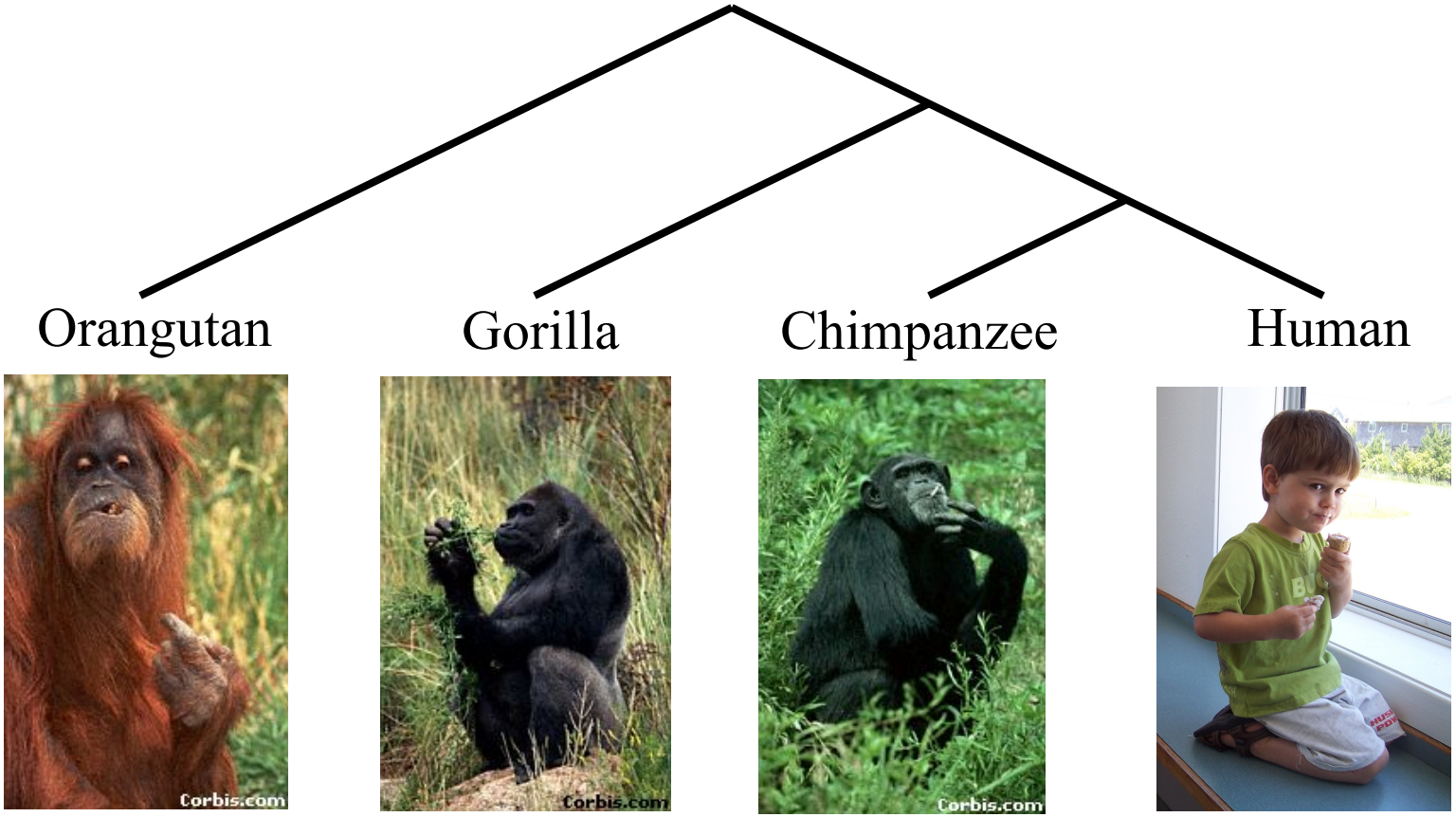
# Large-scale Multiple Sequence Alignment and Phylogenetic Estimation

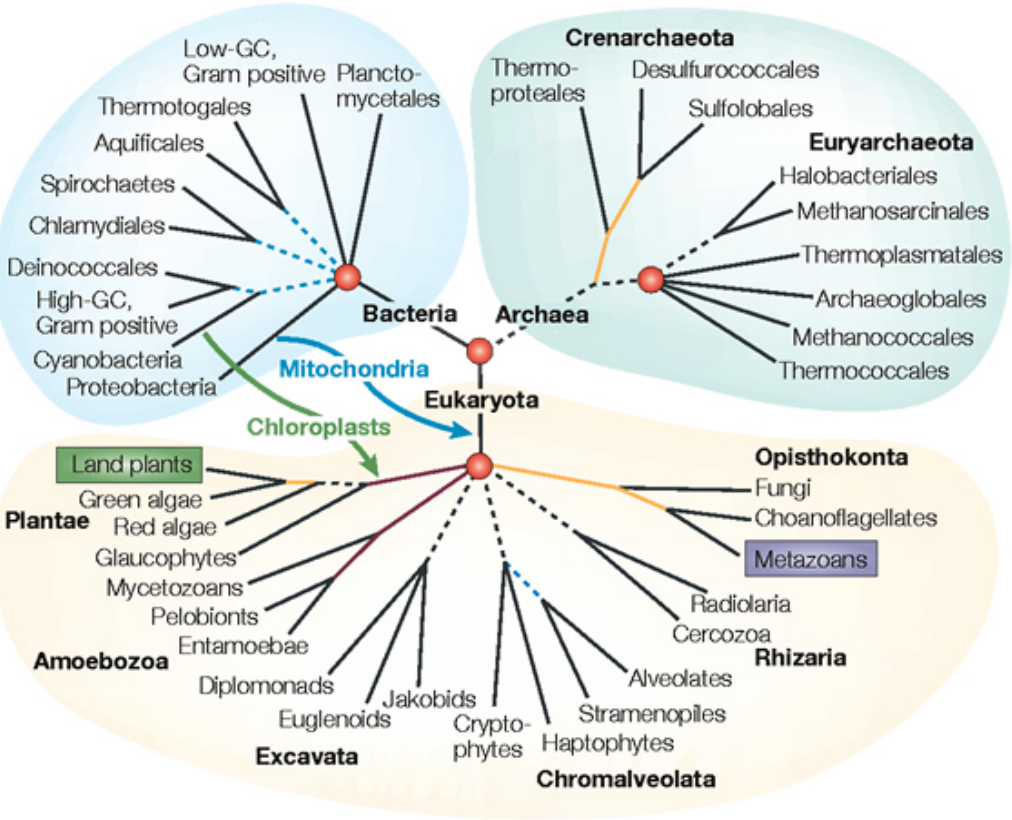
Tandy Warnow
Department of Computer Science
The University of Texas at Austin

# Phylogeny (evolutionary tree)



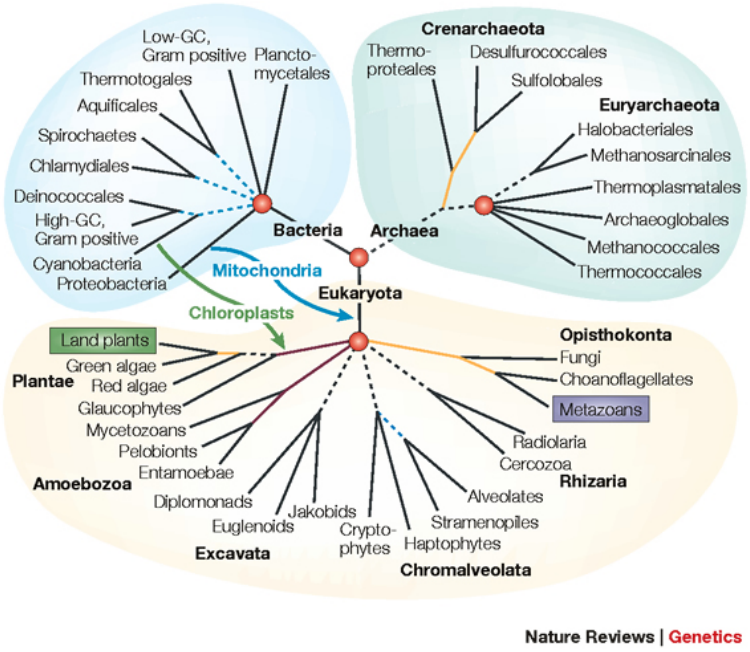

*From the Tree of the Life Website,*
*University of Arizona*

# The “Tree of Life”




Nature Reviews | Genetics

# The Tree of Life: Applications to Biology

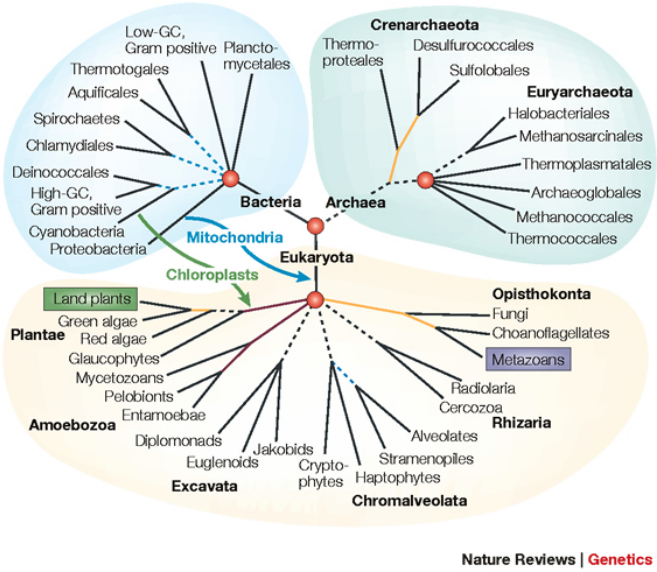



Biomedical applications
- Mechanisms of evolution
- Environmental influences
- Drug Design
- Protein structure and function
- Human migrations

"Nothing in biology makes sense except in the light of evolution"
Dobzhansky

# Phylogenomics

## (Phylogenetic estimation from whole genomes)




Nature Reviews | Genetics

# Computational Phylogenomics

Scientific Challenges:

- Multiple sequence alignment
- Gene tree estimation
- Estimating species trees from incongruent gene trees
- Genome rearrangement phylogeny
- Reticulate evolution
- Metagenomic taxon identification
- Biomolecular structure and function prediction
- Population genetics

Mathematical and computer science approaches:

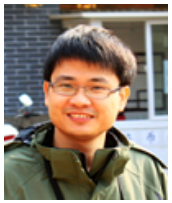
- Probabilistic analysis of algorithms
- Machine learning techniques (e.g., HMMs)
- Graph theory
- Heuristics for NP-hard optimization problems
- Data mining techniques to explore multiple optima
- Parallel computing and HPC
- Massive simulations

# Avian Phylogenomics Project

Erich Jarvis, HHMI

MTP Gilbert, Copenhagen

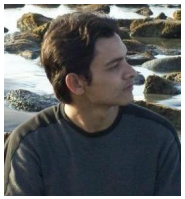
G Zhang, BGI



T. Warnow UT-Austin

S. Mirarab UT-Austin

Md. S. Bayzid UT-Austin



Plus many many other people…

- Approx. 50 species, whole genomes
- 8000+ genes, UCEs
- Gene sequence alignments and trees computed using SATé

**Challenges:**

**Maximum likelihood tree estimation on multi-million-site sequence alignments**

**Massive gene tree incongruence**

# 1kp: Thousand Transcriptome Project

G. Ka-Shu Wong
U Alberta

J. Leebens-Mack
U Georgia

N. Wickett
Northwestern

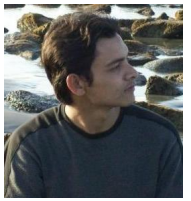
N. Matasci
iPlant

T. Warnow,
UT-Austin

S. Mirarab,
UT-Austin

N. Nguyen,
UT-Austin

Md. S.Bayzid
UT-Austin



Plus many many other people…

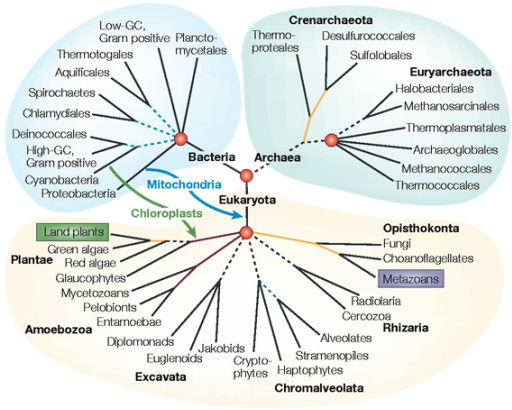
- Plant Tree of Life based on transcriptomes of ~1200 species
- More than 13,000 gene families (most not single copy)
- Gene sequence alignments and trees computed using SATé

**Challenges:**

**Multiple sequence alignment of > 100,000 (highly fragmentary) sequences**

**Gene tree incongruence**

# Estimating the Tree of Life




Nature Reviews | Genetics


*Novel techniques needed* for scalability and accuracy - (HPC is necessary but not sufficient)

NP-hard problems and large datasets

Current methods do not provide good accuracy

Big Data complexity (fragmentary and missing data, heterogeneity, errors)

# Research Agenda

Major scientific goals:

- Develop methods that produce more accurate alignments and phylogenetic estimations for *difficult-to-analyze datasets*
- Produce mathematical theory for statistical inference under complex models of evolution
- Develop novel machine learning techniques to boost the performance of classification methods (e.g., “Disk Covering Methods”, “Bin-and-Conquer” and “HMM Families”)

Software that:

- Can run efficiently on *desktop* computers on large datasets
- Can analyze ultra-large datasets (100,000+) using multiple processors
- Is freely available in *open source* form, with biologist-friendly GUIs

Current topics:

- Ultra-large multiple sequence alignment and tree estimation
- Estimating species trees from incongruent gene trees
- Metagenomic taxon identification

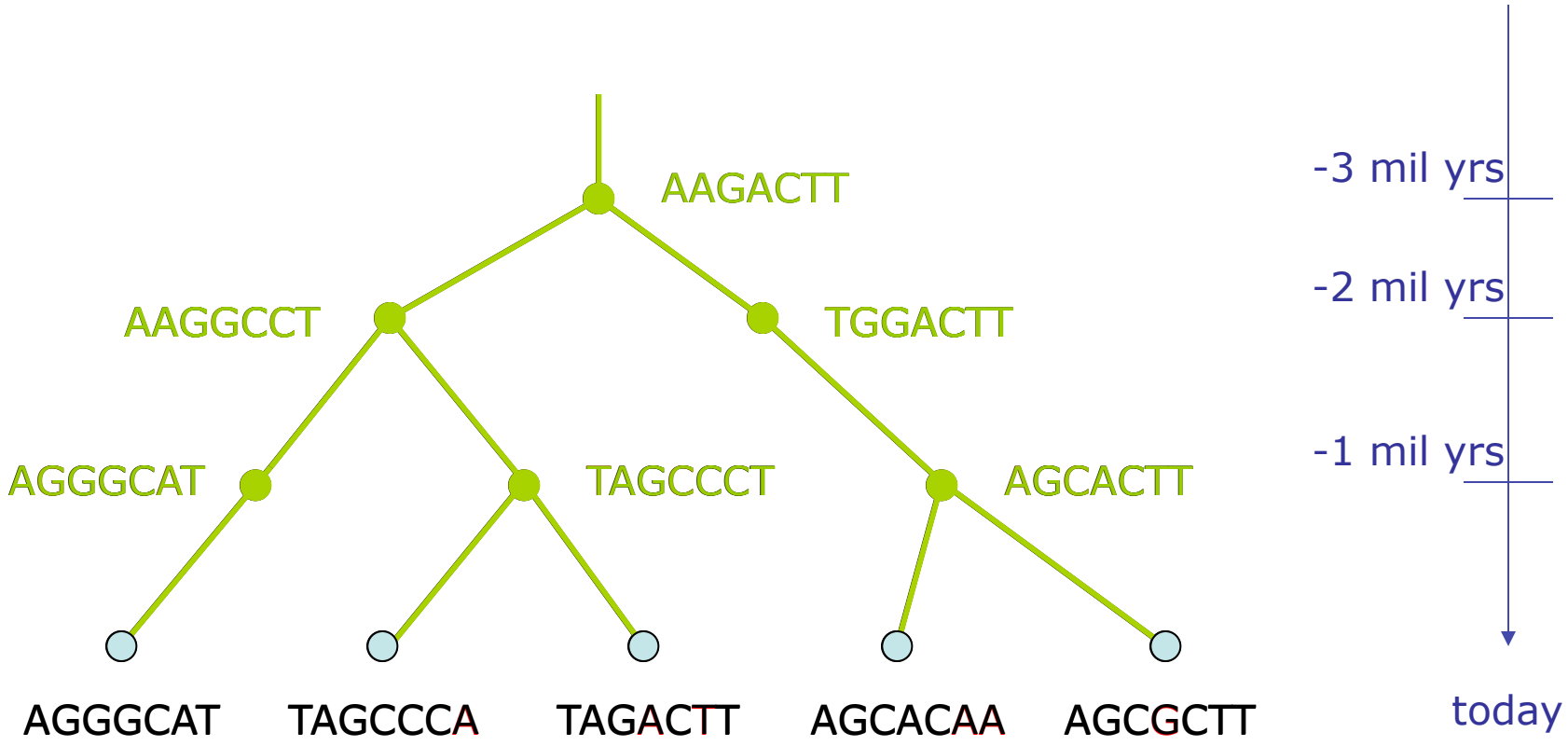
# This Talk

1. Gene Tree Estimation: phylogeny estimation under Markov models of evolution, and "absolute fast converging methods"
2. Ultra-large Multiple Sequence Alignment and Phylogeny Estimation (up to *1,000,000 sequences*) using "HMM Families" (new technique)
3. Application of HMM Families to Taxon Identification of Metagenomic Data and Phylogenetic Placement
4. Discussion: Statistical Inference and Machine Learning on Big Data

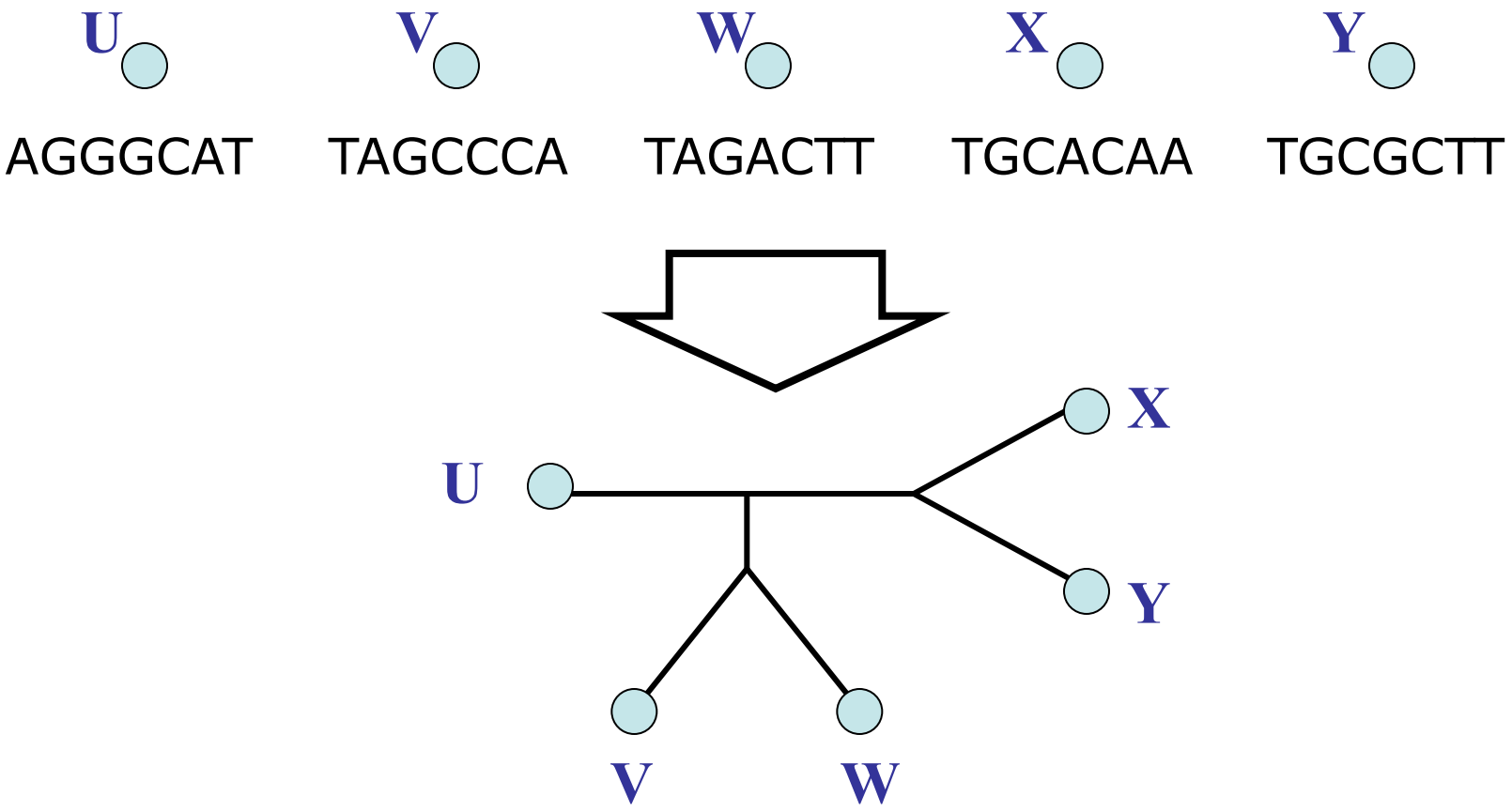
# I: Gene Tree Estimation

- Markov models of sequence evolution
- Statistical consistency and sequence length requirements
- Absolute fast convergence
- DCM1-boosting

# DNA Sequence Evolution



# Phylogeny Problem



# Markov Model of Site Evolution

Simplest (Jukes-Cantor, 1969):

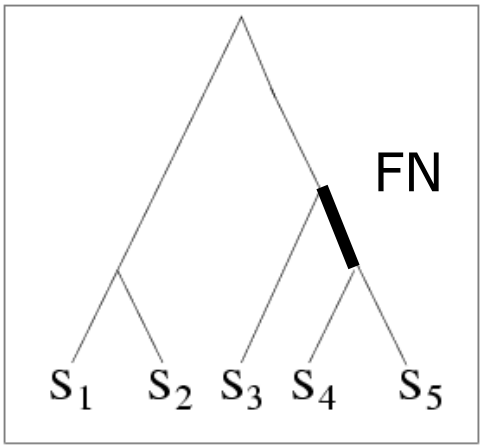
- The model tree T is binary and has substitution probabilities p(e) on each edge e.
- The state at the root is randomly drawn from {A,C,T,G} (nucleotides)
- If a site (position) changes on an edge, it changes with equal probability to each of the remaining states.
- The evolutionary process is Markovian.

More complex models (such as the General Markov model) are also considered, often with little change to the theory.

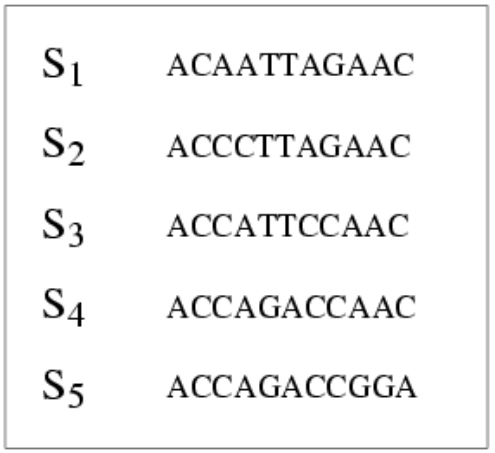
# Questions

- Is the model tree identifiable?
- Which estimation methods are statistically consistent under this model?
- How much data does the method need to estimate the model tree correctly (with high probability)?
- What is the computational complexity of an estimation problem?

# Simulation Study



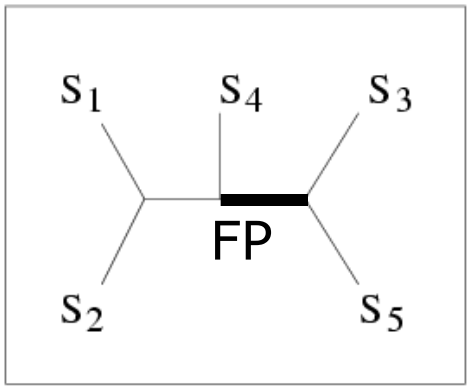

TRUE TREE




DNA SEQUENCES

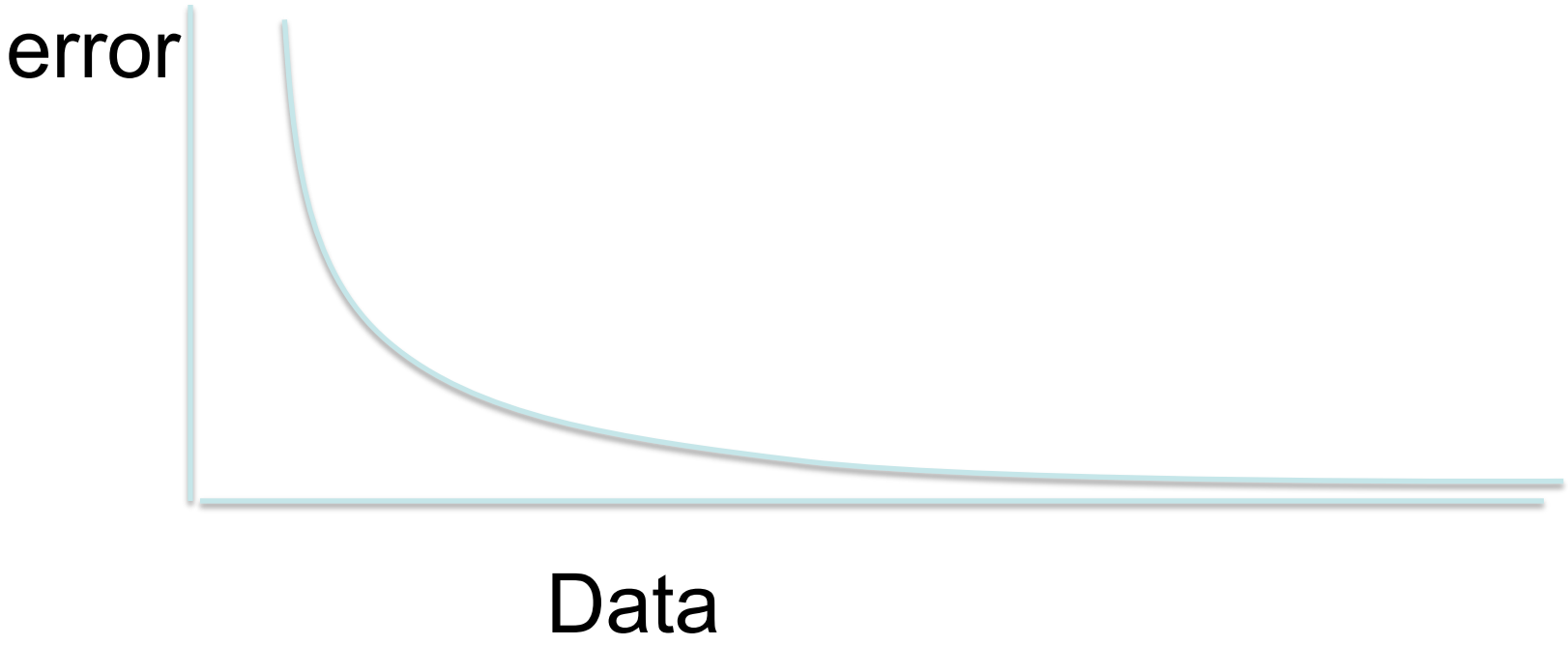
FN: false negative
(missing edge)
FP: false positive
(incorrect edge)

**50% error rate**

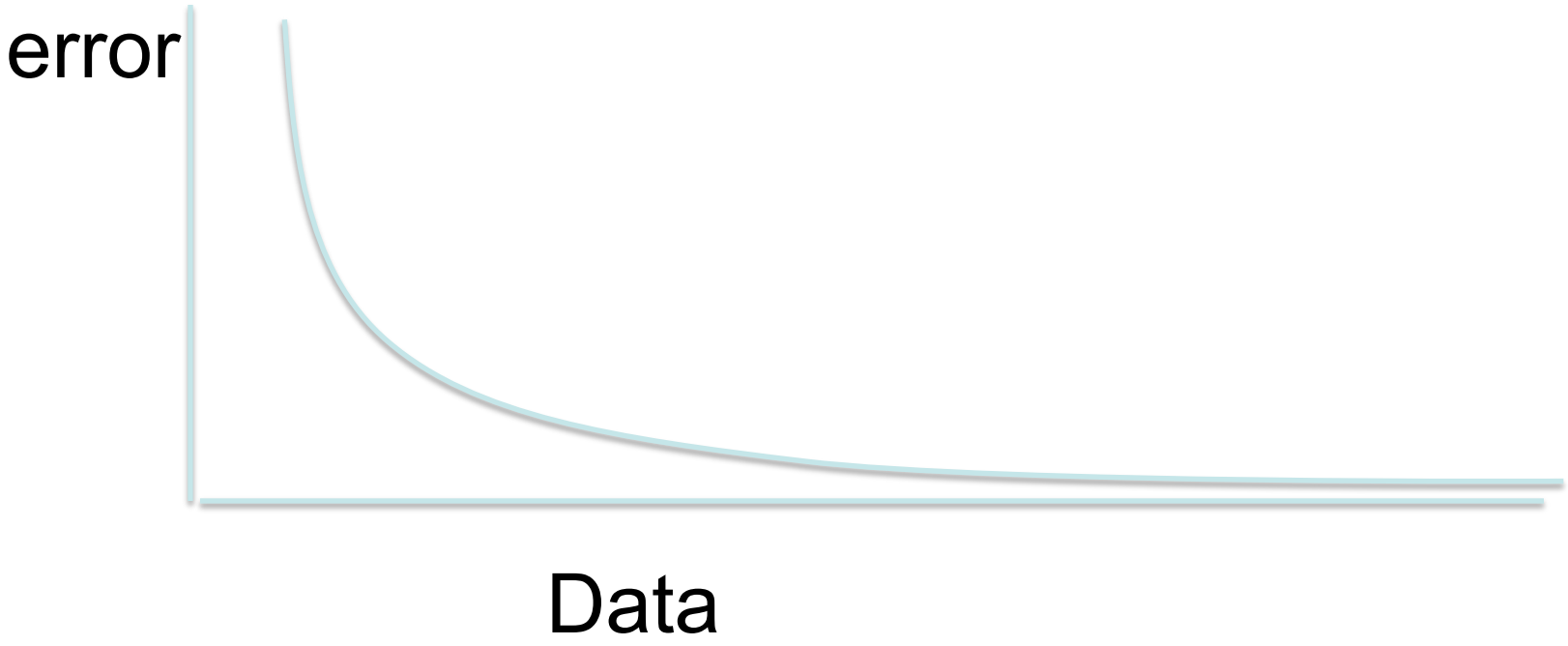



INFERRED TREE

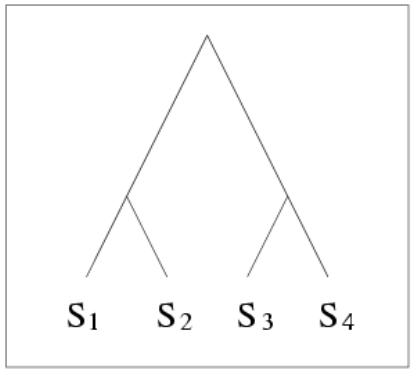
# Statistical Consistency



# Statistical Consistency




Data are sites in an alignment




TRUE TREE

| | |
|---|---|
| $S_1$ | ACAATTAGAAC |
| $S_2$ | ACCCTTAGAAC |
| $S_3$ | ACCATTCCAAC |
| $S_4$ | ACCAGACCAAC |

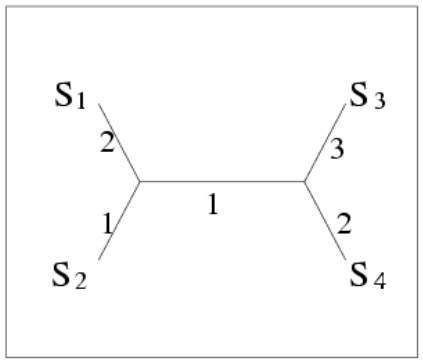
DNA SEQUENCES

STATISTICAL ESTIMATION OF PAIRWISE DISTANCES

| | $S_1$ | $S_2$ | $S_3$ | $S_4$ |
|---|---|---|---|---|
| $S_1$ | 0 | 3 | 6 | 5 |
| $S_2$ | | 0 | 5 | 4 |
| $S_3$ | | | 0 | 5 |
| $S_4$ | | | | 0 |

DISTANCE MATRIX

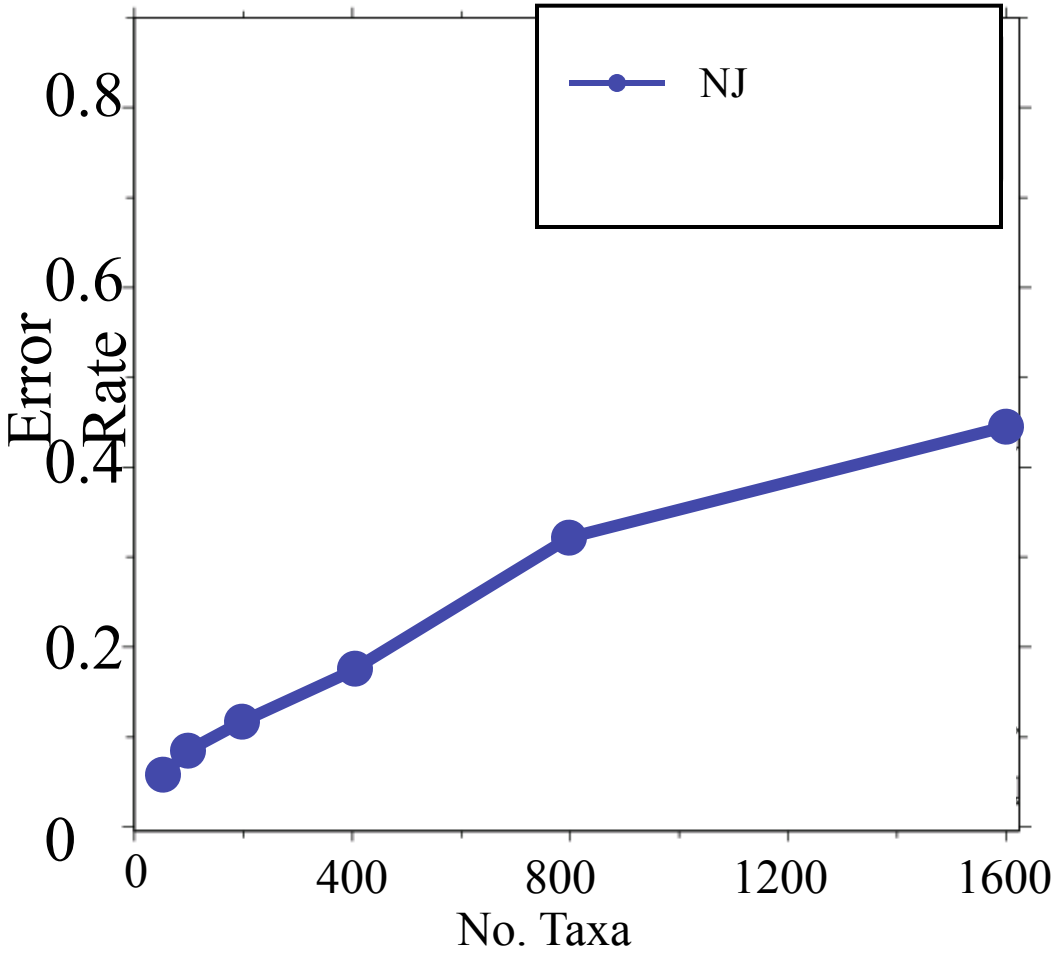
METHODS SUCH AS NEIGHBOR JOINING




INFERRED TREE

Neighbor Joining (and many other distance-based methods) are statistically consistent under Jukes-Cantor

# Neighbor Joining on large diameter trees




Simulation study based upon fixed edge lengths, K2P model of evolution, sequence lengths fixed to 1000 nucleotides.

Error rates reflect proportion of incorrect edges in inferred trees.

*[Nakhleh et al. ISMB 2001]*

# "Convergence rate" or sequence length requirement

The sequence length (number of sites) that a phylogeny reconstruction method M needs to reconstruct the true tree with probability at least $1-\varepsilon$ depends on

- M (the method)
- $\varepsilon$
- f = min p(e),
- g = max p(e), and
- n = the number of leaves

We fix everything but n.

Theorem (Erdos et al. 1999, Atteson 1999):

Various distance-based methods (including Neighbor joining) will return the true tree with high probability given sequence lengths that are *exponential* in the evolutionary diameter of the tree (hence, exponential in n).

Proof:

- the method returns the true tree if the estimated distance matrix is close to the model tree distance matrix
- the sequence lengths that suffice to achieve bounded error are exponential in the evolutionary diameter.

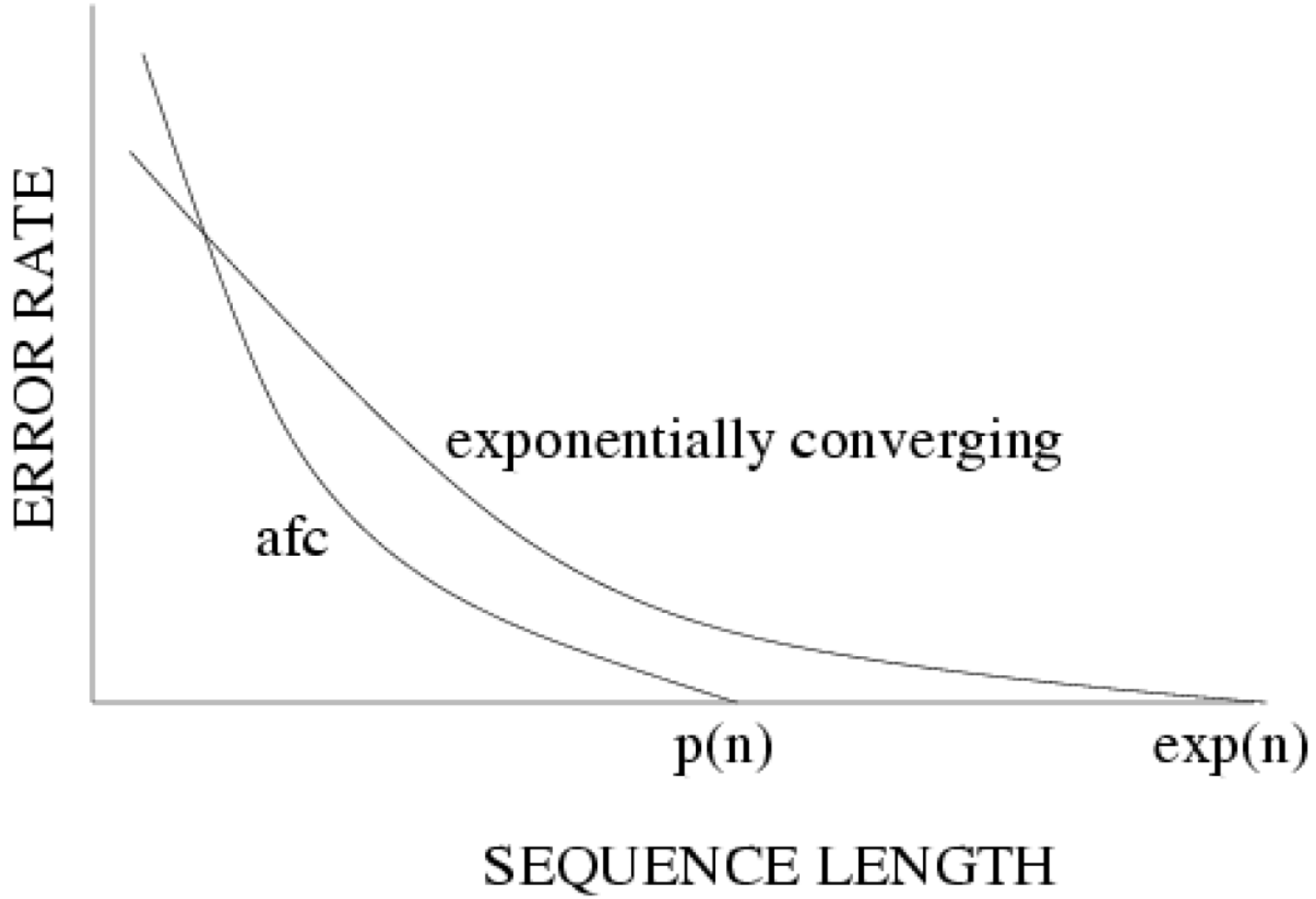
# *afc* methods (Warnow et al., 1999)

A method M is "absolute fast converging", or *afc*, if for all positive f, g, and $\varepsilon$, there is a polynomial p(n) such that $Pr(M(S)=T) > 1-\varepsilon$, when S is a set of sequences generated on T of length at least p(n).

Notes:

1. The polynomial p(n) will depend upon M, f, g, and $\varepsilon$.
2. The method M is not "told" the values of f and g.

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

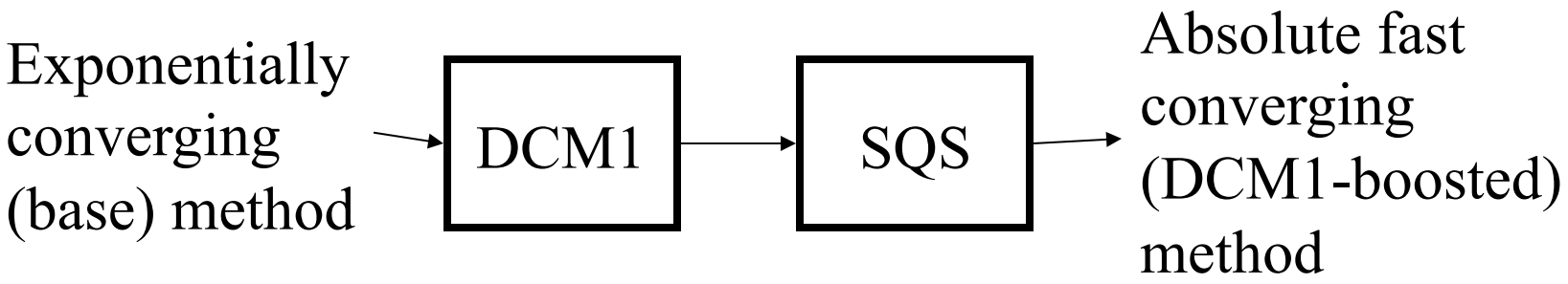


# Fast-converging methods (and related work)

- 1997: Erdos, Steel, Szekely, and Warnow (ICALP).
- 1999: Erdos, Steel, Szekely, and Warnow (RSA, TCS);
  Huson, Nettles and Warnow (J. Comp Bio.)
- 2001: Warnow, St. John, and Moret (SODA);
  Nakhleh, St. John, Roshan, Sun, and Warnow (ISMB)
  Cryan, Goldberg, and Goldberg (SICOMP);
  Csuros and Kao (SODA);
- 2002: Csuros (J. Comp. Bio.)
- 2006: Daskalakis, Mossel, Roch (STOC),
  Daskalakis, Hill, Jaffe, Mihaescu, Mossel, and Rao (RECOMB)
- 2007: Mossel (IEEE TCBB)
- 2008: Gronau, Moran and Snir (SODA)
- 2010: Roch (Science)
- 2013: Roch (in preparation)

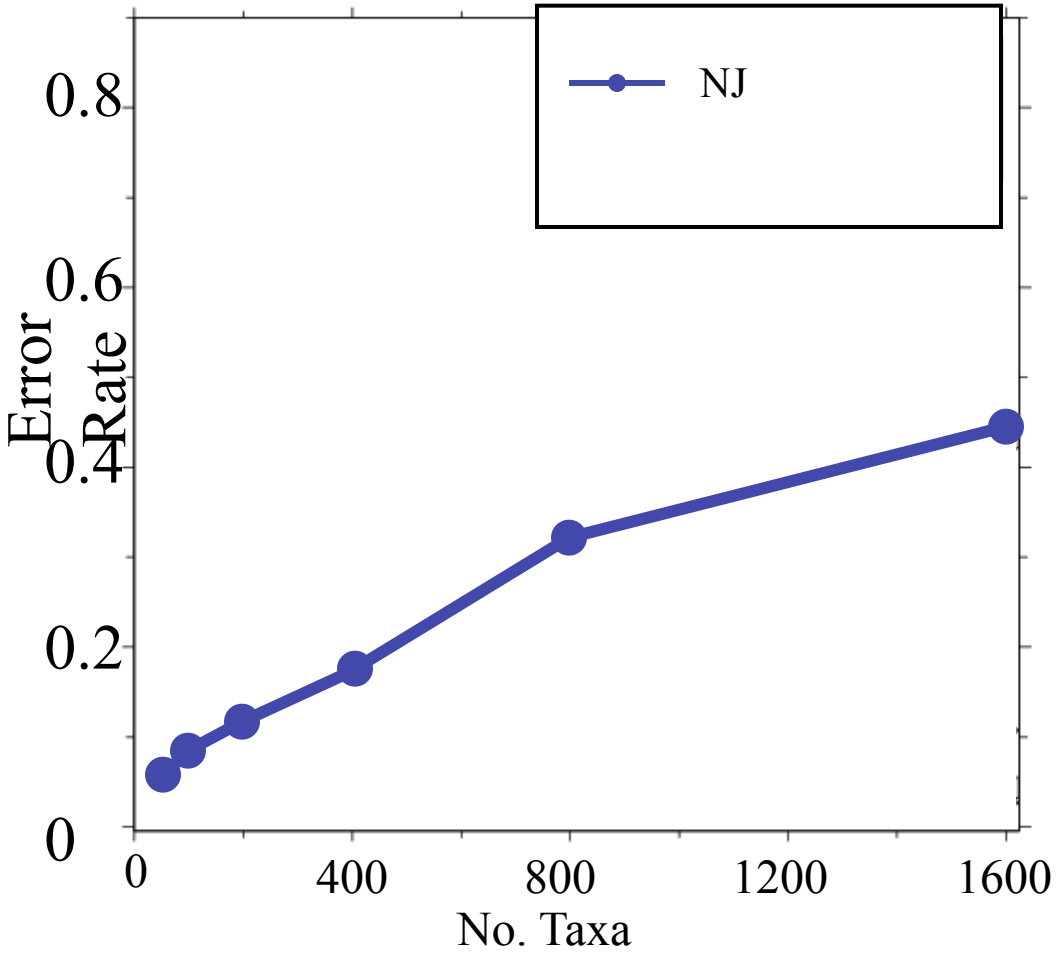
# DCM1-boosting:

*Warnow, St. John, and Moret, SODA 2001*




- The DCM1 phase produces a collection of trees (one for each threshold), and the SQS phase picks the "best" tree.
- *How to compute a tree for a given threshold:*
  - *Handwaving description*: erase all the entries in the distance matrix above that threshold, and obtain the threshold graph. Then add edges to get a chordal graph. Use the base method to estimate a tree on each maximal clique. Combine the trees together.
  - Note the *use of* chordal graph theory and algorithms.

# Neighbor Joining on large diameter trees



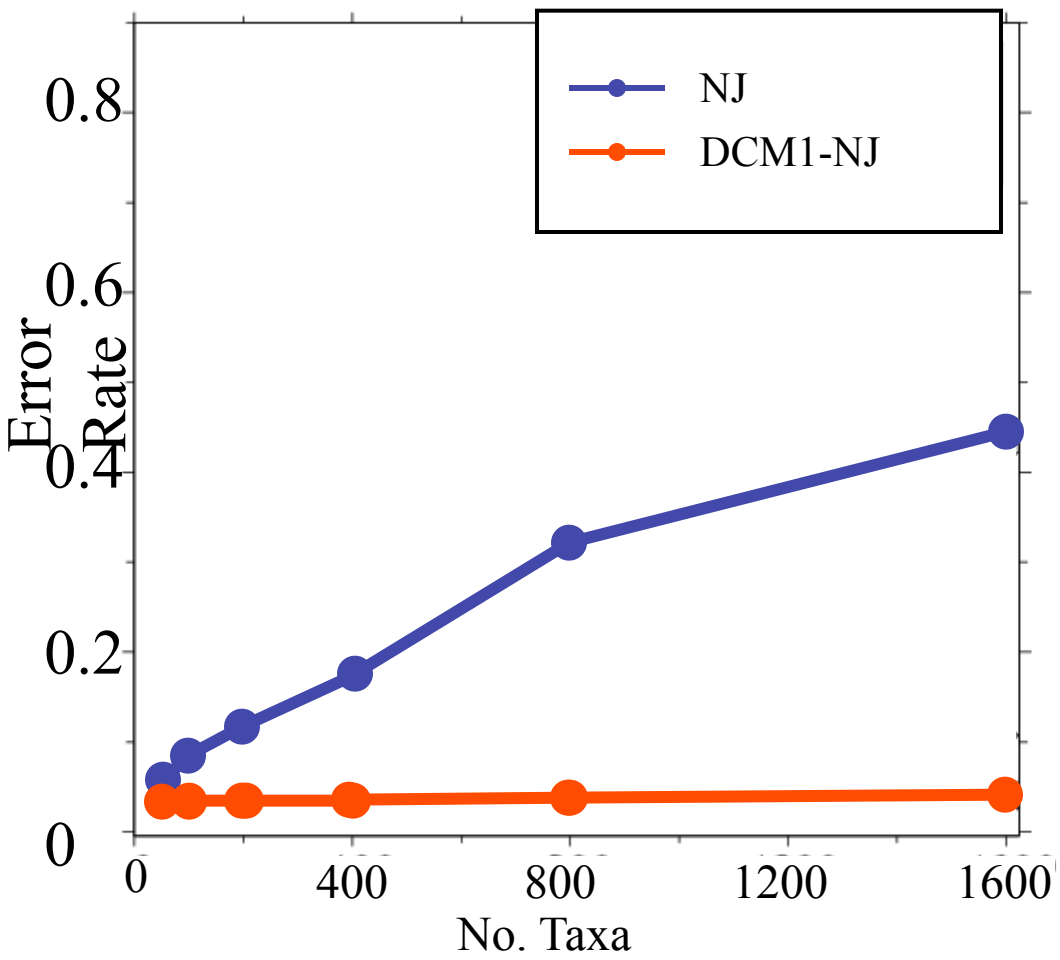

Simulation study based upon fixed edge lengths, K2P model of evolution, sequence lengths fixed to 1000 nucleotides.

Error rates reflect proportion of incorrect edges in inferred trees.

*[Nakhleh et al. ISMB 2001]*

# DCM1-boosting distance-based methods

*[Nakhleh et al. ISMB 2001]*



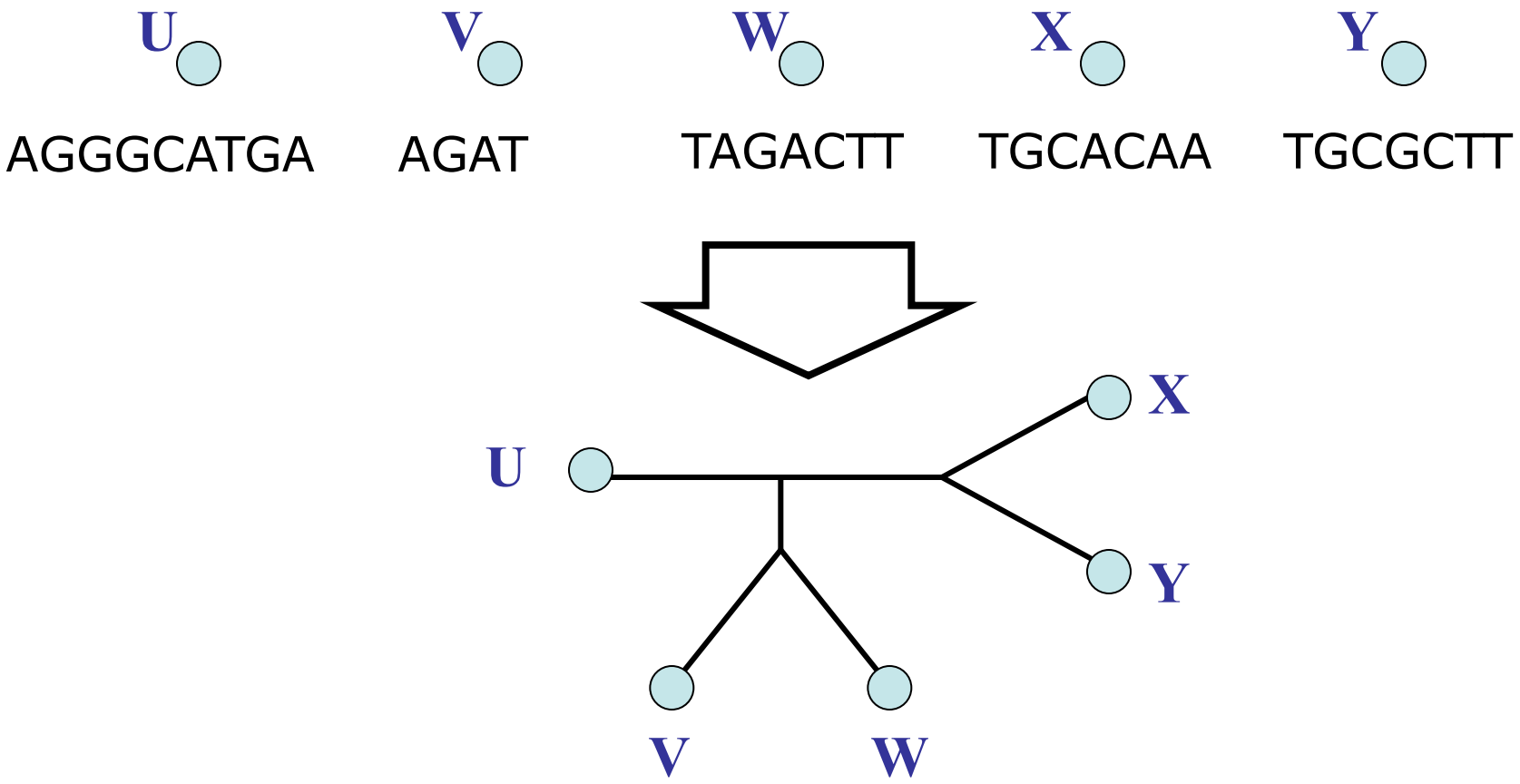

Theorem (Warnow et al., SODA 2001): DCM1-NJ converges to the true tree from polynomial length sequences. Hence DCM1-NJ is afc.

Proof: uses chordal graph theory and probabilistic analysis of algorithms

# Questions

- Is the model tree identifiable?
- Which estimation methods are statistically consistent under this model?
- How much data does the method need to estimate the model tree correctly (with high probability)?
- What is the computational complexity of an estimation problem?

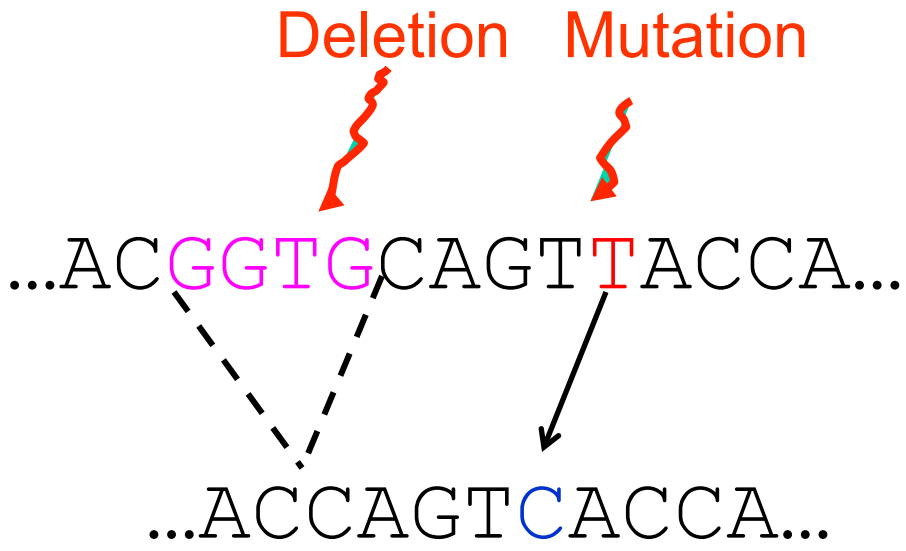
# The “real” problem

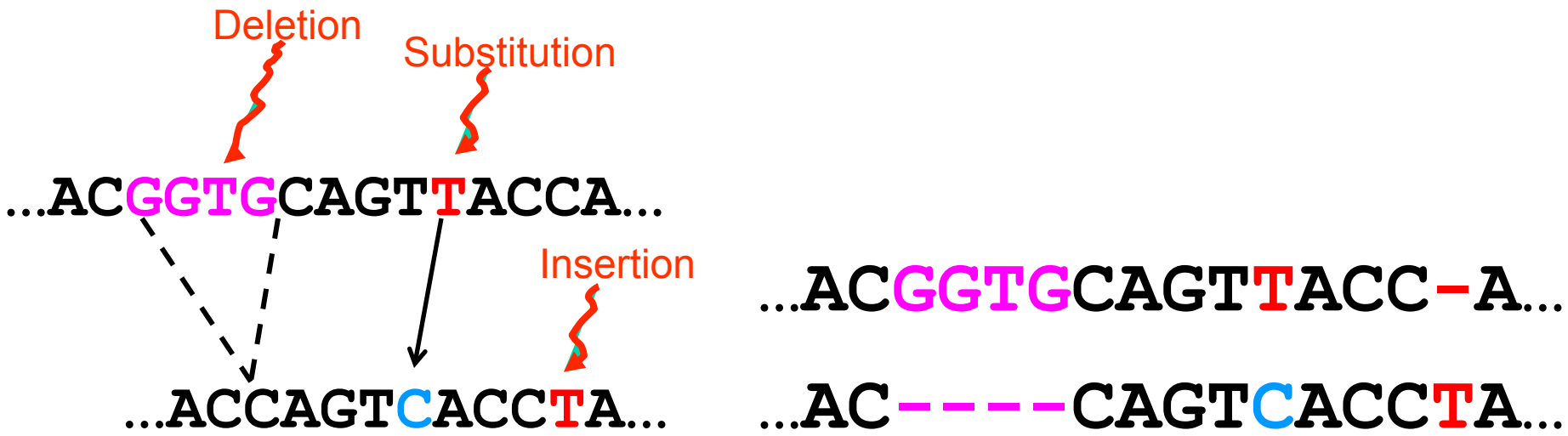


# II: Multiple Sequence Alignment

- Indels, and why we need to align sequences
- Poor performance of standard methods on large datasets
- SATé (Liu et al., Science 2009 and Systematic Biology 2012)
- UPP (Nguyen, Mirarab, and Warnow, in preparation)
- The “HMM Families” Technique

# Indels (insertions and deletions)






**The true multiple alignment**

- **Reflects historical substitution, insertion, and deletion events**
- **Defined using transitive closure of pairwise alignments computed on edges of the true tree**

# Input: unaligned sequences

```
S1 = AGGCTATCACCTGACCTCCA
S2 = TAGCTATCACGACCGC
S3 = TAGCTGACCGC
S4 = TCACGACCGACA
```

# Phase 1: Alignment

```
S1 = AGGCTATCACCTGACCTCCA
S2 = TAGCTATCACGACCGC
S3 = TAGCTGACCGC
S4 = TCACGACCGACA
```



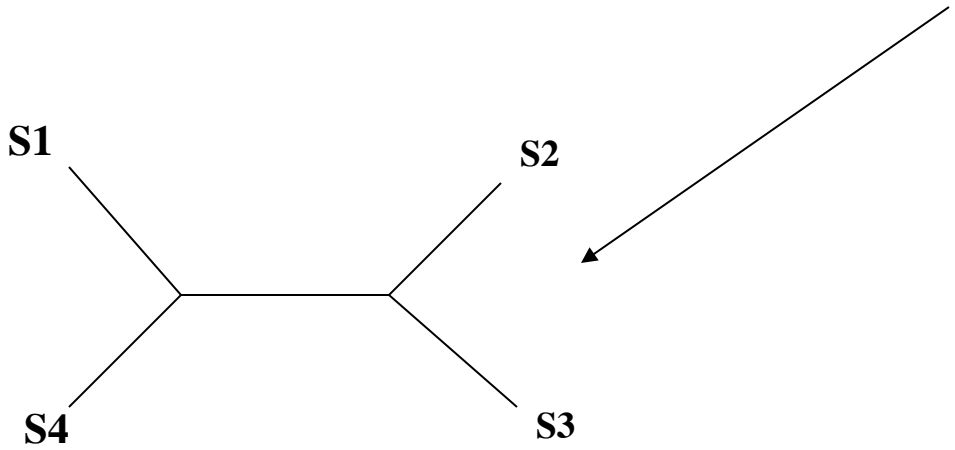
```
S1 = -AGGCTATCACCTGACCTCCA
S2 = TAG-CTATCAC--GACCGC--
S3 = TAG-CT-------GACCGC--
S4 = -------TCAC--GACCGACA
```

# Phase 2: Construct tree

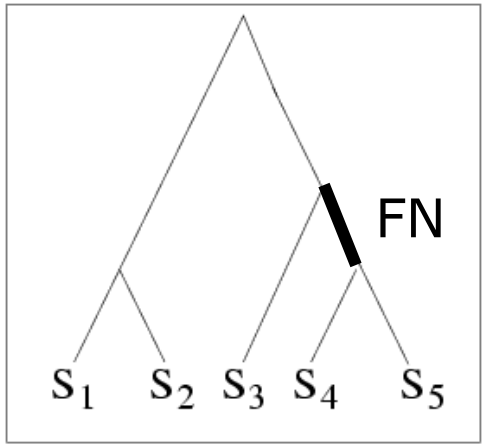
```
S1 = AGGCTATCACCTGACCTCCA
S2 = TAGCTATCACGACCGC
S3 = TAGCTGACCGC
S4 = TCACGACCGACA
```

→

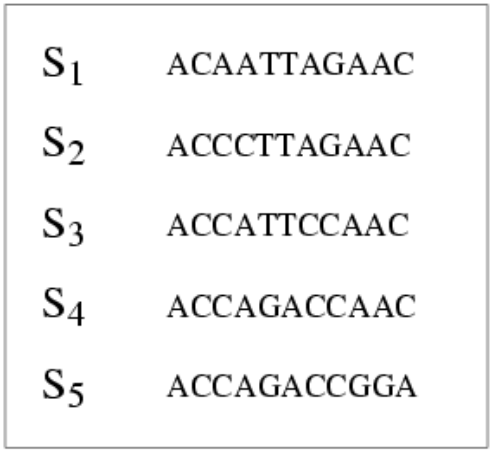
```
S1 = -AGGCTATCACCTGACCTCCA
S2 = TAG-CTATCAC--GACCGC--
S3 = TAG-CT-------GACCGC--
S4 = -------TCAC--GACCGACA
```



# Quantifying Error



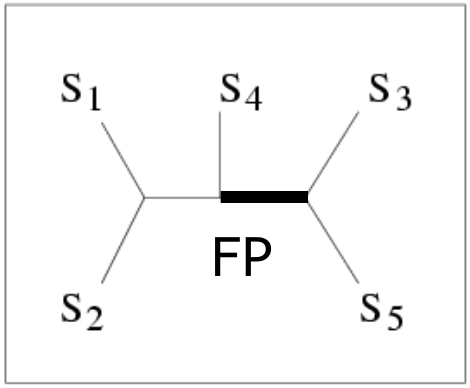

TRUE TREE




DNA SEQUENCES

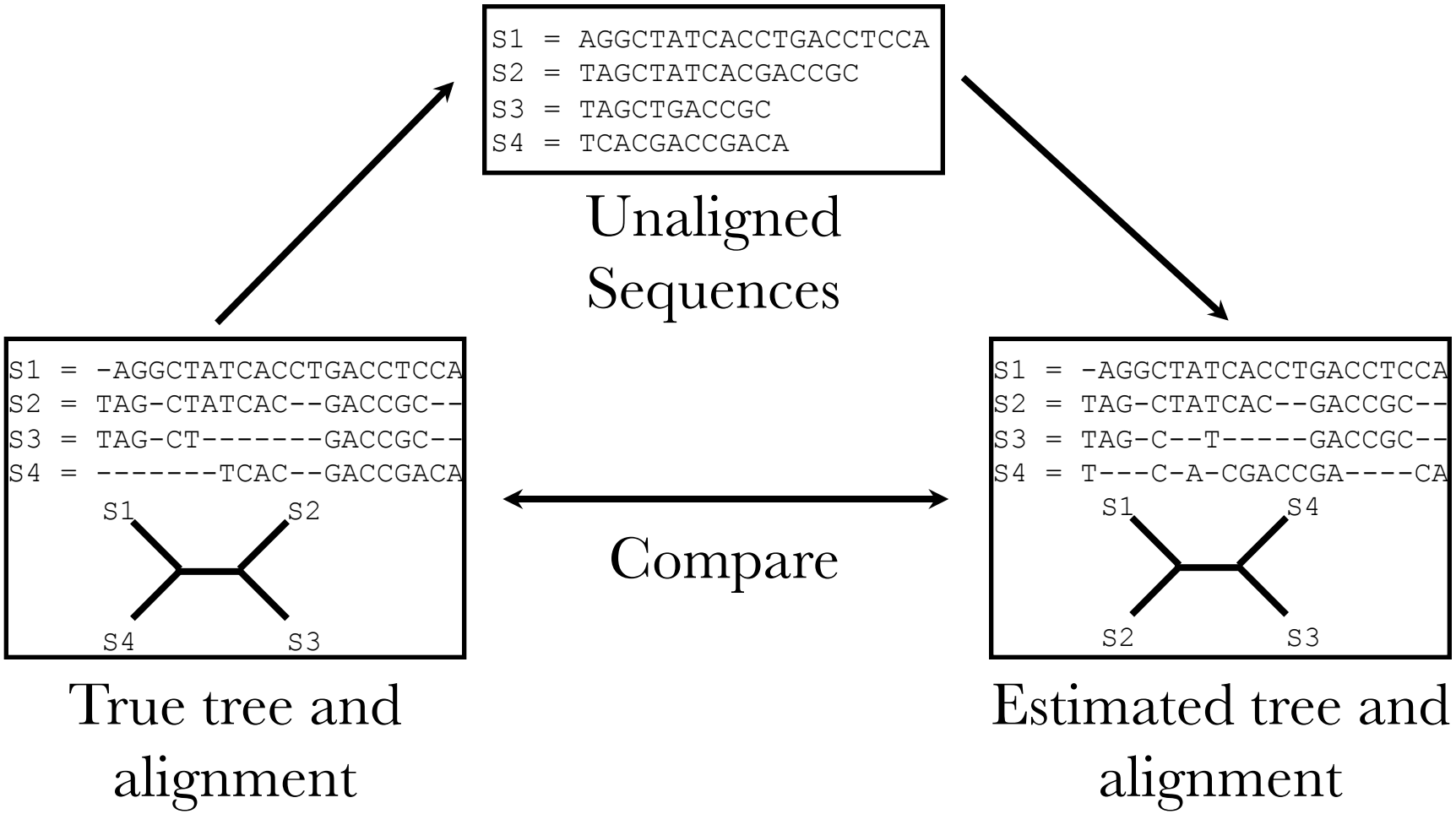
FN: false negative
(missing edge)
FP: false positive
(incorrect edge)

50% error rate




INFERRED TREE

# Simulation Studies



# Two-phase estimation

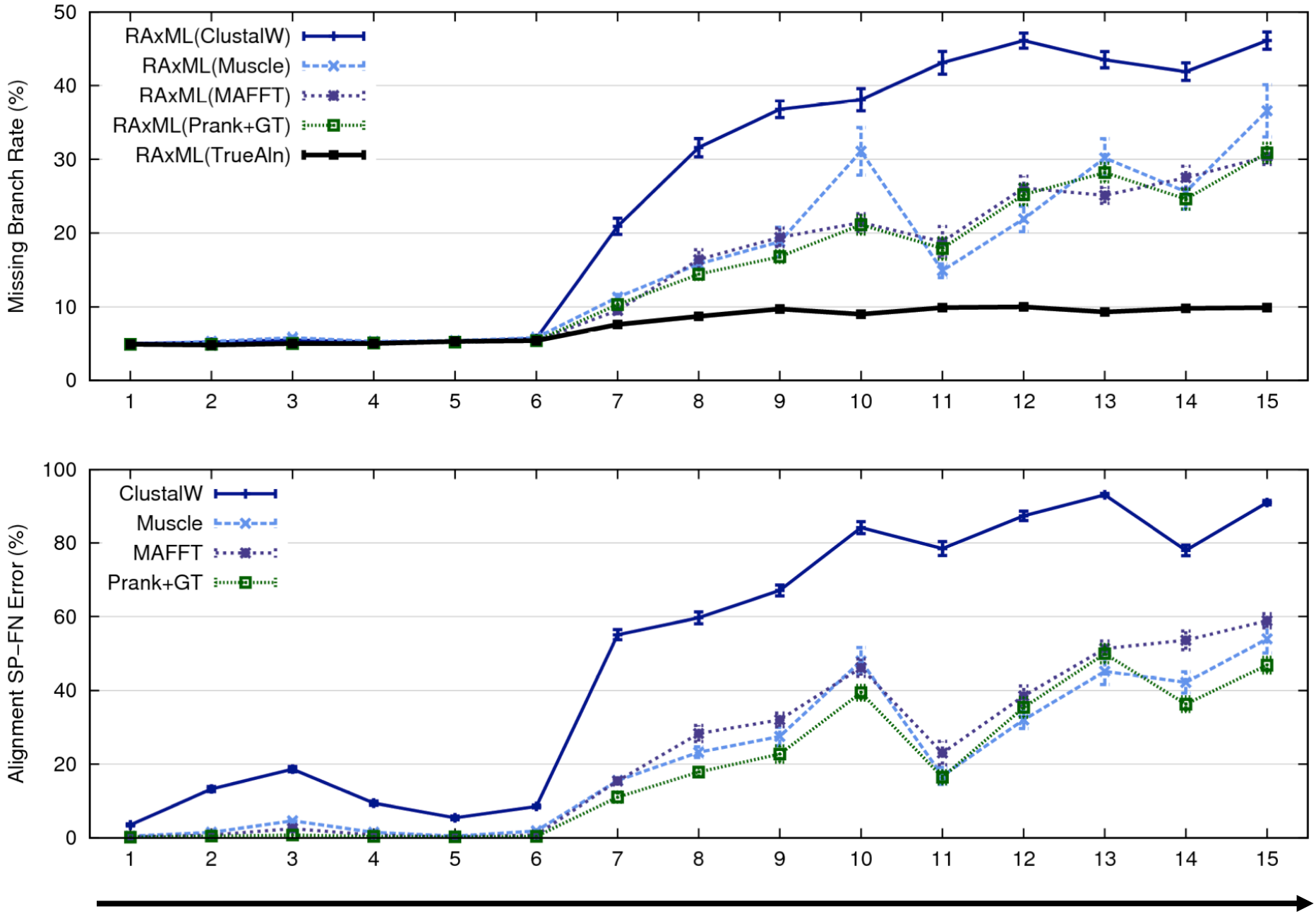
Alignment methods

- Clustal
- POY (and POY*)
- Probcons (and Probtree)
- Probalign
- MAFFT
- Muscle
- Di-align
- T-Coffee
- Prank (PNAS 2005, Science 2008)
- Opal (ISMB and Bioinf. 2007)
- *FSA (PLoS Comp. Bio. 2009)*
- *Infernal (Bioinf. 2009)*
- Etc.

Phylogeny methods

- Bayesian MCMC
- Maximum parsimony
- Maximum likelihood
- Neighbor joining
- FastME
- UPGMA
- Quartet puzzling
- Etc.

***RAxML**: heuristic for large-scale ML optimization*


RAxML(ClustalW)
RAxML(Muscle)
RAxML(MAFFT)
RAxML(Prank+GT)
RAxML(TrueAln)
Missing Branch Rate (%)
0
10
20
30
40
50
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
ClustalW
Muscle
MAFFT
Prank+GT
Alignment SP–FN Error (%)
0
20
40
60
80
100
1000-taxon models, ordered by difficulty (Liu et al., 2009)

# Problems with the two-phase approach

- Current alignment methods fail to return reasonable alignments on large datasets with high rates of indels and substitutions.
- Manual alignment is time consuming and subjective.
- *Systematists discard potentially useful markers* if they are difficult to align.

This issues seriously impact large-scale phylogeny estimation (and Tree of Life projects)

# 1kp: Thousand Transcriptome Project

G. Ka-Shu Wong
U Alberta

J. Leebens-Mack
U Georgia

N. Wickett
Northwestern

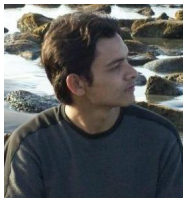
N. Matasci
iPlant

T. Warnow,
UT-Austin

S. Mirarab,
UT-Austin

N. Nguyen,
UT-Austin

Md. S.Bayzid
UT-Austin

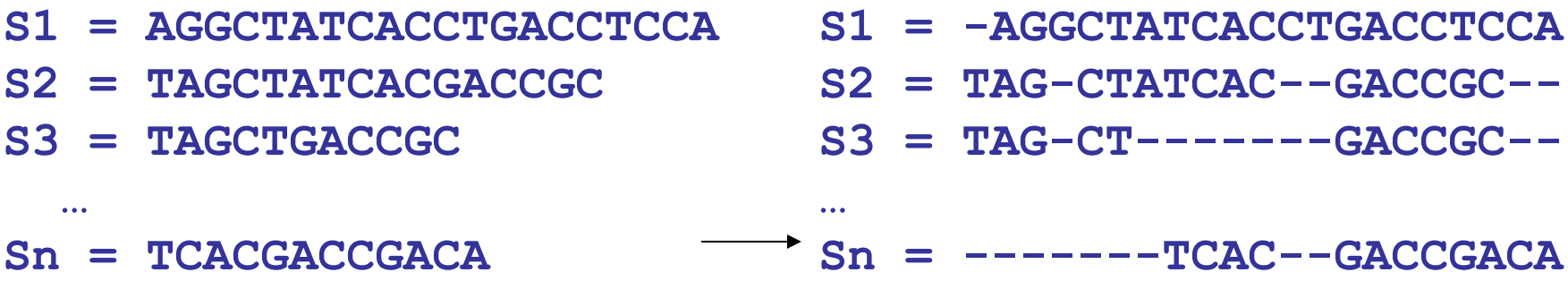


Plus many many other people…

- Plant Tree of Life based on transcriptomes of ~1200 species
- More than 13,000 gene families (most not single copy)

**Challenge:**
**Alignment of datasets with > 100,000 sequences**

# Multiple Sequence Alignment (MSA): *another grand challenge*[1]




*Novel techniques needed* for scalability and accuracy

NP-hard problems and large datasets

Current methods do not provide good accuracy

Few methods can analyze even moderately large datasets

*Many important applications besides phylogenetic estimation*

[1] Frontiers in Massive Data Analysis, National Academies Press, 2013

# Ultra-large Alignment

**SATé** - co-estimating trees and alignments (Science, 2009 and Systematic Biology 2012)

**UPP** - ultra-large alignment estimation using SEPP (unpublished)

Very few other methods for ultra-large alignment

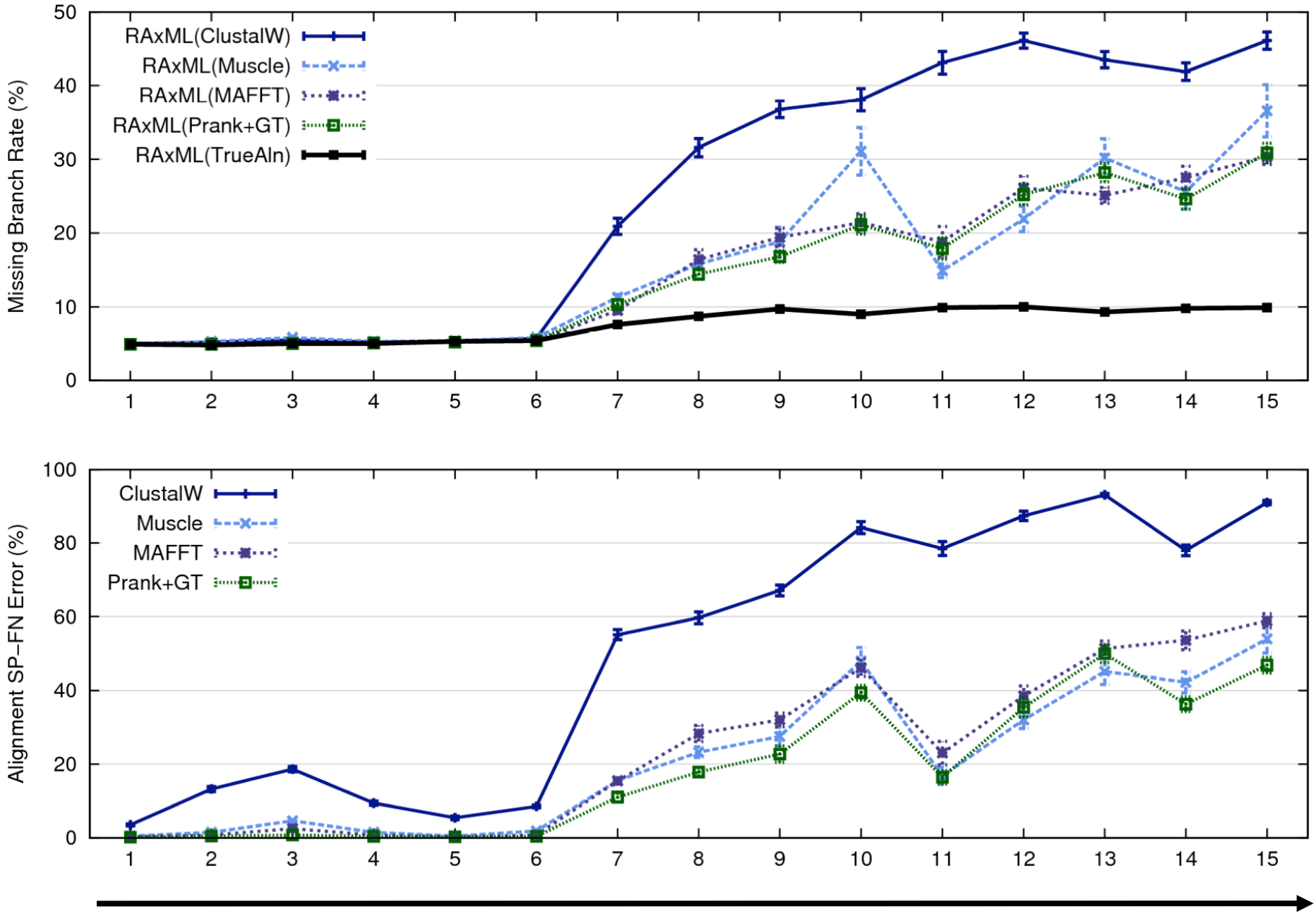
# SATé

Simultaneous Alignment and Tree Estimation

Liu, Nelesen, Raghavan, Linder, and Warnow, *Science*, 19 June 2009, pp. 1561-1564.

Liu et al., Systematic Biology 2012

Public software distribution (open source) through Mark Holder's group at the University of Kansas

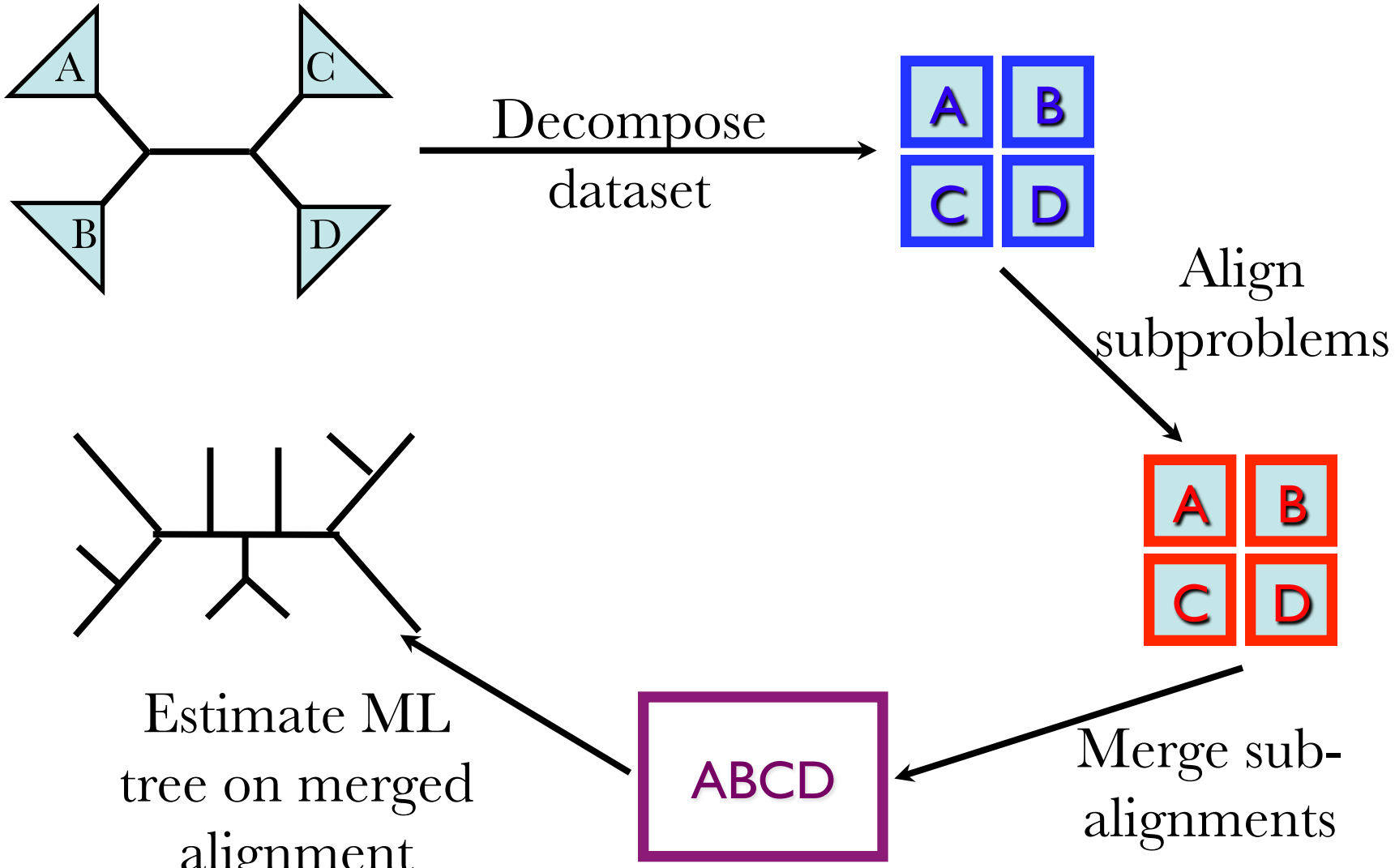

RAxML(ClustalW)
RAxML(Muscle)
RAxML(MAFFT)
RAxML(Prank+GT)
RAxML(TrueAln)
Missing Branch Rate (%)
0
10
20
30
40
50
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
ClustalW
Muscle
MAFFT
Prank+GT
Alignment SP–FN Error (%)
0
20
40
60
80
100
1000-taxon models, ordered by difficulty (Liu et al., 2009)

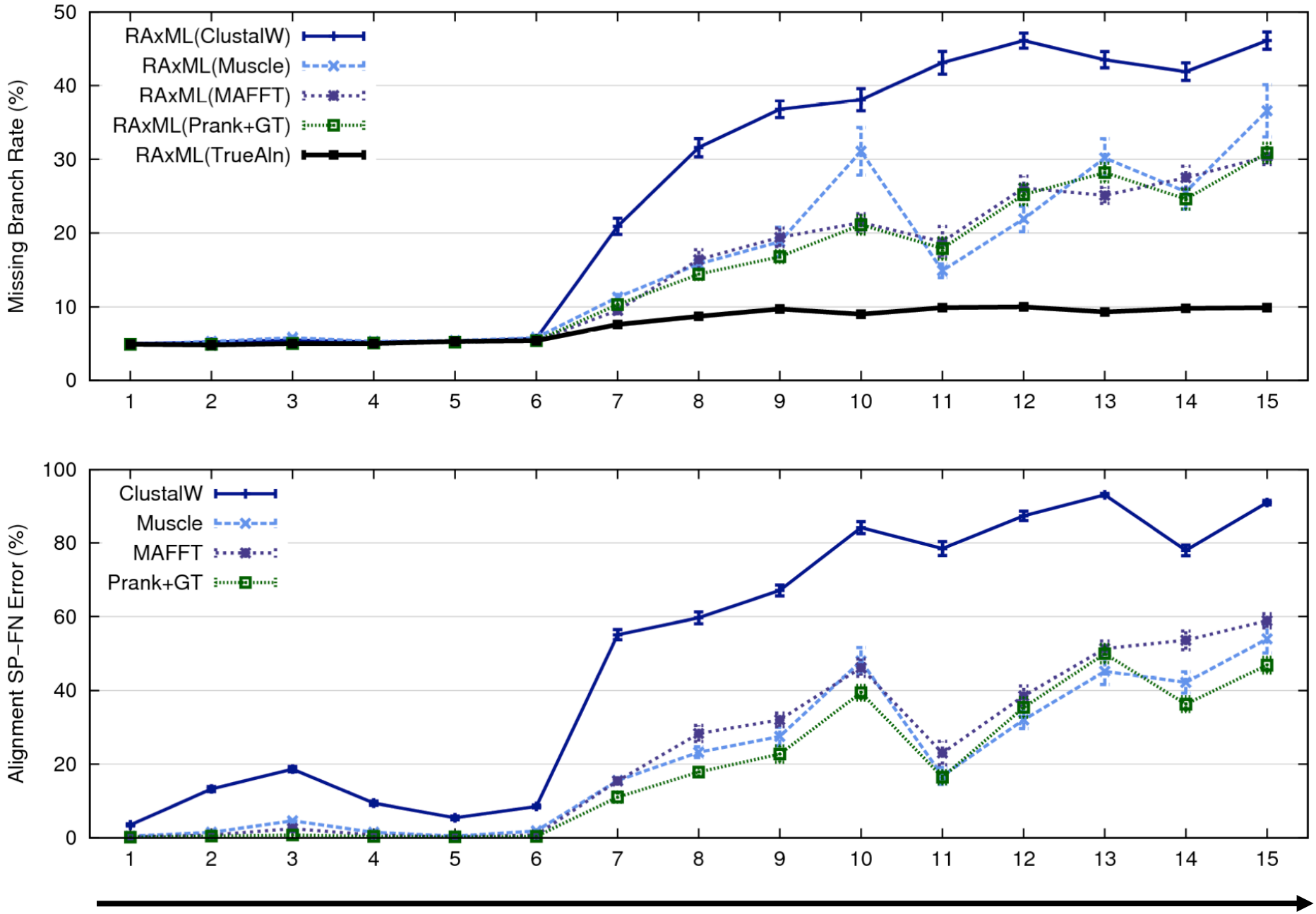
# Two-phase estimation

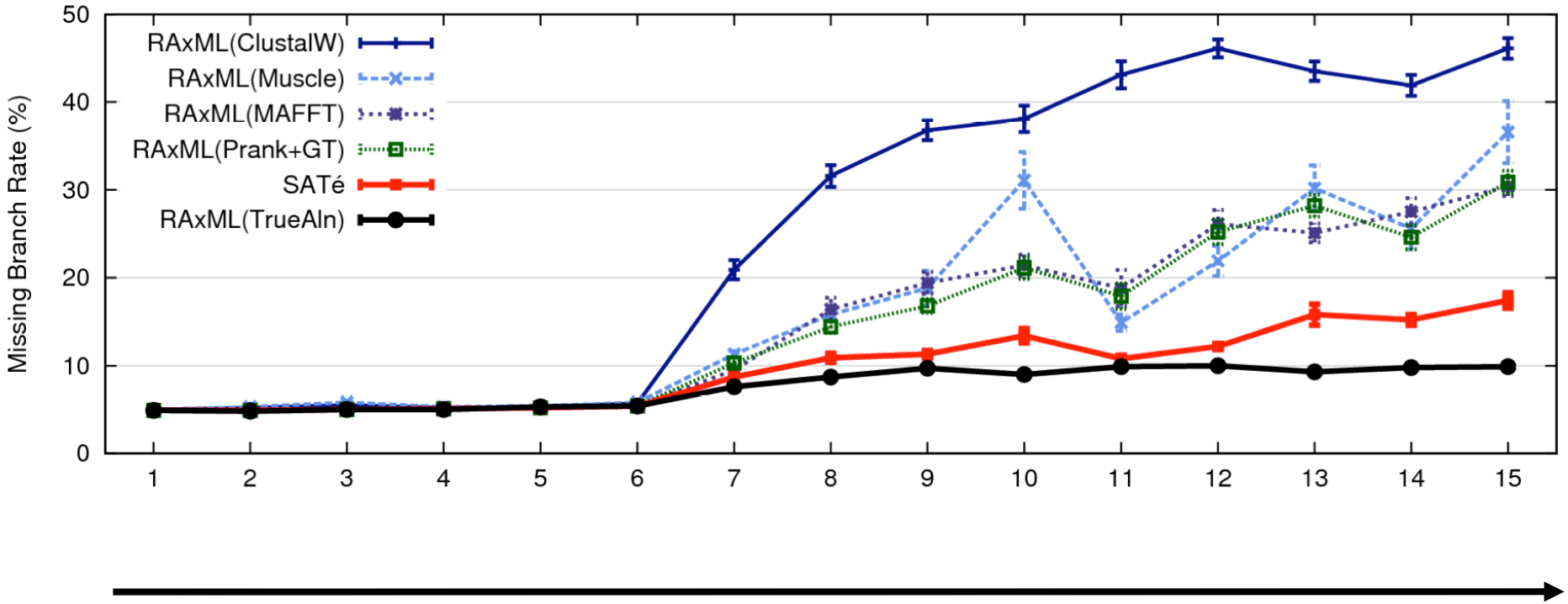
- Alignment error increases with the rate of evolution, and poor alignments result in poor trees.

- Datasets with small enough "evolutionary diameters" are easy to align with high accuracy.

# One SATé iteration (cartoon)



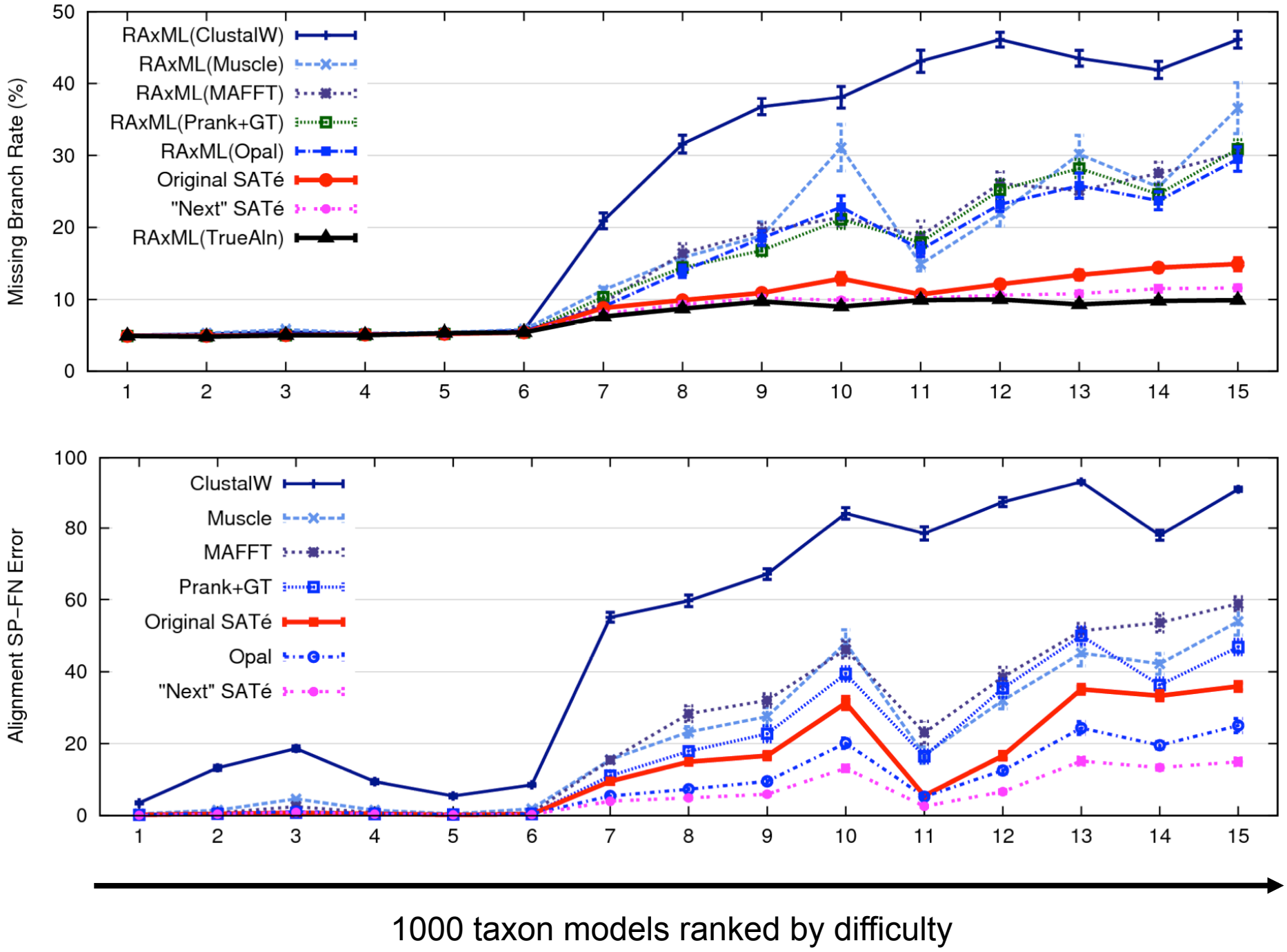

RAxML(ClustalW)
RAxML(Muscle)
RAxML(MAFFT)
RAxML(Prank+GT)
RAxML(TrueAln)
Missing Branch Rate (%)
0
10
20
30
40
50
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
ClustalW
Muscle
MAFFT
Prank+GT
Alignment SP–FN Error (%)
0
20
40
60
80
100
1000-taxon models, ordered by difficulty (Liu et al., 2009)



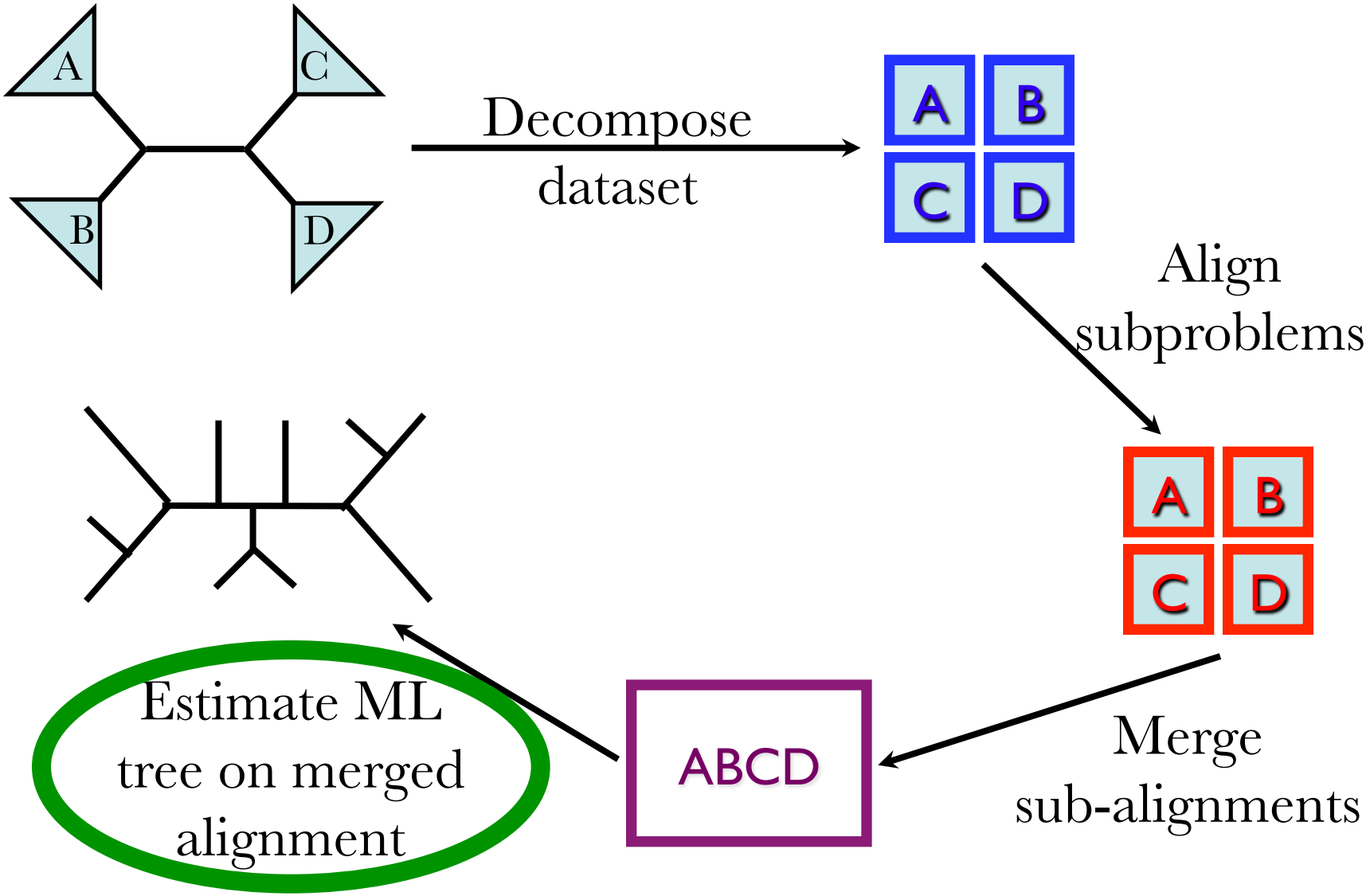

1000 taxon models, ordered by difficulty

24 hour SATé analysis, on desktop machines

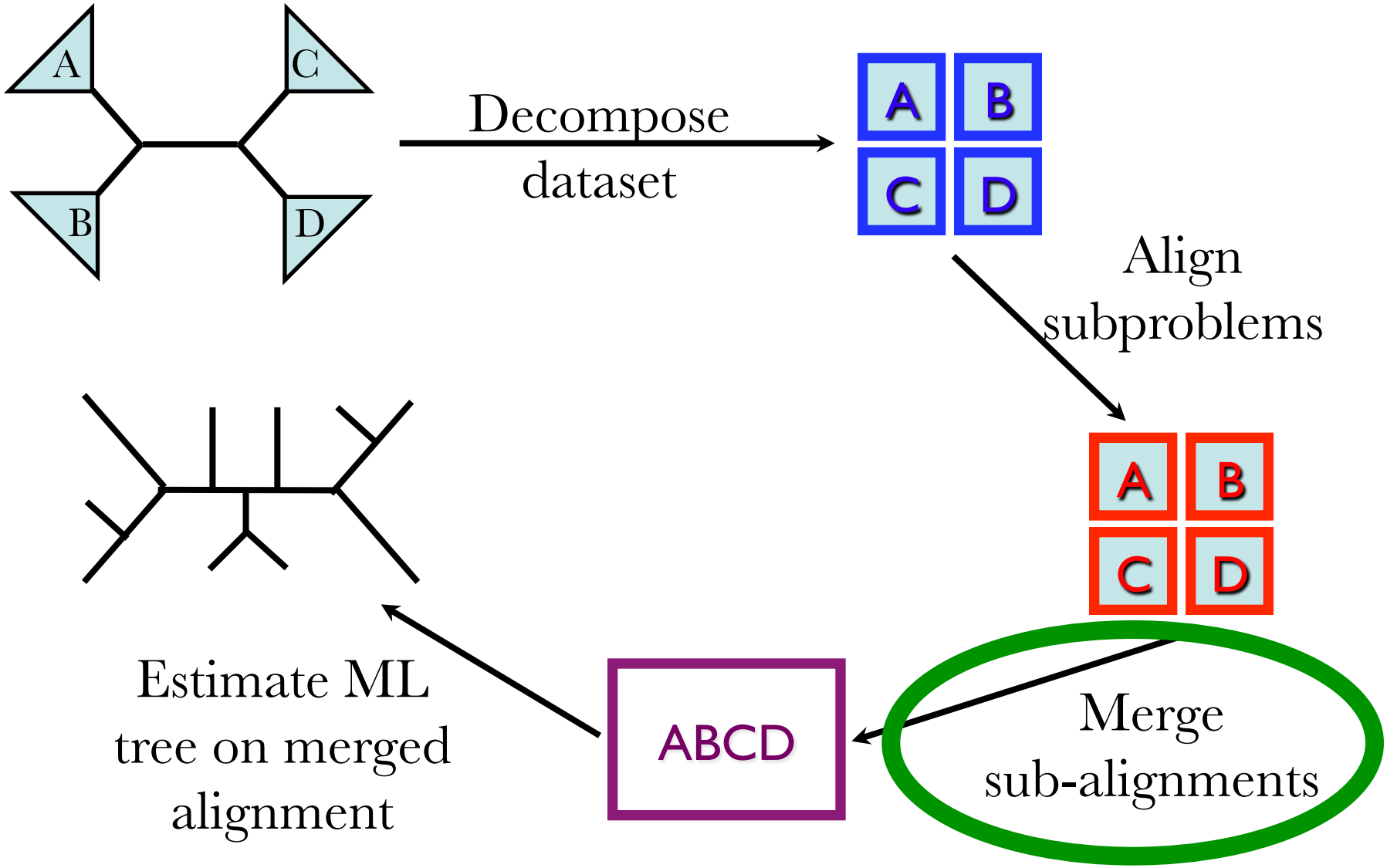
(Similar improvements for biological datasets)


Missing Branch Rate (%)
RAxML(ClustalW)
RAxML(Muscle)
RAxML(MAFFT)
RAxML(Prank+GT)
RAxML(Opal)
Original SATé
"Next" SATé
RAxML(TrueAln)
Alignment SP–FN Error
ClustalW
Muscle
MAFFT
Prank+GT
Original SATé
Opal
"Next" SATé
1000 taxon models ranked by difficulty

# Limitations



# Limitations



# UPP: Ultra-large alignment using SEPP[1]

**Objective: highly accurate multiple sequence alignments and trees on ultra-large datasets**

Authors: Nam Nguyen, Siavash Mirarab, and Tandy Warnow

In preparation – expected submission Fall 2013

[1] SEPP: SATe-enabled phylogenetic placement, Nguyen, Mirarab, and Warnow, PSB 2012

# UPP: basic idea

Input: set S of unaligned sequences

Output: alignment on S

- Select random subset X of S
- Estimate “backbone” alignment A and tree T on X
- Independently align each sequence in S-X to A
- Use transitivity to produce multiple sequence alignment A* for entire set S

# Input: Unaligned Sequences

```
S1  = AGGCTATCACCTGACCTCCAAT
S2  = TAGCTATCACGACCGCGCT
S3  = TAGCTGACCGCGCT
S4  = TACTCACGACCGACAGCT
S5  = TAGGTACAACCTAGATC
S6  = AGATACGTCGACATATC
```

# Step 1: Pick random subset (backbone)

```
S1  = AGGCTATCACCTGACCTCCAAT
S2  = TAGCTATCACGACCGCGCT
S3  = TAGCTGACCGCGCT
S4  = TACTCACGACCGACAGCT
S5  = TAGGTACAACCTAGATC
S6  = AGATACGTCGACATATC
```

# Step 2: Compute backbone alignment

```
S1  = -AGGCTATCACCTGACCTCCA-AT
S2  = TAG-CTATCAC--GACCGC--GCT
S3  = TAG-CT-------GACCGC--GCT
S4  = TAC----TCAC--GACCGACAGCT
S5  = TAGGTAAAACCTAGATC
S6  = AGATAAAACTACATATC
```

# Step 3: Align each remaining sequence to backbone

First we add S5 to the backbone alignment

```
S1  = -AGGCTATCACCTGACCTCCA-AT-
S2  = TAG-CTATCAC--GACCGC--GCT-
S3  = TAG-CT-------GACCGC--GCT-
S4  = TAC----TCAC--GACCGACAGCT-
S5  = TAGG---T-A-CAA-CCTA--GATC
```

# Step 3: Align each remaining sequence to backbone

Then we add S6 to the backbone alignment

```
S1  = -AGGCTATCACCTGACCTCCA-AT-
S2  = TAG-CTATCAC--GACCGC--GCT-
S3  = TAG-CT-------GACCGC--GCT-
S4  = TAC----TCAC--GACCGACAGCT-
S6  = -AG---AT-A-CGTC--GACATATC
```

# Step 4: Use transitivity to obtain MSA on entire set

```
S1  = -AGGCTATCACCTGACCTCCA-AT--
S2  = TAG-CTATCAC--GACCGC--GCT--
S3  = TAG-CT-------GACCGC--GCT--
S4  = TAC----TCAC--GACCGACAGCT--
S5  = TAGG---T-A—CAA-CCTA--GATC-
S6  = -AG---AT-A-CGTC--GACATAT-C
```

# UPP: details

Input: set S of unaligned sequences

Output: alignment on S

- Select random subset X of S
- Estimate “backbone” alignment A and tree T on X
- Independently align each sequence in S-X to A
- Use transitivity to produce multiple sequence alignment A* for entire set S

# UPP: details

Input: set S of unaligned sequences

Output: alignment on S

- Select random subset X of S
- Estimate “backbone” alignment A and tree T on X
- Independently align each sequence in S-X to A
- Use transitivity to produce multiple sequence alignment A* for entire set S

# How to align sequences to a backbone alignment?

Standard machine learning technique:

- Build HMM (Hidden Markov Model) for backbone alignment, and use it to align remaining sequences

We use HMMER (Sean Eddy, HHMI) for this purpose

# Using HMMER

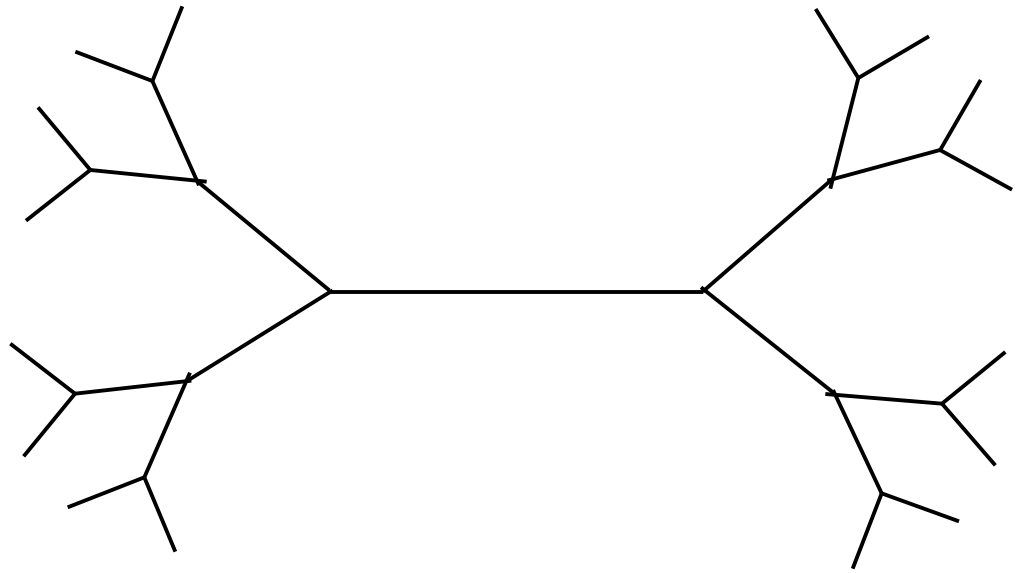
Using HMMER works well…

# Using HMMER

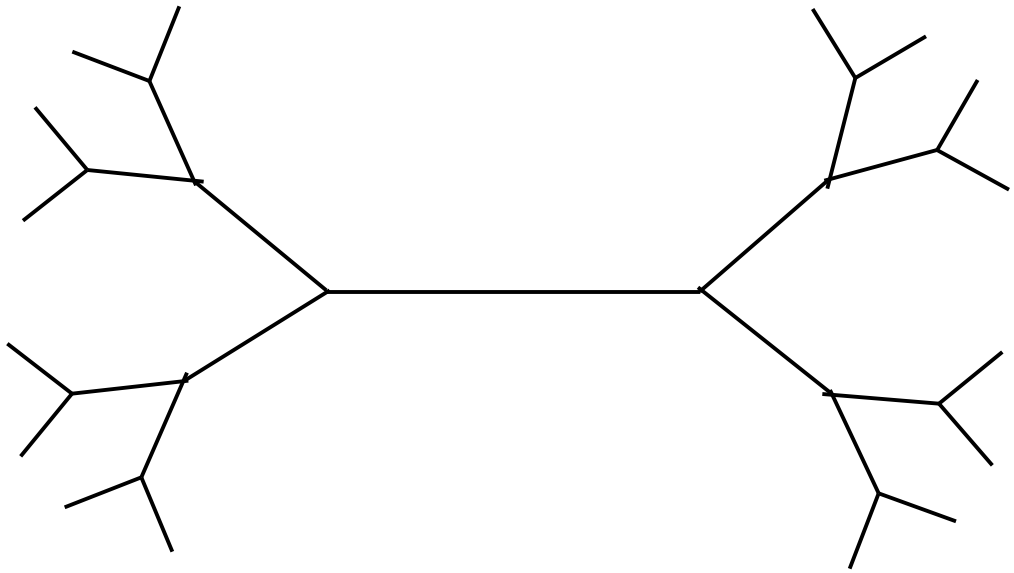
Using HMMER works well…except when the dataset is big!

Using HMMER to add sequences to an existing alignment

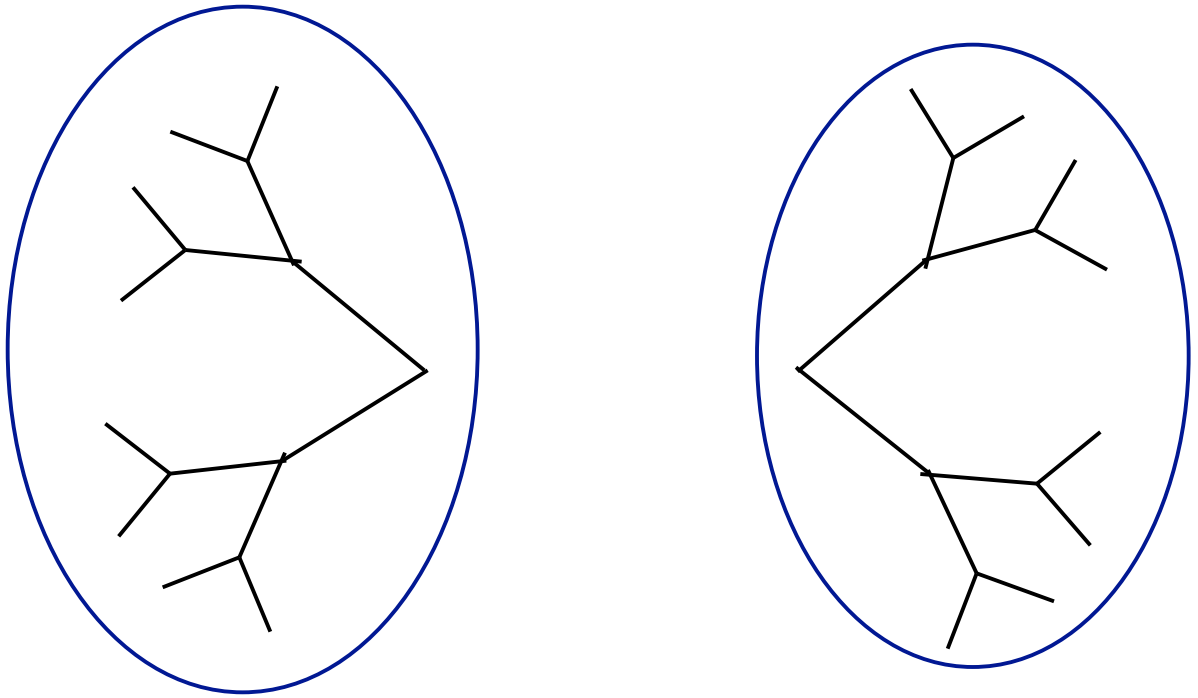
1) build one HMM for the backbone alignment
2) Align sequences to the HMM, and insert into backbone alignment



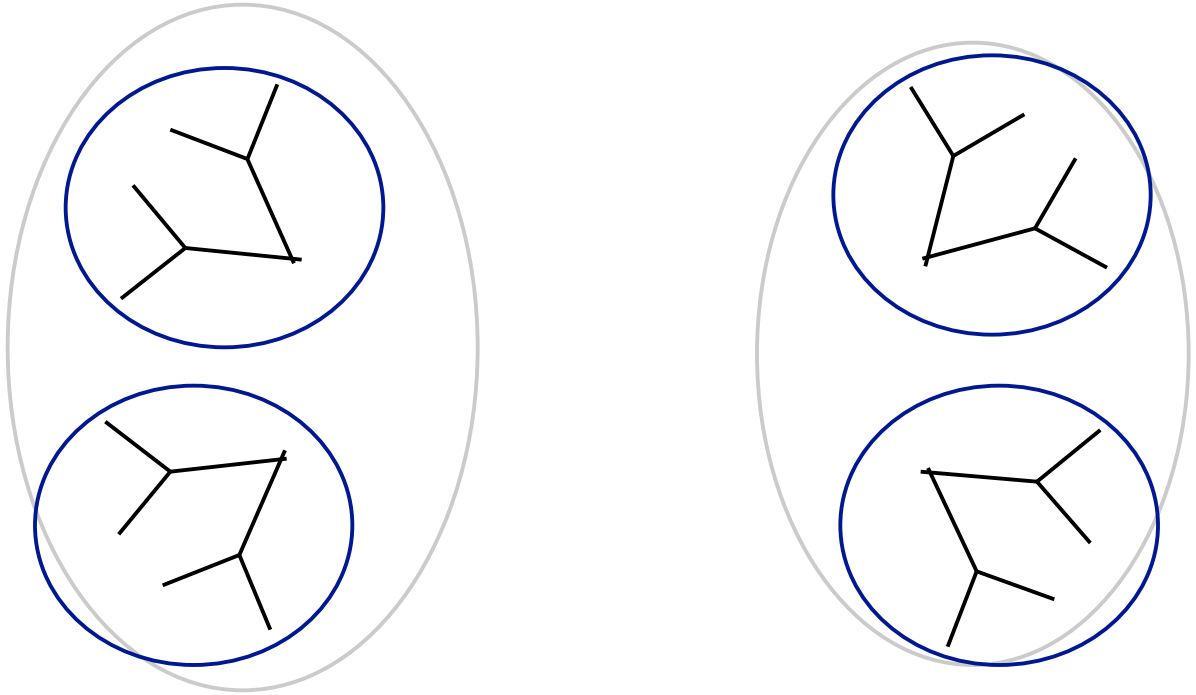
# One Hidden Markov Model for the entire alignment?



# Or 2 HMMs?



# Or 4 HMMs?



# UPP(x,y)

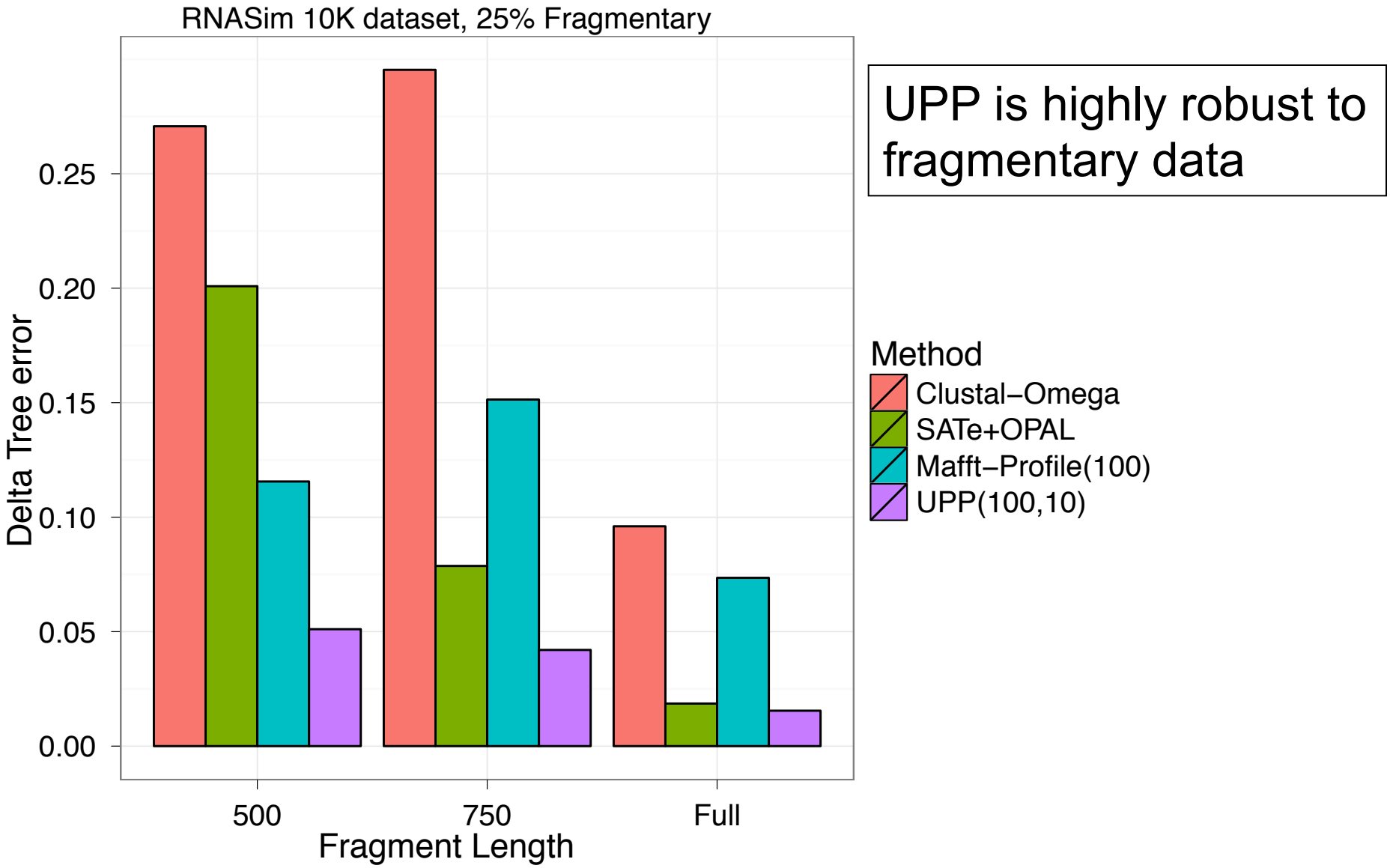
- Pick random subset X of size **x**
- Compute alignment A and tree T on X
- Use SATé decomposition on T to partition X into small “alignment subsets” of at most **y** sequences
- Build HMM on each alignment subset using HMMBUILD
- For each sequence s in S-X,
  - use HMMALIGN to produce alignment of s to each subset alignment and note the score of each alignment.
  - Pick the subset alignment that has the best score, and align s to that subset alignment.
  - Use transitivity to align s to the backbone alignment.

# UPP design

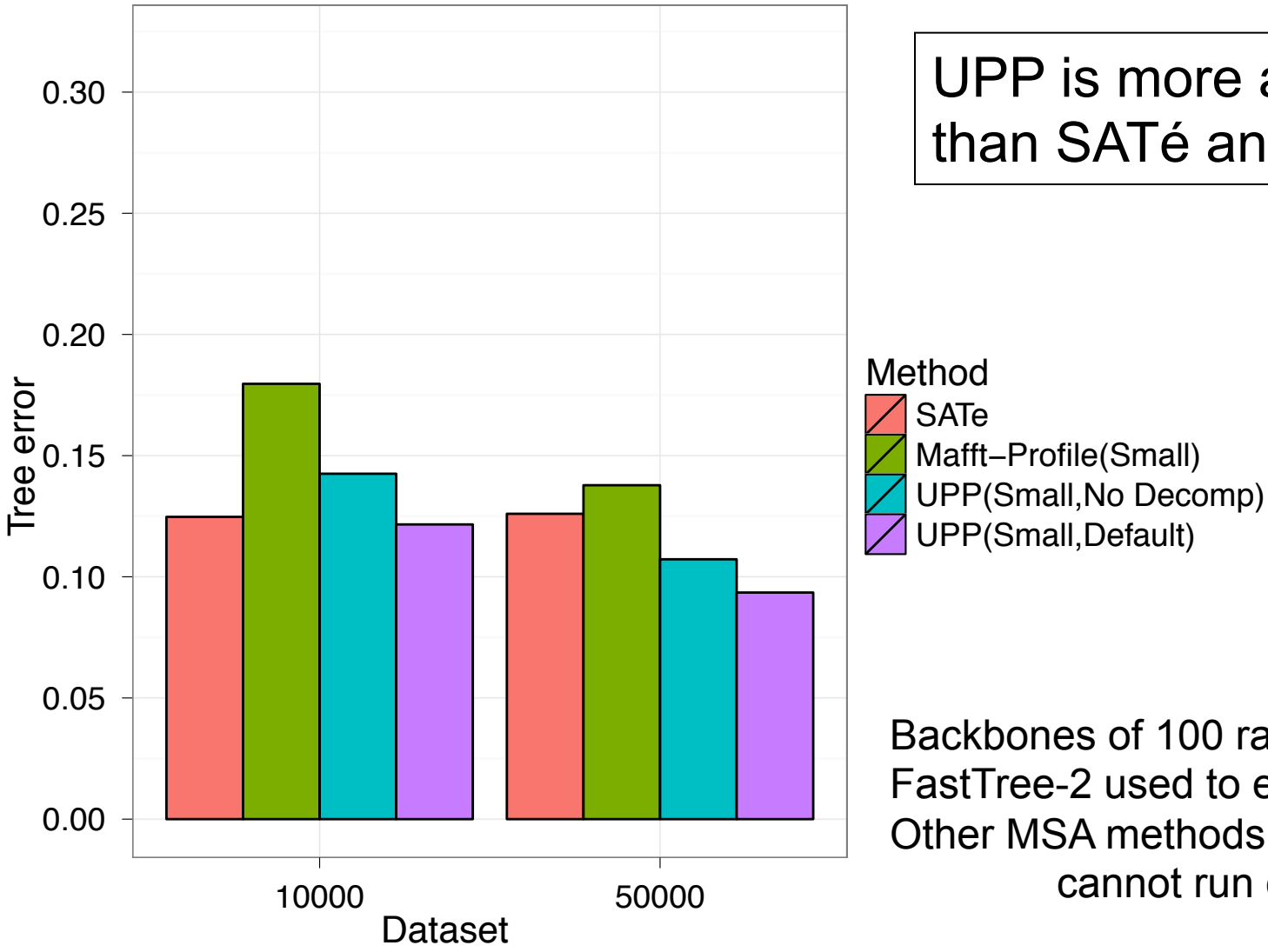
- Size of backbone matters – small backbones are sufficient for most datasets (except for ones with very high rates of evolution). Random backbones are fine.
- Number of HMMs matters, and depends on the rate of evolution and number of taxa.
- Backbone alignment and tree matter; we use SATé.

# Evaluation of UPP

- Simulated Datasets: 10,000 to 1,000,000 sequences (RNASim, Junhyong Kim, U Penn)
- Biological datasets with reference alignments (Gutell's CRW data with up to 28,000 sequences)
- MSA methods: MAFFT-profile, Clustal-Omega, SATé, Muscle, and others
- ML Tree estimation: FastTree-2
- Criteria: Alignment error (SP-FN and SP-FP), tree error, and time


RNASim 10K dataset, 25% Fragmentary
Delta Tree error
0.25
0.20
0.15
0.10
0.05
0.00
500
750
Full
Fragment Length
UPP is highly robust to fragmentary data
Method
Clustal−Omega
SATe+OPAL
Mafft−Profile(100)
UPP(100,10)

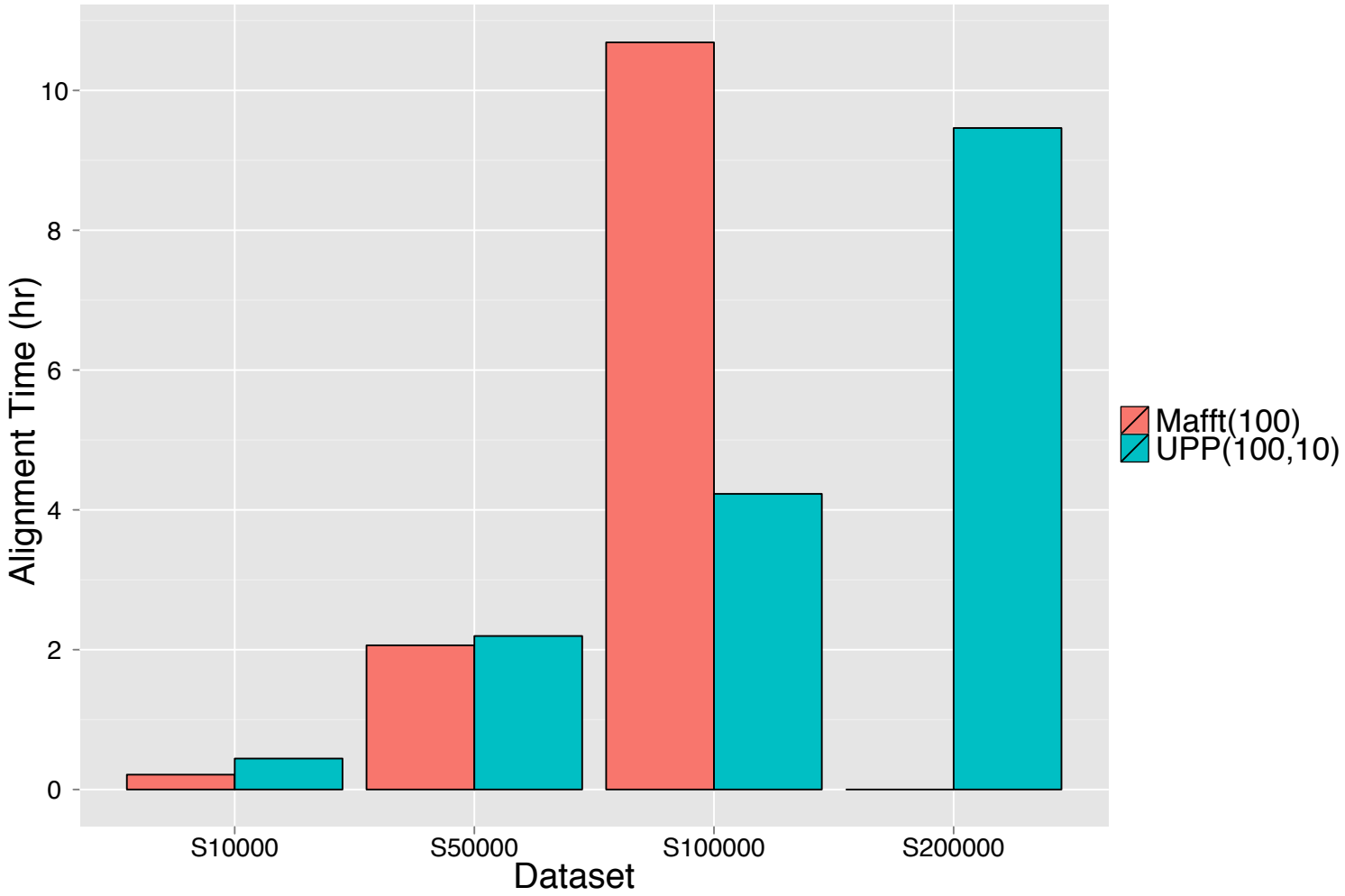
# Tree Error on 10K and 50K RNASim datasets



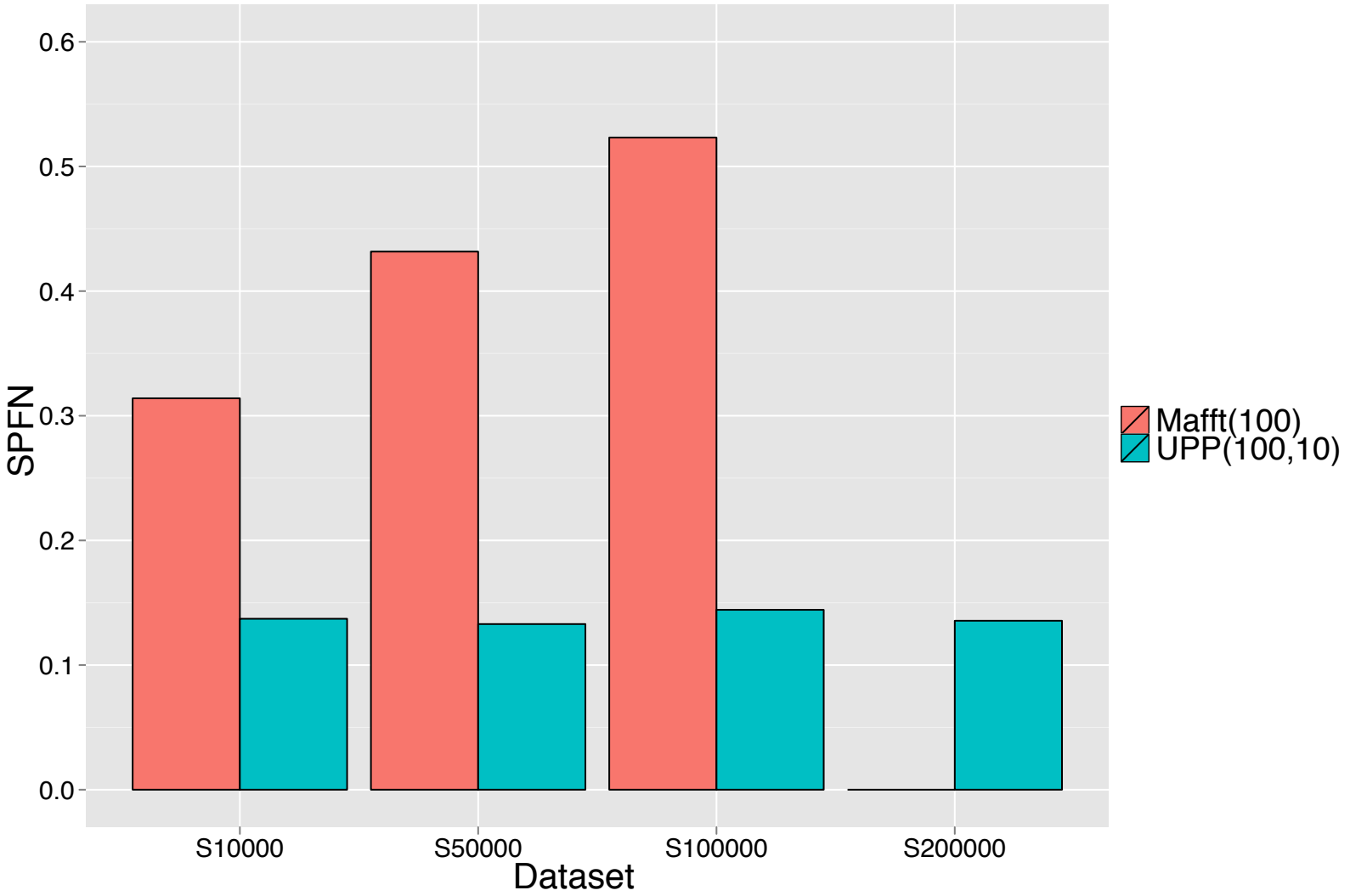

UPP is more accurate than SATé and MAFFT

Backbones of 100 random sequences
FastTree-2 used to estimate ML trees
Other MSA methods less accurate or cannot run on these data

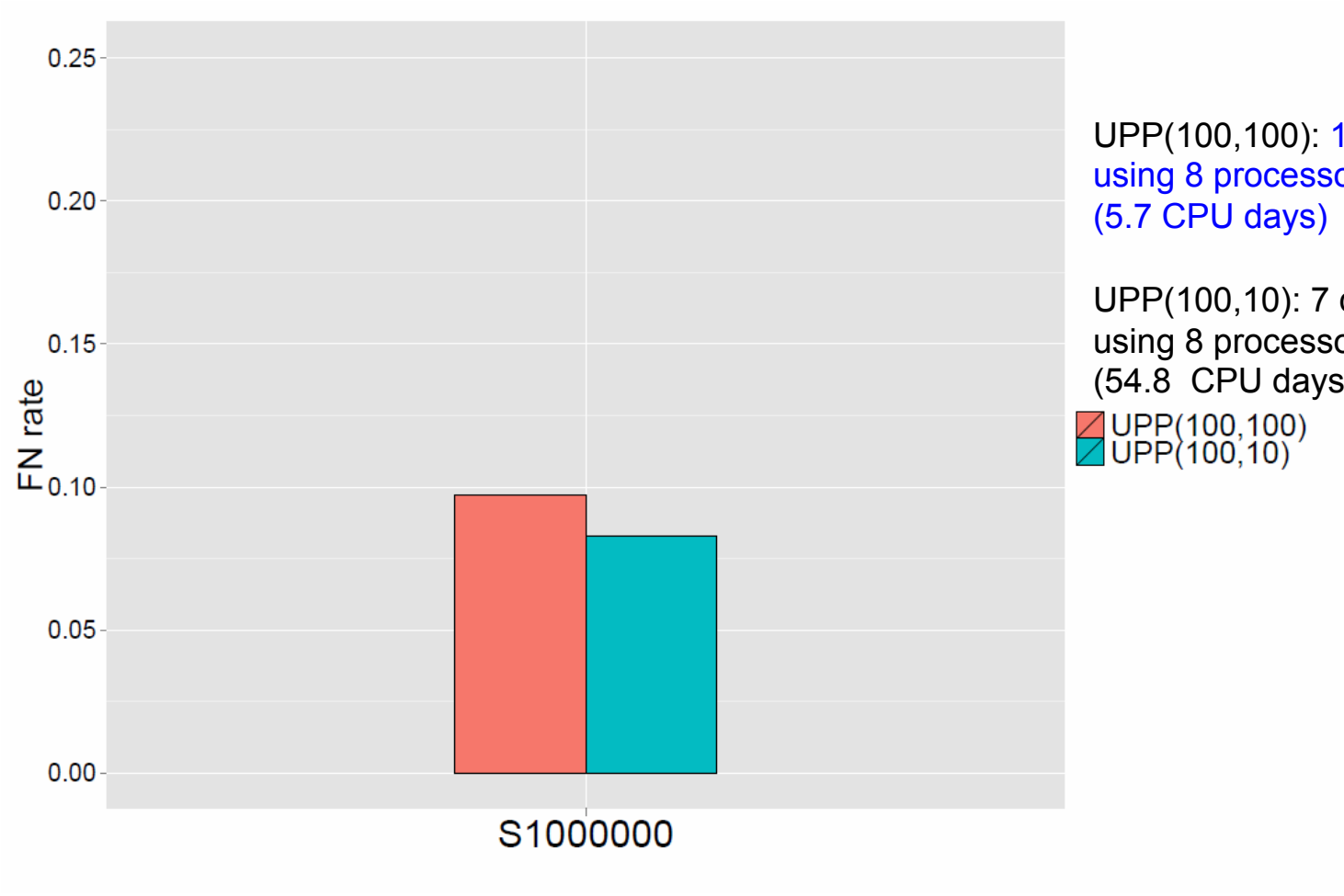
# UPP vs. MAFFT-profile Running Time



# UPP vs. MAFFT-profile Alignment Error



# One Million Sequences: Tree Error




Note: UPP Decomposition improves accuracy

# UPP performance

- UPP is very fast, parallelizable, and scalable. UPP can analyze very large datasets (up to 1,000,000 sequences so far).
- UPP is highly robust to fragmentary datasets, where it has by far the best accuracy of all methods.
- On full length sequences:
  - UPP is generally the only method that can run on very large datasets in reasonable timeframes.
  - UPP is more accurate than all other methods on the largest datasets (50,000 sequences and up) and most of the smaller datasets.
  - On small enough datasets (under 1000 sequences or so), UPP alignments are comparable to SATé but SATé produces slightly better trees. UPP produces more accurate alignments than the other alignment methods, and the next most accurate method is MAFFT-profile.

# UPP “HMM Family” technique

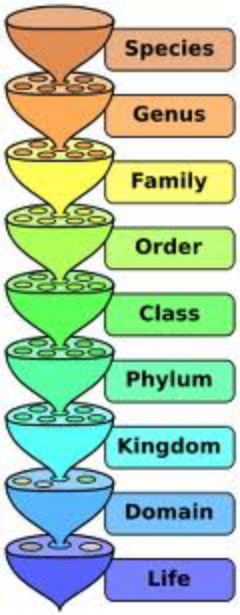
- Uses multiple HMMs to represent a multiple sequence alignment (each on a different subset of the sequences).
- Random decompositions are not as helpful as tree-based decomposition.
- UPP decompositions do not necessarily produce “clades”.

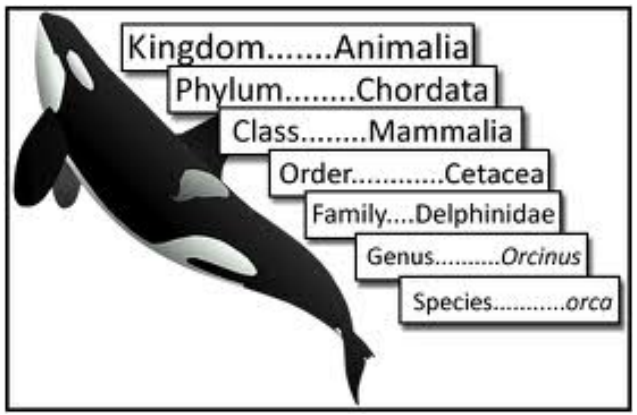
# Other uses of HMM Families

- SEPP: SATé-enabled phylogenetic placement (PSB 2012)
- TIPP: Taxonomic Identification using SEPP (in preparation, collaboration with Mihai Pop, Maryland)

# Part III: Metagenomic Taxon Identification

Objective: classify short reads in a metagenomic sample

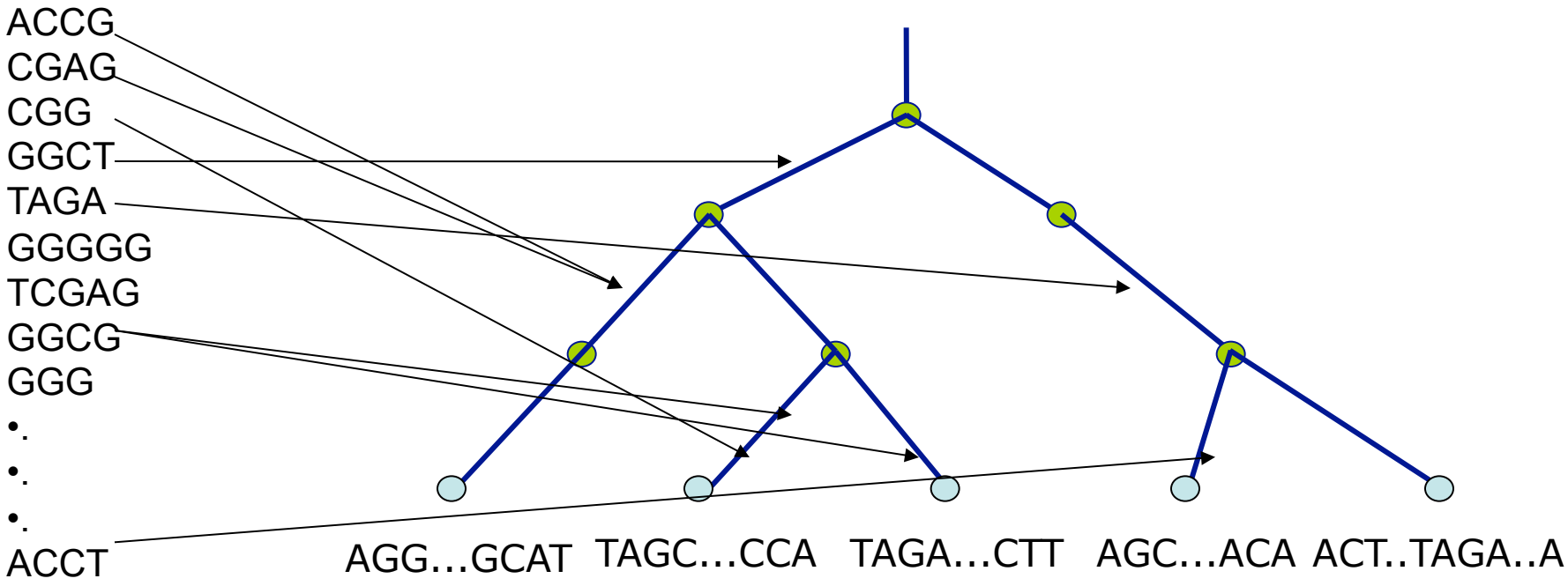


# Phylogenetic Placement

Fragmentary sequences from some gene

Full-length sequences for same gene, and an alignment and a tree



# SEPP

- **SEPP: SATé-enabled Phylogenetic Placement**, by Mirarab, Nguyen, and Warnow

- Pacific Symposium on Biocomputing, 2012 (special session on the Human Microbiome)

# Phylogenetic Placement

Step 1: Align each query sequence to backbone alignment

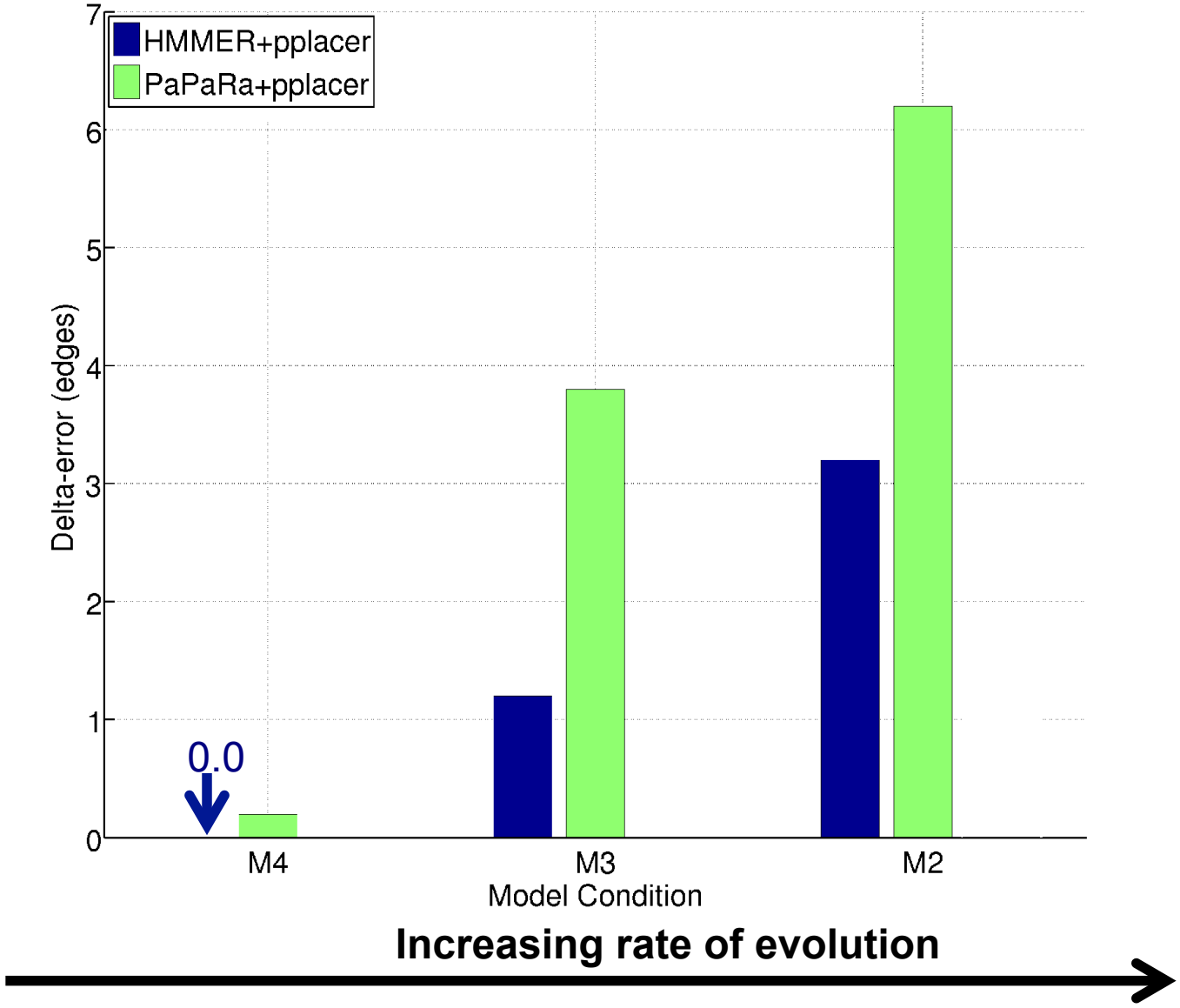
Step 2: Place each query sequence into backbone tree, using extended alignment

# Phylogenetic Placement

- Align each query sequence to backbone alignment
  - HMMALIGN (Eddy, Bioinformatics 1998)
  - PaPaRa (Berger and Stamatakis, Bioinformatics 2011)
- Place each query sequence into backbone tree
  - Pplacer (Matsen et al., BMC Bioinformatics, 2011)
  - EPA (Berger and Stamatakis, Systematic Biology 2011)

Note: pplacer and EPA use maximum likelihood, and are reported to have the same accuracy.

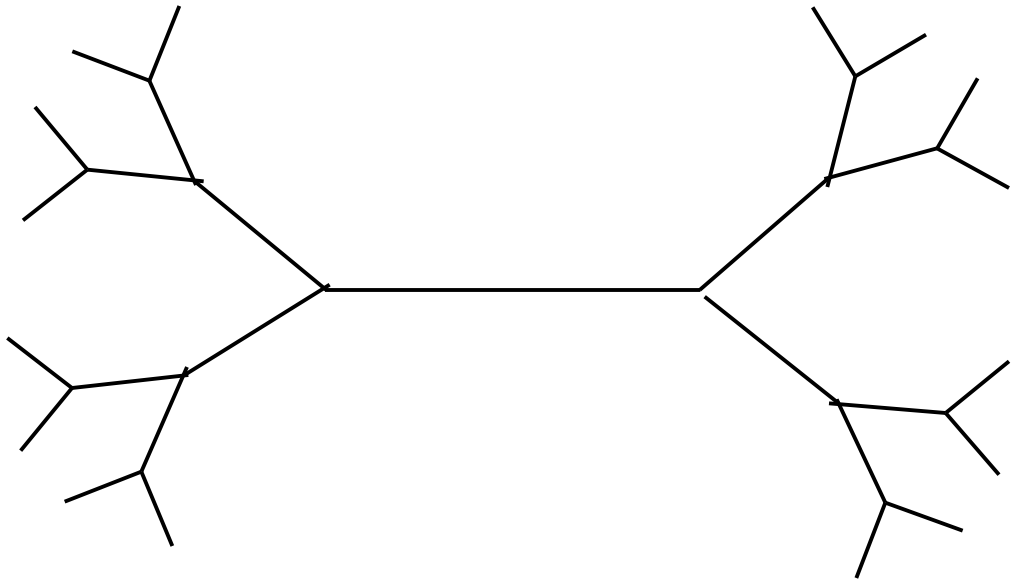
# HMMER vs. PaPaRa placement error



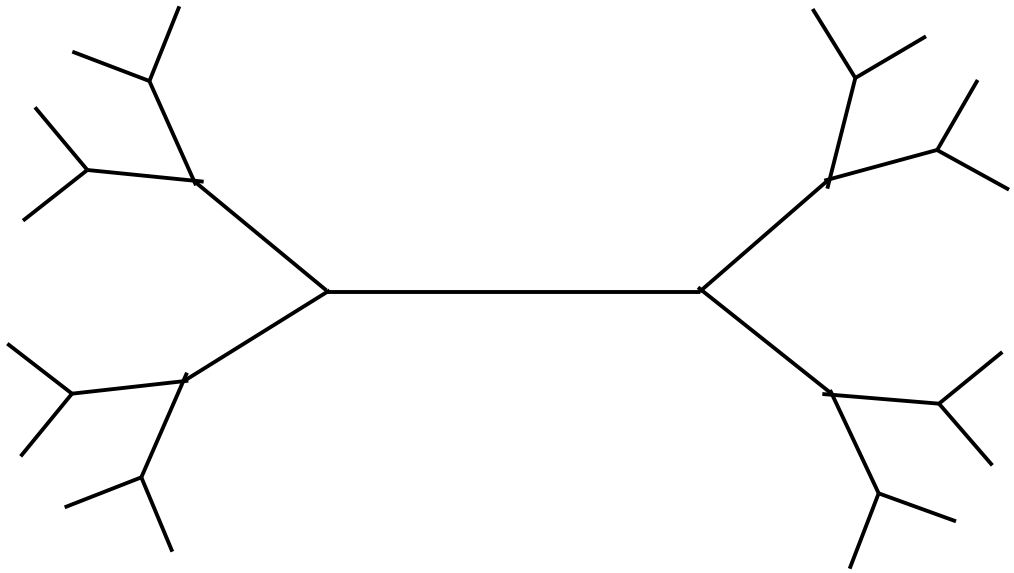
# HMMER+pplacer

Steps:

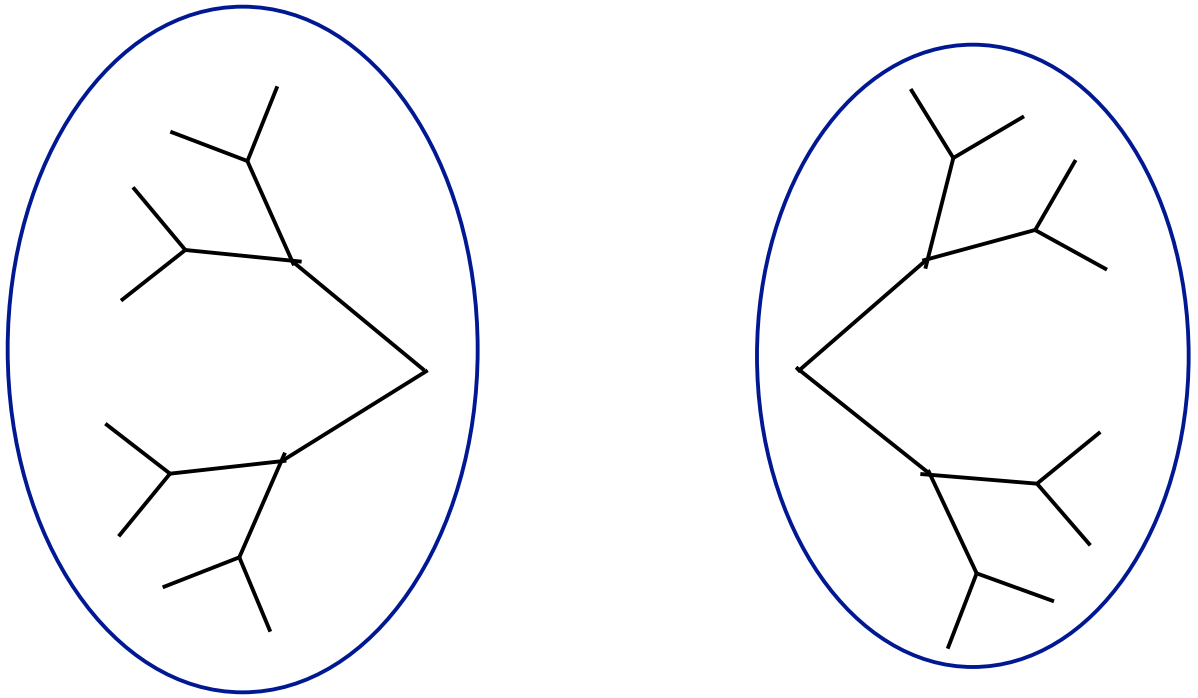
1) Build one HMM for the entire alignment
2) Align fragment to the HMM, and insert into alignment
3) Insert fragment into tree to optimize likelihood



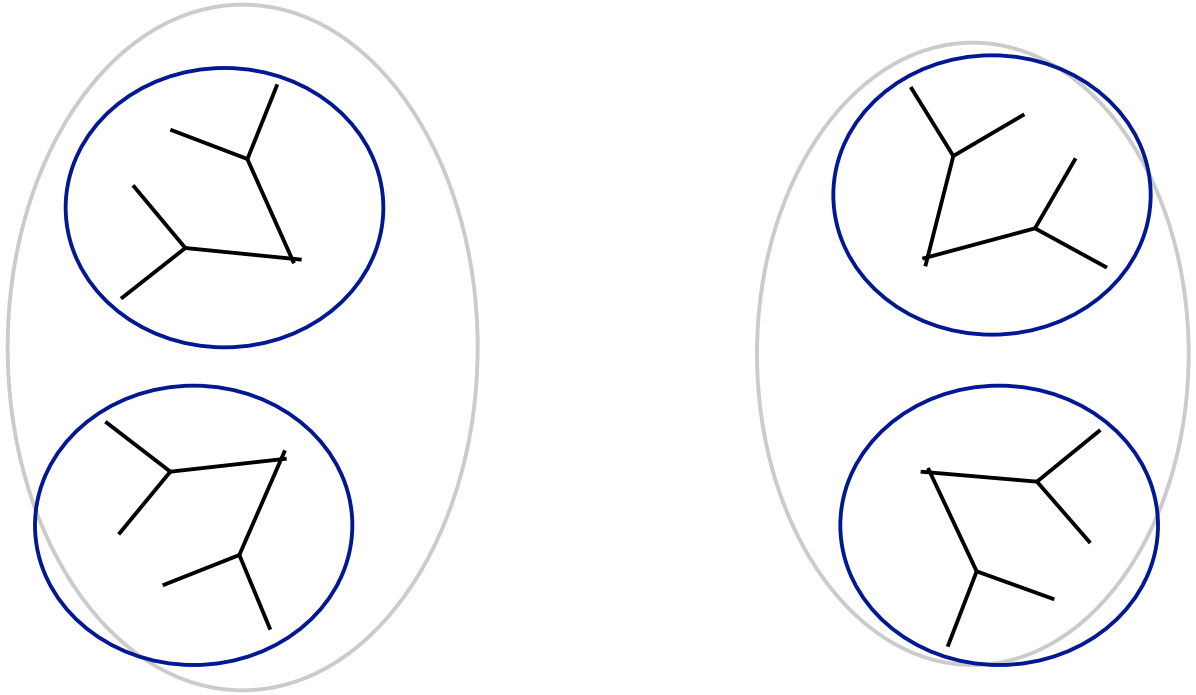
# One Hidden Markov Model for the entire alignment?



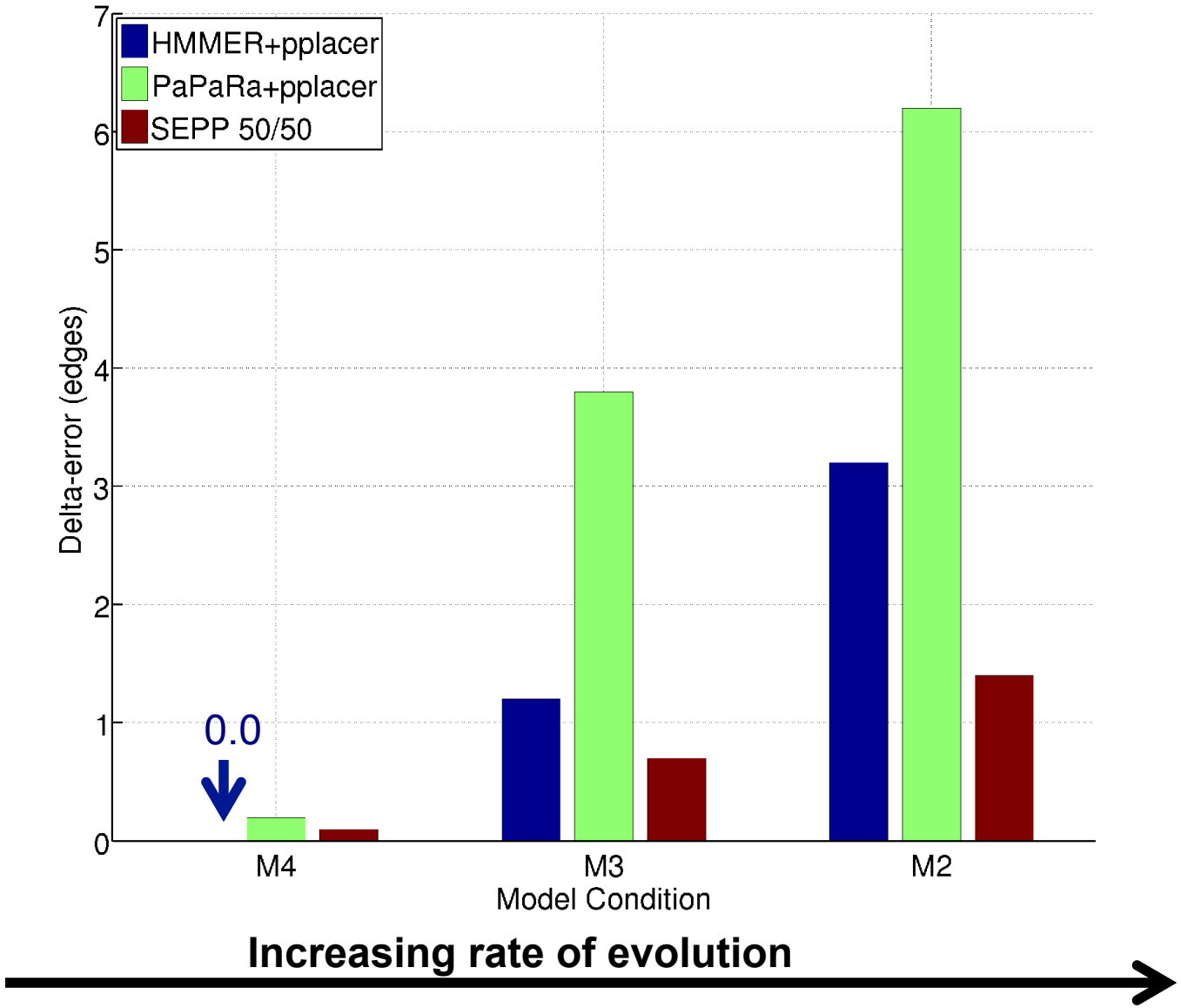
# Or 2 HMMs?



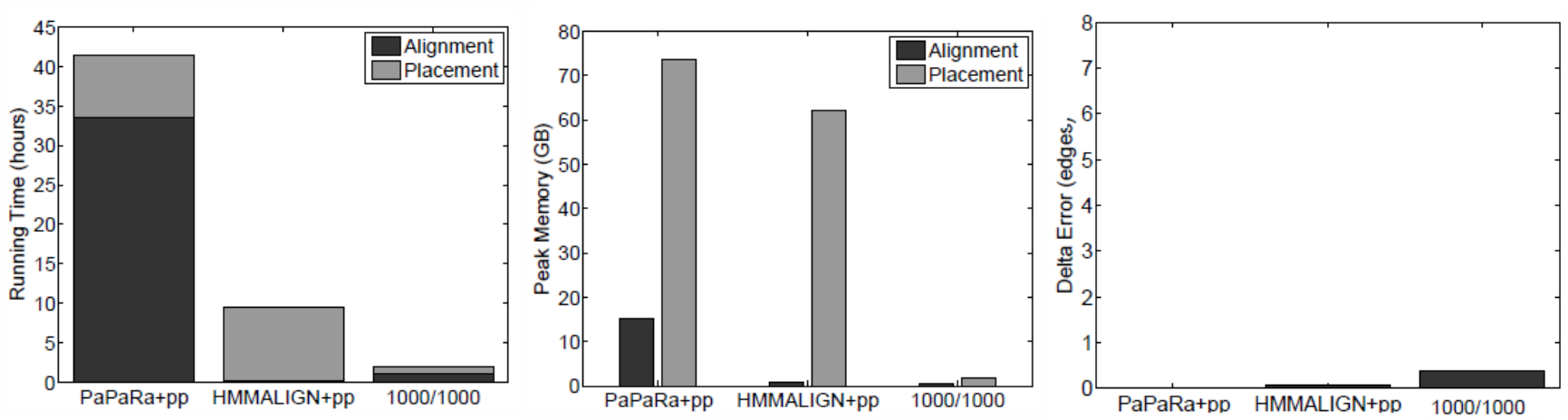
# Or 4 HMMs?



# SEPP(10%), based on ~10 HMMs



# SEPP (10%) on Biological Data




16S.B.ALL dataset, 13k curated backbone tree, 13k total fragments

For 1 million fragments:

PaPaRa+pplacer: ~133 days

HMMALIGN+pplacer: ~30 days

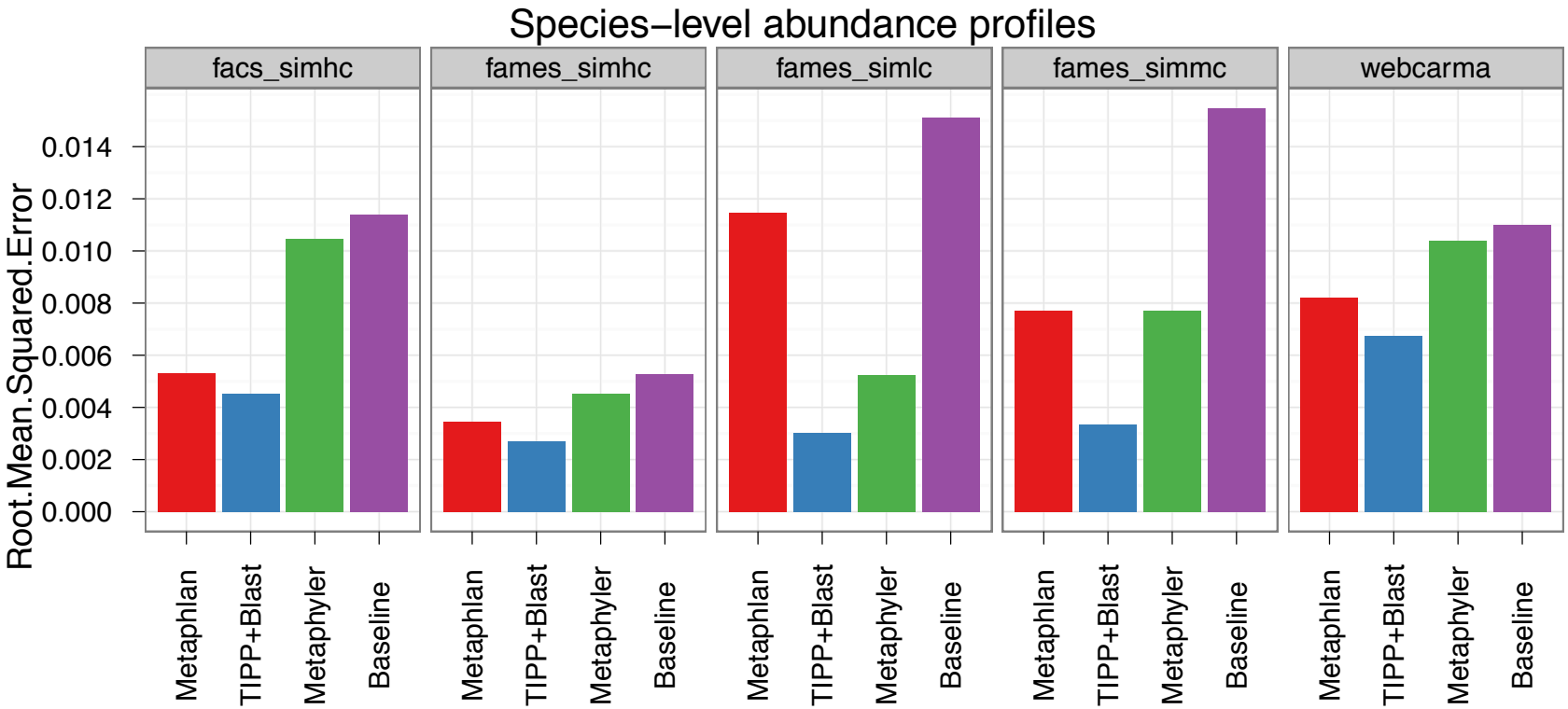
SEPP 1000/1000:  ~6 days

# TIPP: SEPP + statistics

Using SEPP as a taxon identification technique has high recall but low precision (classifies almost everything)

TIPP: dramatically reduces false positive rate with small reduction in true positive rate, by considering uncertainty in alignment (HMMER) and placement (pplacer)

We show a comparison of TIPP to Metaphyler and Metaphlan on 5 simulated datasets.




- FACs HC: Fragments simulated from 19 bacterial genomes, all in equal abundance (Stranneheim et al. 2010)

- FAMeS: Fragments simulated from 113 bacterial and archael genomes, under 3 different abundance complexity profiles. (Mavromatis et al. 2007)

- WebCarma: Fragments simulated from 25 bacterial genomes, all in equal abundance (Gerlach and Stoye 2011).

# Summary: 5 Phylogenetic "boosters"

- DCM1: absolute fast converging method
- SATé: co-estimation of alignments and trees
- UPP: ultra-large multiple sequence alignment
- TIPP: taxonomic identification of short reads
- SEPP: phylogenetic placement

Each method can be used with different "base methods" to produce improved accuracy and/or scalability.

Three of these methods use the HMM Family technique.

# Other Research in my lab

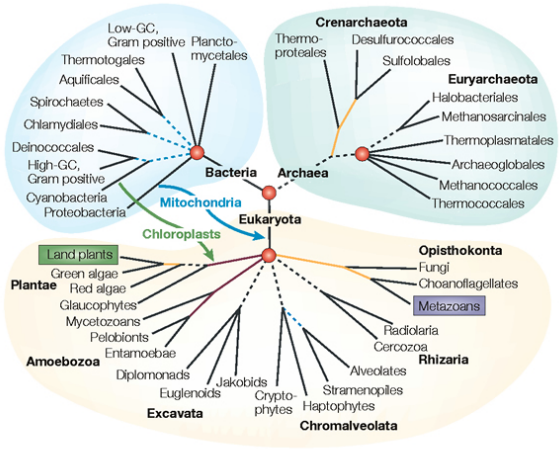
Method development for

- Species tree estimation from incongruent genes
- Reticulate phylogeny (HGT and hybridization)
- Alignment-free phylogeny estimation
- Supertree estimation
- Genome rearrangement phylogeny
- Historical Linguistics

Techniques:

- Statistical estimation under Markov models of evolution
- Graph theory and algorithms
- Machine learning and data mining
- Heuristics for NP-hard optimization problems
- High performance computing
- Massive simulations

# Estimating the Tree of Life

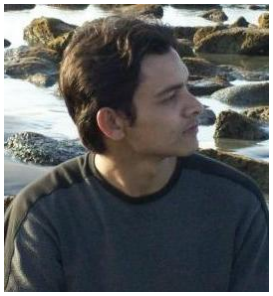



Nature Reviews | Genetics


*New algorithmic techniques*
*New methods*
*New questions*
*New theory*
*Open source software*

# Warnow Laboratory





PhD students: Siavash Mirarab, Nam Nguyen, and Md. S. Bayzid

Undergrad: Keerthana Kumar

Lab Website: http://www.cs.utexas.edu/users/phylo

**Funding**: Guggenheim Foundation,Packard Foundation, NSF, Microsoft Research New England, David Bruton Jr. Centennial Professorship, and TACC (Texas Advanced Computing Center). HHMI graduate fellowship to Siavash Mirarab and Fulbright graduate fellowship to Md. S. Bayzid.