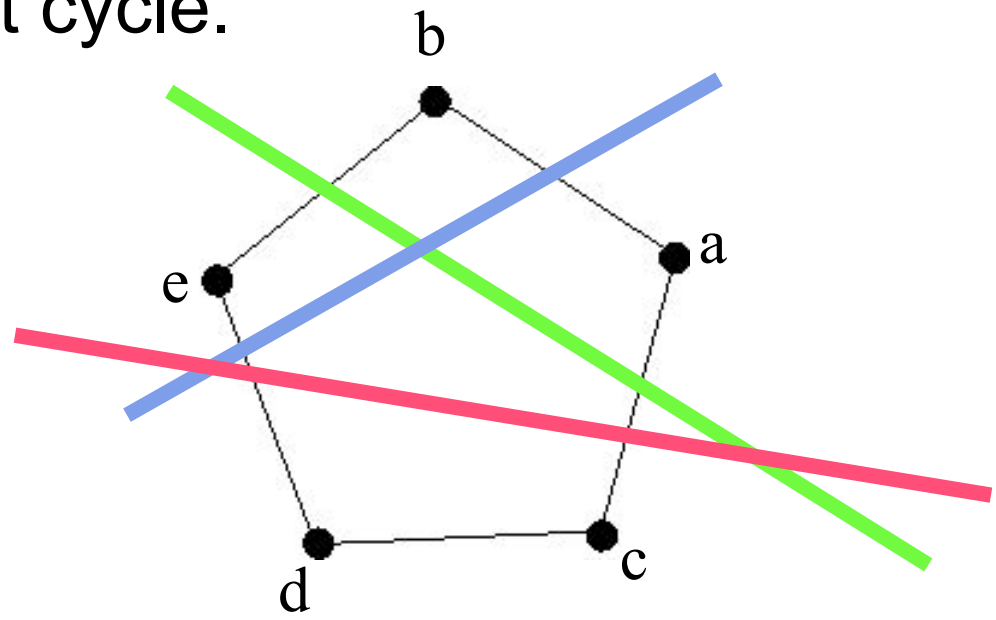


Why restricting to special classes of split systems?

- For n taxa, there are $2^{n-1}-1$ possible splits so the space of all split systems has dimension $2^{n-1}-1$.
- The space of all distance function has dimension $0.5(n-1)n$.
- Weakly compatible split systems can contain up to $0.5(n-1)n$ splits.

Circular Split Systems

A split system with taxa set X is *circular* if the elements of X can be arranged on a cycle such that every split can be obtained by cutting two edges of that cycle.



Circular Split Systems

- Circular split systems, together with a non-negative weight associated to each split, can be visualized by phylogenetic networks with some nice properties (planar, all taxa in the outer face).

- Weighted split systems that correspond to phylogenetic trees are circular.

NeighborNet

- NeighborNet (Bryant, Moulton 2004) is a successful method that constructs weighted circular split systems from distance data in two steps:
  - Constructing a cyclic ordering of the taxa
  - Associating weights to all splits that agree with that cyclic ordering

Quartets

- A *quartet* is a split of four taxa into two pairs.
- For four taxa *a,b,c,d*, there are three possible quartets *ab*|*cd*, *ac*|*bd*, *ad*|*bc*.
- Trees, circular split systems, etc. can be reconstructed from their quartets.
- A weighted split system defines weights for all possible quartets.

Distances or Quartets?

- Mihaescu, Levy, Pachter, *Why Neighbor-Joining Works,* Algorithmica (2009) 54: 1–24

"We show that the neighbor-joining algorithm is a robust quartet method for constructing trees from distances."

- The most widely used distance-based methods to construct phylogenetic networks also use quartet weights to select splits.
- Why not compute quartet weights directly from the raw data?

Distances or Quartets?

St. John, Warnow, Moret, Vawter (2003), *Performance study of phylogenetic methods: (unweighted) quartet methods and neighbor joining*, J. Algorithms **48**: 174–193

"Quartet-based methods are much less accurate than the simple and efficient method of neighbor-joining."

Quartet weights from sequences

- We have implemented 2 methods, one using statistical geometry, one using maximum likelihood.
- Run a sequence based tree reconstruction method on all sets of 4 taxa.

QNet

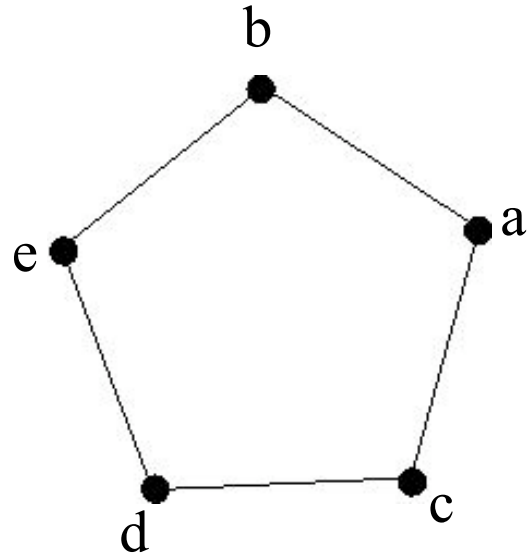
QNet (SG, Forslund, Moulton, Mol. Biol. Evol. 2007) is the quartet equivalent of NeighborNet.

Input: Non-negative weights for all quartets

Task: Find a weighted cyclic split system such that the induced quartet weights are as close to the input weights as possible.

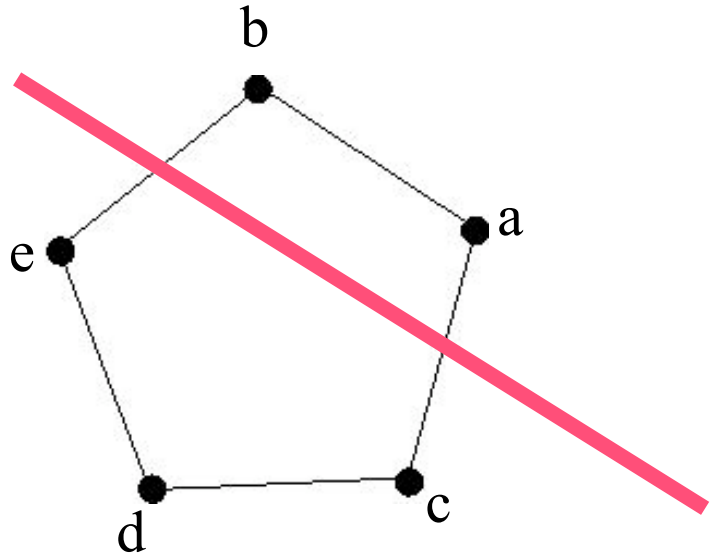
Quartets in a cyclic ordering

An X-quartet *ab*|*cd* is *displayed* by a cyclic ordering of X if there is a cyclic interval containing *a* and *b* but neither *c* nor *d*.

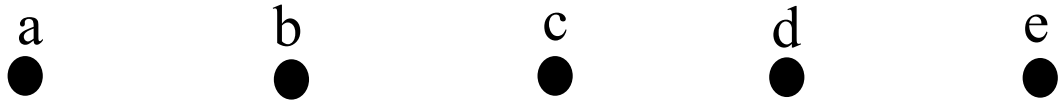


Quartets in a cyclic ordering

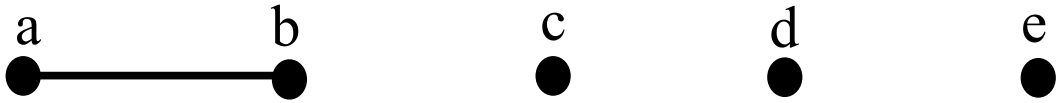
An X-quartet *ab*|*cd* is *displayed* by a cyclic ordering of X if there is a cyclic interval containing *a* and *b* but neither *c* nor *d*.



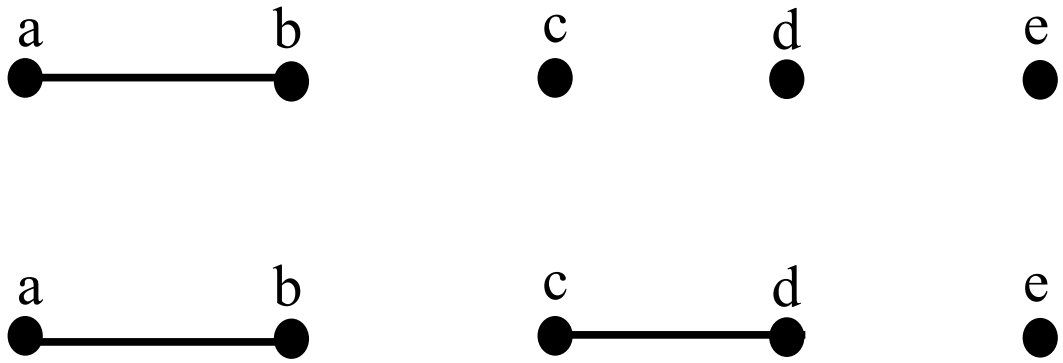
Constructing a cyclic ordering



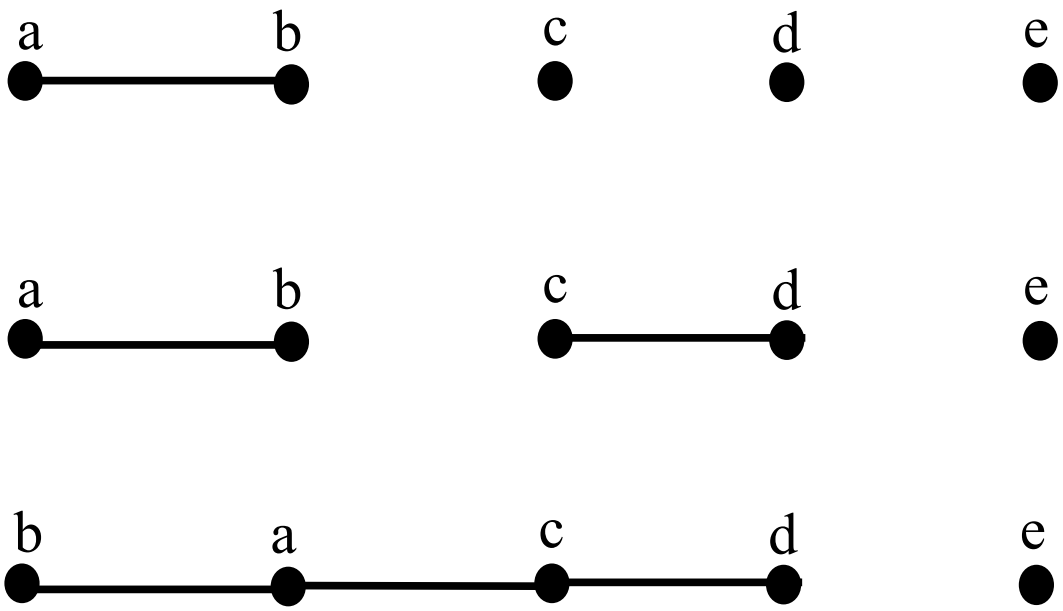
Constructing a cyclic ordering



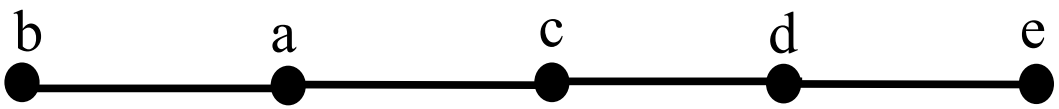
Constructing a cyclic ordering



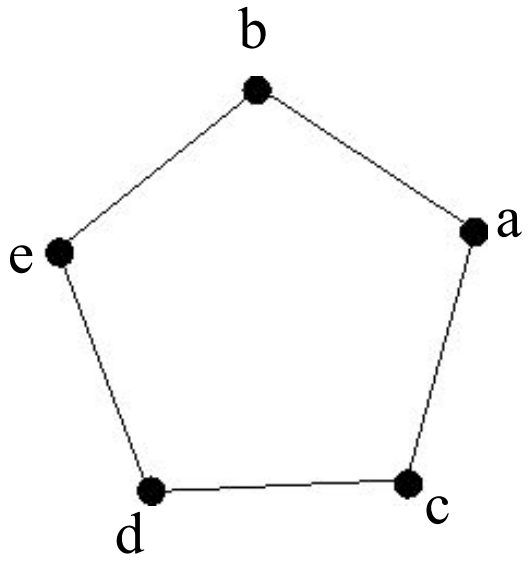
Constructing a cyclic ordering

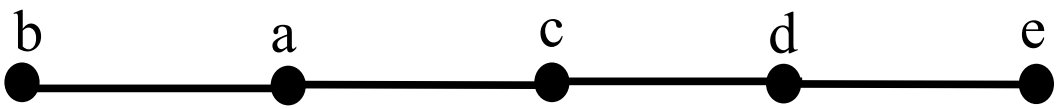


Constructing a cyclic ordering



Constructing a cyclic ordering





The first selection criterion

Let $P_1,...,P_k$ be the paths and let

$$w(P_iP_j \mid P_lP_m) = \sum_{x_r \in P_r} \frac{w(x_ix_j \mid x_lx_m)}{\mid P_i \mid\mid P_j \mid\mid P_l \mid\mid P_m \mid}$$

Choose two paths P_s,P_t such that

$\sum_{\mid\{i,j,s,t\}\mid=4} w(P_iP_j \mid P_sP_t)$ is maximal.

The second selection criterion

- Choose two end vertices u in P_s and v in P_t such that the sum of the weights of all *new* quartets obtained by joining u and v is maximal.

- New quartets: $bc|ad,\ bv|ac,\ \ldots$

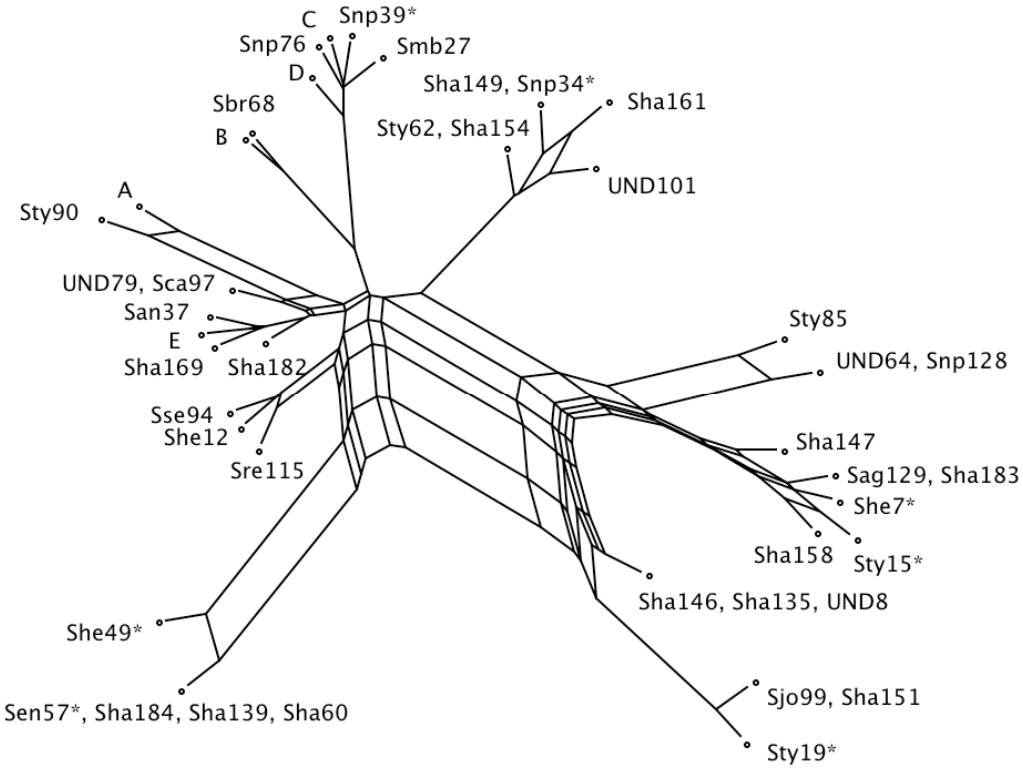
Split weights

After constructing a cyclic ordering, we apply non-negative least squares to find the split weights such that the quartet weights induced by the weighted split system and the input quartet weights are as similar as possible.

Consistency

Theorem (SG, Moulton, Spillner, 2009):
If the input quartet weights correspond to a weighted circular split system then QNet reconstructs it.

QNet


C
Snp39*
Snp76
Smb27
D
Sha149, Snp34*
Sha161
Sbr68
Sty62, Sha154
B
UND101
A
Sty90
UND79, Sca97
San37
E
Sty85
UND64, Snp128
Sha169
Sha182
Sse94
Sha147
She12
Sag129, Sha183
Sre115
She7*
Sha158
Sty15*
Sha146, Sha135, UND8
She49*
Sjo99, Sha151
Sen57*, Sha184, Sha139, Sha60
Sty19*

Supertrees

- Given a collection of phylogenetic trees with overlapping but non-identical taxa sets (typical example: gene trees);
- Find one big tree on the union of all taxa sets that contains as much of the input information as possible.
- If there are conflicting signals in the input, then only the strongest ones can be displayed.

Supernetworks

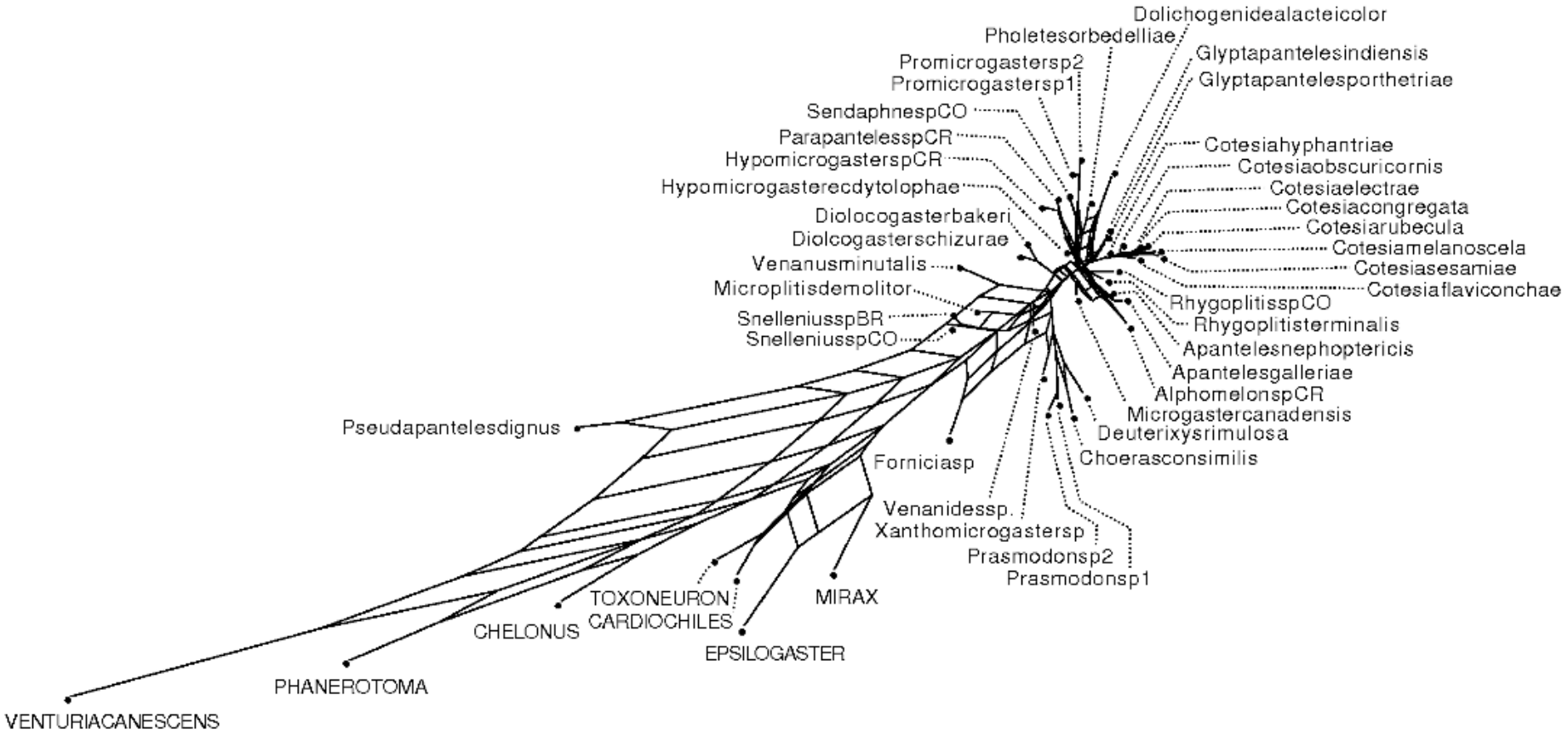
- In order to get a more complete overview of the signals in the input trees, one can construct a network rather than a tree.
- There have been two methods to construct supernets, Z-closure and Q-imputation.
- We used QNet to develop a third one which we call SuperQ.

SG, A. Spillner, K. Forslund, V. Moulton *Constructing Phylogenetic Supernetworks from Quartets,* WABI 2008.

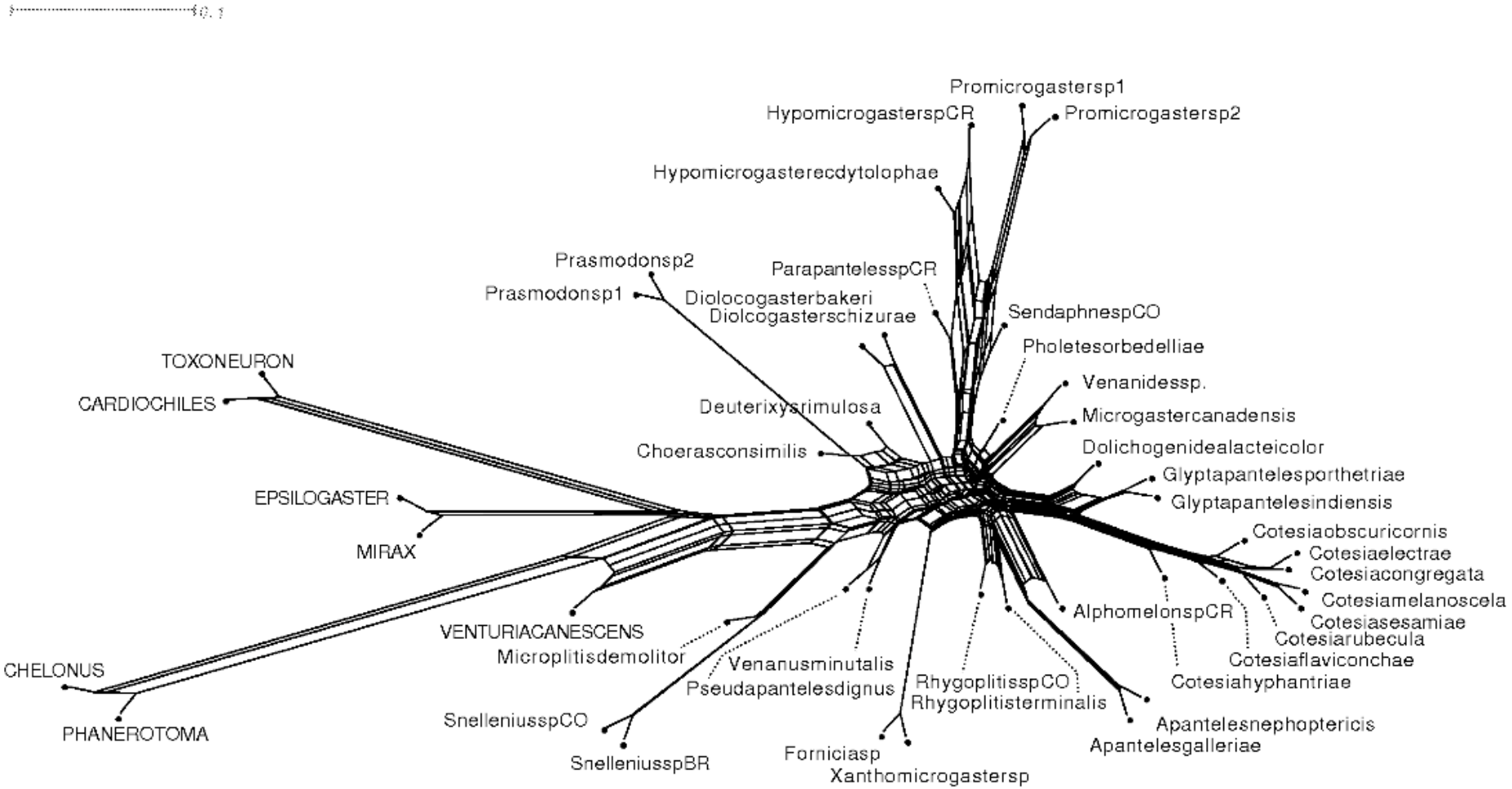
Using QNet for supernets

- First we chop the input trees into quartets.
- Then we compute the average quartet weight for every possible quartet.
- Finally, we construct the QNet.

The Z-closure network on 45 wasp species



The SuperQ network for the wasp data set



How many splits?

- NNet and QNet tend to produce too many splits.
- Split decomposition tends to produce too few splits.
- It would be nice to have a canonical method where the number of splits is just right.

Split decomposition for quartets

We are working on a quartet version of split decomposition that can reconstruct more general than weakly compatible split systems.

D-splits

- Given a distance function *d* on X, a split of $X=\{1,\ldots,n\}$ into the two disjoint subsets *A* and *X*-*A* is a *d-split* if and only if for all $i,j \in A$ and $k,l \in X\text{-}A$, one has

$$d(i,j)+d(k,l) < \max\{d(i,k)+d(j,l),\ d(i,l)+d(j,k)\}$$

The isolation index

- For every d-split $S=A|X\text{-}A$, we define the isolation index α_S of S by

$$\alpha_S := \min_{i,j \in A,\ k,l \in X-A} \{\max\{d(i,k)+d(j,l),\ d(i,l)+d(j,k)\} - d(i,j)+d(k,l)\}$$

- The isolation index is the weight of the split S.

Properties of split decomposition

- The dissimilarity function is transformed into a quartet weight function where at least one of the 3 quartets per quadruple gets weight 0.
- If the input is a metric, then the distance induced by the output split system is never larger than the input distance.
- The method is canonical (order independent, no tie-breaking, no parameters).

Our approach

- We start with weighted quartets, all 3 quartets of a quadruple can be positive.
- If the input quartet weights have a certain property (corresponding to the triangle inequality), then the quartet weights induced by the output are not larger than the input weights.
- The method is canonical (order independent, no tie-breaking, no parameters).

What have we done?

- We have slightly reformulated the split decomposition method (and shown equivalence).
- This new formulation can be generalized from distances (considered as weighted (1,1)-splits) to other (k,k)-splits as the input.
- We have implemented the method and we are running experiments.

Thank you!