Supertree Estimation

Tandy Warnow

February 26, 2017

Supertree Estimation
Tandy Warnow

Today's material:

- Review of Tree Compatibility (Chapter 3)
- Intro to supertree methods (Chapter 7)

Species tree estimation

Multiple challenges:

1. NP-hard problems on big datasets
2. Heterogeneity (different trees for different parts of genome)

Supertree estimation addresses the first problem, but the second problem is more challenging!

Supertree estimation

- Input: set $\mathcal{T}$ of trees on subsets of S, species set
- Output: tree T on full set S, optimizing some criterion

Supertree optimization problems

- Find T minimizing the Robinson-Foulds distance to the trees in $\mathcal{T}$
- Find T minimizing the quartet distance to the trees in $\mathcal{T}$
- Represent every tree in $\mathcal{T}$ with a $0,1$-matrix (one column for every edge), concatenate the matrices, and then solve maximum parsimony
- Represent every tree in $\mathcal{T}$ with a $0,1$-matrix (one column for every edge), concatenate the matrices, and then solve CFN maximum likelihood
- Compute a matrix M of average leaf-to-leaf distances between species, and find an additive matrix close to M (minimizing some criterion)

All these problems are NP-hard.

Tree Compatibility

A set $\mathcal{T}$ of trees is said to be **compatible** if there is a supertree that induces each tree in $\mathcal{T}$.

Tree Compatibility problem: are the trees in $\mathcal{T}$ compatible?

- If all the trees are rooted, then the problem can be solved using Aho, Sagiv, Szymanski, and Ullman (Section 3.3 from textbook) in polynomial time.
- If the trees are unrooted, the problem is NP-hard (Section 3.5 from textbook).

Because unrooted tree compatibility is NP-hard, it is trivial to show that most optimization problems for supertree construction from unrooted source trees are NP-hard.

But they are still hard for supertree construction from rooted source trees!

Popular Supertree "methods"

- MRP: Matrix Representation with Parsimony
- MRL: Matrix Representation with Likelihood
- Robinson-Foulds Supertrees

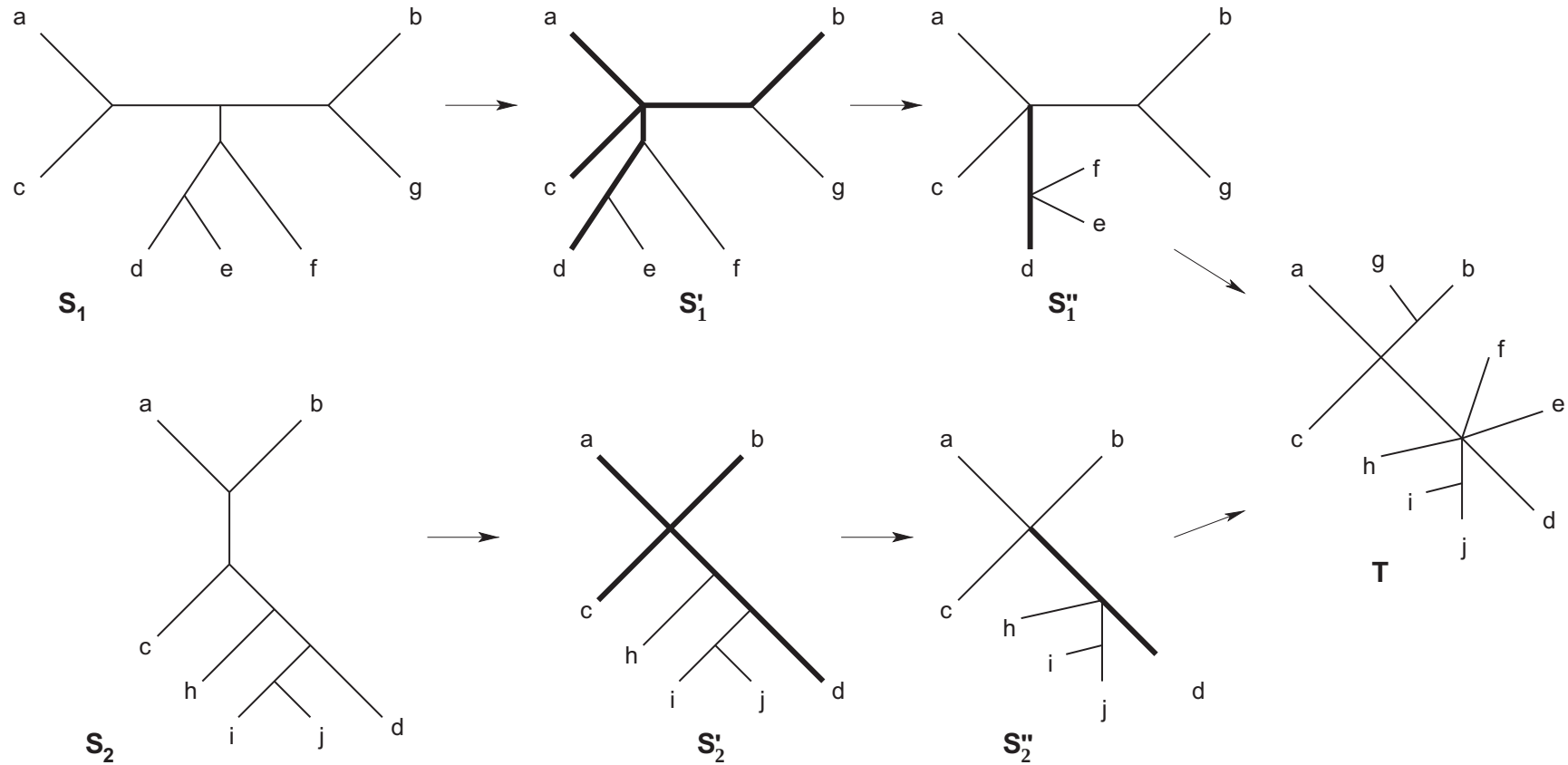
All of these are NP-hard problems, so heuristics are used to find "good" solutions.

SuperFine: a supertree method "booster"

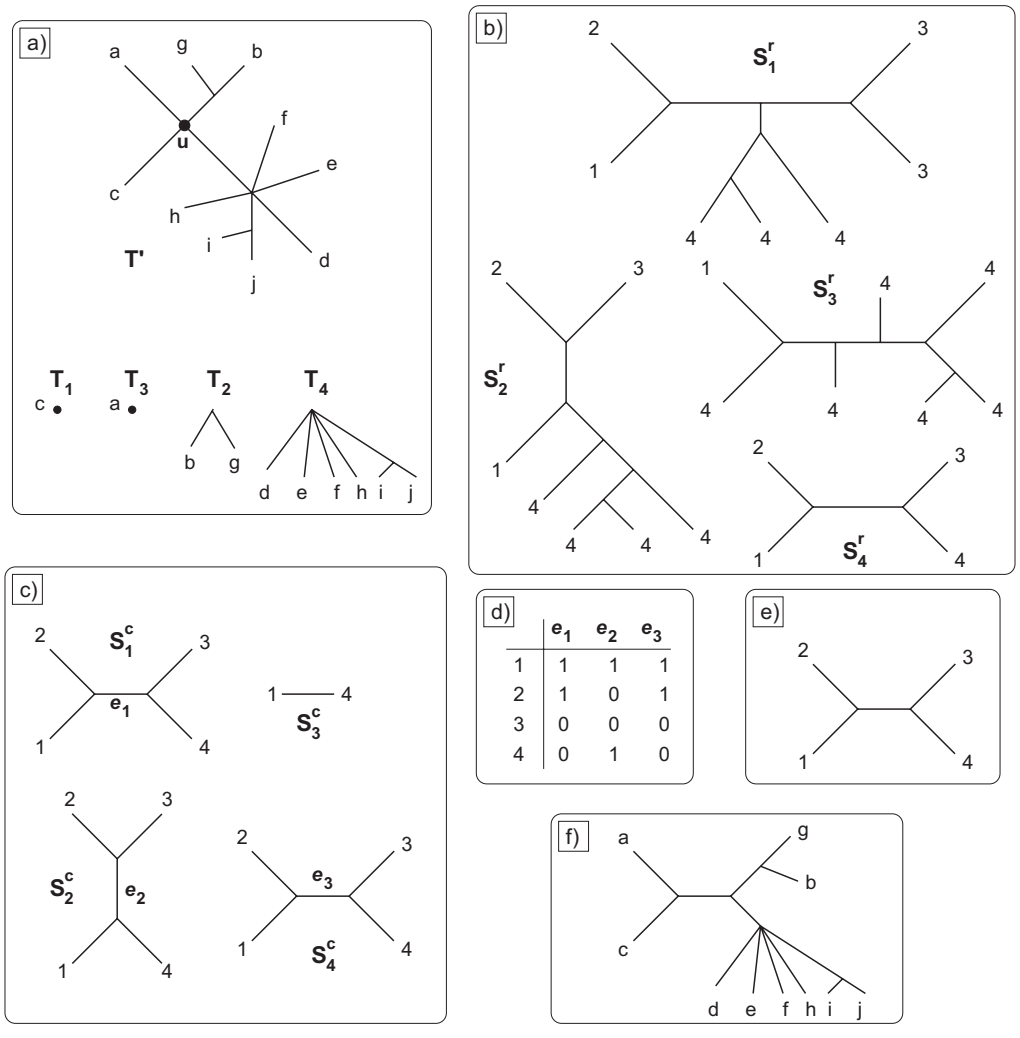
SuperFine (Swenson et al. 2012) is a meta-method for improving the speed and accuracy of supertree methods:

- Compute a constraint tree using the Strict Consensus Merger
- For each polytomy v (node of degree $d > 3$):
  - Compute an encoding of the source trees into trees with at most d leaves
  - Run preferred supertree method on the new source trees, obtaining $t(v)$
  - Refine polytomy v with the computed supertree $t(v)$

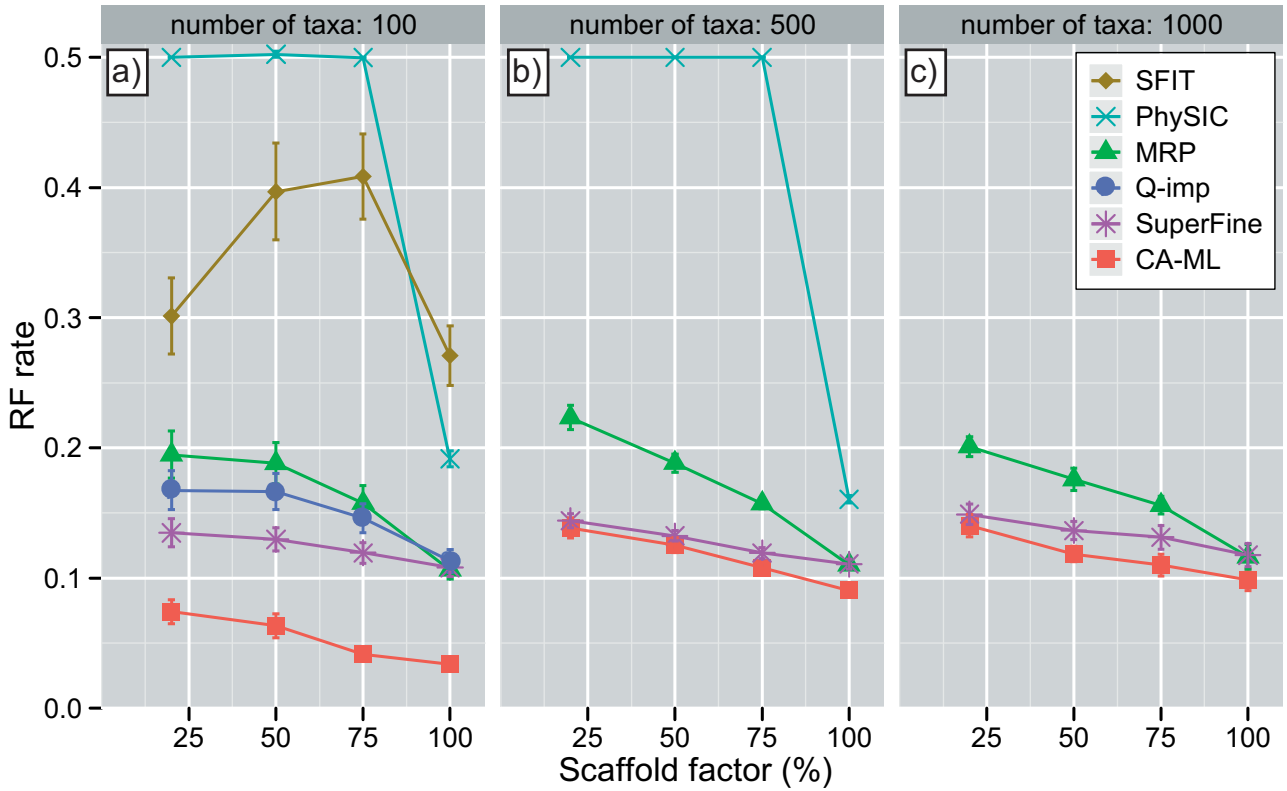
Phase 1: Strict Consensus Merger (SCM)



Phase 2: Refining the SCM tree using MRP



SuperFine+MRP vs MRP, CA-ML, and other methods



Robinson-Foulds Supertrees

Finding the supertree T that minimizes the Robinson-Foulds (RF) distance to the source trees is the Robinson-Foulds Supertree problem.

MulRF (Chaudhary et al., 2014) and PluMiST (Kupczok, 2011) are two methods for Robinson-Foulds Supertrees.

Robinson-Foulds Supertrees are (sort of) approximations to the Maximum Likelihood Supertree (Steel and Rodrigo, 2008) problem (see Bryant and Steel 2009 for more).

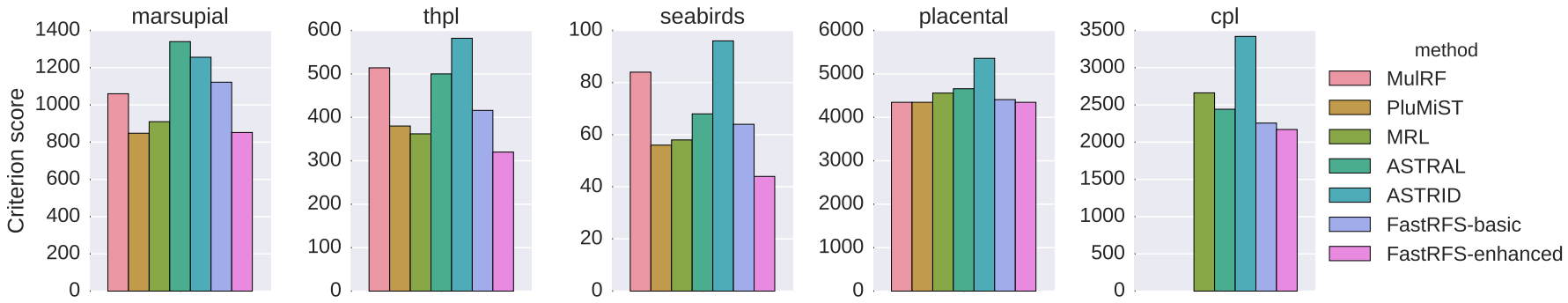
Constrained optimization

Bipartition-constrained Robinson-Foulds Supertree Problem:

- Input: Set $\mathcal{T}$ of source trees and set X of bipartitions on S
- Output: Tree T on S that draws its bipartition set $C(T)$ from X, and that minimizes the RF distance to $\mathcal{T}$ among all such supertrees

FastRFS (Vachaspati and Warnow, Bioinformatics 2016) solves this problem in polynomial time, using dynamic programming.

FastRFS criterion scores



The CPL dataset has 2228 species, and so represents the largest and most difficult dataset; MulRF and PluMiST could not complete on it.

Method	500	500	500	500
Scaffold %	20	50	75	100
# Replicates	8	10	10	10
ASTRAL	15.3	14.8	12.7	11.2
ASTRAL-enhanced	14.8	14.1	12.6	11.2
ASTRID	26.0	50.1	45.4	**10.5**
MRL	15.4	14.3	12.1	11.2
MulRF	46.9	40.3	27.4	12.6
PluMiST	35.4	29.5	22.4	10.9
FastRFS-basic	14.5	14.3	12.4	11.1
FastRFS-enhanced	**14.3**	**13.9**	**12.0**	10.8

Table : Average supertree topology estimation error on simulated datasets.

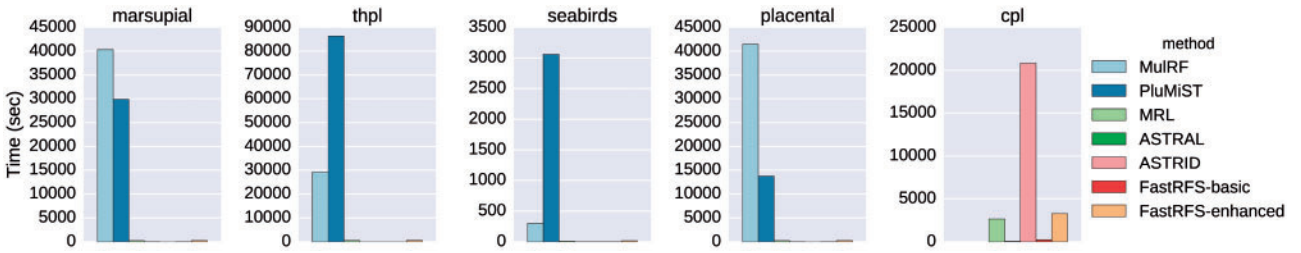


Fig. 2. Sequential running times (in seconds) on biological data of supertree methods. MulRF and PluMiST could not be run on the CPL dataset, due to its large size; hence no values are shown for those methods on that dataset

Summary about supertree methods

- Supertree methods are essential techniques for large-scale phylogeny estimation, in part because divide-and-conquer is necessary.
- New approaches provide good accuracy but are limited to relatively small datasets.
- SuperFine can help, but only when the constraint tree computed by the Strict Consensus Merger is not too unresolved.
- New supertree methods are needed!
- Potential directions: distance-based supertree methods, constrained approaches, others?