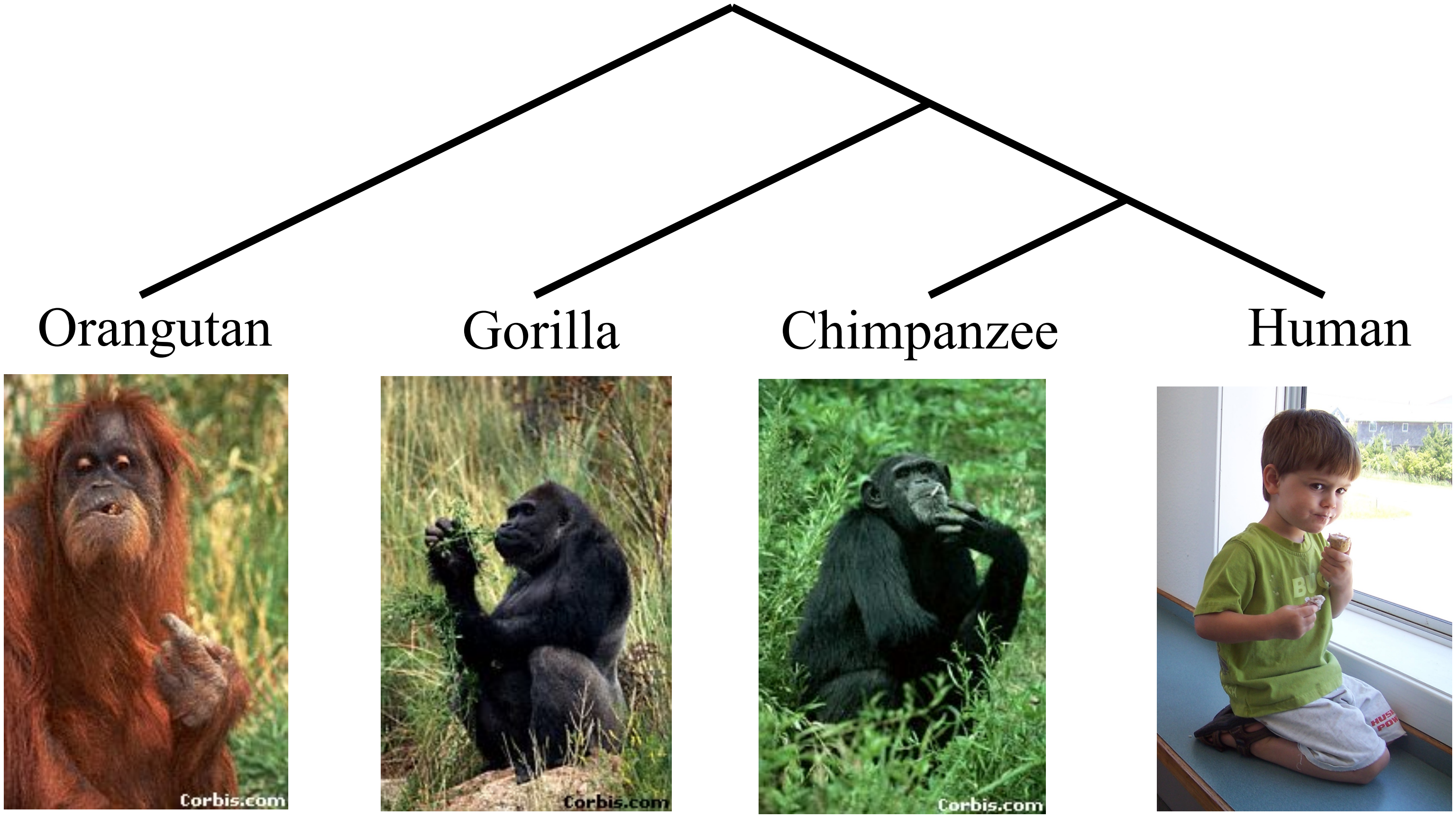
Algorithms for Ultra-large Multiple Sequence Alignment and Phylogeny Estimation

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

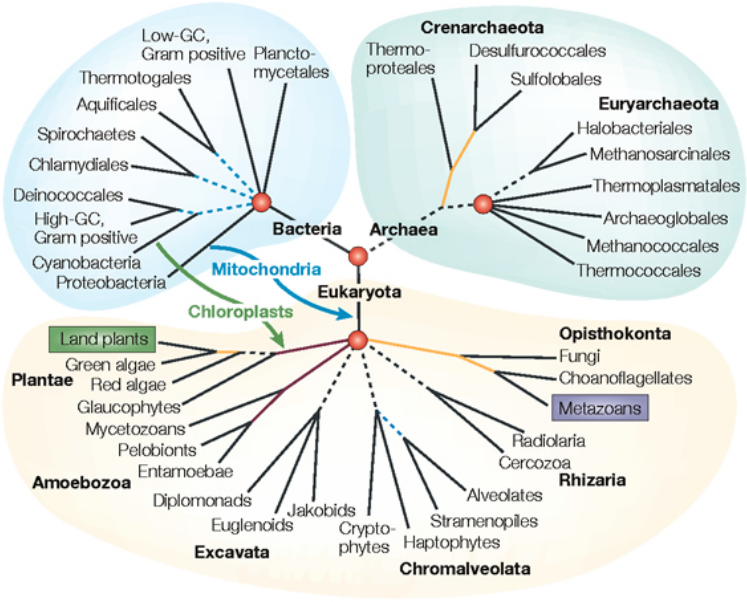
Phylogeny (evolutionary tree)



Orangutan Gorilla Chimpanzee Human

From the Tree of the Life Website,
University of Arizona

The Tree of Life: Applications to Biology

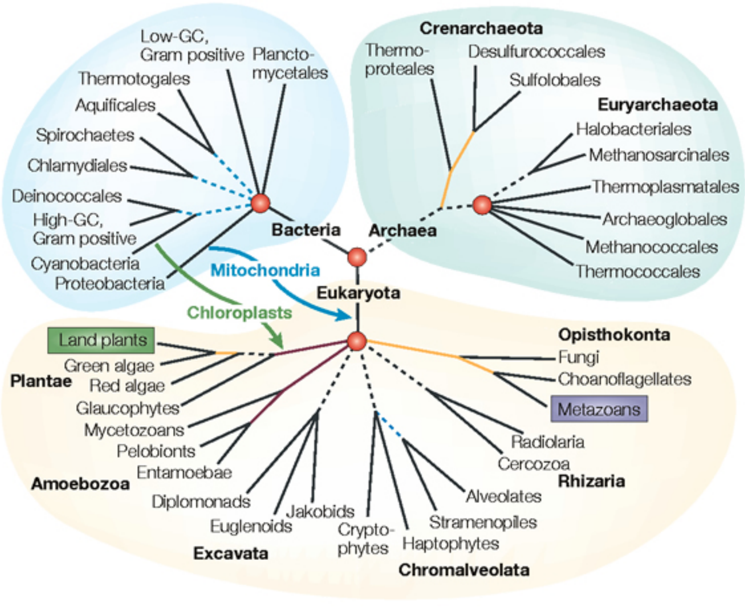


Nature Reviews | Genetics

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

The Tree of Life: a *Grand Challenge*



Nature Reviews | Genetics

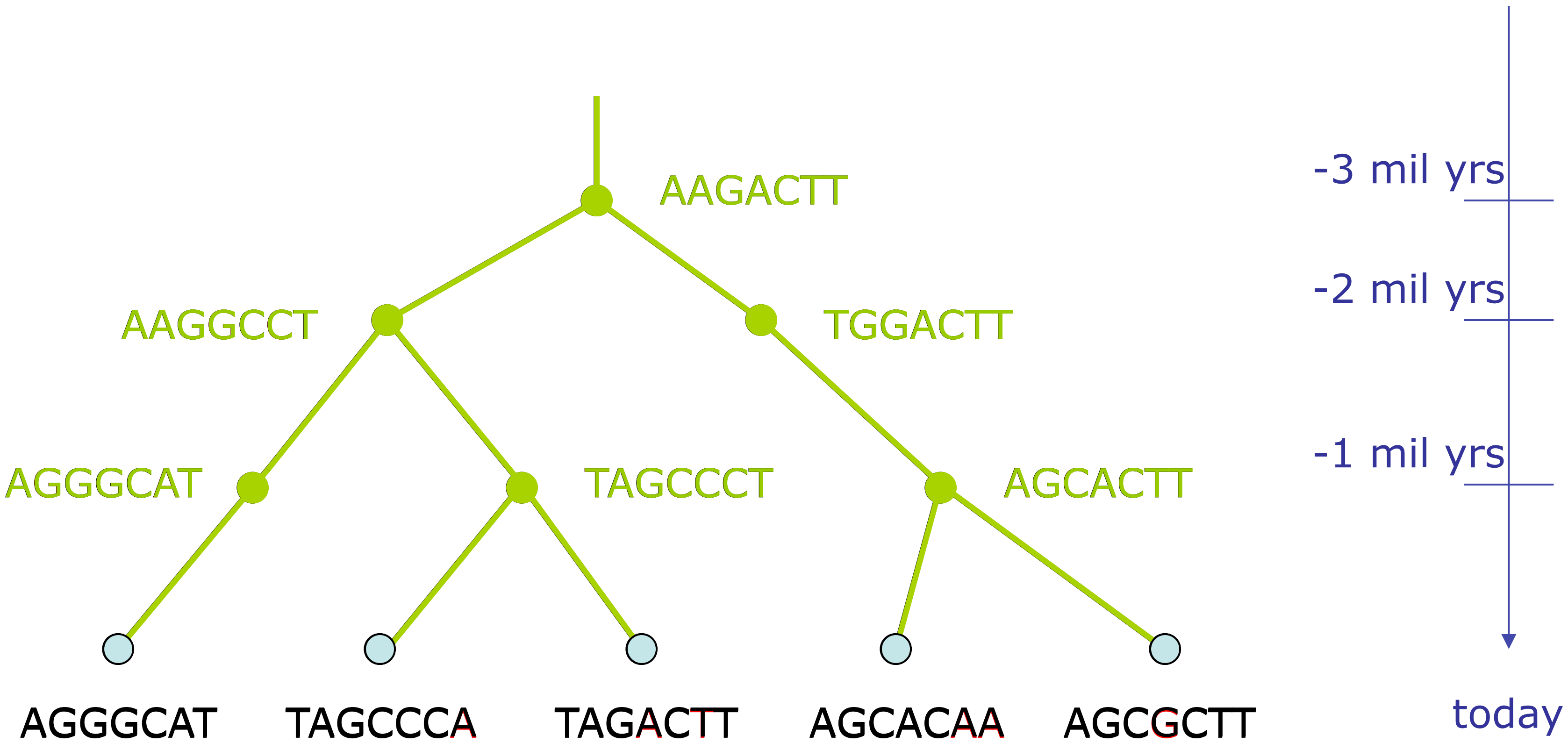
Novel techniques needed for scalability and accuracy

NP-hard problems and large datasets

Current methods do not provide good accuracy

HPC is insufficient

DNA Sequence Evolution



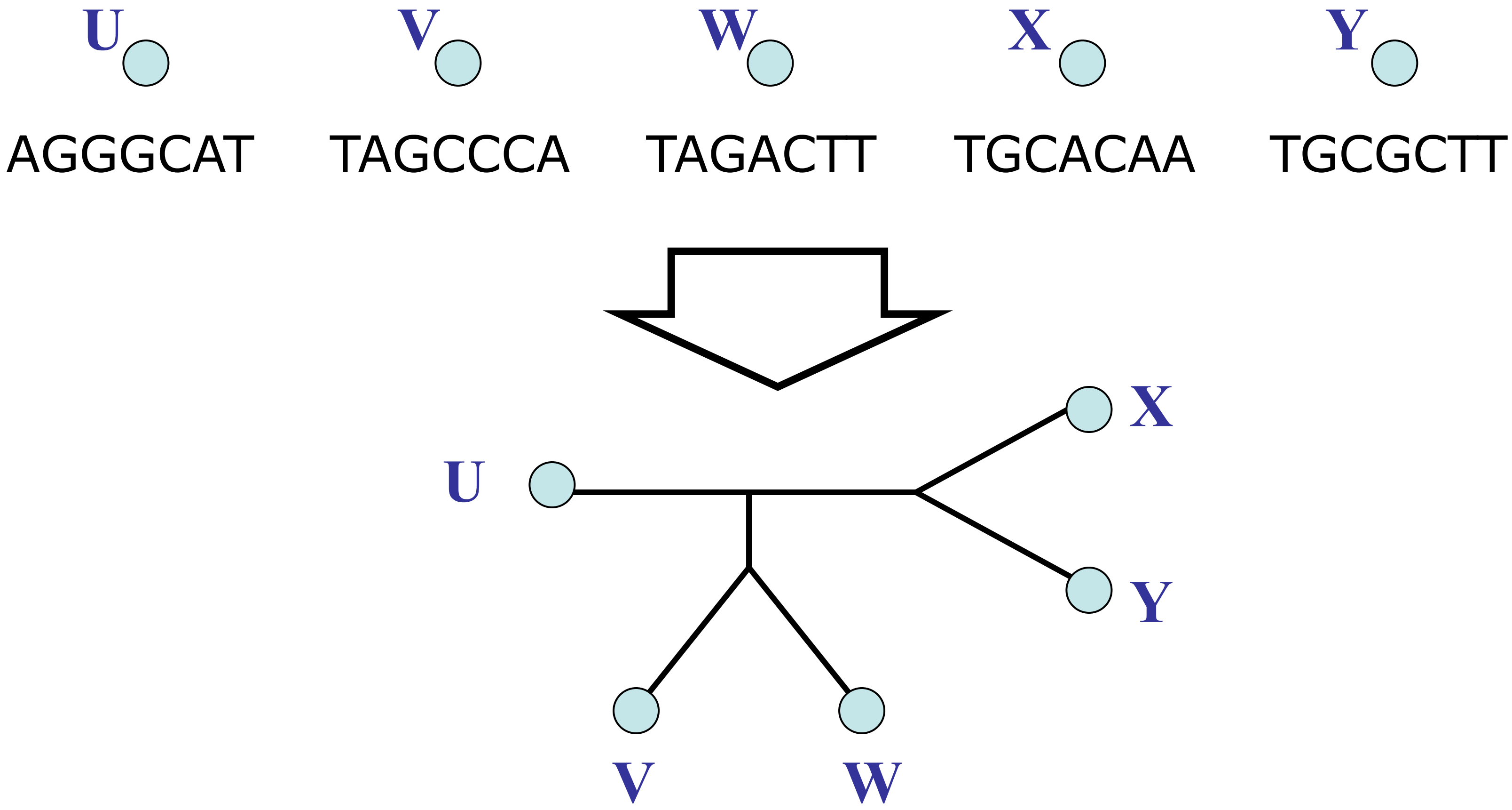
Markov Model of Site Evolution

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.

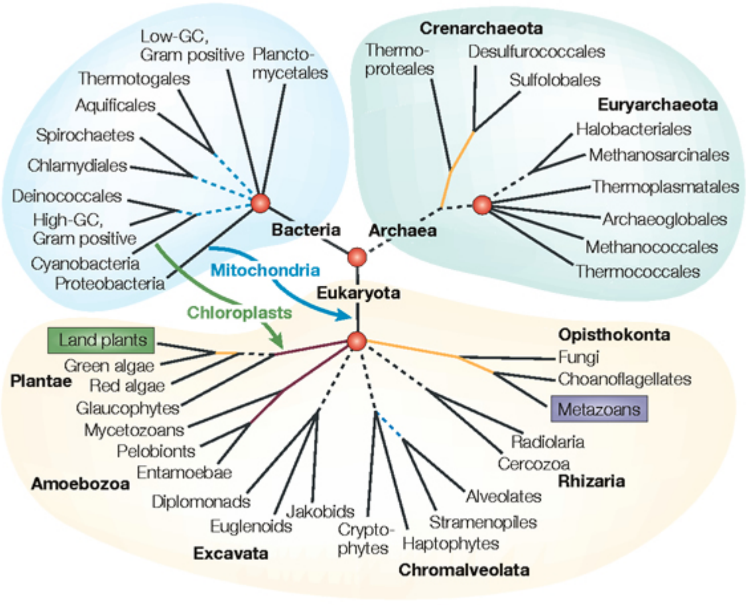
More complex single site evolution models (such as the General Markov model) are also considered, often with little change to the theory.

Phylogeny Problem



U	V	W	X	Y
AGGGCAT	TAGCCCA	TAGACTT	TGCACAA	TGCGCTT

The Tree of Life: a *Grand Challenge*



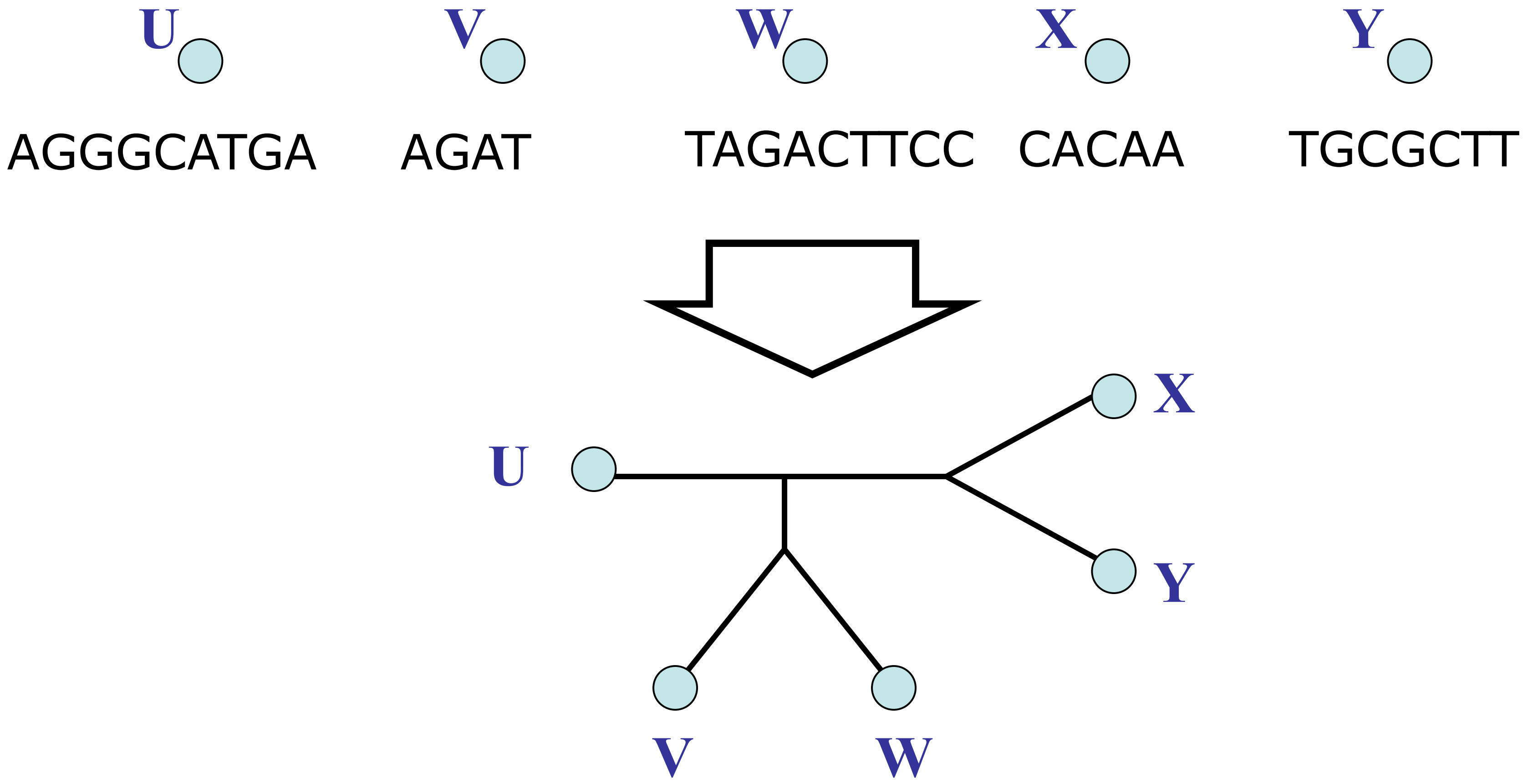
Nature Reviews | Genetics

Most well known problem:

Given set of DNA sequences, find the Maximum Likelihood Tree

NP-hard, but lots of software (RAxML, FastTree, GARLI, PhyML…)

The “real” problem



U	V	W	X	Y
AGGGCATGA	AGAT	TAGACTTCC	CACAA	TGCGCTT

Input: unaligned sequences

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

Phase 1: Alignment

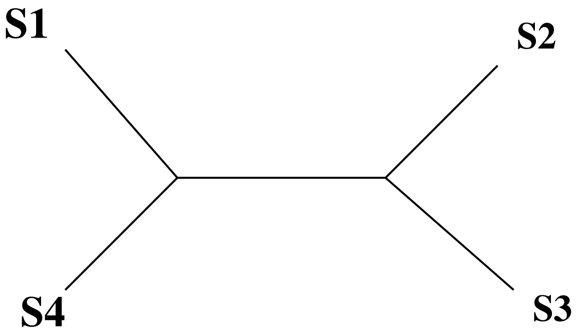
```
S1 = AGGCTATCACCTGACCTCCA          S1 = -AGGCTATCACCTGACCTCCA
S2 = TAGCTATCACGACCGC              S2 = TAG-CTATCAC--GACCGC--
S3 = TAGCTGACCGC            →      S3 = TAG-CT-------GACCGC--
S4 = TCACGACCGACA                  S4 = -------TCAC--GACCGACA
```

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
```



Steps in a phylogenetic estimation

- Identify gene sequences in each genome for each species
- Compute multiple sequence alignment (MSA)
- Compute gene tree (phylogenetic tree on the MSA)

Steps in a phylogenetic estimation

1. Select genes and set of species
2. For each gene:
   - Identify gene sequences in each genome for each species
   - Compute multiple sequence alignment (MSA)
   - Compute gene tree (phylogenetic tree on the MSA)
3. Combine gene trees into species tree

Steps in a phylogenetic estimation

1. Select genes and set of species
2. For each gene:
   - Identify gene sequences in each genome for each species
   - Compute multiple sequence alignment (MSA)
   - Compute gene tree (phylogenetic tree on the MSA)
3. Combine gene trees into species tree

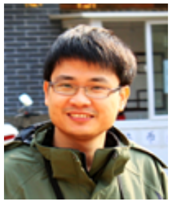
Tomorrow's talk

Avian Phylogenomics Project

Erich Jarvis,
HHMI

MTP Gilbert,
Copenhagen

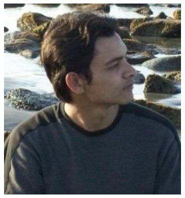
G Zhang,
BGI



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 (Liu et al., Science 2009 and Systematic Biology 2012)

Challenges:

Maximum likelihood on multi-million-site sequence alignments

Massive gene tree incongruence

Steps in a phylogenetic estimation

1. Select genes and set of species
2. For each gene:
   - Identify gene sequences in each genome for each species
   - Compute multiple sequence alignment (MSA)
   - Compute gene tree (phylogenetic tree on the MSA)
3. Combine gene trees into species tree

1kp: Thousand Transcriptome Project

G. Ka-Shu Wong
U Alberta

J. Leebens-Mack
U Georgia

N. Wickett
Northwestern

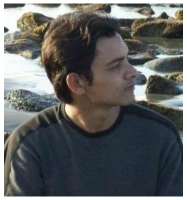
N. Matasci
iPlant

T. Warnow,
UT-Austin

S. Mirarab,
UT-Austin

N. Nguyen,
UT-Austin

Md. S.Bayzid
UT-Austin

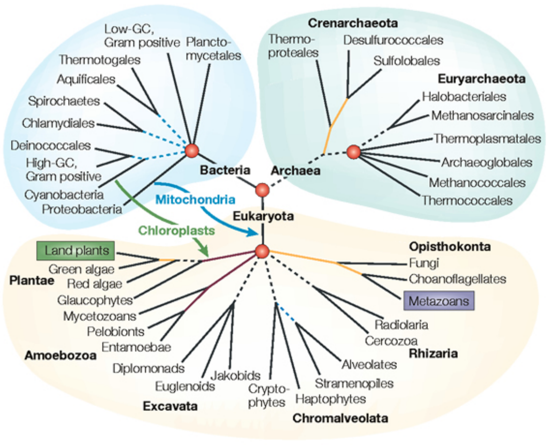


Plus many many other people...

- 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 (Liu et al., Science 2009 and Systematic Biology 2012)

Challenges:
Multiple sequence alignments of > 100,000 sequences
Gene tree incongruence

The Tree of Life: *Multiple Challenges*



Nature Reviews | Genetics

Large datasets:

100,000+ sequences

10,000+ genes

“BigData” complexity

Orthology prediction

Multiple sequence alignment

Maximum likelihood tree estimation

Bayesian tree estimation

Alignment-free phylogeny estimation

Supertree estimation

Estimating species trees from incongruent gene trees

Genome rearrangements

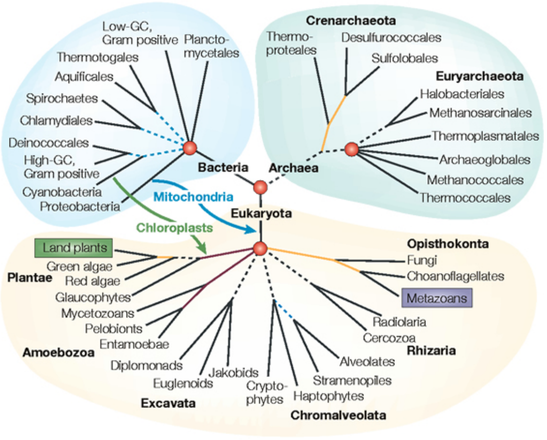
Reticulate evolution

Visualization of large trees and alignments

Databases of sets of trees

Data mining techniques to explore multiple optima

The Tree of Life: *Multiple Challenges*



Nature Reviews | Genetics

Large datasets:
100,000+ sequences
10,000+ genes
"BigData" complexity

Orthology prediction
Multiple sequence alignment
Maximum likelihood tree estimation
Bayesian tree estimation
Alignment-free phylogeny estimation
Supertree estimation
Estimating species trees from incongruent gene trees
Genome rearrangements
Reticulate evolution
Visualization of large trees and alignments
Databases of sets of trees
Data mining techniques to explore multiple optima

Today’s talk

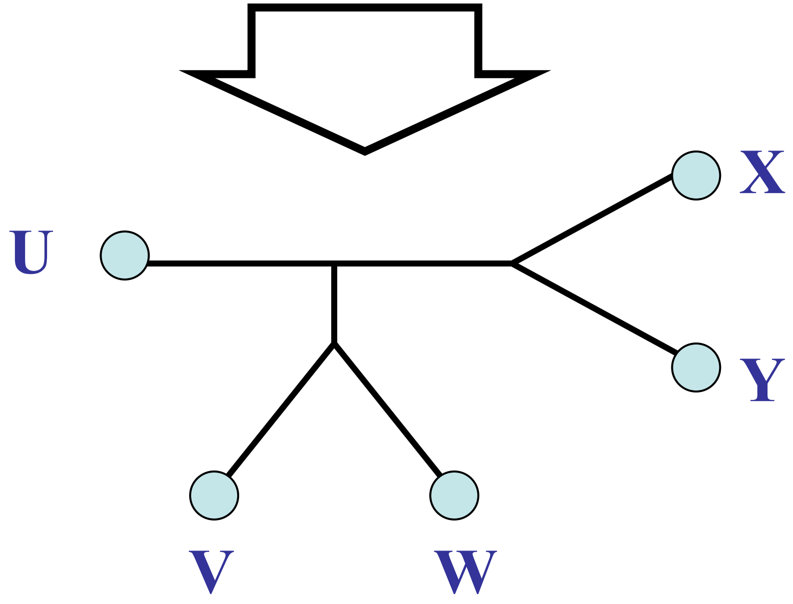
- Challenges in alignment estimation
- SATé – co-estimating alignments and trees (Science 2009 and Systematic Biology 2012)
- DACTAL – divide-and-conquer trees (almost) without alignments (RECOMB 2012)
- UPP – ultra-large alignment estimation using SEPP (in preparation)

Focus on *practical performance* for large-scale analysis.

Part I: Challenges in alignment estimation

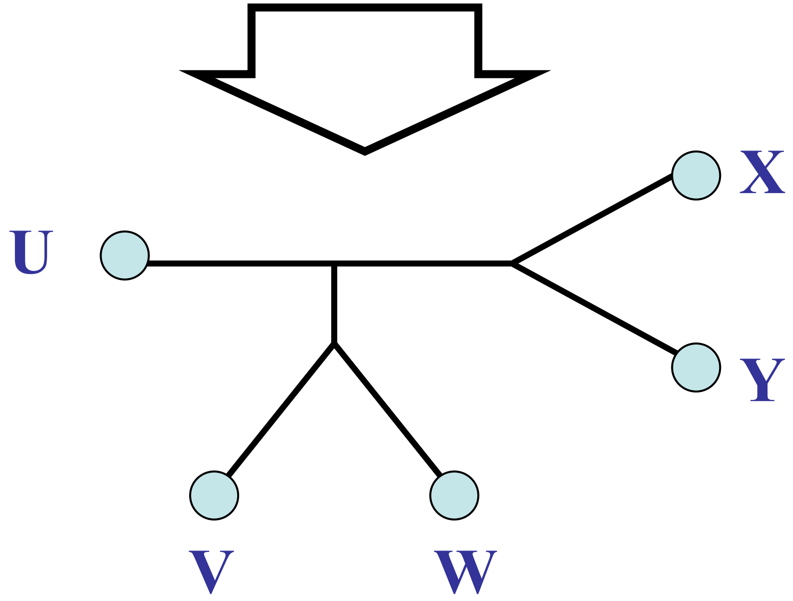
Phylogeny Problem

U	V	W	X	Y
AGGGCAT	TAGCCCA	TAGACTT	TGCACAA	TGCGCTT

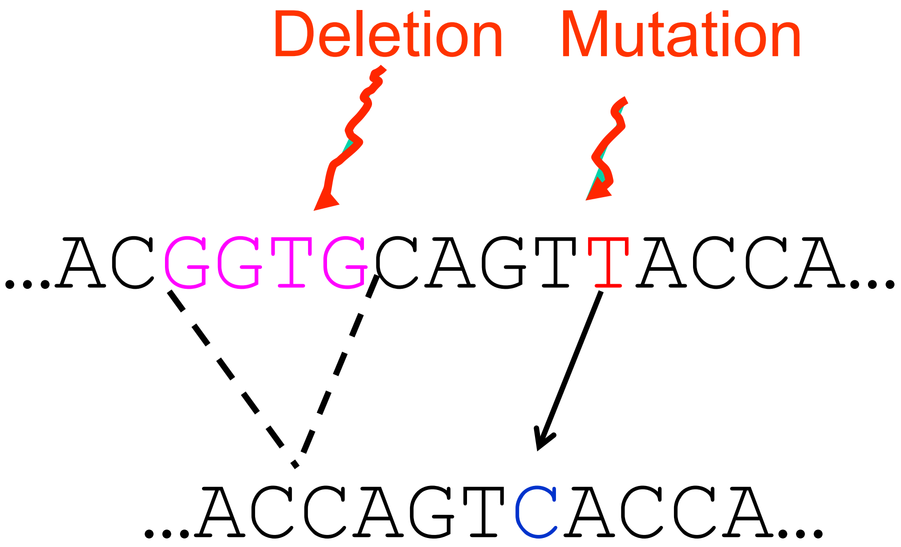


The “real” problem

U	V	W	X	Y
AGGGCATGA	AGAT	TAGAC	TGCAAA	TGCGCTTT



Not just substitutions, but also “Indels”

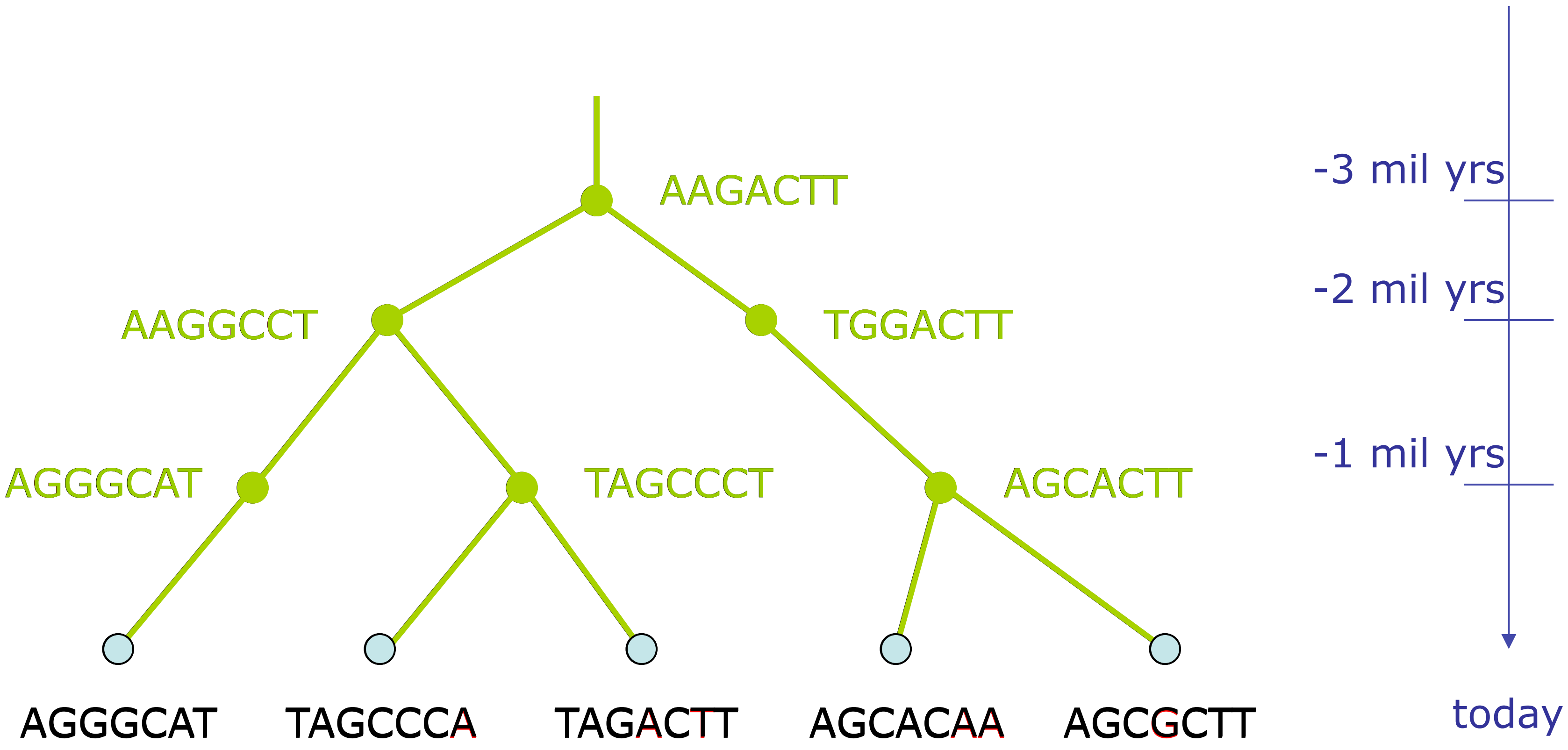


Deletion Mutation

…ACGGTGCAGTTACCA…

…ACCAGTCACCA…

DNA Sequence Evolution



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.

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.

New models need to consider indels

Markov Model of Site Evolution

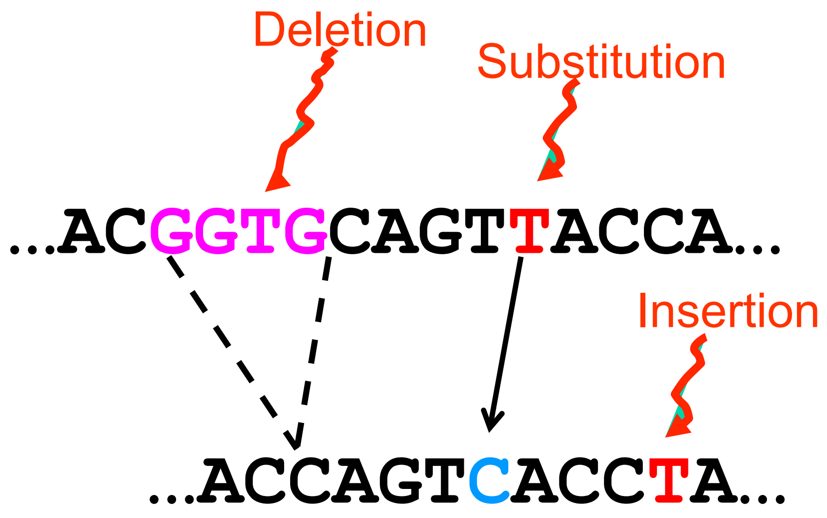
Simplest (Jukes-Cantor, 1969):

- 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.

New models need to consider indels

Limited progress

New mathematical questions



```
...ACGGTGCAGTTACC-A...
...AC----CAGTCACCTA...
```

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
```

→

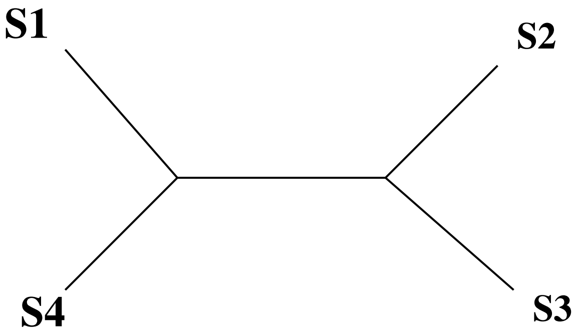
```
S1 = -AGGCTATCACCTGACCTCCA
S2 = TAG-CTATCAC--GACCGC--
S3 = TAG-CT-------GACCGC--
S4 = -------TCAC--GACCGACA
```

Phase 2: Construct tree

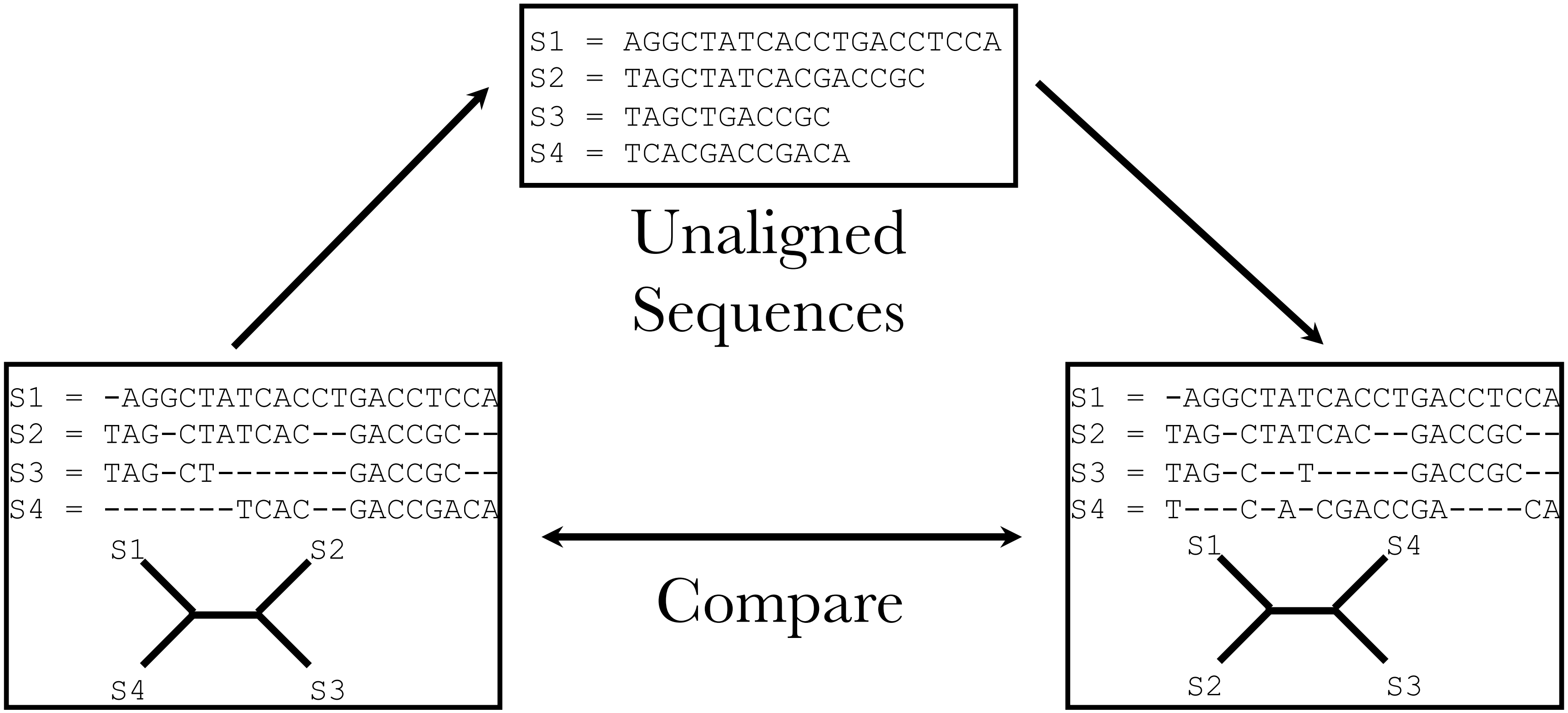
```
S1 = AGGCTATCACCTGACCTCCA
S2 = TAGCTATCACGACCGC
S3 = TAGCTGACCGC
S4 = TCACGACCGACA
```

→

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



Simulation Studies



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

Unaligned Sequences

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

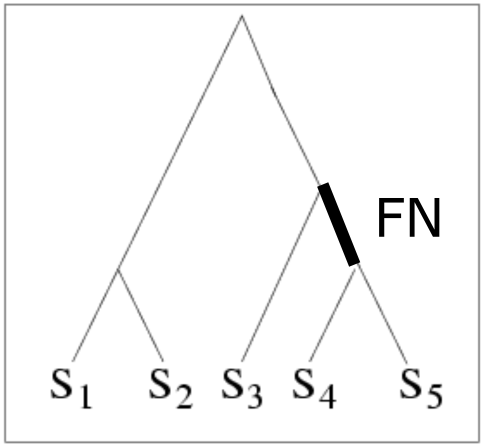
True tree and alignment

Compare

```
S1 = -AGGCTATCACCTGACCTCCA
S2 = TAG-CTATCAC--GACCGC--
S3 = TAG-C--T-----GACCGC--
S4 = T---C-A-CGACCGA----CA
```

Estimated tree and alignment

Quantifying Error



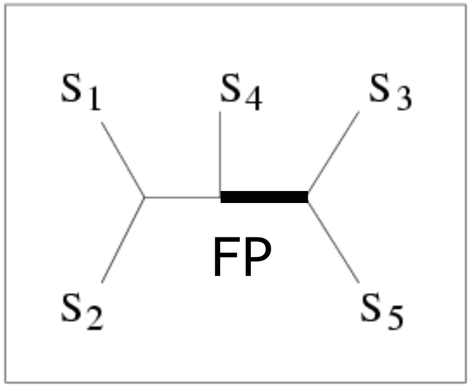
TRUE TREE

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% error rate



INFERRED TREE

Two-phase estimation

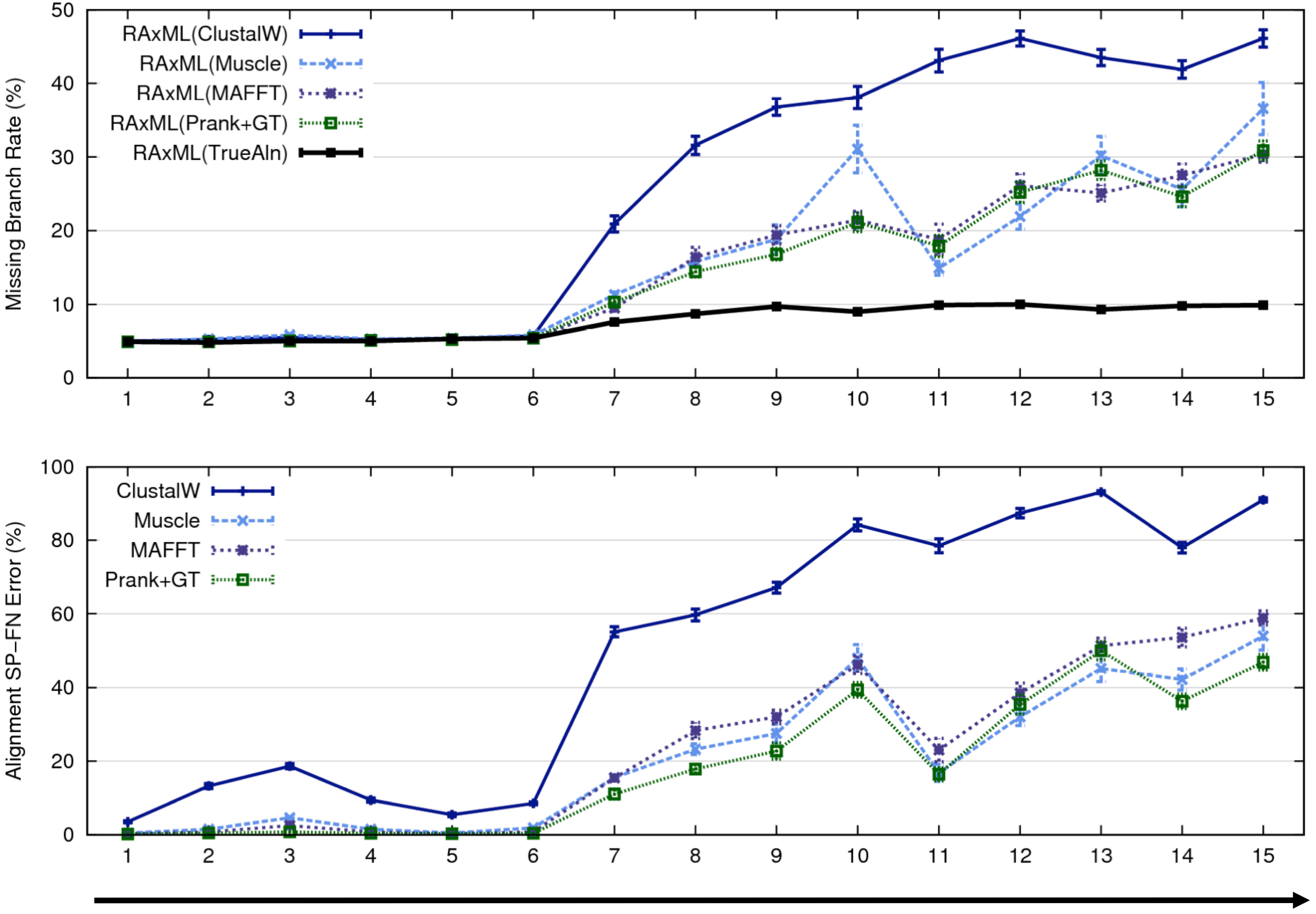
Alignment methods

- 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*



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)

Large-scale MSA: *another grand challenge*[1]

```
S1 = AGGCTATCACCTGACCTCCA
S2 = TAGCTATCACGACCGC
S3 = TAGCTGACCGC
  …
Sn = TCACGACCGACA
```

⇒

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

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

Part II: 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

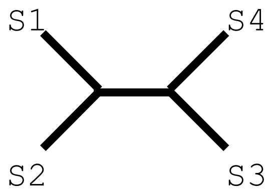
Co-estimation

Input: Unaligned Sequences

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

Estimated tree and alignment

```
S1 = -AGGCTATCACCTGACCTCCA
S2 = TAG-CTATCAC--GACCGC--
S3 = TAG-C--T-----GACCGC--
S4 = T---C-A-CGACCGA----CA
```

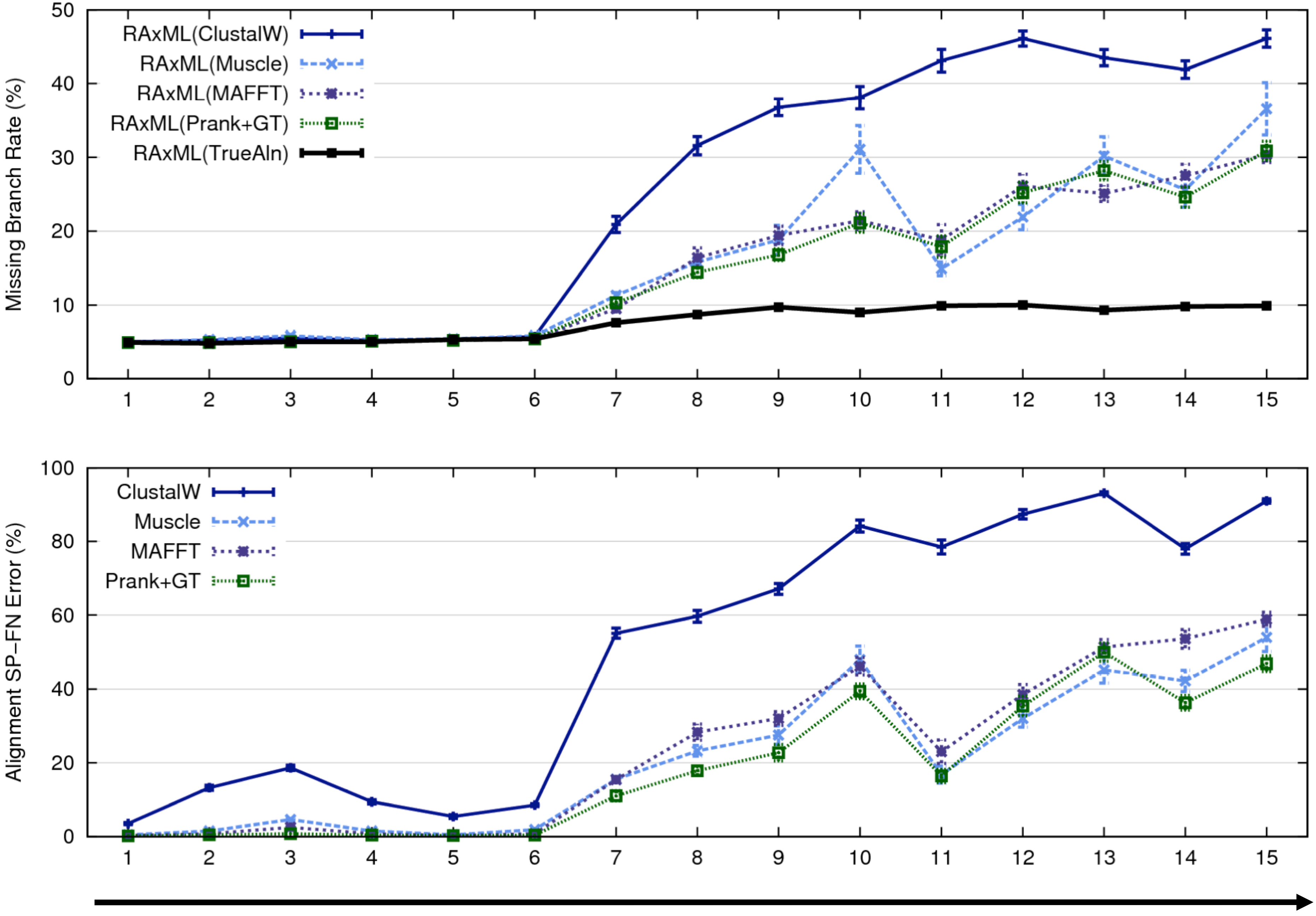


Co-estimation makes sense, but…

- Existing statistical co-estimation methods (e.g., BAliPhy) are extremely computationally intensive and do not scale.

- Existing models are too simple

Can we do better?



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.

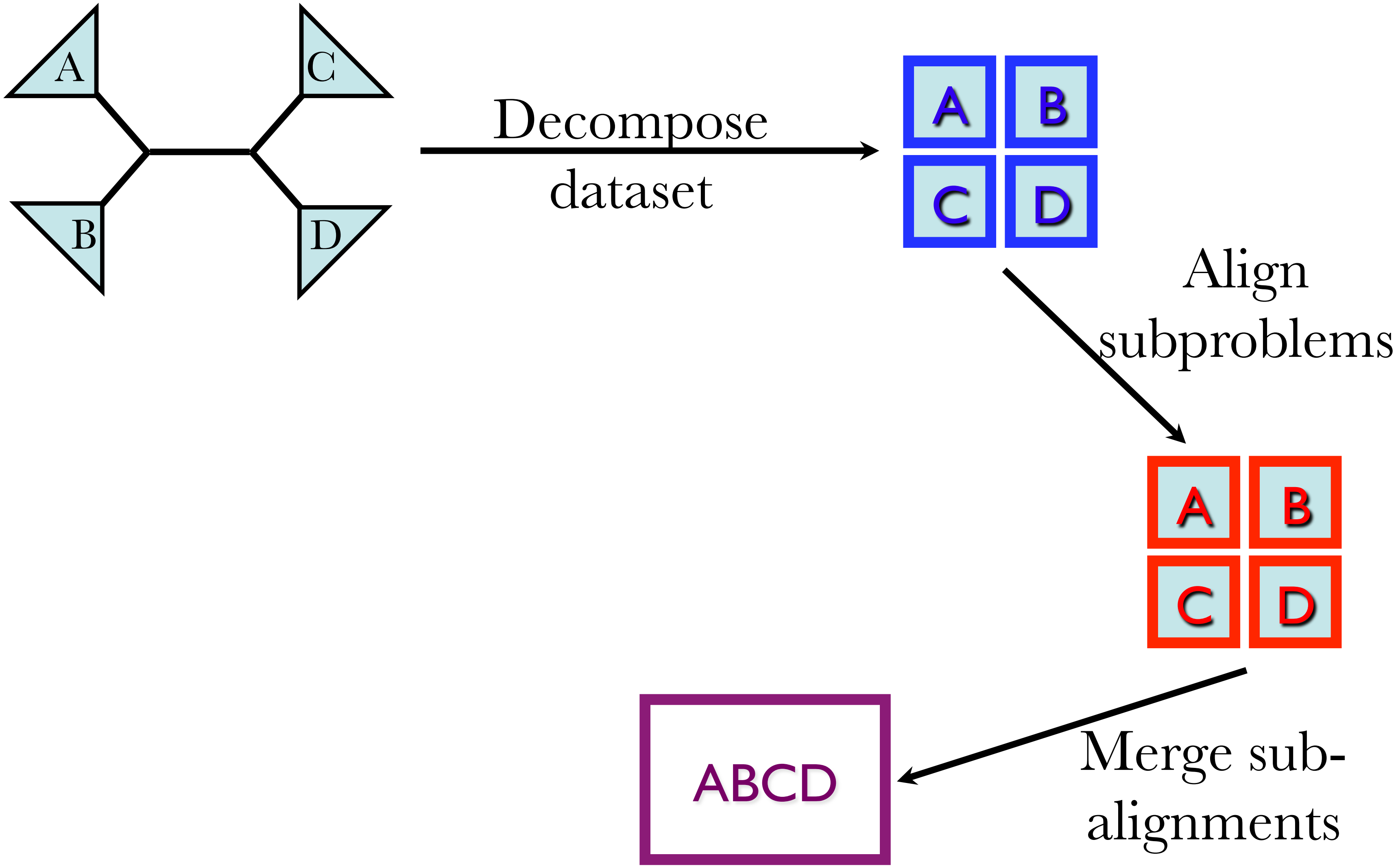
- Datasets with small enough “evolutionary diameters” are easy to align with high accuracy.

Alignment on the tree

- Idea: better (more accurate) alignments will be found if we align subsets with smaller diameters, and then combine alignments on these subsets

- Approach: use the tree topology to divide-and-conquer

Re-alignment on a tree (Cartoon)



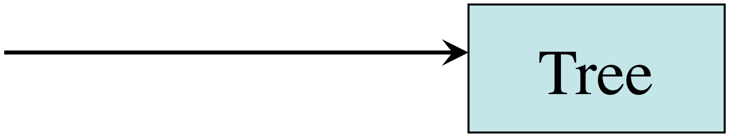
Decompose dataset

Align subproblems

Merge sub-alignments

SATé Algorithm

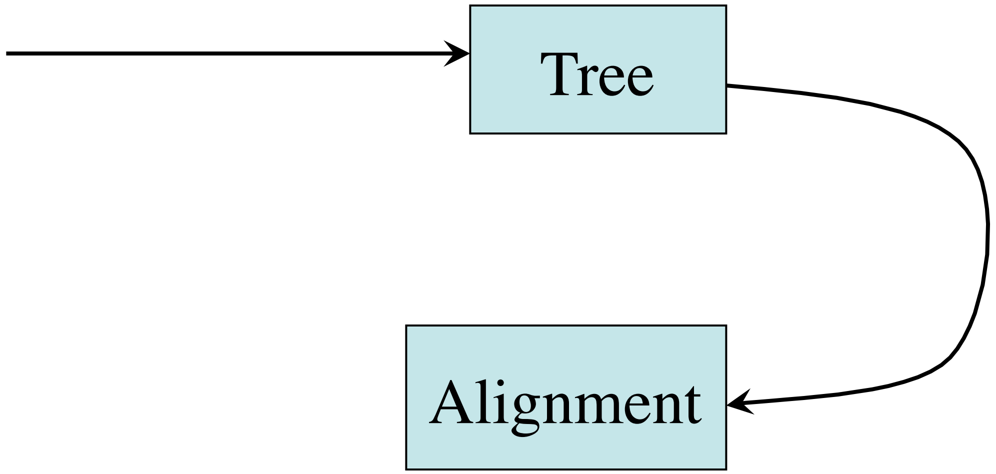
Obtain initial alignment
and estimated ML tree



Tree

SATé Algorithm

Obtain initial alignment
and estimated ML tree

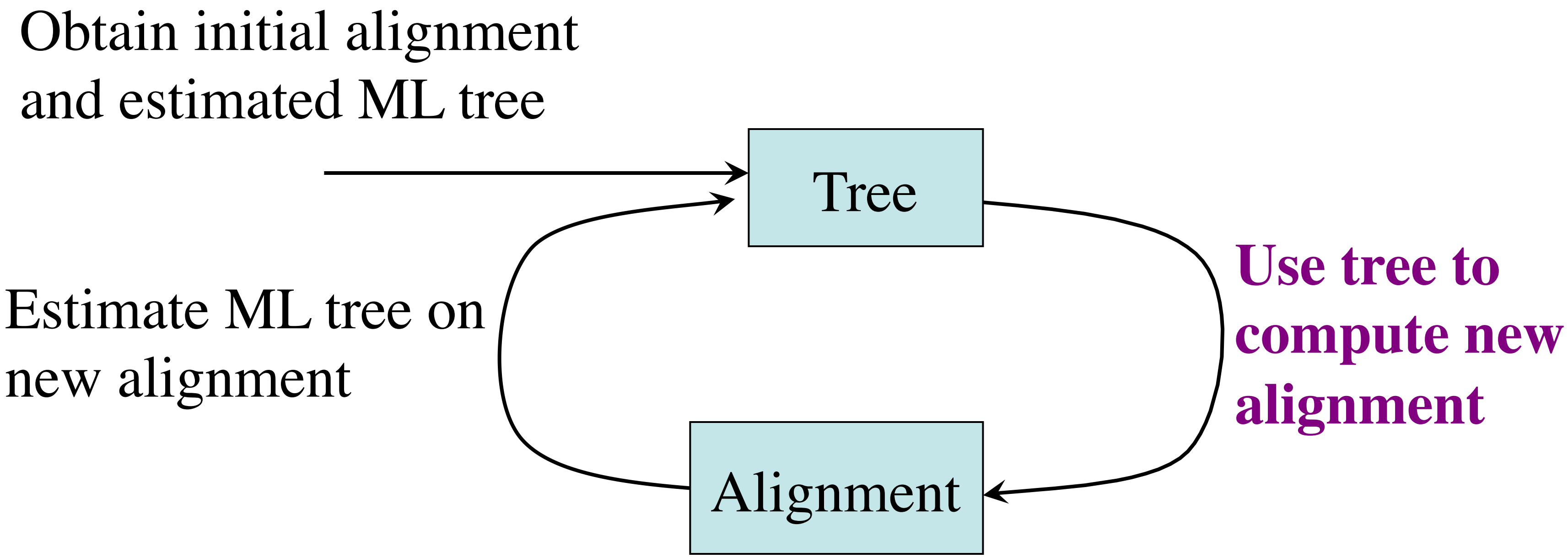


Tree

Use tree to
compute new
alignment

Alignment

SATé Algorithm



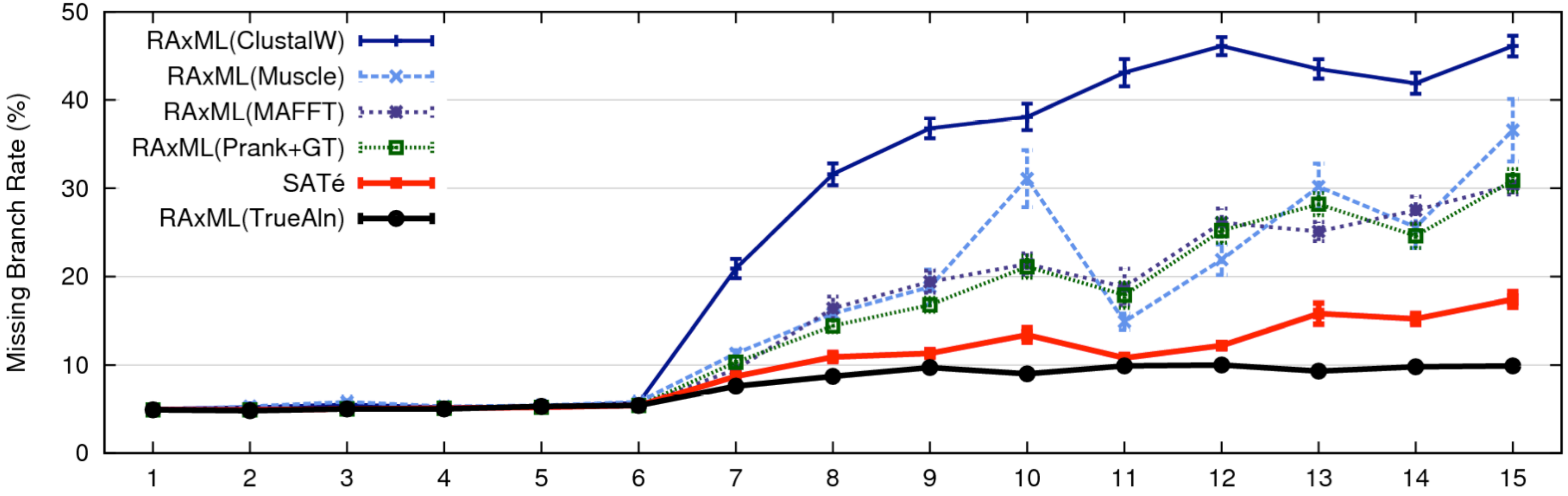
Obtain initial alignment and estimated ML tree

Tree

Use tree to compute new alignment

Alignment

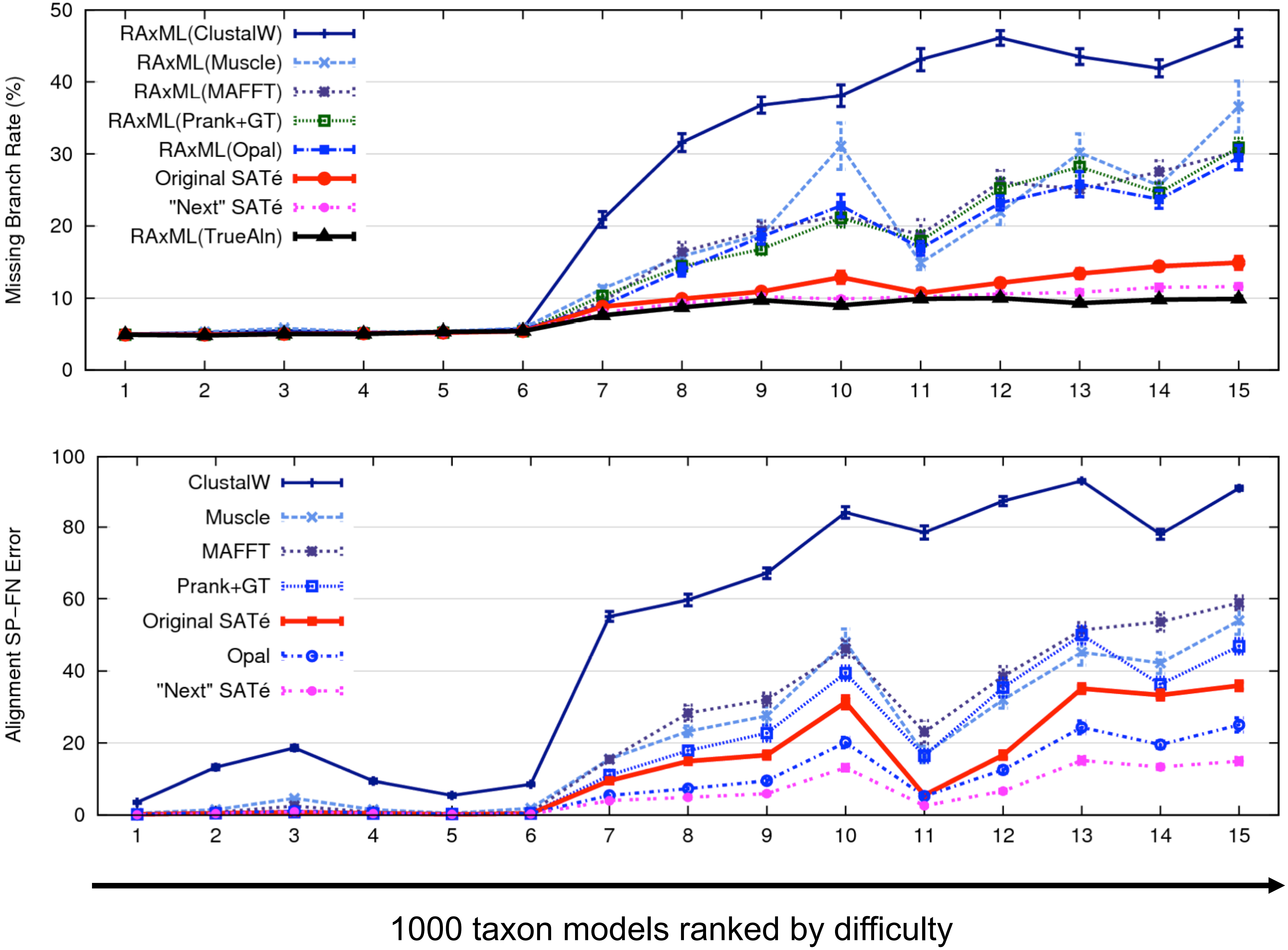
Estimate ML tree on new alignment



1000 taxon models, ordered by difficulty

24 hour SATé analysis, on desktop machines

(Similar improvements for biological datasets)


Missing Branch Rate (%)
50
40
30
20
10
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
RAxML(ClustalW)
RAxML(Muscle)
RAxML(MAFFT)
RAxML(Prank+GT)
RAxML(Opal)
Original SATé
"Next" SATé
RAxML(TrueAln)
Alignment SP–FN Error
100
80
60
40
20
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
ClustalW
Muscle
MAFFT
Prank+GT
Original SATé
Opal
"Next" SATé
1000 taxon models ranked by difficulty

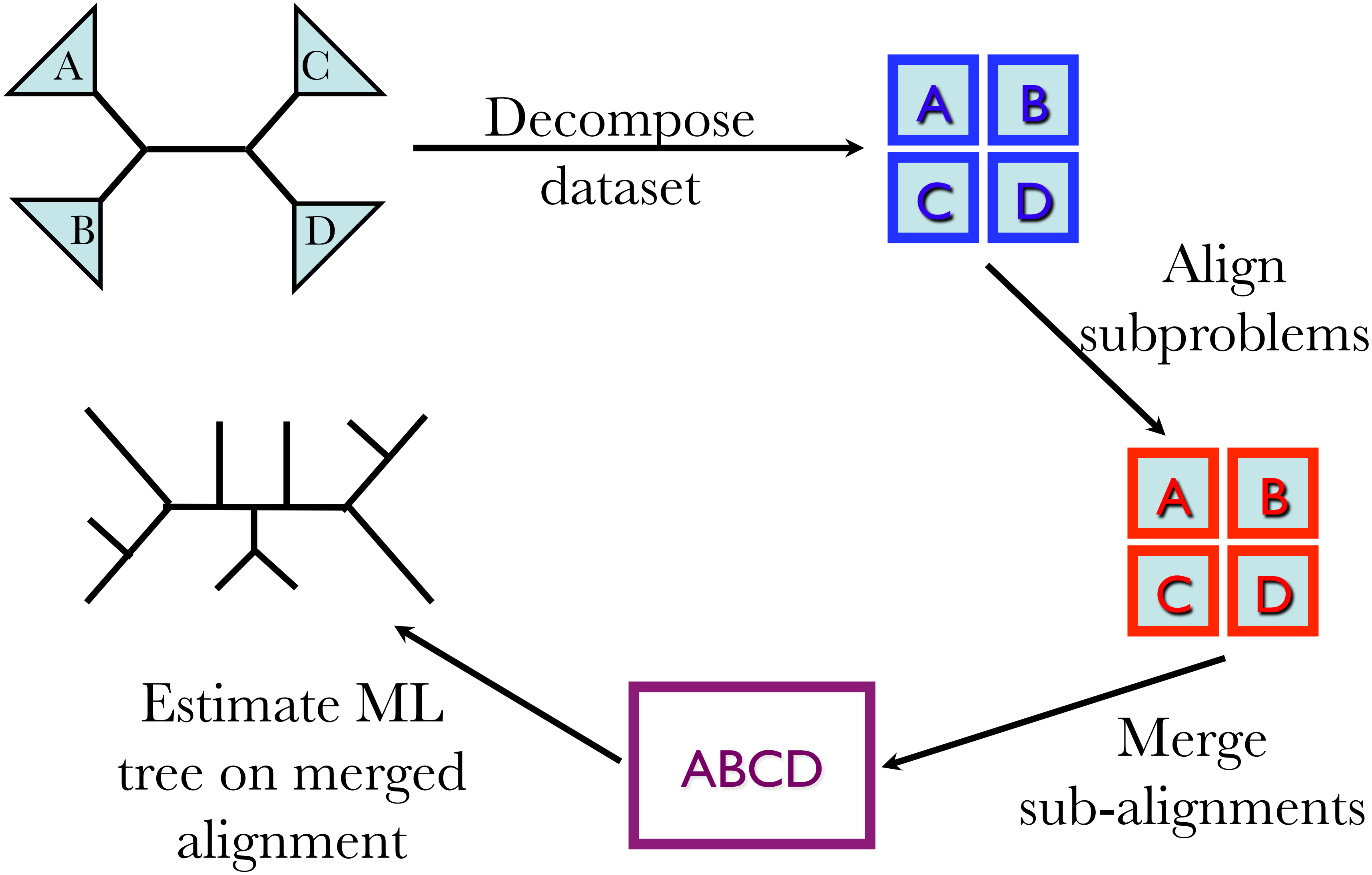
Performance

- SATé "boosts" the base methods**.** Results shown are for SATé used with MAFFT. Similar improvements seen for use with other MSA methods (e.g., Prank, Opal, Muscle, ClustalW).

Performance

- SATé "boosts" the base methods**.** Results shown are for SATé used with MAFFT. Similar improvements seen for use with other MSA methods (e.g., Prank, Opal, Muscle, ClustalW).
- Biological datasets: Similar results on large benchmark datasets (structurally-based rRNA alignments)

One Iteration



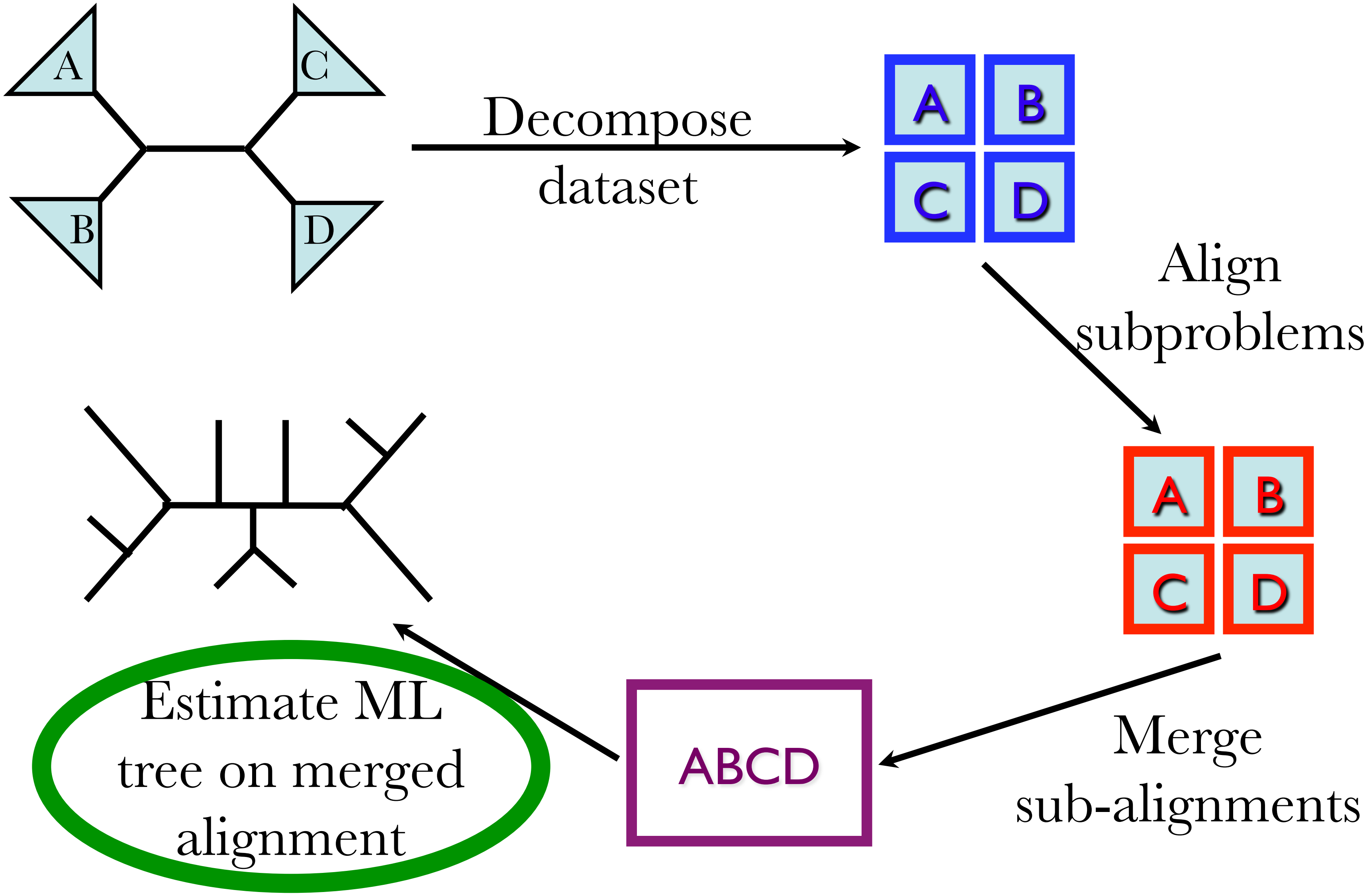
Decompose dataset

Align subproblems

Merge sub-alignments

Estimate ML tree on merged alignment

Limitations



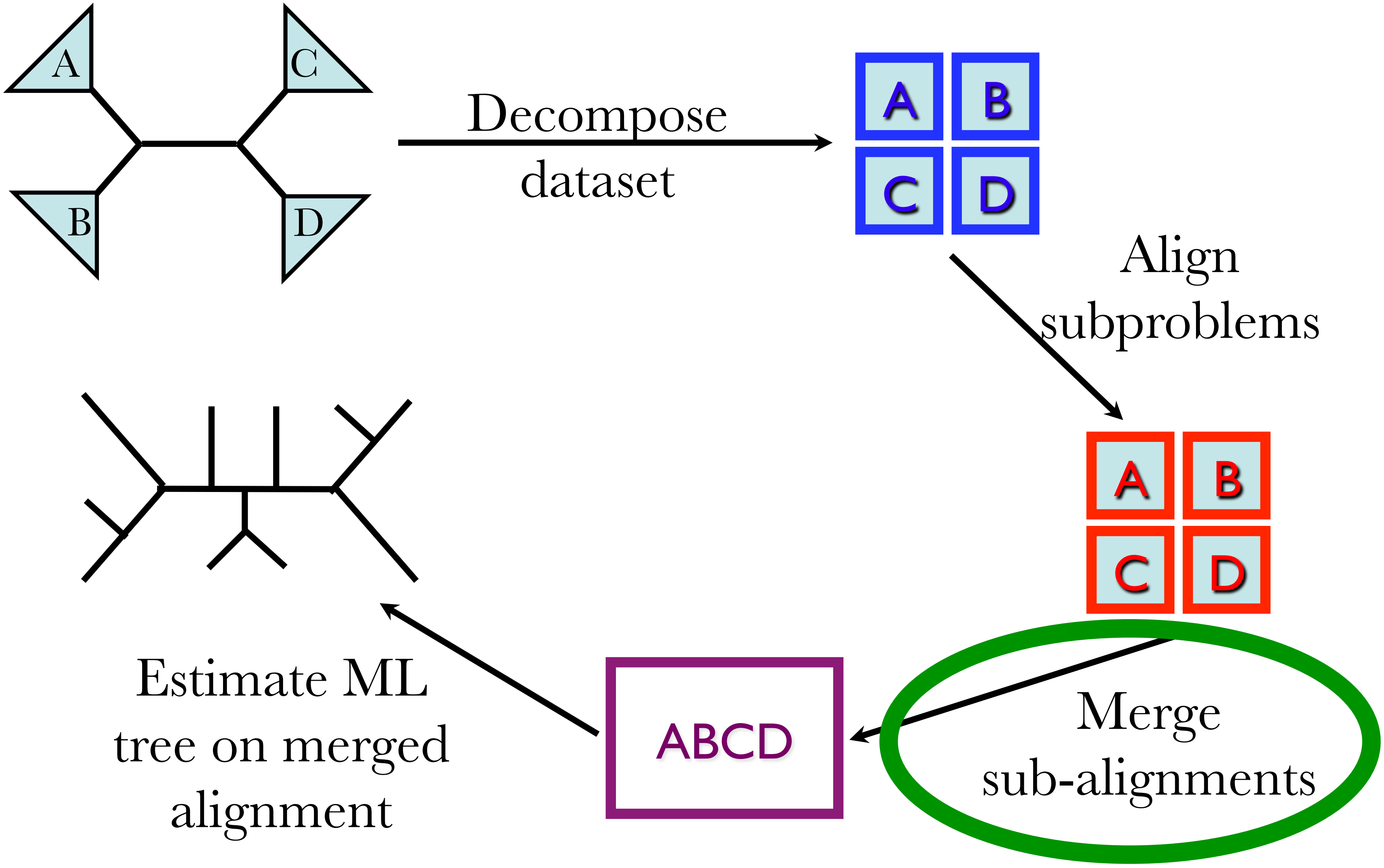
Decompose dataset

Align subproblems

Merge sub-alignments

Estimate ML tree on merged alignment

Limitations



Decompose dataset

Align subproblems

Merge sub-alignments

Estimate ML tree on merged alignment

Trees without alignments?

- Estimating very large alignments with high accuracy is very difficult – some datasets are considered “unalignable”.

- Running maximum likelihood on a large alignment is very computationally intensive.

Part III: DACTAL

(Divide-And-Conquer Trees (without) ALignments)

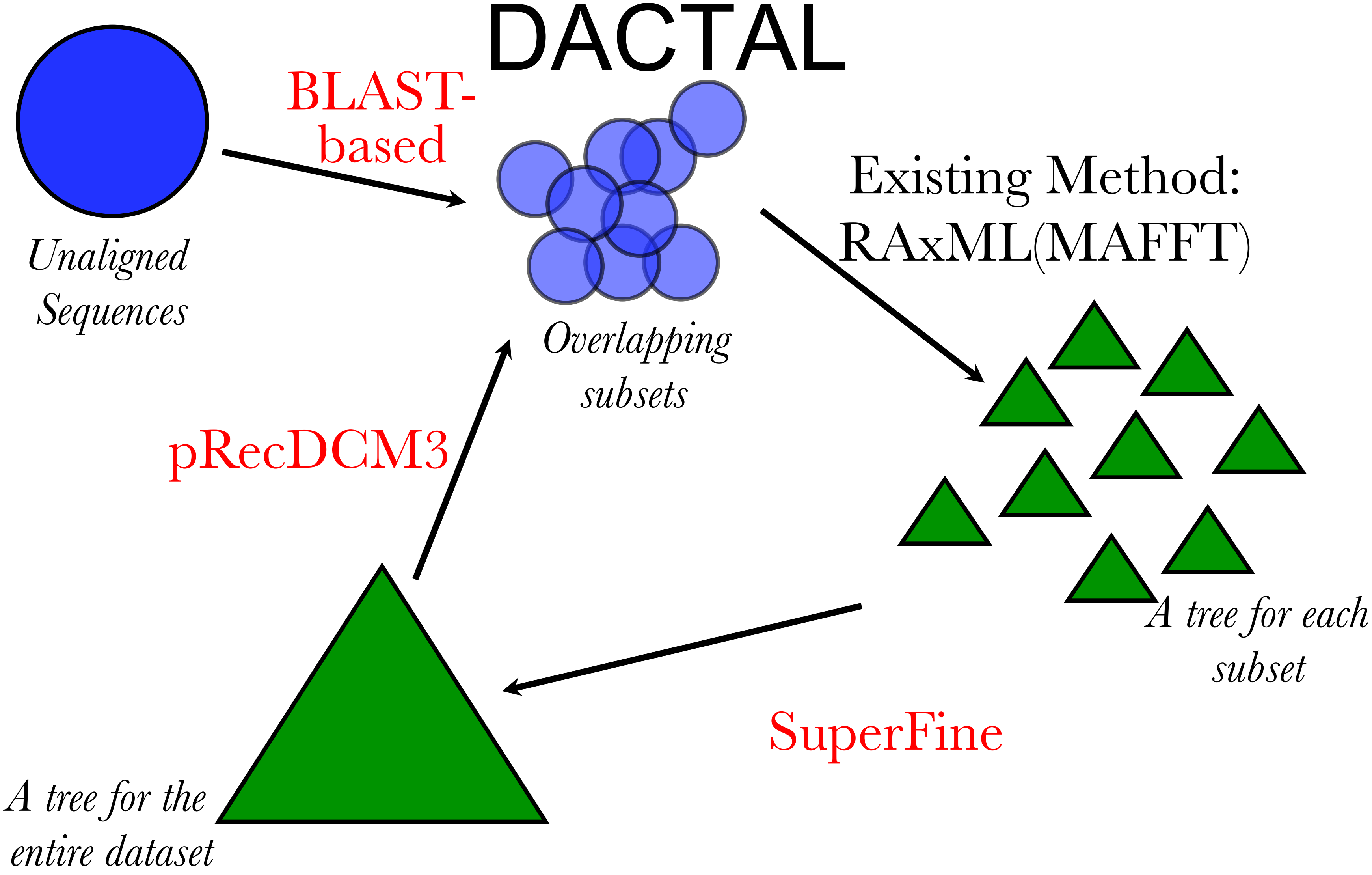
- Input: set S of unaligned sequences
- Output: tree on S (but no alignment)

(Nelesen, Liu, Wang, Linder, and Warnow, RECOMB 2012 and Bioinformatics 2012)

DACTAL

Objective: To produce a highly accurate estimation of a very large tree without requiring a multiple sequence alignment of the full dataset.

DACTAL



Unaligned Sequences

BLAST-based

Overlapping subsets

Existing Method: RAxML(MAFFT)

A tree for each subset

SuperFine

A tree for the entire dataset

pRecDCM3

SuperFine: supertree “booster”

- Phase 1: construct the Strict Consensus Merger supertree (Huson, Nettles, and Warnow, RECOMB 1999). The SCM tree is generally highly unresolved, but it solves the NP-hard Tree Compatibility Problem for some special cases.

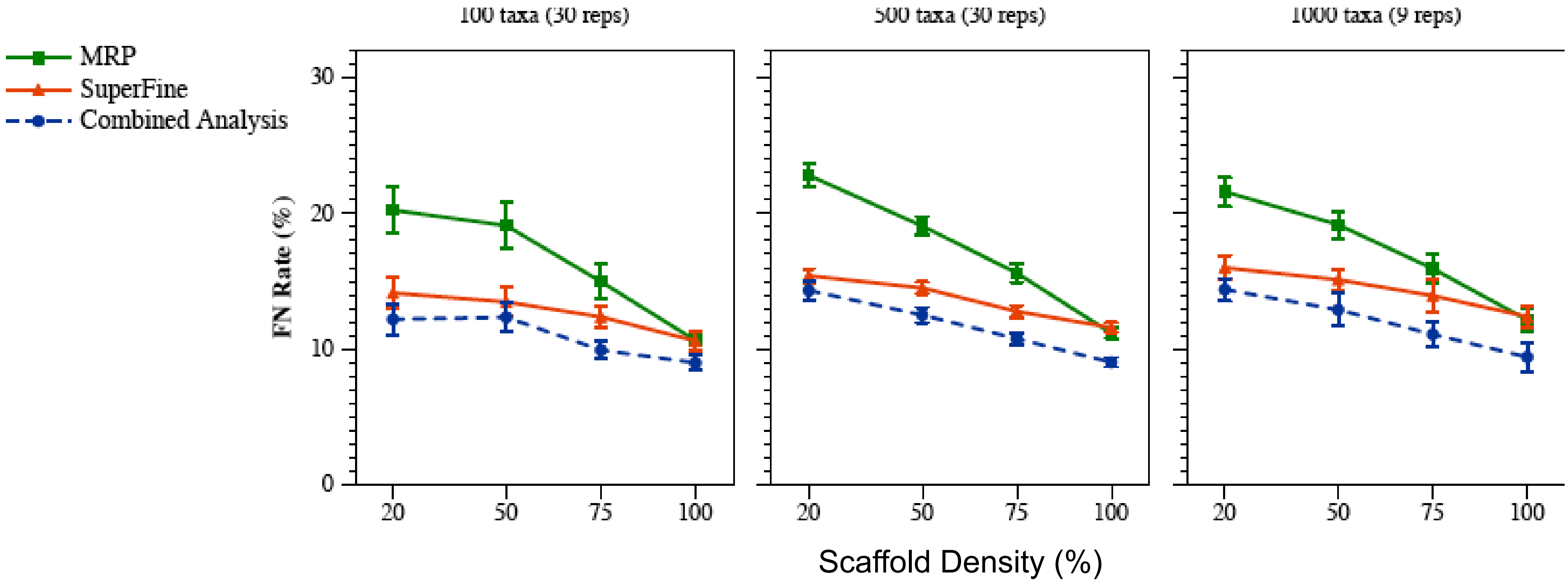
- Phase 2: Refine the tree by resolving each high degree node using a “base” supertree method (e.g., MRP).

Examples: SuperFine+MRP -- boosts MRP; but also
SuperFine+QMC, SuperFine+MRL, etc.

Swenson et al., Systematic Biology, 2012

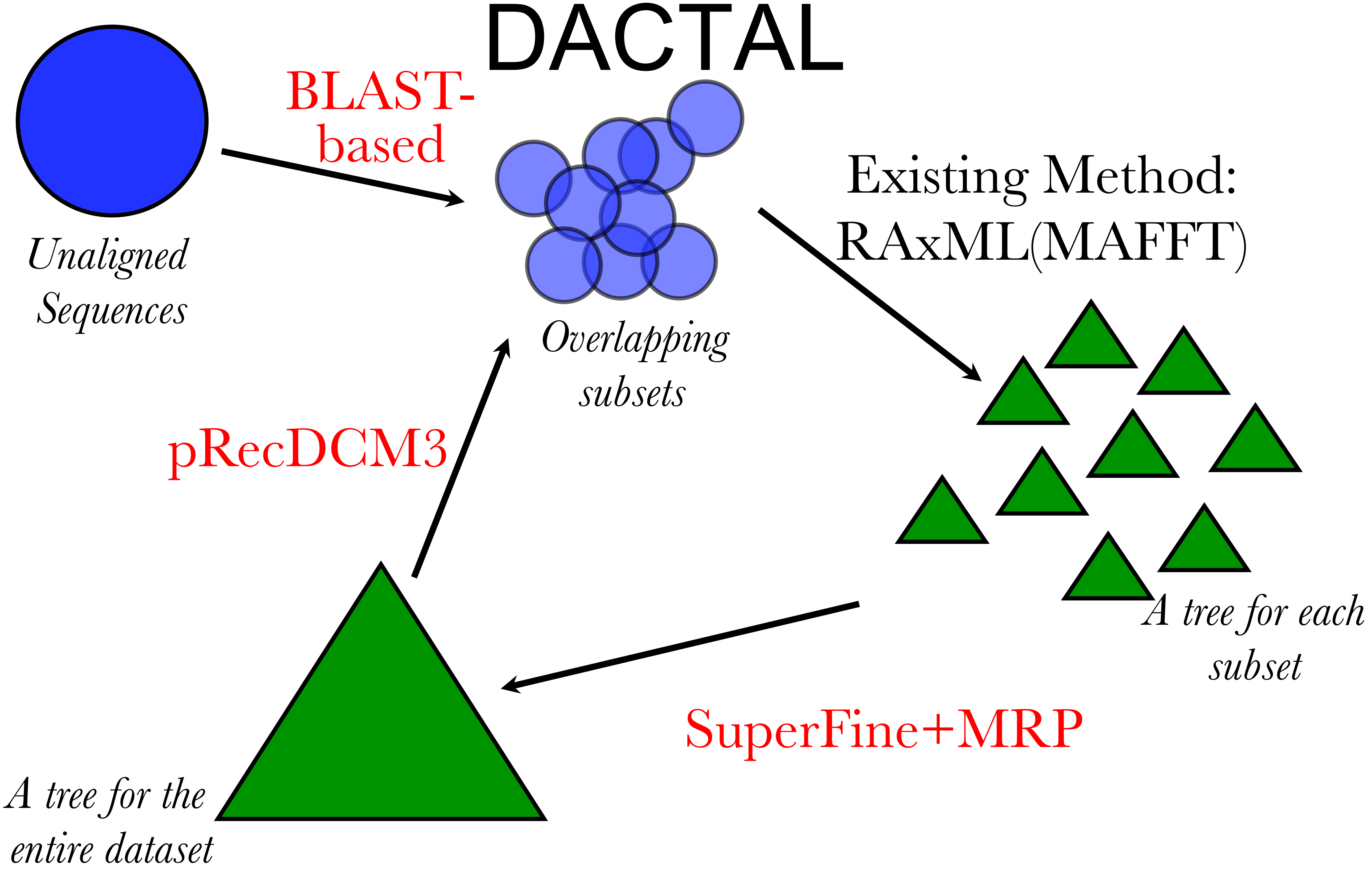
Nguyen et al., Algorithms for Molec Biol, 2012

SuperFine+MRP vs. MRP



(Swenson et al., Syst. Biol. 2012)

DACTAL



Unaligned Sequences

BLAST-based

Overlapping subsets

Existing Method: RAxML(MAFFT)

A tree for each subset

SuperFine+MRP

A tree for the entire dataset

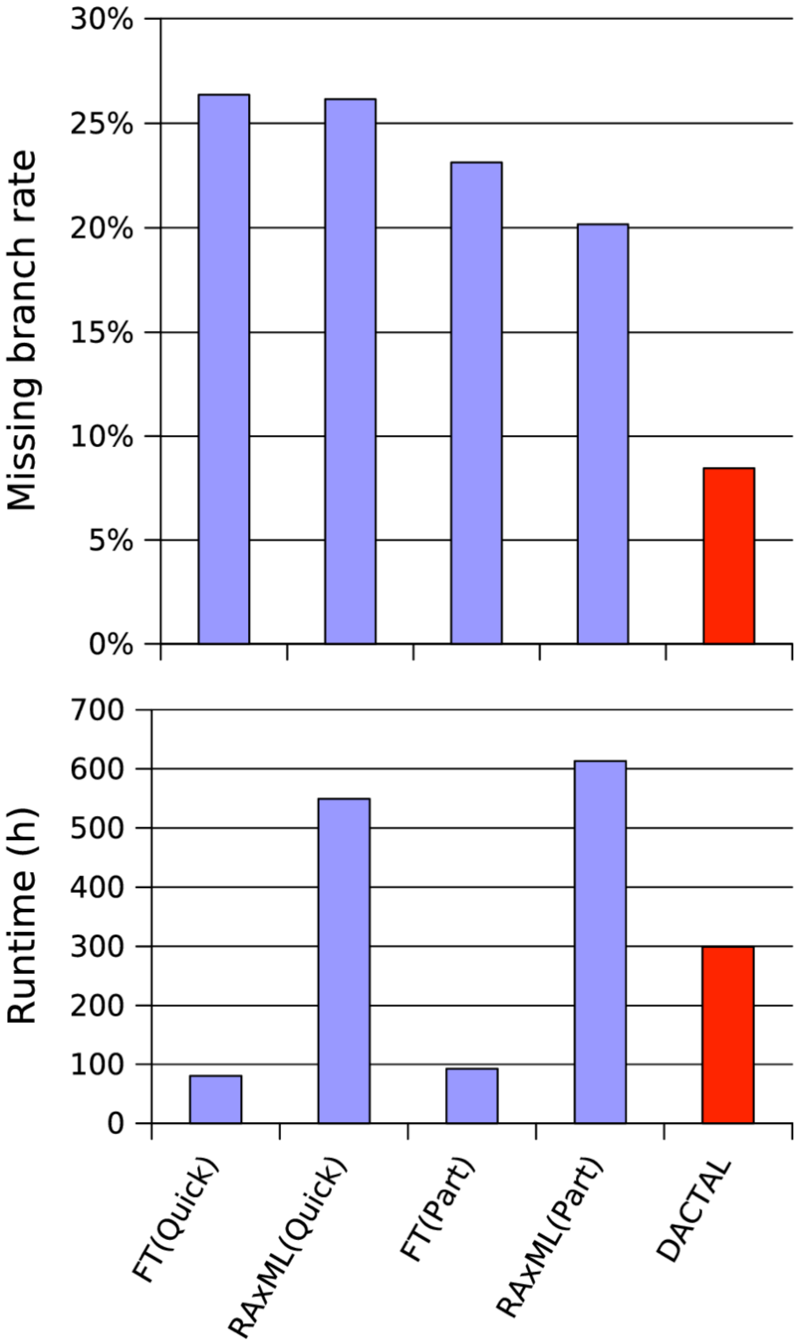
pRecDCM3

Performance on biological datasets

Average performance on three 16S RNA datasets with curated alignments based upon secondary structure, with 6323 to 27,643 sequences

Reference trees are 75% RAxML bootstrap trees

DACTAL is run with 5 iterations, starting from FastTree(PartTree)



Part IV: 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

HMMER (Sean Eddy, HHMI) leading software for this purpose

Using HMMER

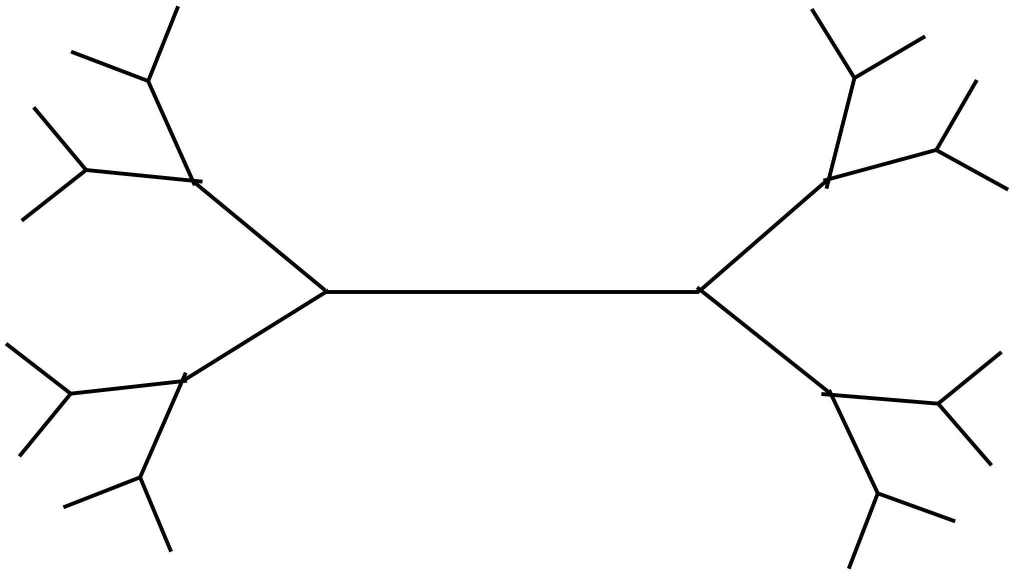
Using HMMER works well…

Using HMMER

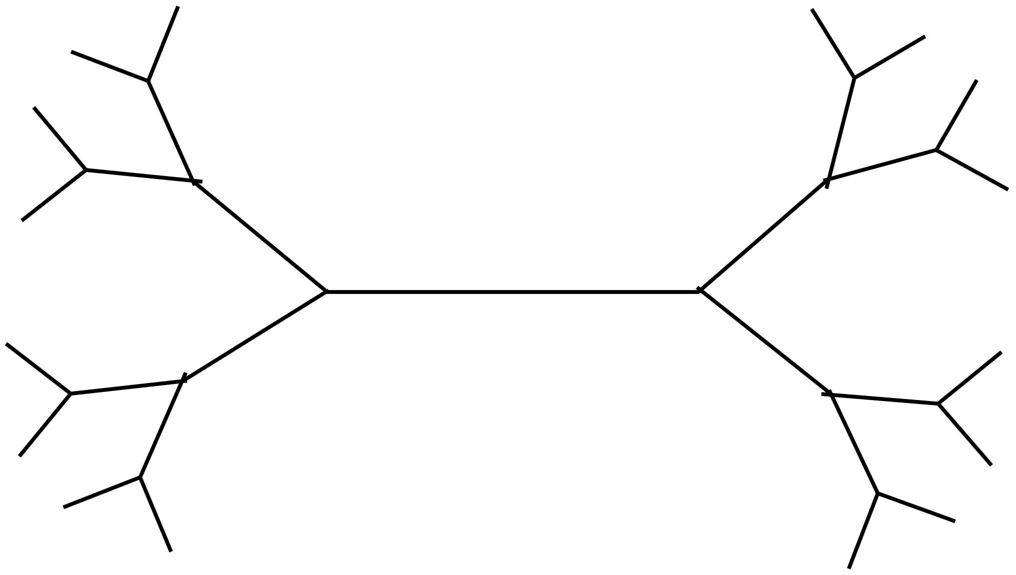
Using HMMER works well…except when the dataset is big!

Using HMMER to add sequences to an existing alignment

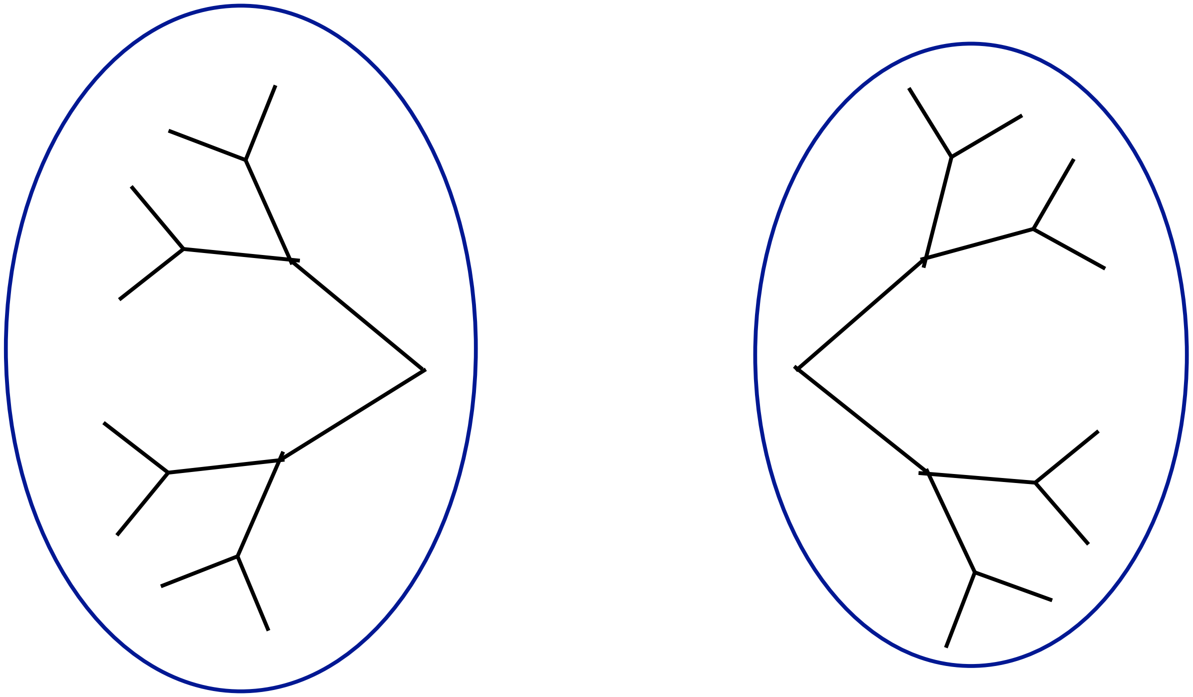
1) build one HMM for the backbone alignment
2) Align sequences to the HMM, and insert into backbone alignment



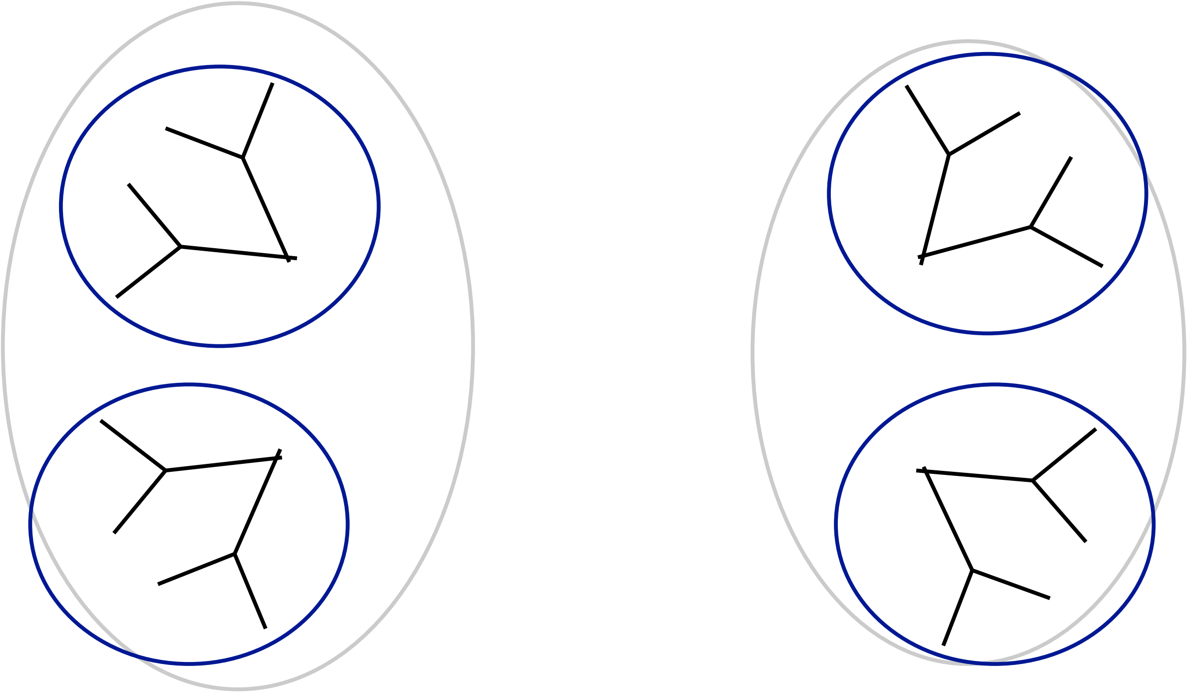
One Hidden Markov Model for the entire alignment?



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

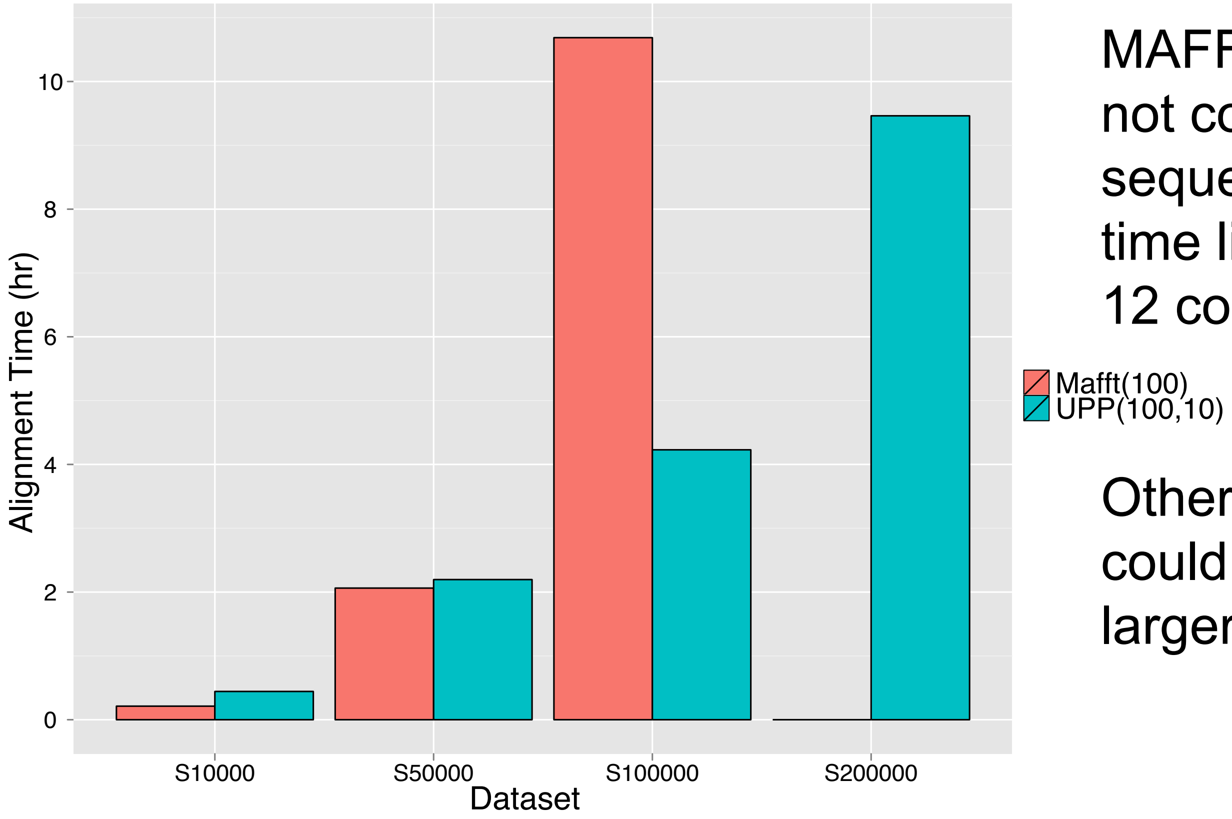
- Size of backbone matters – small backbones are sufficient for most datasets (except for ones with very high rates of evolution). Random backbones are fine.
- Number of HMMs matters, and depends on the rate of evolution and number of taxa.
- Backbone alignment and tree matter; we use SATé.

Evaluation of UPP

- Simulated Datasets: 1,000 to 1,000,000 sequences (RNASim, Junhyong Kim, Penn)
- Biological datasets: up to 28,000 rRNA sequences with structural reference alignments (CRW, Robin Gutell, Texas)
- Methods: MAFFT-profile, UPP(x,y) and UPP(x,x) (“HMMER”), all on the SATé backbone alignment. Also, MAFFT-parttree, Muscle, Opal, Clustal-quicktree, and SATé.
- Criteria: Alignment error (SP-FN and SP-FP), tree error, and time

MAFFT-profile is the MSA method with the best accuracy of standard methods.

UPP vs. MAFFT Running Time



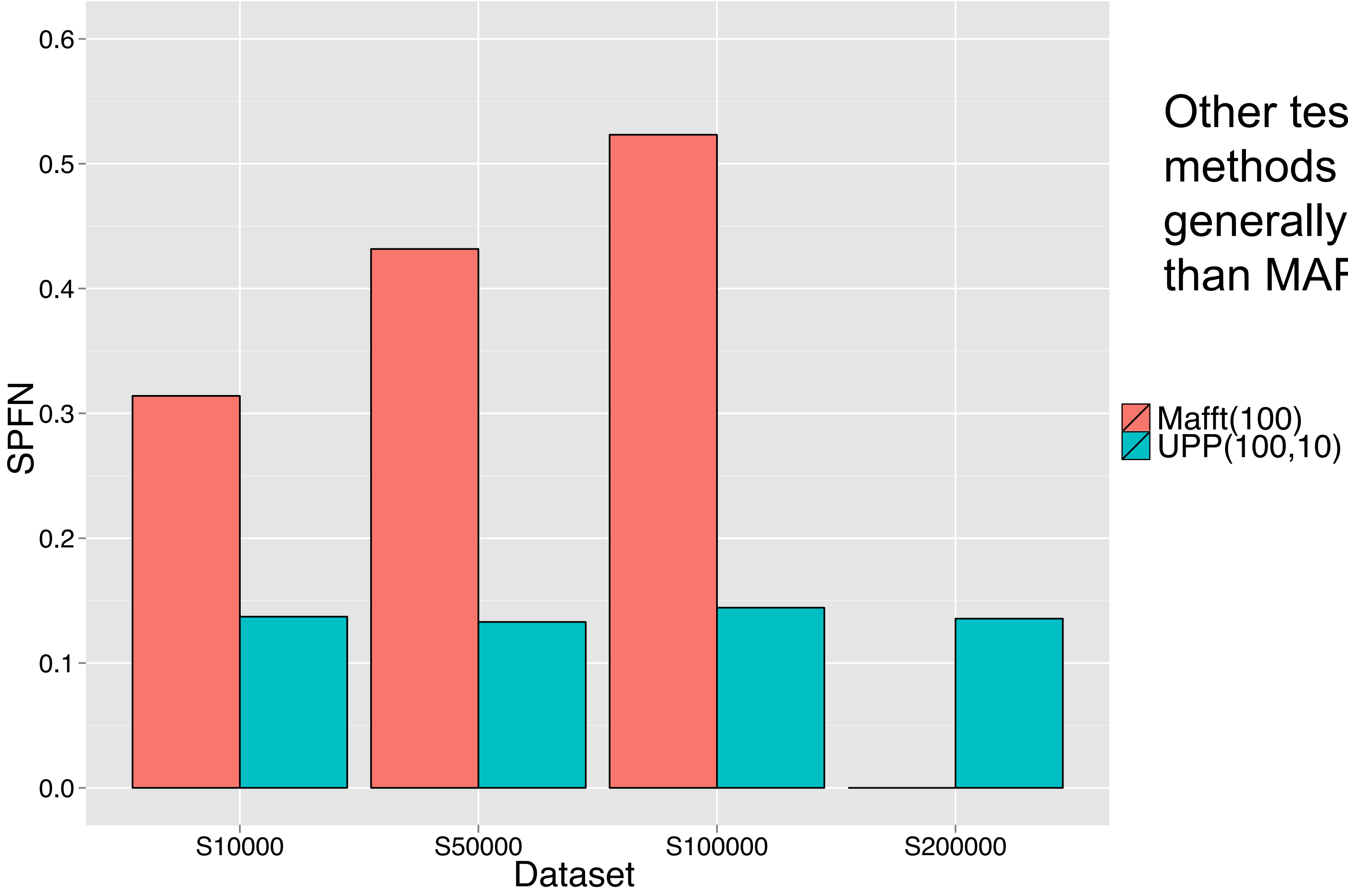
MAFFT-profile did not complete on 200K sequences within the time limit (24 hours on 12 cores.)

Other MSA methods could not run on the larger data sets.

RNASim data, 10K to 1,000K sequences

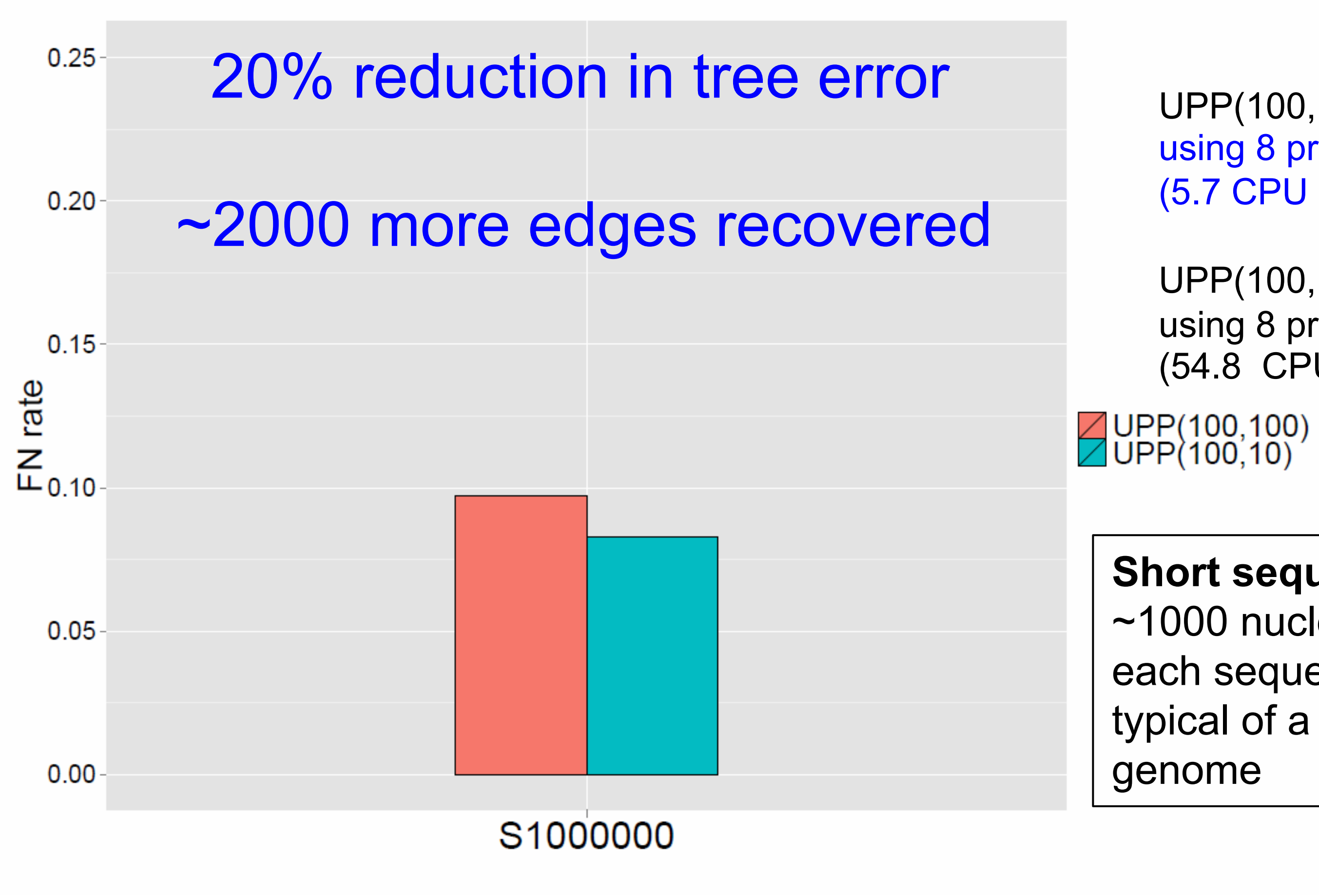
Elapsed time on 12-core machine

UPP vs. MAFFT Alignment Error



Other tested methods were generally worse than MAFFT.

One Million Sequence Alignment: Tree Error



20% reduction in tree error

~2000 more edges recovered

UPP(100,100): 1.6 days using 8 processors (5.7 CPU days)

UPP(100,10): 7 days using 8 processors (54.8 CPU days)

Short sequences: ~1000 nucleotides in each sequence, so typical of a gene, not a genome

Similar improvements on all datasets.
Thus, using multiple HMMs improves tree accuracy.

UPP performance

- Speed: UPP is very fast, parallelizable, and scalable.
- UPP vs. standard MSA methods: UPP alignments are more accurate on large datasets (with 1000+ taxa), and trees on UPP alignments are more accurate than trees on standard alignments.
- UPP vs. SATé: UPP can analyze larger datasets and is much faster; UPP has about the same alignment accuracy, but produces slightly less accurate trees (data not shown).
- UPP vs. PASTA (new method, in prep.): Both can analyze the same datasets, but PASTA is slower. Both have about the same alignment accuracy, but PASTA produces slightly more accurate trees (like SATé).

Other uses of multiple HMMs

- SEPP: Phylogenetic Placement of short reads into existing tree (Nguyen, Mirarab, and Warnow, PSB 2012)

- TIPP: taxon identification of metagenomic sequences (in preparation, Nguyen et al. 2013)

Part V: Discussion

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

Software that:

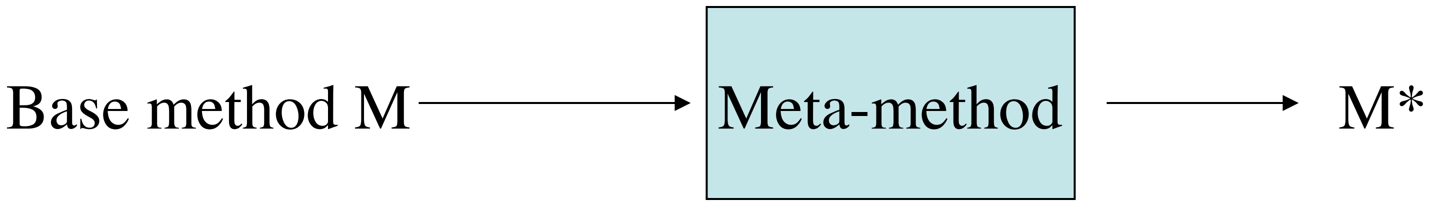
- Can run efficiently on *desktop* computers on large datasets
- Can analyze ultra-large datasets (100,000+) using multiple processors
- Is freely available in *open source* form, with biologist-friendly GUIs

4 methods

- SATé: co-estimation of alignments and trees
- SuperFine: supertree estimation
- DACTAL: trees without alignments
- UPP: ultra-large multiple sequence alignment

Meta-Methods

- Meta-methods “boost” the performance of base methods (e.g., for phylogeny or alignment estimation).



Base method M → Meta-method → M*

Phylogenetic “boosters”

Goal: improve accuracy, speed, robustness, or theoretical guarantees of base methods

Techniques: divide-and-conquer, iteration, chordal graph algorithms, and “bin-and-conquer”

Examples:

- DCM-boosting for distance-based methods (1999)
- DCM-boosting for heuristics for NP-hard problems (1999)
- SATé-boosting for alignment methods (2009 and 2012)
- SuperFine-boosting for supertree methods (2012)
- DACTAL: almost alignment-free phylogeny estimation methods (2012)
- SEPP-boosting for phylogenetic placement of short sequences (2012)
- UPP-boosting for alignment methods (in preparation)
- PASTA-boosting for alignment methods (in preparation)
- TIPP-boosting for metagenomic taxon identification (in preparation)
- Bin-and-conquer for coalescent-based species tree estimation (2013)

Algorithmic Strategies

- Divide-and-conquer
- Chordal graph decompositions
- Iteration
- Multiple HMMs
- “Bin-and-conquer”

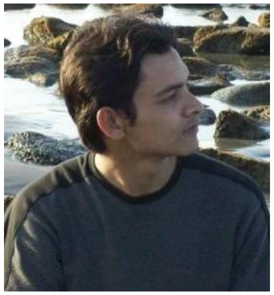
Computational Phylogenetics

Interesting combination of

- statistical estimation under Markov models of evolution
- mathematical modelling
- graph theory and combinatorics
- machine learning and data mining
- heuristics for NP-hard optimization problems
- high performance computing

Testing involves massive simulations

Warnow Laboratory

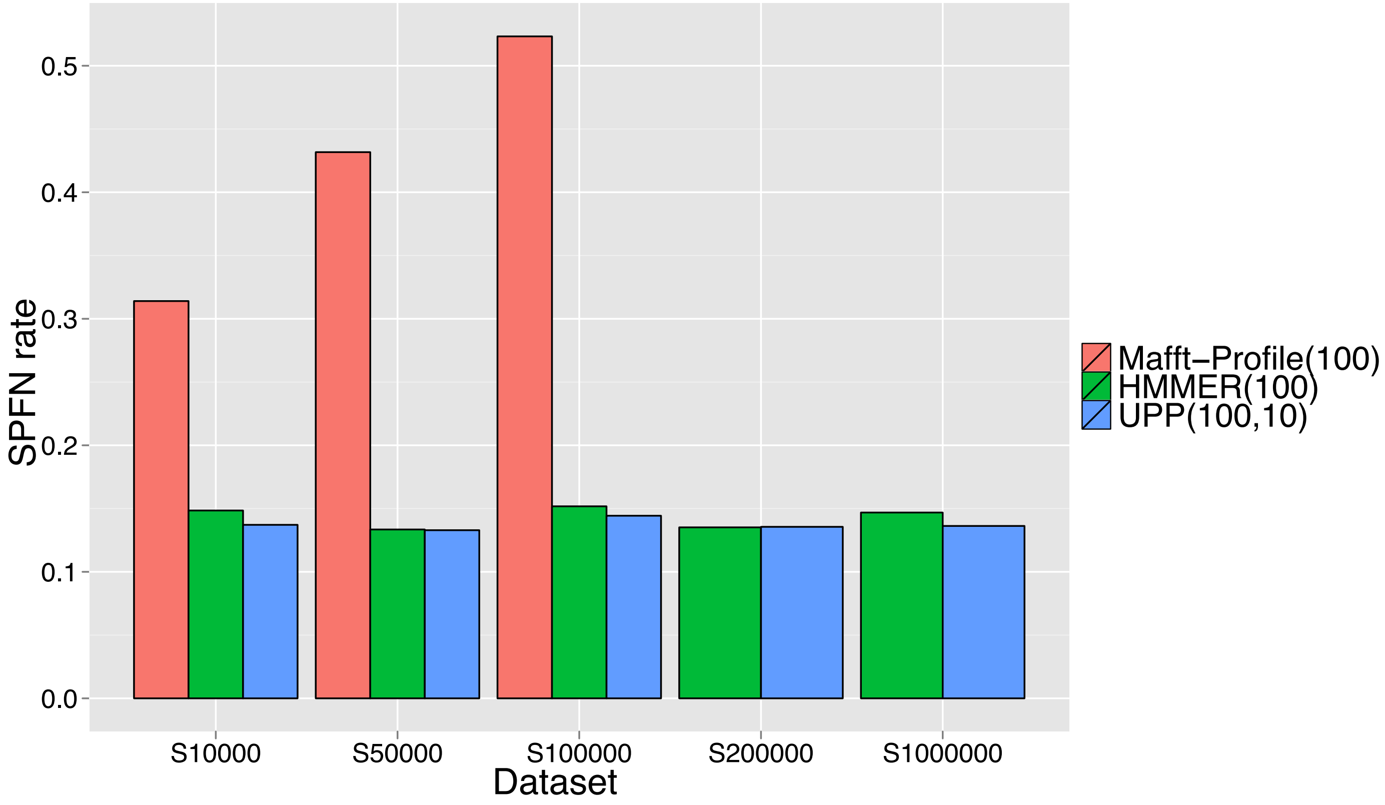


PhD students: Siavash Mirarab[1], Nam Nguyen, and Md. S. Bayzid[2]
Undergrad: Keerthana Kumar
Lab Website: http://www.cs.utexas.edu/users/phylo

Funding: Guggenheim Foundation, Packard Foundation, NSF, Microsoft Research New England, David Bruton Jr. Centennial Professorship, and TACC (Texas Advanced Computing Center)

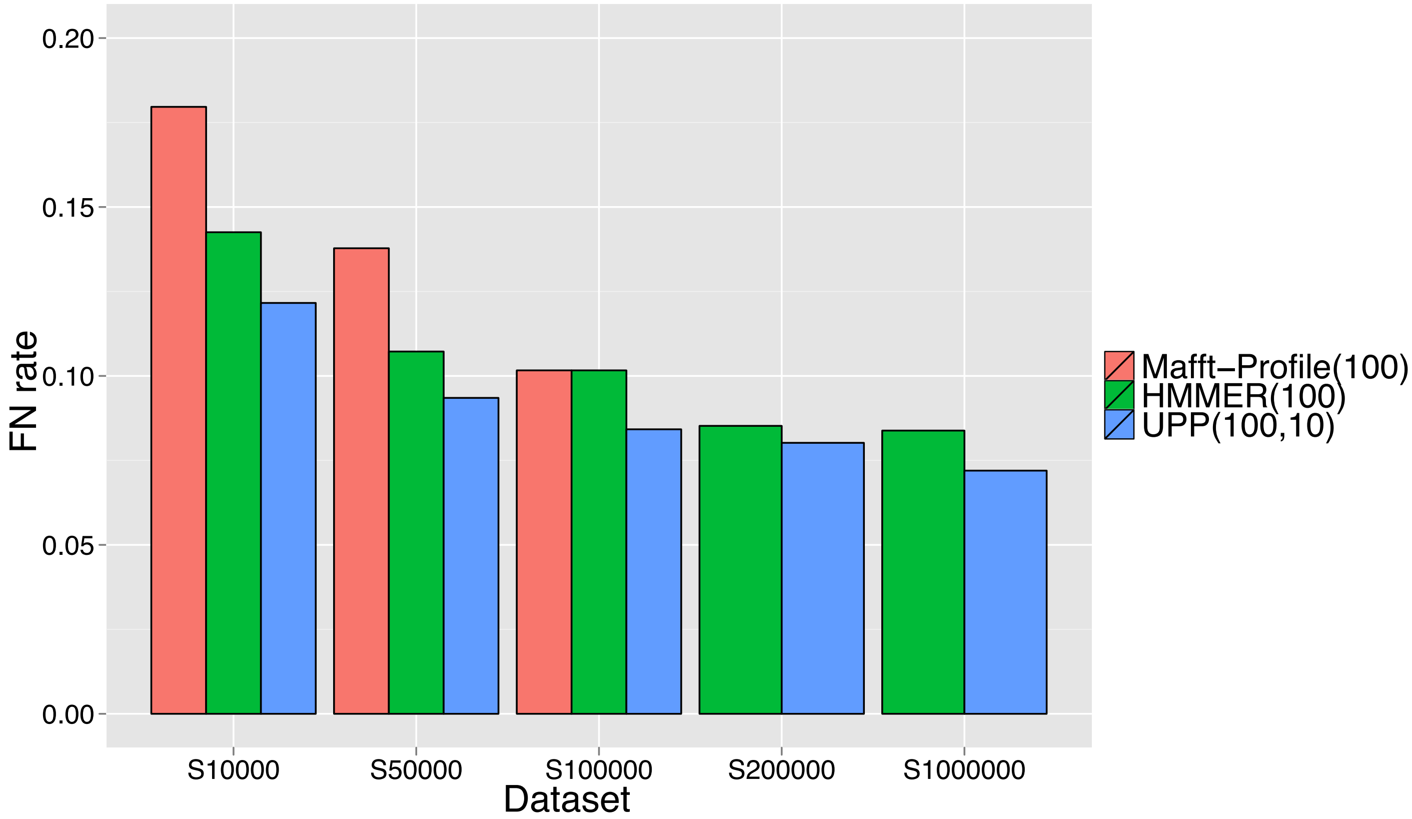
[1] HHMI International Predoctoral Fellow, [2] Fulbright Predoctoral Fellow

UPP vs. HMMER vs. MAFFT (alignment error)



MAFFT-profile alignment strategy not as accurate as UPP(100,10) or UPP(100,100).

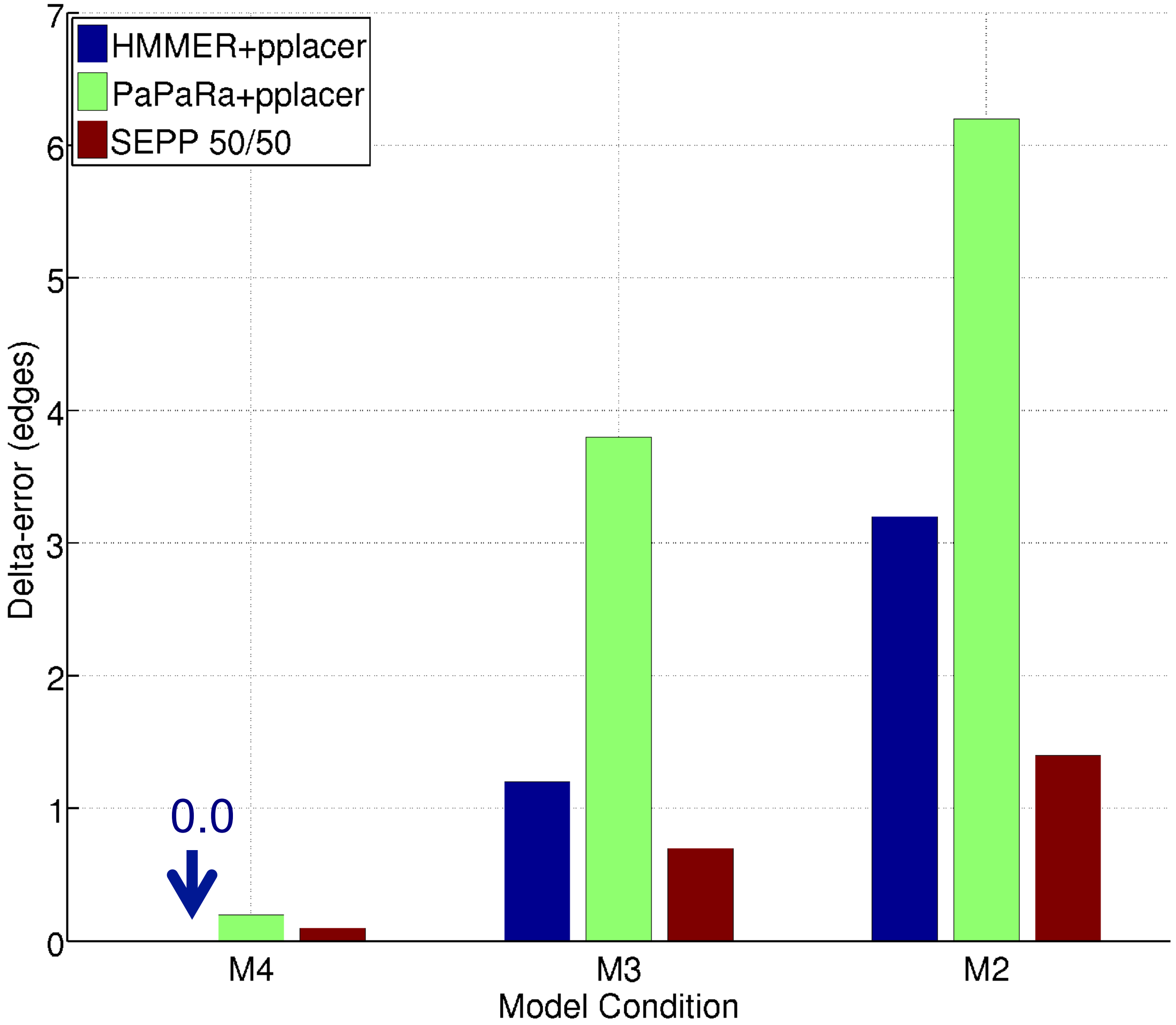
UPP vs. HMMER vs. MAFFT (tree error)



ML on UPP(100,10) and UPP(100,100) alignments both produce
produce better trees than MAFFT.

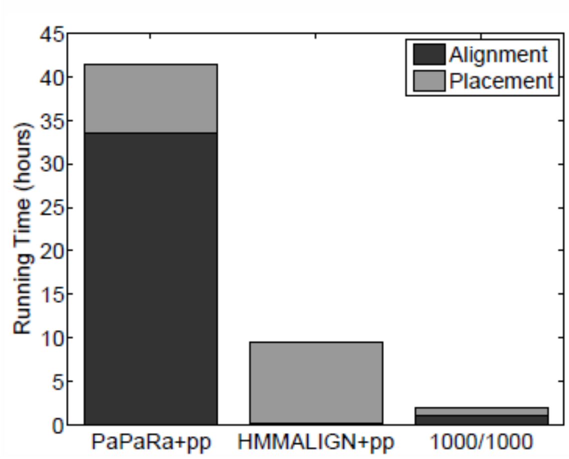
Decomposition into a family of HMMs improves resultant trees.

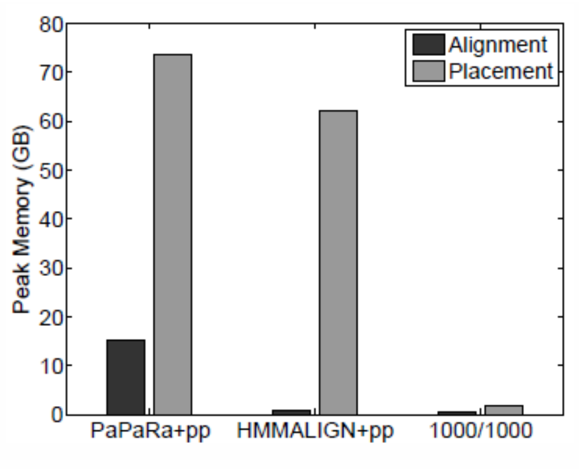
SEPP(10%), based on ~10 HMMs

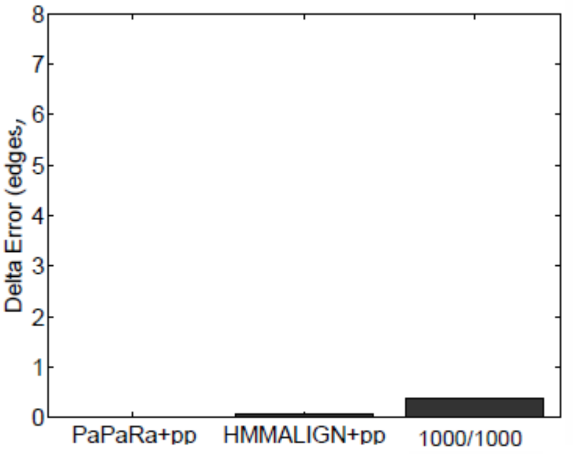


Increasing rate of evolution

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

Major Challenges:
large datasets, fragmentary sequences

- **Multiple sequence alignment**: Few methods can run on large datasets, and alignment accuracy is generally poor for large datasets with high rates of evolution.
- **Gene Tree Estimation:** standard methods have *poor accuracy* on even moderately large datasets, and the most accurate methods are enormously *computationally intensive* (weeks or months, high memory requirements).
- **Species Tree Estimation**: gene tree incongruence makes accurate estimation of species tree challenging.

Both phylogenetic estimation and multiple sequence alignment are also impacted by ***fragmentary data.***

DACTAL performance

- DACTAL faster and matches or improves upon accuracy of SATé-I for datasets with 1000 or more taxa.
- DACTAL outperforms two-phase methods, and the biggest gains are on the very large datasets.