Graph-theoretic algorithms to improve phylogenomic analyses

Tandy Warnow and Pranjal Vachaspati
University of Illinois at Urbana-Champaign

AITF Project: CCF-1535977

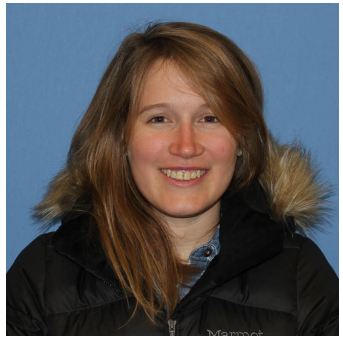
Tandy Warnow

Chandra Chekuri

Satish Rao

Pranjal Vachaspati

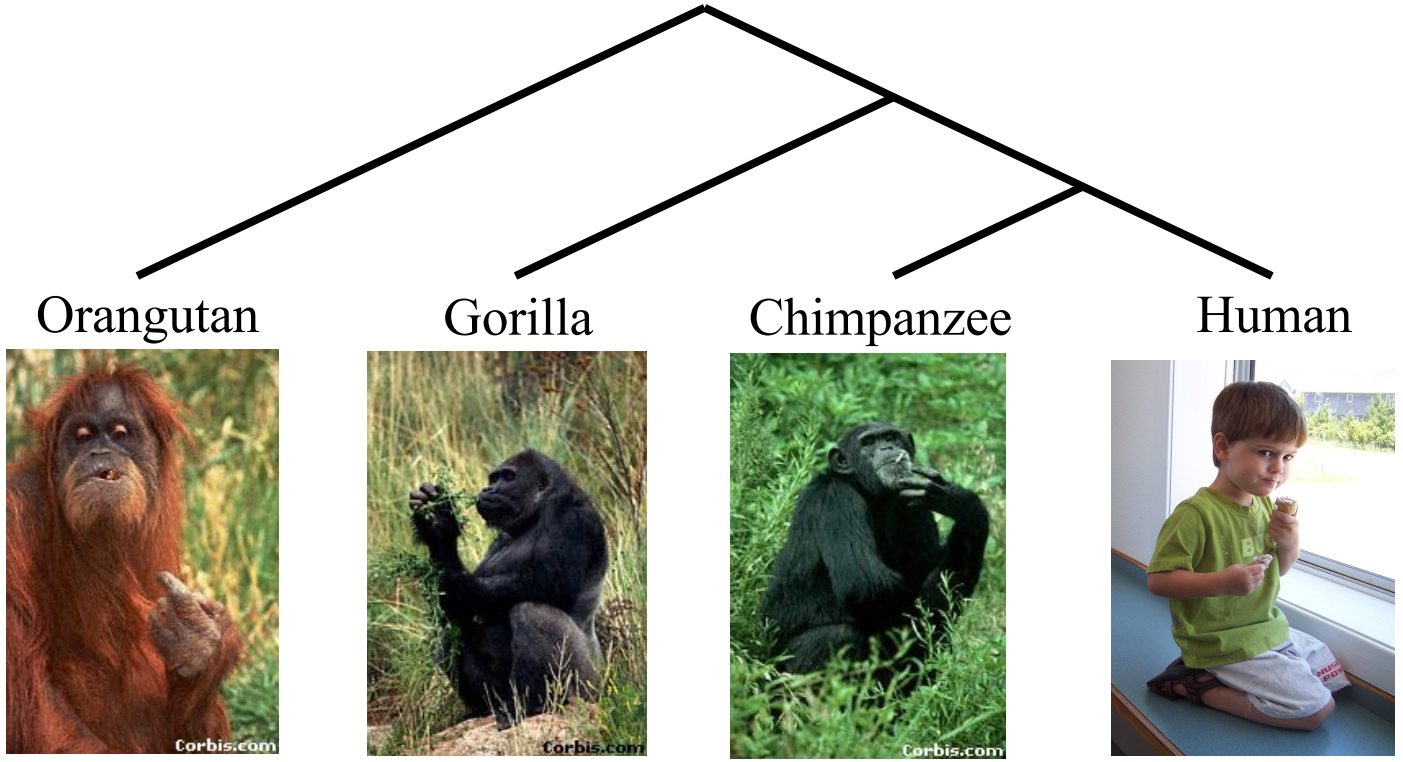
Sarah Christensen



Erin Molloy

Richard Zhang

Species Tree



From the Tree of the Life Website,
University of Arizona

Applications to Biology

- “Nothing in biology makes sense except in the light of evolution” – T. Dobhzhansky (1973)

- “Nothing in evolution makes sense except in the light of phylogeny” - The Society of Systematic Biologists

Evolution informs about everything in biology

- Big genome sequencing projects just produce data so what?
- Evolutionary history relates all organisms and genes, and helps us understand and predict
  - interactions between genes (genetic networks)
  - drug design
  - predicting functions of genes
  - influenza vaccine development
  - origins and spread of disease
  - origins and migrations of humans

Phylogenomic pipeline

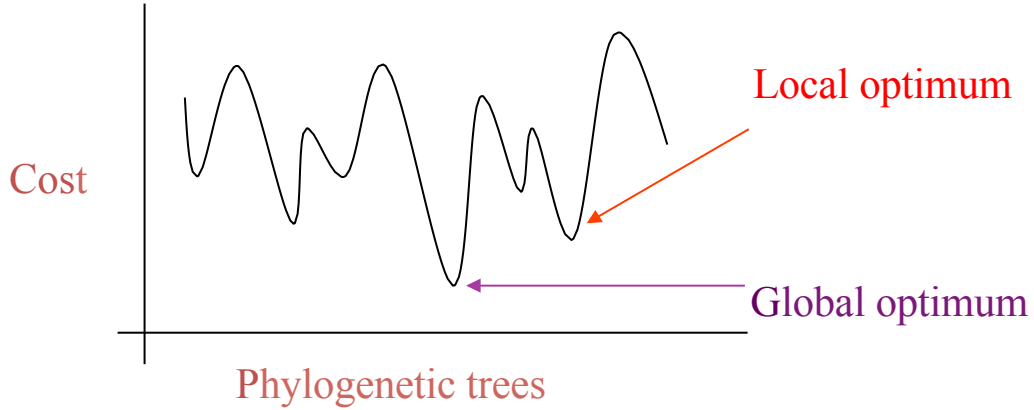
- Select taxon set and markers
- Gather and screen sequence data, possibly identify orthologs
- Compute multiple sequence alignments for each locus
- Compute species tree or network:
  - Compute gene trees on the alignments and combine the estimated gene trees, OR
  - Estimate a tree from a concatenation of the multiple sequence alignments
- Get statistical support on each branch (e.g., bootstrapping)
- Estimate dates on the nodes of the phylogeny
- Use species tree with branch support and dates to understand biology

Phylogenomic pipeline

- Select taxon set and markers
- Gather and screen sequence data, possibly identify orthologs
- Compute multiple sequence alignments for each locus
- Compute species tree or network:
  - Compute gene trees on the alignments and combine the estimated gene trees, OR
  - Estimate a tree from a concatenation of the multiple sequence alignments
- Get statistical support on each branch (e.g., bootstrapping)
- Estimate dates on the nodes of the phylogeny
- Use species tree with branch support and dates to understand biology

Phylogenetic reconstruction methods

1 Hill-climbing heuristics for hard optimization criteria (Maximum Parsimony and Maximum Likelihood)



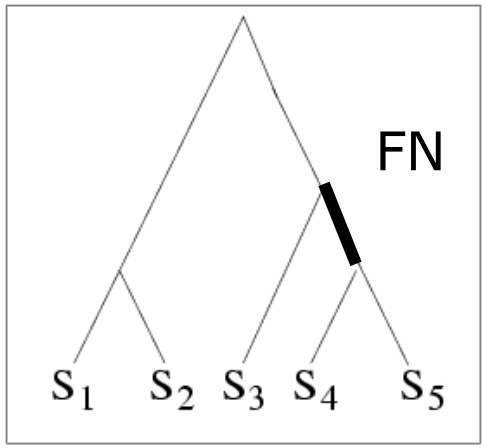
2 Polynomial time distance-based methods: Neighbor Joining, FastME, etc.

3. Bayesian methods

Performance criteria

- Running time
- Space
- Statistical performance issues (e.g., statistical consistency) with respect to a Markov model of evolution
- "Topological accuracy" with respect to the underlying ***true tree or true alignment,*** typically studied in simulation
- Accuracy with respect to a particular criterion (e.g. maximum likelihood score), on real data

Quantifying Error



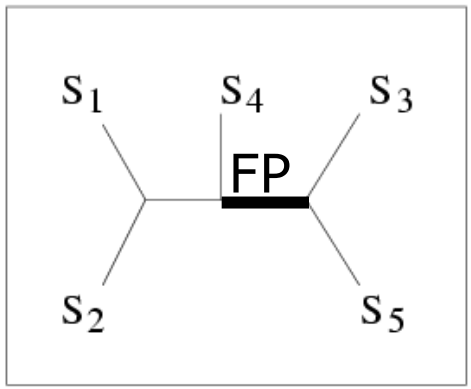
TRUE TREE

S_1	ACAATTAGAAC
S_2	ACCCTTAGAAC
S_3	ACCATTCCAAC
S_4	ACCAGACCAAC
S_5	ACCAGACCGGA

DNA SEQUENCES

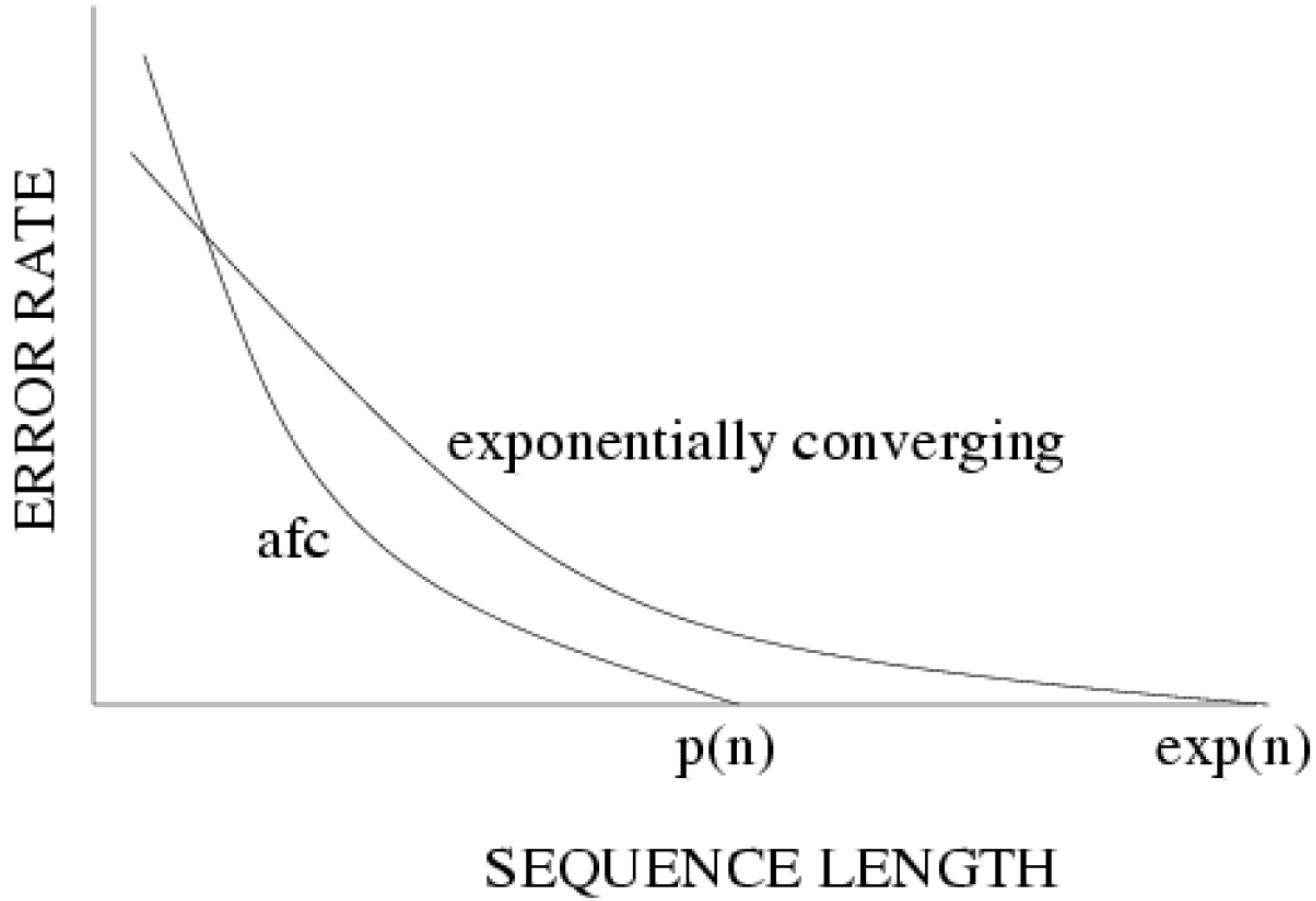
FN: false negative
(missing edge)
FP: false positive
(incorrect edge)

50% Robinson-Foulds error rate

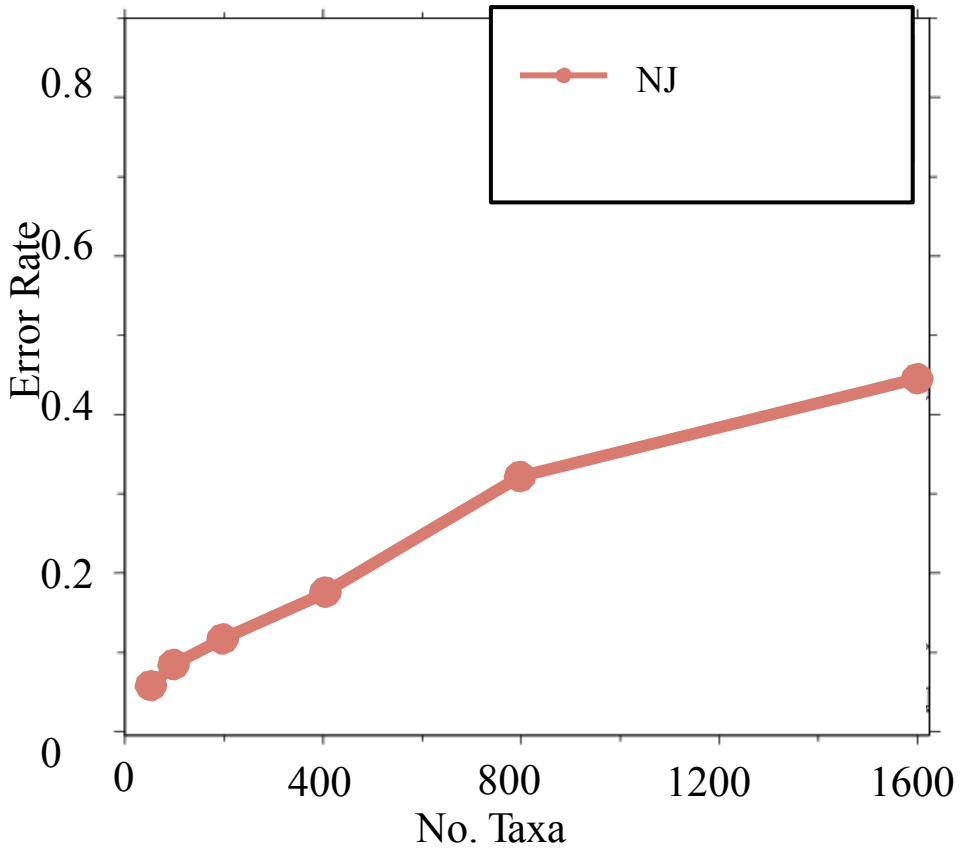


INFERRED TREE

Statistical consistency, exponential convergence, and absolute fast convergence (afc)

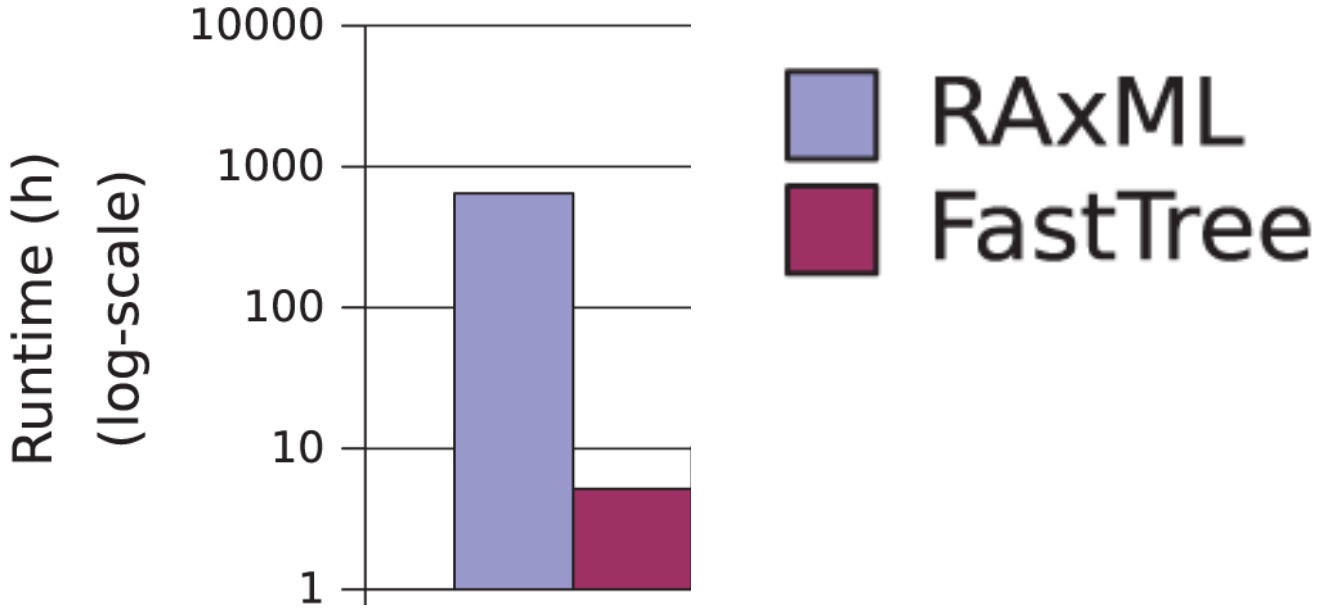


Neighbor joining has poor performance on large diameter trees *[Nakhleh et al. ISMB 2001]*



Theorem (Atteson):
Exponential sequence length requirement for Neighbor Joining!

RAxML is the “best” ML code – but it is very slow on large datasets



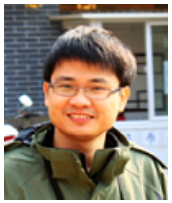
Analyses on biological dataset (16S.B.ALL) from Gutell Lab, with 27,643 sequences. Results shown the structural alignment, using three different ML methods.

Avian Phylogenomics Project

Erich Jarvis, HHMI

MTP Gilbert, Copenhagen

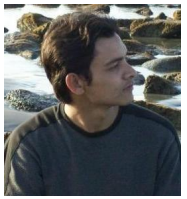
G Zhang, BGI



T. Warnow UIUC/UT-Austin

S. Mirarab UT-Austin

Md. S. Bayzid, UT-Austin



Plus many many other people…

- 48 species, whole genomes
- 14,000 genomic regions and "gene trees"

Science, December 2014 (Jarvis, Mirarab, et al., and Mirarab et al.)

Two main challenges

- Computationally intensive concatenation analysis: 200 CPU years
- Gene tree heterogeneity: needed new method (statistical binning)

1kp: Thousand Transcriptome Project

G. Ka-Shu Wong
U Alberta

J. Leebens-Mack
U Georgia

N. Wickett
Northwestern

N. Matasci
iPlant

T. Warnow,
UIUC

S. Mirarab,
UT-Austin

N. Nguyen,
UT-Austin

Plus many many other people…

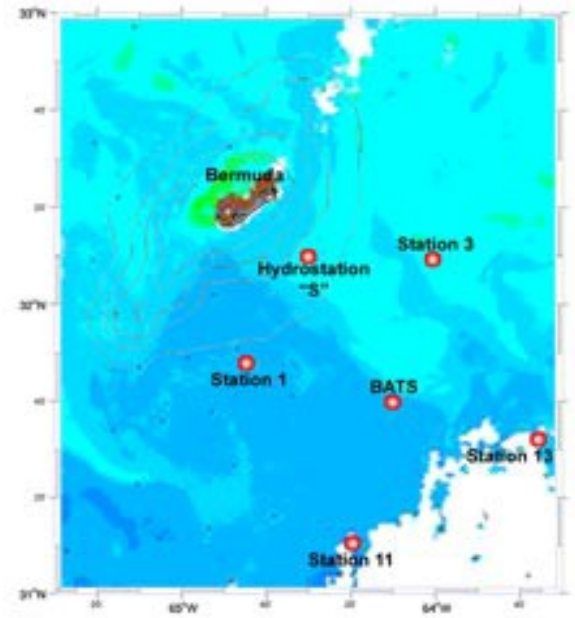
- Plant Tree of Life based on transcriptomes of ~1200 species
- More than 13,000 gene families (most not single copy)
- First paper: PNAS 2014 (~100 species and ~800 loci)

- First challenges: gene tree heterogeneity (new method: ASTRAL)
- Upcoming Challenges: alignments and trees on ~1200 species

Metagenomics:

Venter et al., Exploring the Sargasso Sea:

Scientists Discover One Million New Genes in Ocean Microbes

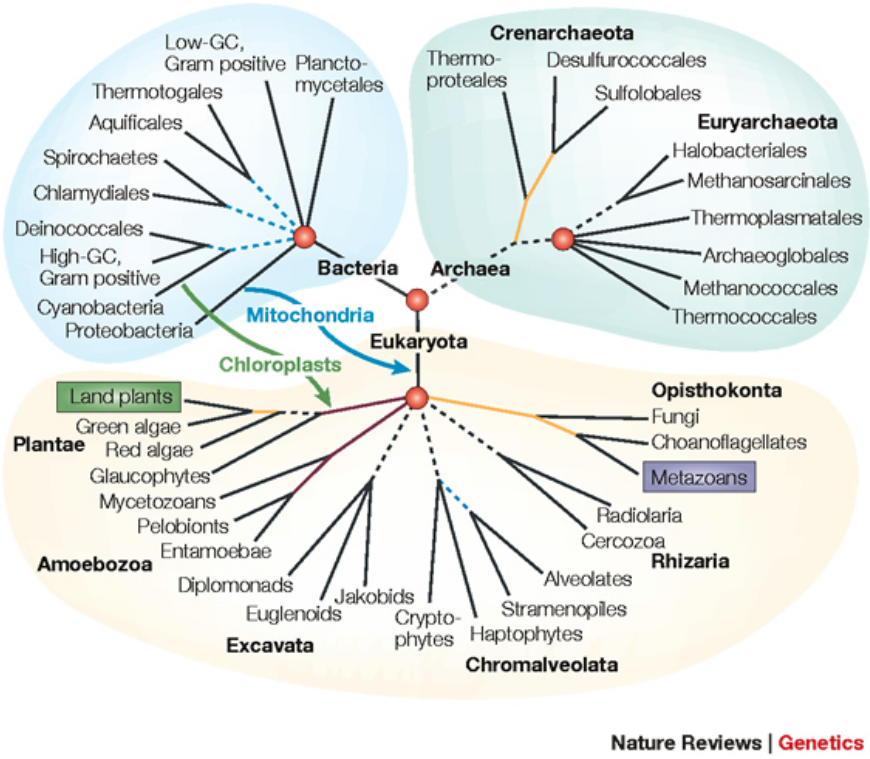


Two dimensions

- Number of species – not adequately addressed by any methods, and size also becomes a big issue (large alignments with >200Gb)

- Number of genes (resulting in very long sequences from combining sequence datasets) – gene tree heterogeneity requires new methods

Constructing the Tree of Life: Hard Computational Problems



NP-hard problems

Large datasets

- 1,000,000+ sequences
- thousands of genes

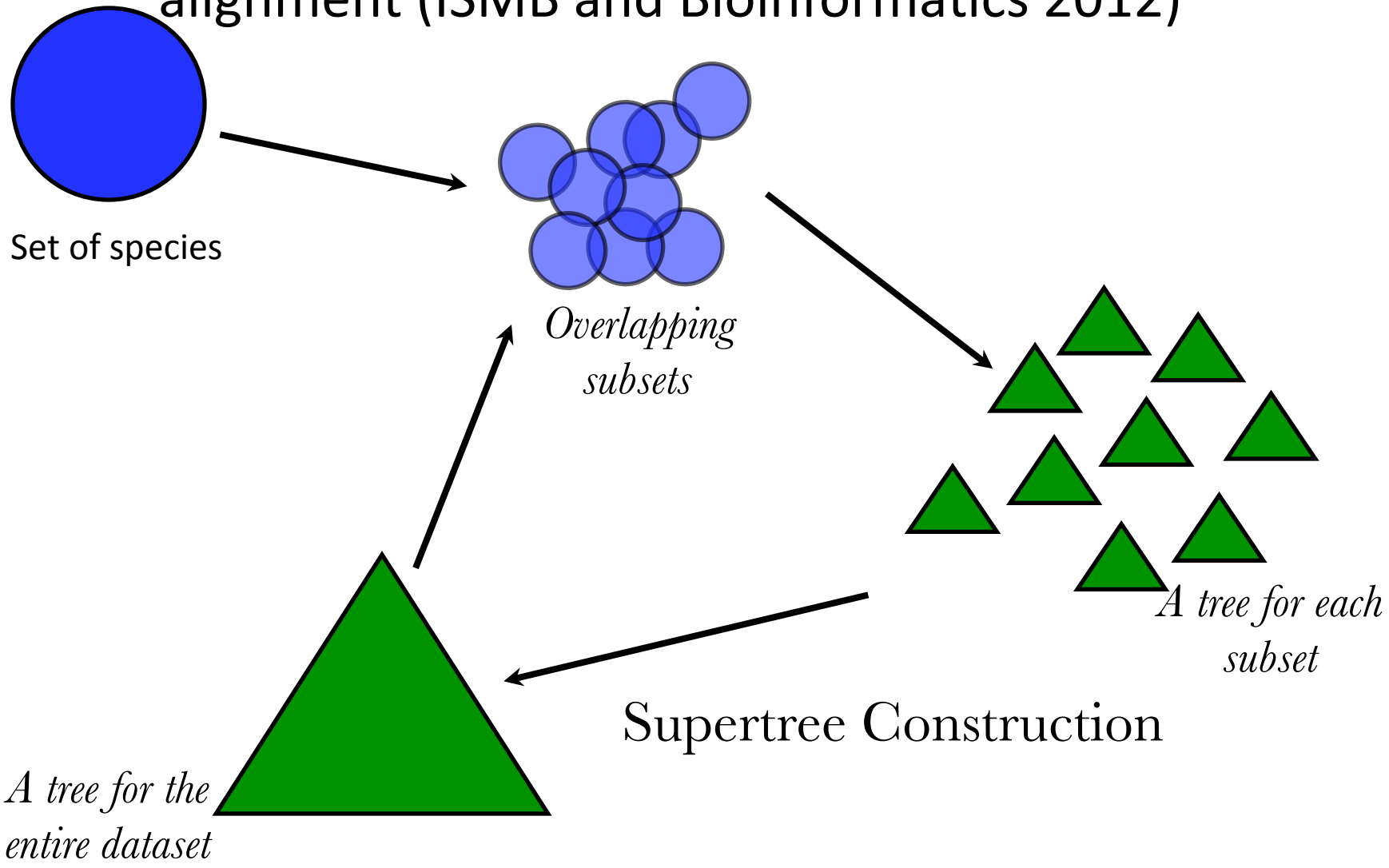
"Big data" complexity:

- model misspecification
- heterogeneity across genome
- fragmentary sequences
- errors in input data
- streaming data

Research Strategies

- Improved algorithms through:
  - Divide-and-conquer
  - "Bin-and-conquer"
  - Iteration
  - Bayesian statistics
  - Hidden Markov Models
  - Graph theory
  - Combinatorial optimization
- Statistical modelling
- Massive Simulations
- High Performance Computing

DACTAL: divide-and-conquer trees (almost) without alignment (ISMB and Bioinformatics 2012)



Results on Three Biological Datasets

DACTAL more accurate than standard methods, and faster than SATé (Liu et al., Science 2009)

CRW: Comparative RNA database, structural alignments

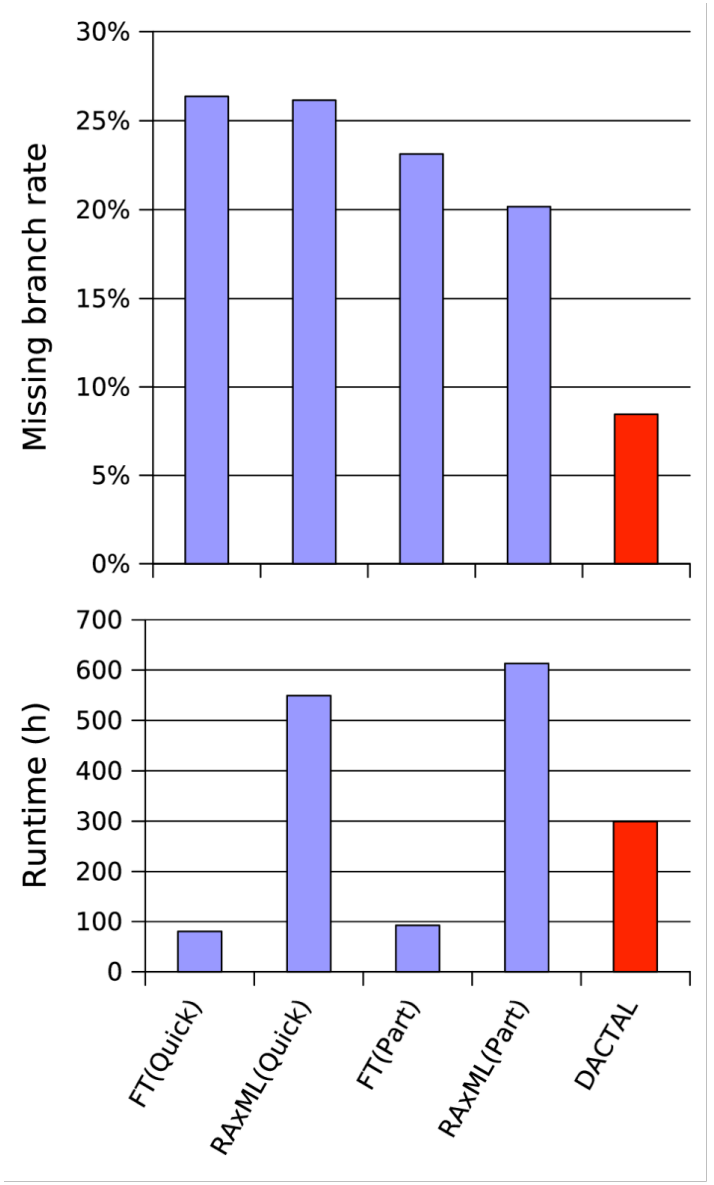
3 datasets with 6,323 to 27,643 sequences

Reference trees: 75% RAxML bootstrap trees

DACTAL (shown in red) run for 5 iterations starting from FT(Part)

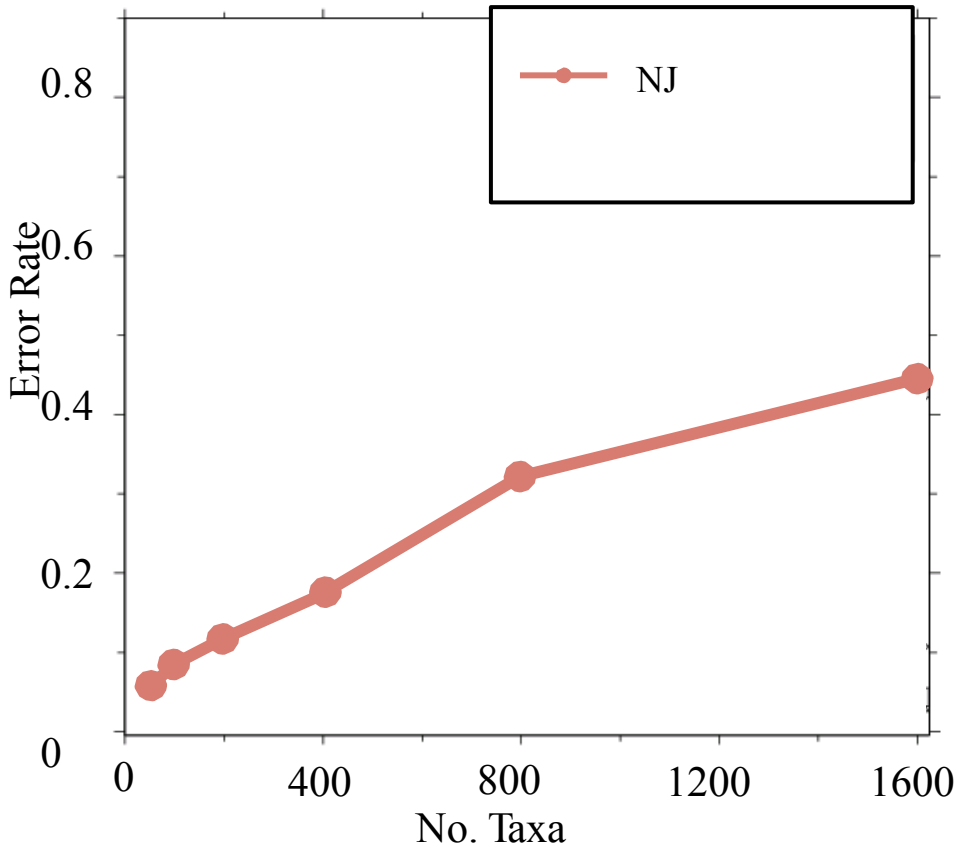
SATé-1 fails on the largest dataset

SATé-2 runs but is not more accurate than DACTAL, and takes longer



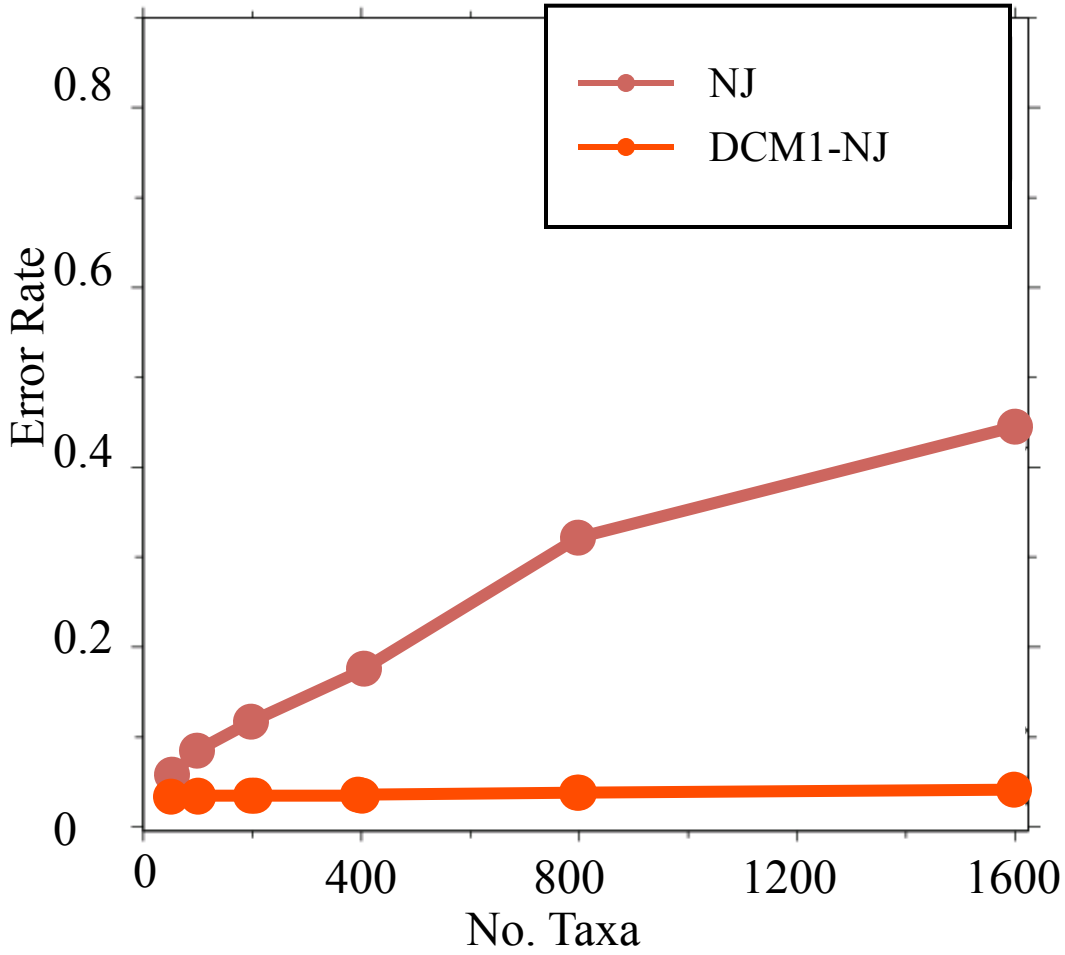
Neighbor joining has poor performance on large diameter trees

[Nakhleh et al. ISMB 2001]



Theorem (Atteson):
Exponential sequence length requirement for Neighbor Joining!

Chordal graph algorithms enables phylogeny estimation w.h.p. from *polynomial length* sequences



- Theorem (Warnow et al., SODA 2001): DCM1-NJ correct with high probability given sequences of length $O(\ln n\, e^{O(\ln n)})$
- Simulation study from Nakhleh et al. ISMB 2001

Supertree Estimation

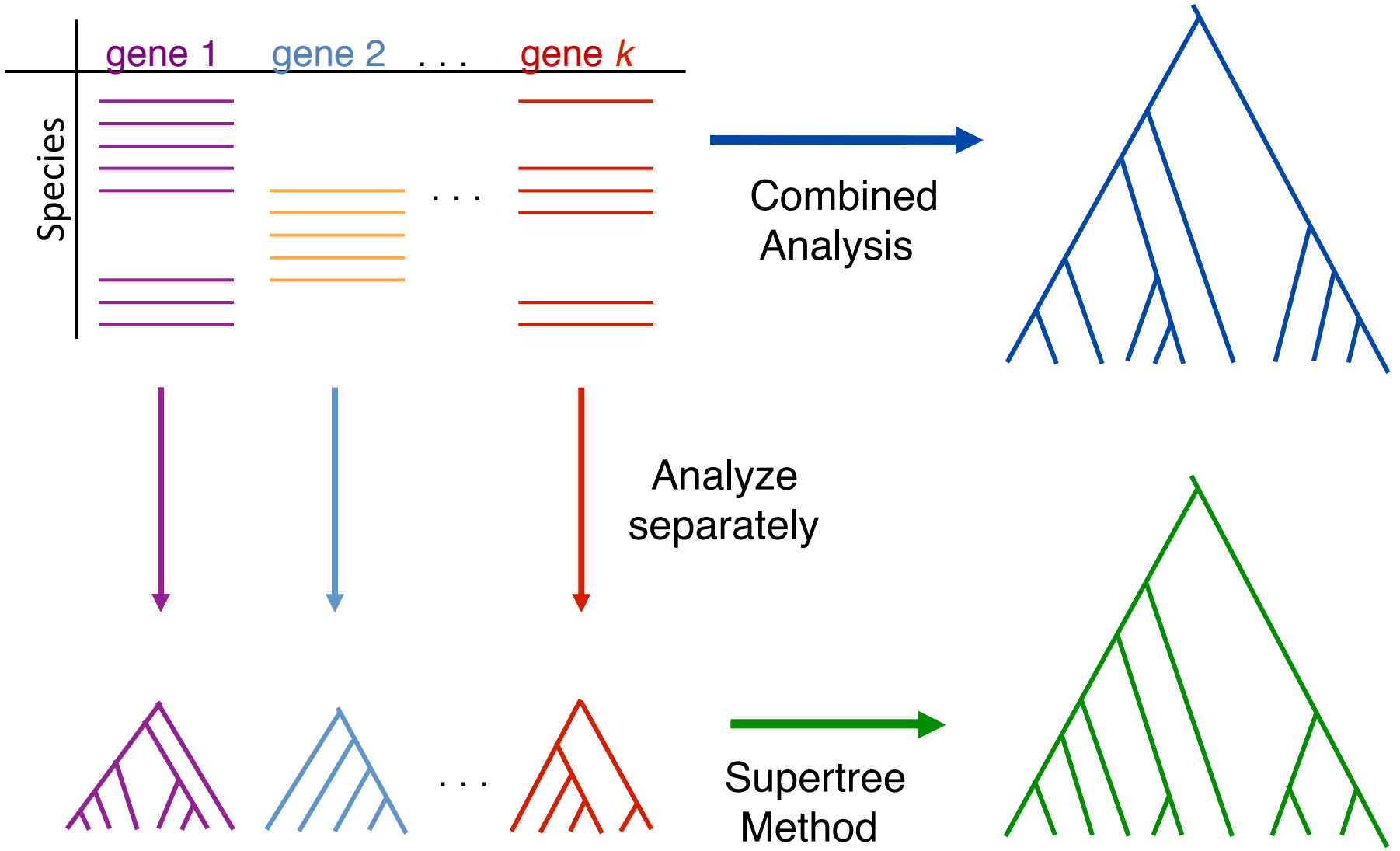
- Purposes:
  - Divide-and-conquer tree estimation
  - Combining analyses performed by other research groups

Many Supertree Methods

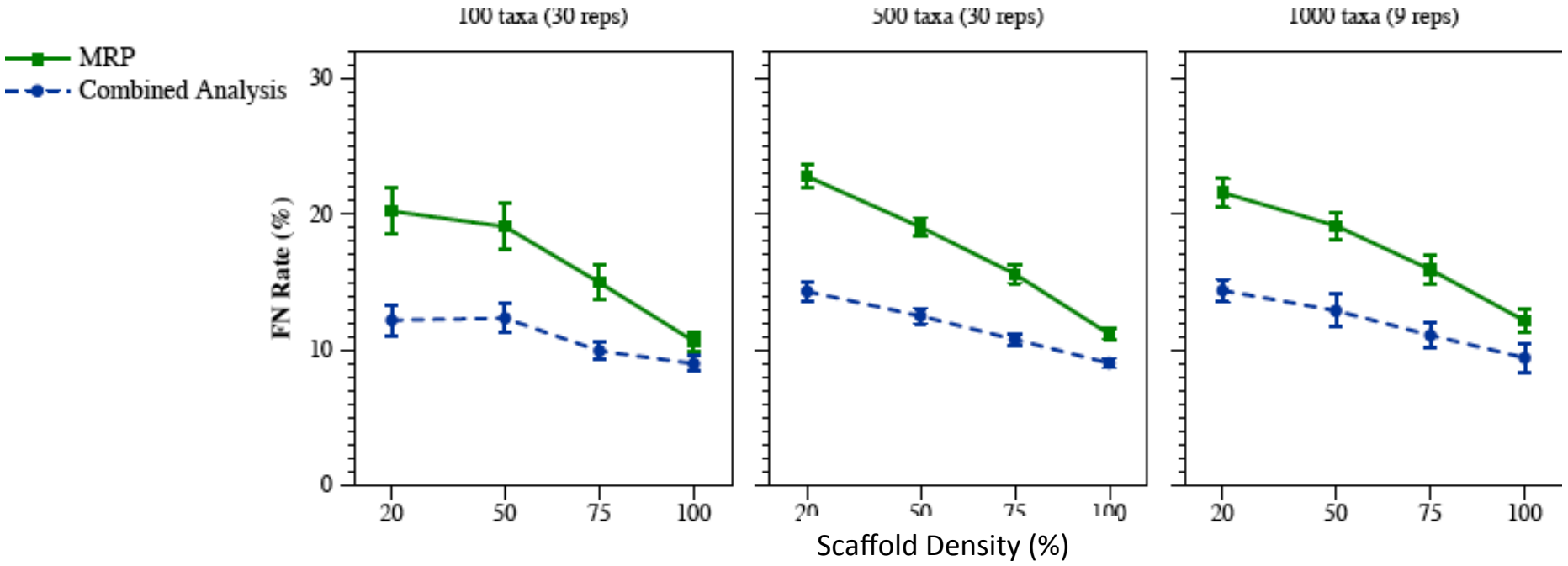
Matrix Representation with Parsimony
(Most commonly used and until recently the most accurate)

- MRP
- MRL
- MRF
- MRD
- Robinson-Foulds Supertrees
- Min-Cut
- Modified Min-Cut
- Semi-strict Supertree
- QMC
- Q-imputation
- SDM
- PhySIC
- Majority-Rule Supertrees
- Maximum Likelihood Supertrees
- and many more ...

Two competing approaches



MRP vs. RAxML on combined dataset



Challenges in Supertree Estimation

Challenges:

- Tree compatibility is NP-complete (therefore, even if subtrees are correct, supertree estimation is hard)
- Estimated subtrees have error
- MRP and MRL– two leading supertree methods - create huge binary matrices and analyze them using heuristics for NP-hard optimization problems. This cannot run on any large input.
- The best current methods (MRP, ML) are also not as accurate as RAxML on combined dataset.

We need new supertree methods that have excellent accuracy and can analyze large datasets!

Maximum Likelihood Supertrees

Steel and Rodrigo, Systematic Biology:

Given set of source trees, find a supertree that maximizes the probability of generating the source trees under a statistical model of tree generation

Robinson-Foulds Supertrees: non-parametric version of ML Supertrees.

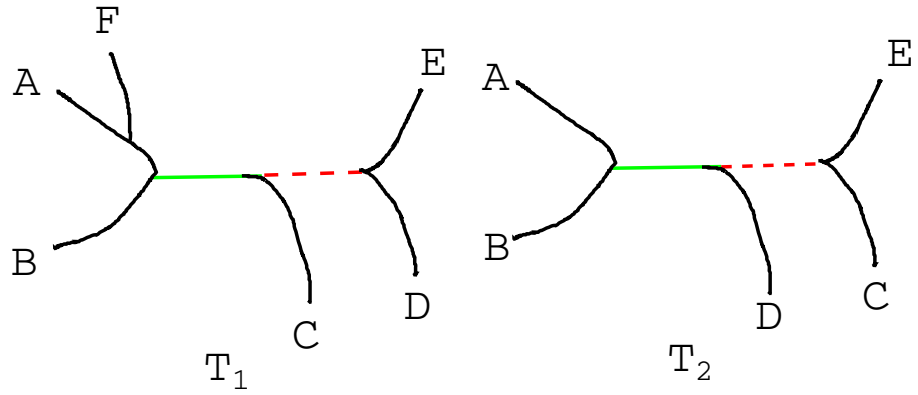
The RF Supertree optimization problem

- Input: Set $\mathcal{T}$ of source trees
- Output: *RF Supertree* T that minimizes the total RF distance to $\mathcal{T}$

- The Robinson-Foulds (RF) distance between a binary supertree T and a binary source tree t on a taxon subset s is

$$RF(T, t) = |bipartitions(T|_s) \setminus bipartitions(t)|$$

where $T|_s$ is T restricted to the taxa in s

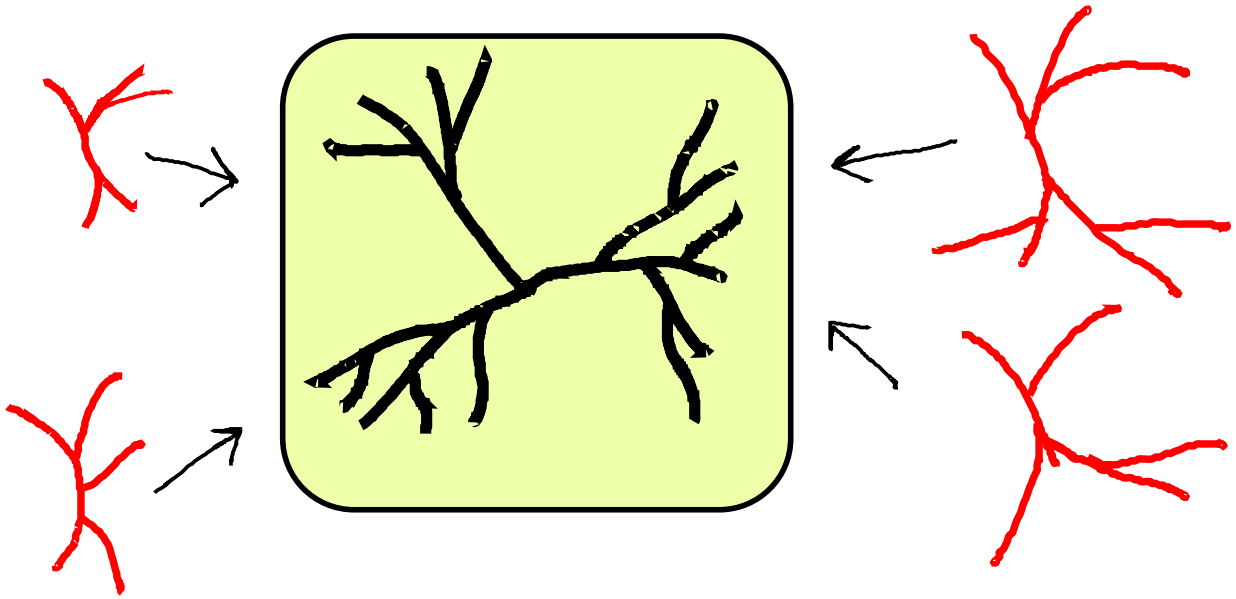


- RF distance is 1

The RF Supertree optimization problem

- Input: Set $\mathcal{T}$ of source trees
- Output: *RF Supertree T* that minimizes the total RF distance to $\mathcal{T}$

NP-hard!



Constrained Robinson-Foulds Supertree

- Input: Set $\mathcal{T}$ of source trees and set X of bipartitions on species set S (so each source tree has leaves in S)

- Output: Tree T on S that draws its bipartitions from X, and that minimizes the total RF distance to the source trees in $\mathcal{T}$.

The criterion score of a supertree is its total RF distance to the source trees.

FastRFS

- Theorem: FastRFS solves the Constrained Robinson-Foulds Supertree problem exactly in $O(|X|^2nk)$ time, where $n=|S|$ and $k=|\mathcal{T}|$.

- Proof: Uses dynamic programming, and constructs the tree from the bottom-up based on halves of the bipartitions in X.

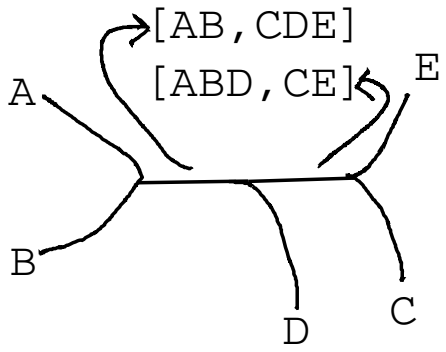
Published in Bioinformatics 2016, selected papers from RECOMB Comparative Genomics.

Exact constrained search used before for different problems

- Approach initially suggested in Hallet and Lagergren (2000) for dup-loss model
- Similar approach used for quartet support maximization in Bryant and Steel (2001) and ASTRAL (Mirarab et al., 2014), minimizing deep coalescences (Than and Nakhleh, 2009)

Choosing the constraint set *X*

- FastRFS finds the best scoring tree with every bipartition in the set *X*
- We can look at the input trees to generate the set *X*



- We can also add bipartitions from a tree *M* estimated with a different method
- If that tree is added, the FastRFS tree will have a score at least as good as *M*

Enhancing FastRFS with other supertree methods

FastRFS-basic:

- By default, FastRFS uses ASTRAL-2 to generate the constraint set X from the input trees
- This finds a tree with a score at least as good as the ASTRAL-2 tree

We define **FastRFS-enhanced**:

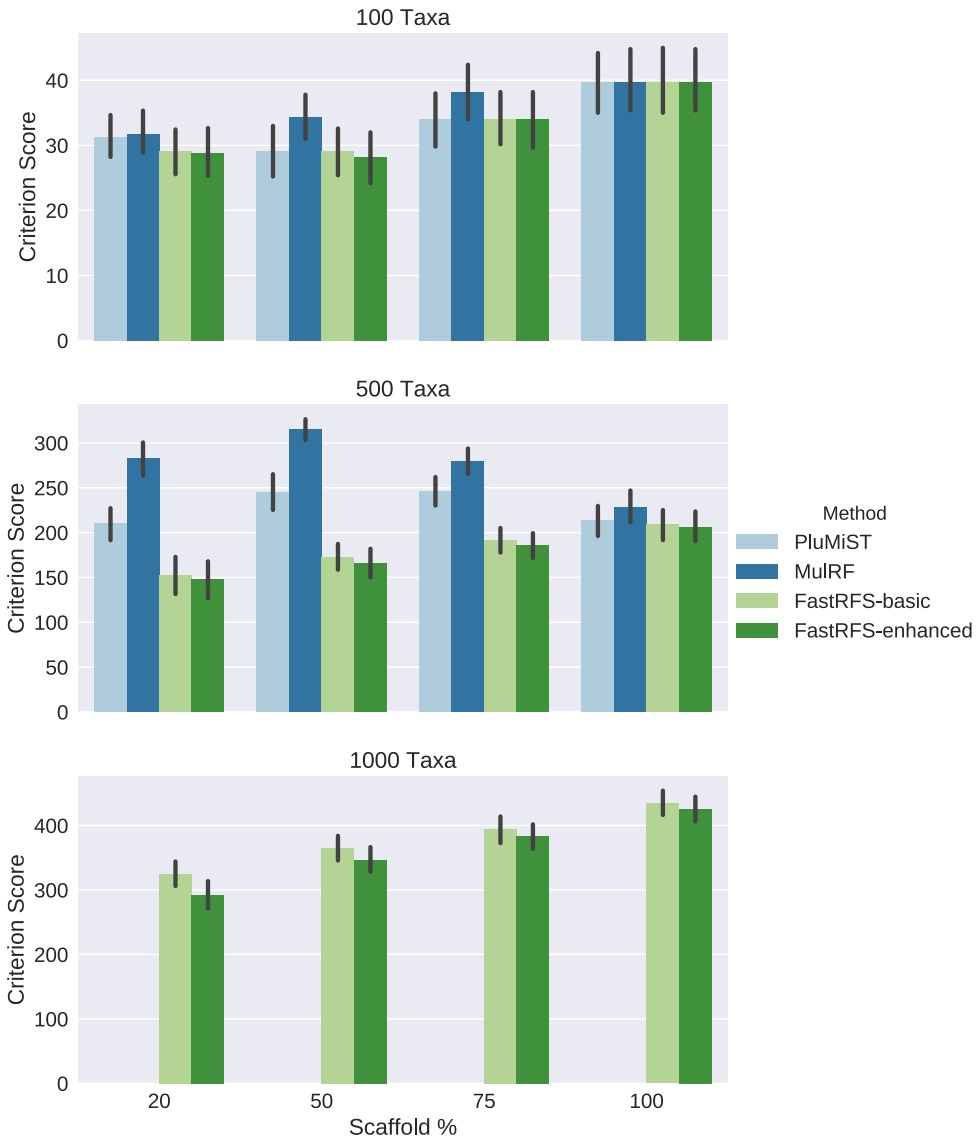
- Always add the MRL tree
- Use the ASTRID tree if ASTRID can run quickly

ASTRID runs quickly if every pair of taxa appears in at least one source tree

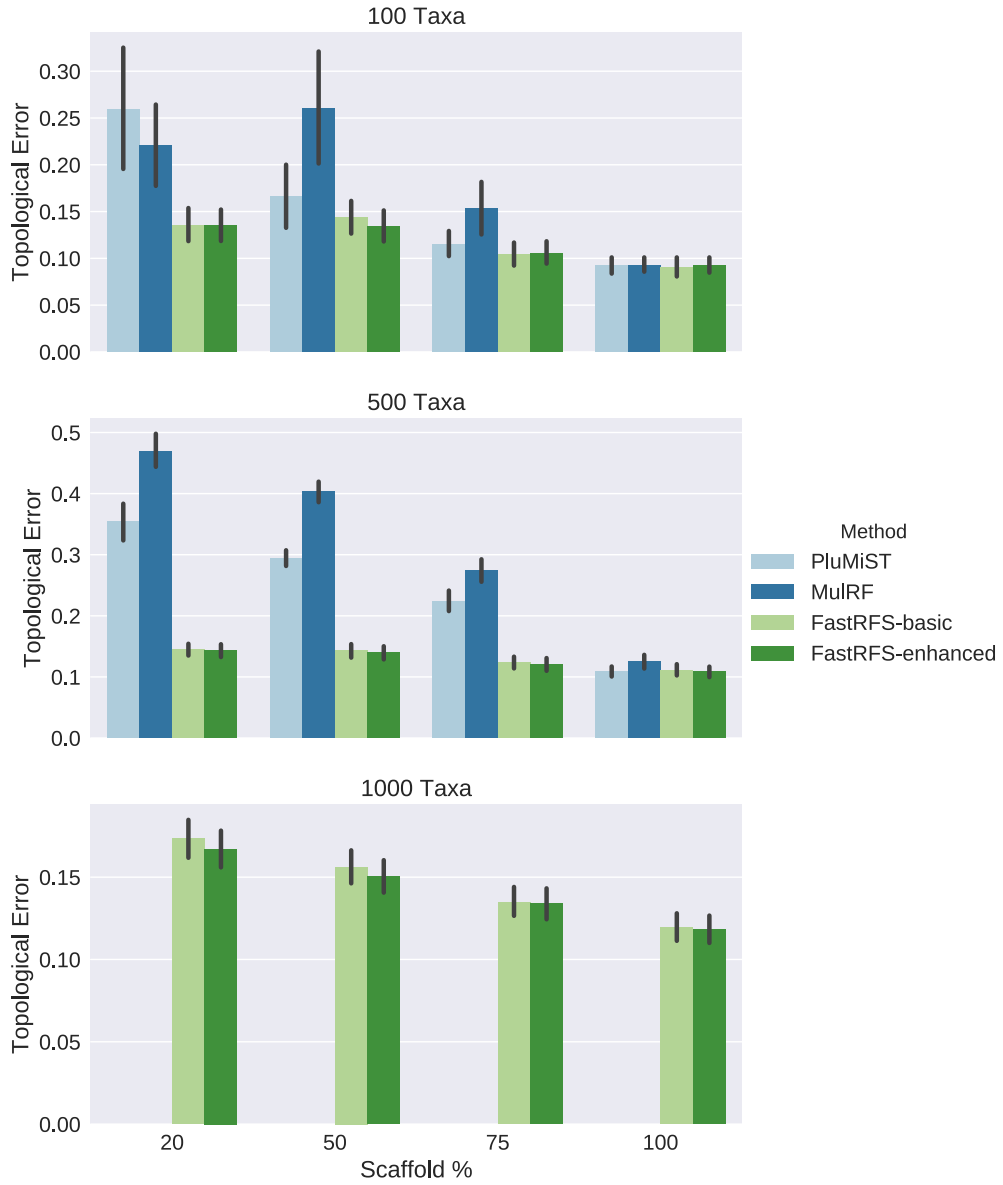
Performance study

- We compared FastRFS-basic and FastRFS-enhanced to leading supertree methods for Robinson-Foulds Supertrees (PluMiST and MulRF) on biological and simulated data with respect to
  - Criterion scores
  - Tree error (on simulated data)
  - Running time

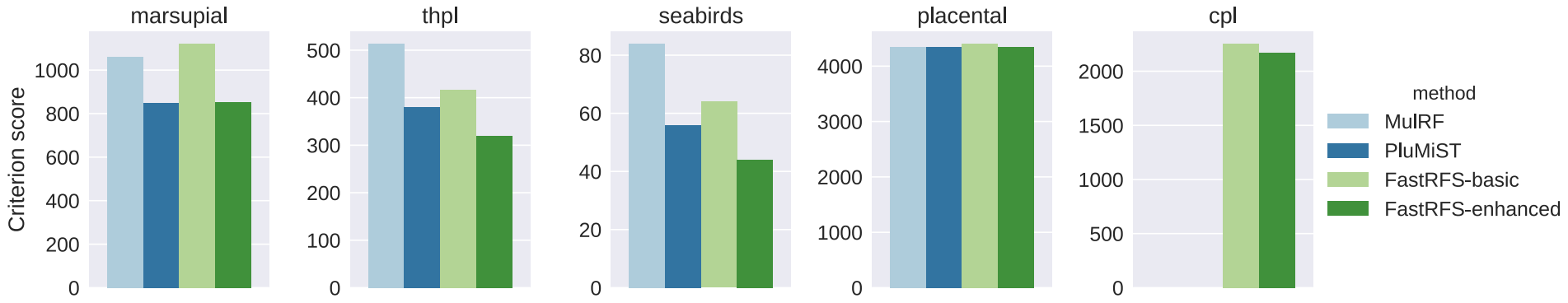
Robinson-Foulds Supertree Criterion Scores



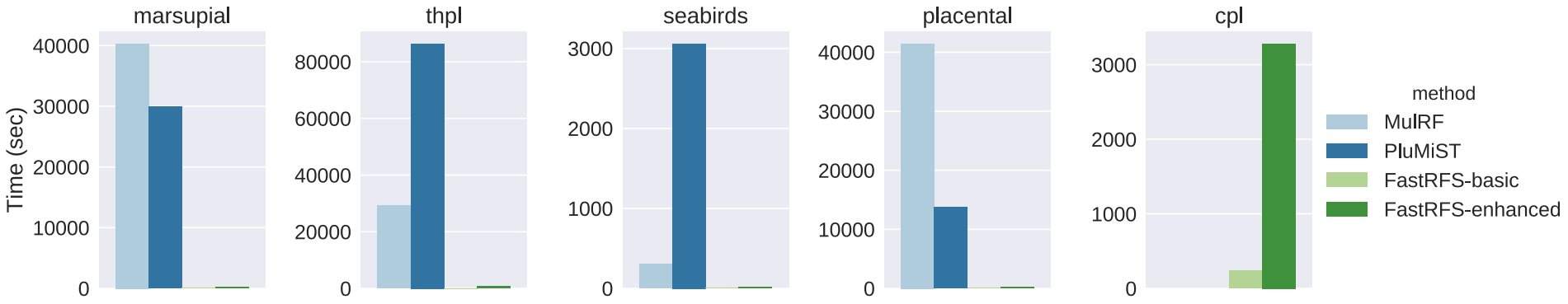
Tree Error on Simulated Datasets



Robinson-Foulds Supertree Criterion Scores on biological datasets



Running times on biological datasets



Running times on five biological supertree datasets.
The CPL dataset has 2228 species, and is too large for PluMiST and MulRF to run.

Summary

- FastRFS is a fast and highly accurate supertree method, with greatly improved topological accuracy and criterion scores compared to alternative approaches for Robinson-Foulds Supertrees.
- FastRFS also is more topologically accurate than other leading supertree methods (data not shown, see paper).
- The main challenge is computing a set X of bipartition constraints from the input.

Future Work

- Test FastRFS within DACTAL and other divide-and-conquer strategies, and evaluate it as a starting point for Maximum Likelihood Supertrees.
- Explore whether constraining the search space makes other NP-hard optimization problems tractable.
- Analyses of biological datasets (e.g., collaborations with Genome 10K, Avian Phylogenetics Project, and Thousand Plant Transcriptome Project)