Advances in Large-scale Multiple Sequence Alignment

Tandy Warnow
The Siebel School of Computing and Data Science
The University of Illinois at Urbana-Champaign
http://tandy.cs.illinois.edu

Supported by NSF grant IIBR Informatics 2006069

Multiple Sequence Alignment (MSA): *a scientific grand challenge*[1]

```
S1 = AGGCTATCACCTGACCTCCA        S1 = -AGGCTATCACCTGACCTCCA
S2 = TAGCTATCACGACCGC            S2 = TAG-CTATCAC--GACCGC--
S3 = TAGCTGACCGC                 S3 = TAG-CT-------GACCGC--
   …                             …
Sn = TCACGACCGACA        ⟶      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

What are MSAs used for?

- Inferring evolutionary histories
- Predicting biomolecular (RNA, protein) structure
- Genome annotation and assembly
- And others

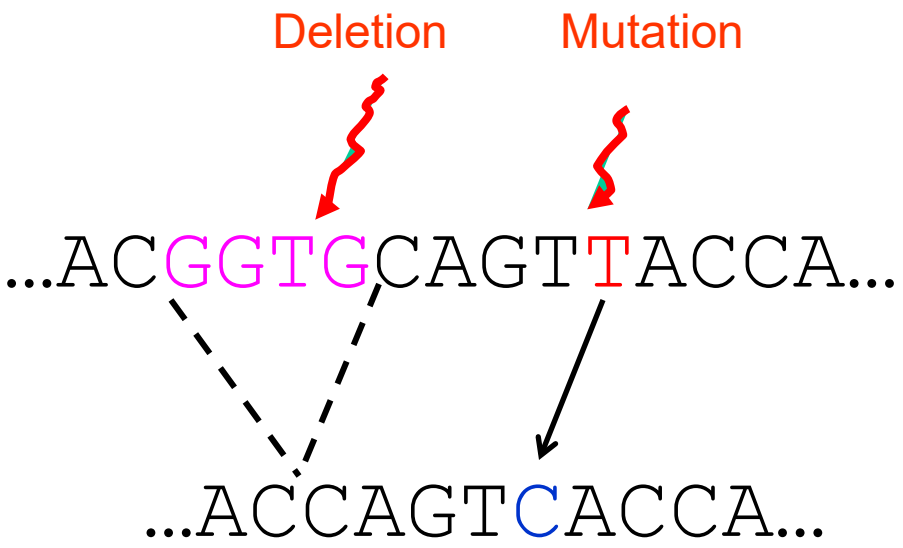
Phylogenomic pipeline

- Select taxon set and markers
- Gather and screen sequence data, possibly identify orthologs
- Compute multiple sequence alignments for each locus, and construct gene trees
- Compute species tree or network:
  - 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

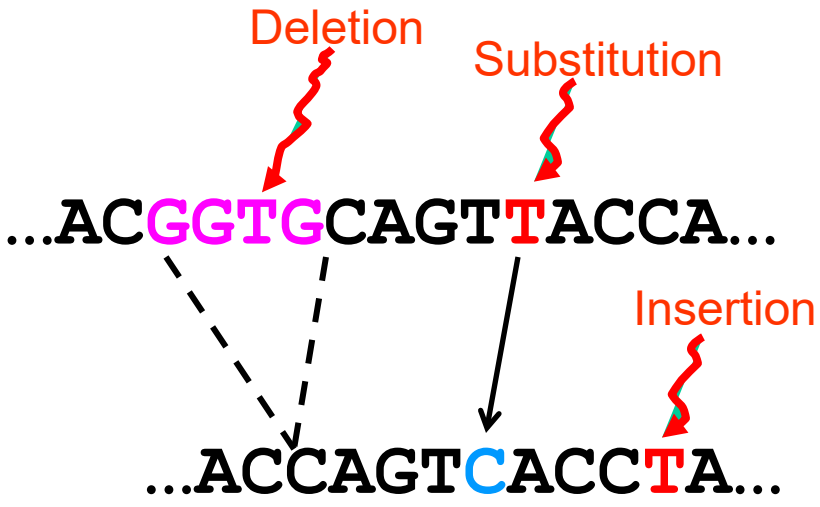
Phylogenomic pipeline

- Select taxon set and markers
- Gather and screen sequence data, possibly identify orthologs
- Compute multiple sequence alignments for each locus, and construct gene trees
- Compute species tree or network:
  - 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

Indels (insertions and deletions)



Homology: two letters (nucleotides or amino-acids) that are related by descent from a common ancestor

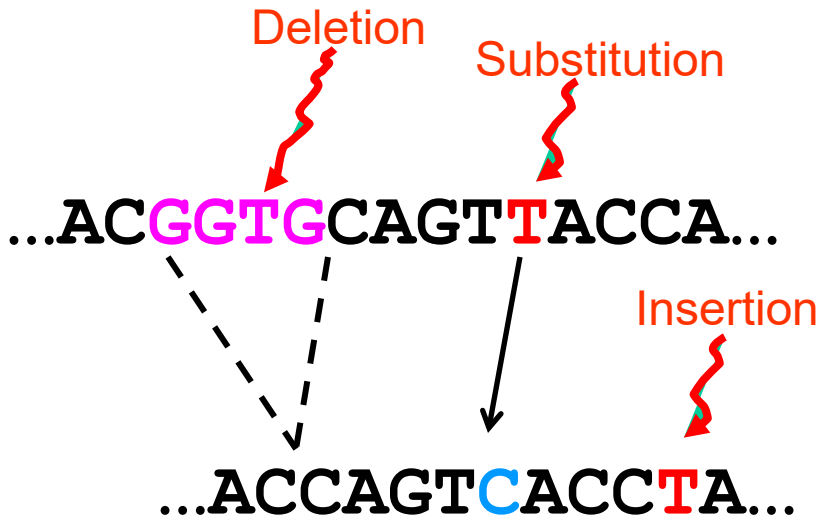


...ACGGTGCAGTTACC-A...

...AC----CAGTCACCTA...

The true pairwise alignment

- Reflects historical substitution, insertion, and deletion events
- Letters (nucleotides or amino acids) in the same column are supposed to be homologs



Then two deletions (one at front, long one at end)

CCAGT

...ACGGTGCAGTTACC-A...

...AC----CAGTCACCTA...

...-C----CAGT------...

The true multiple alignment

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

Pairwise alignment

- Global alignment: finding the lowest-cost edit transformation, solved using Needleman-Wunsch (dynamic programming)
- Polynomial time!
- Allows for variations in cost function and similarity scores, still polynomial time

Multiple Sequence Alignment

- Optimization problems extend pairwise alignment
  - Minimizing sum-of-pairs costs
  - Minimizing tree length
  - Likelihood-based approaches (e.g., Bayesian estimation)
- Optimization problems are NP-hard
- Bayesian estimation is even less scalable

Standard approaches?

- Standard methods use a variety of techniques, such as extending pairwise alignments with:
  - Star alignment
  - Progressive alignment
  - Ensemble methods, including “Consistency”
  - Supervised learning

Progressive alignment

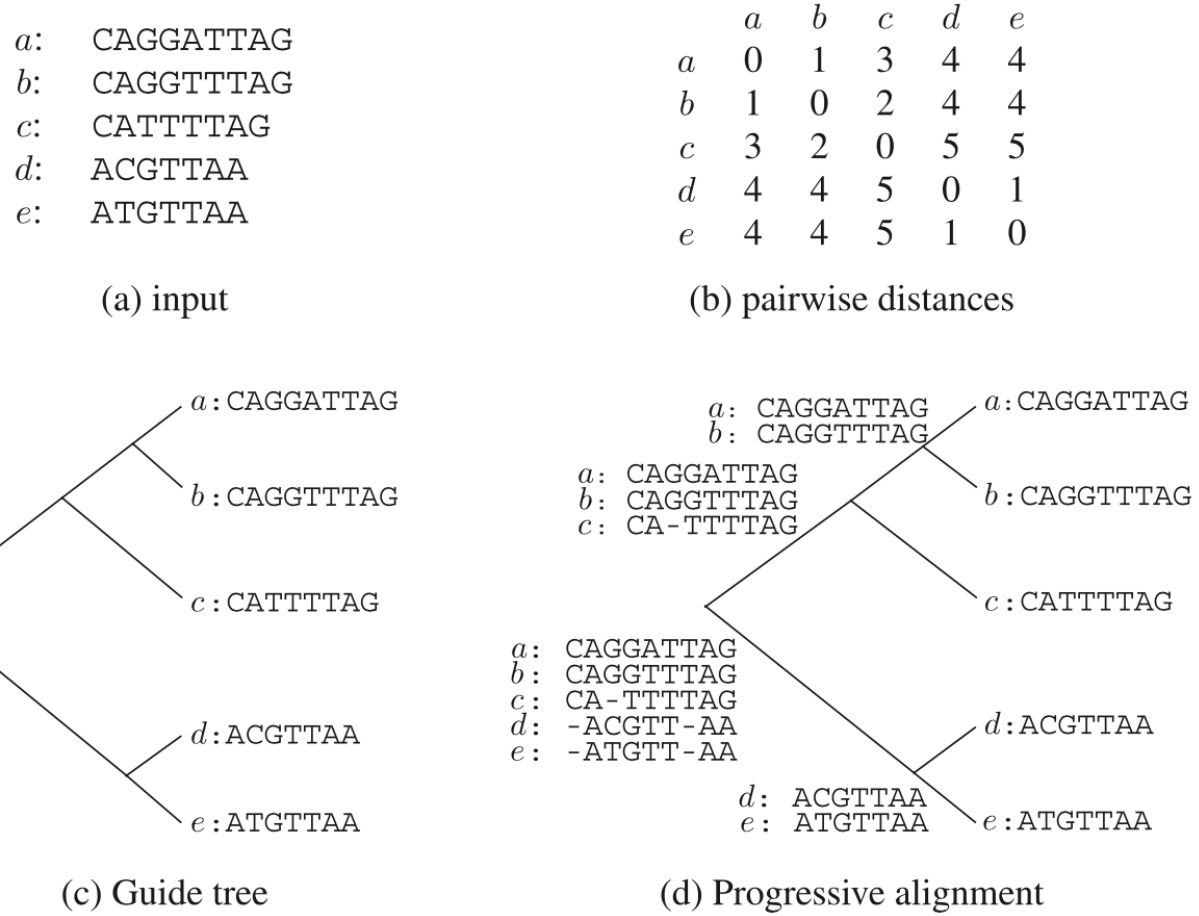
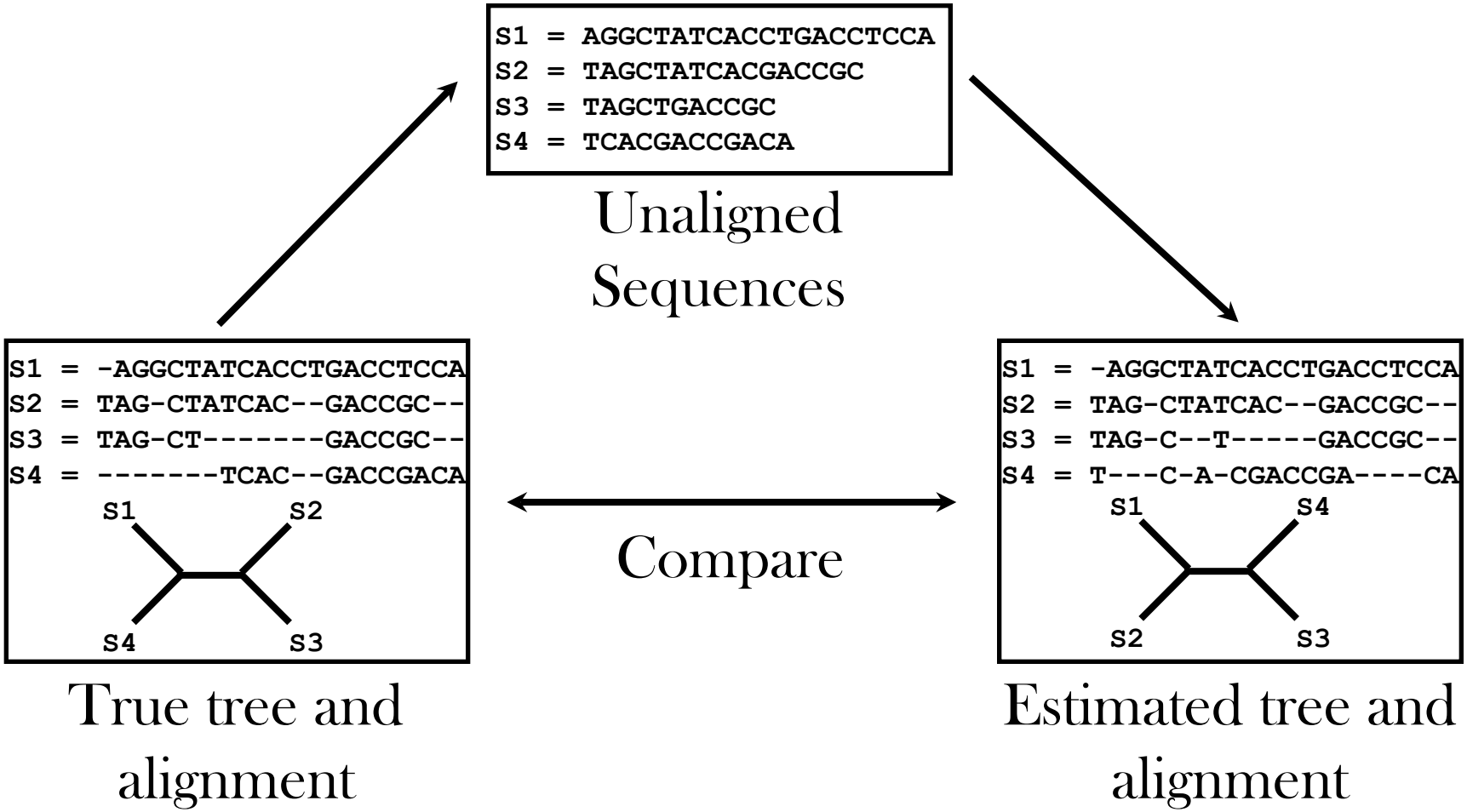


Figure 2.9 from Huson et al. (2010)

Simulation Studies



MSA+Tree estimation

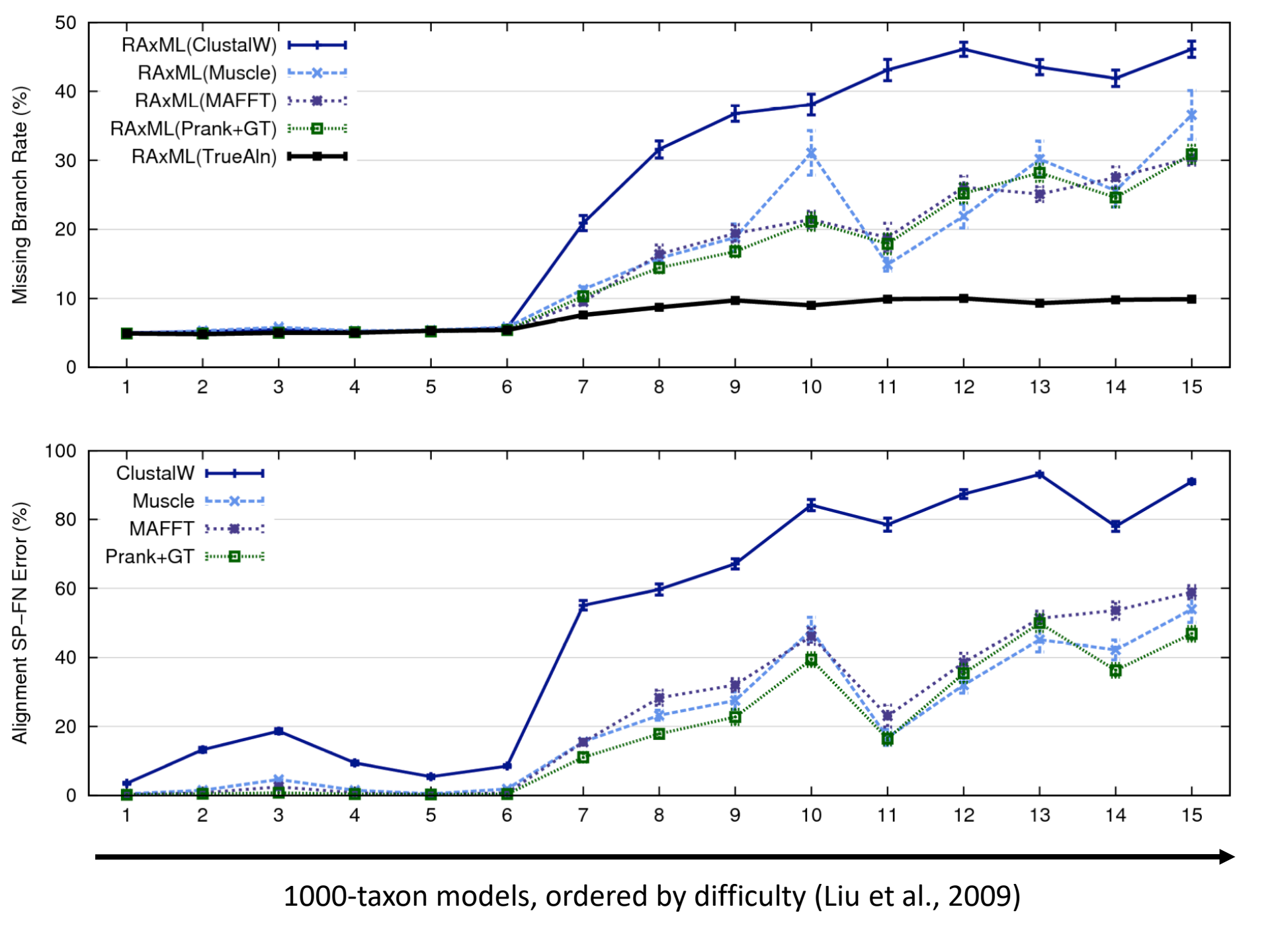
Alignment methods

- 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)

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)

Challenge:
Alignment of datasets with > 100,000 sequences with many very short sequences

What makes for an “easy” MSA?

- MSA is easy when the input is a small set of very similar sequences
  - All nearly the same length
  - Very few substitutions
  - Very few “indels”
- But large datasets are difficult, even when they are otherwise relatively “easy”

Large-scale MSA

Challenges

- High evolutionary rates
- Sequence length heterogeneity (e.g., fragments)
- Very long sequences

This talk

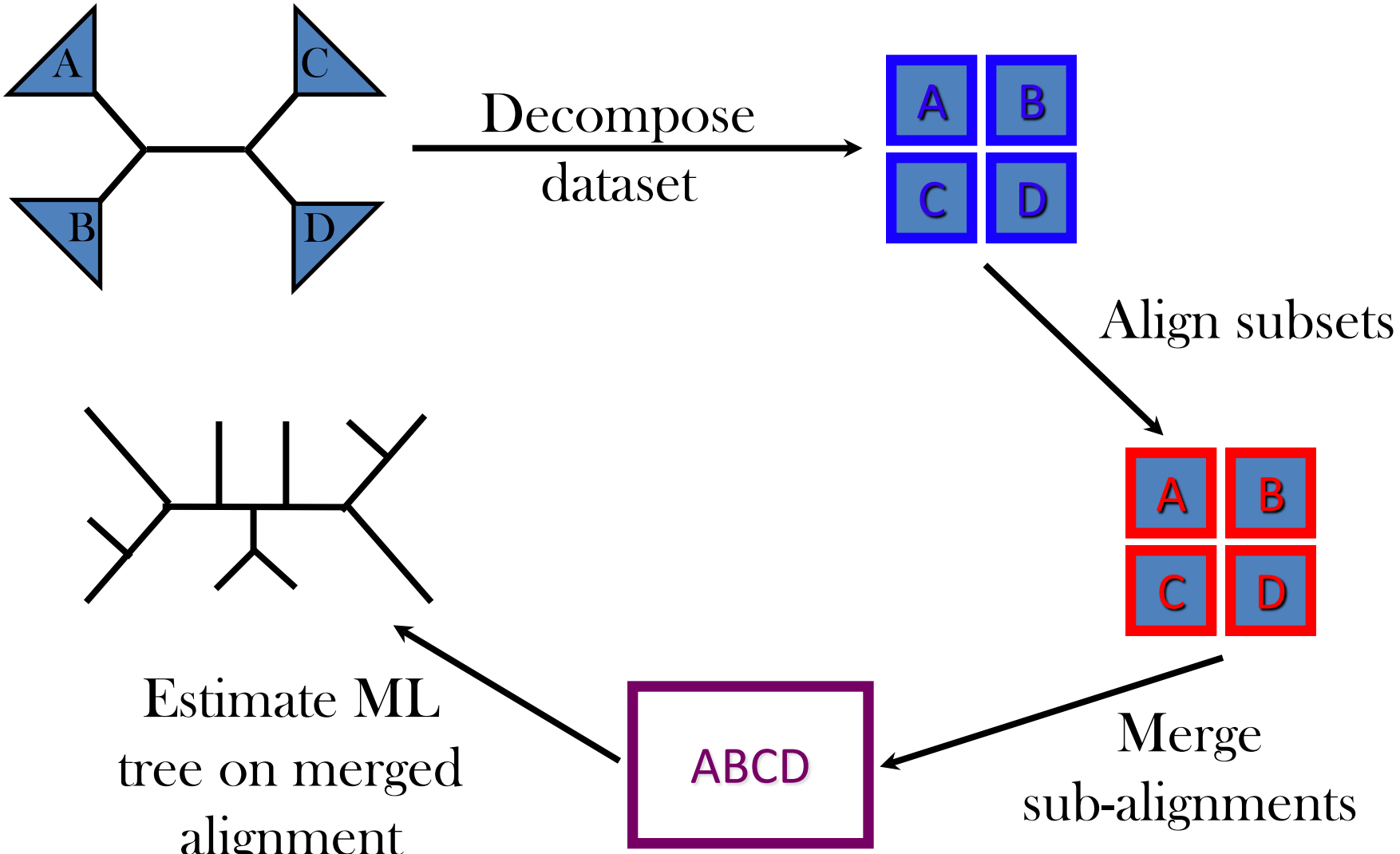
- Part 1: Divide-and-conquer: boosting MSA methods to large datasets
  - SATé, PASTA, MAGUS, and Recursive MAGUS
- Part 2: Adding sequences into alignments using Ensembles of profile HMMs
  - UPP, WITCH, and EMMA
- Part 3: Statistical alignment (e.g., BAli-Phy)
  - do we need to converge?
  - can divide-and-conquer improve scalability?
- Part 4: Discussion

Part I: Divide-and-Conquer

Divide-and-conquer

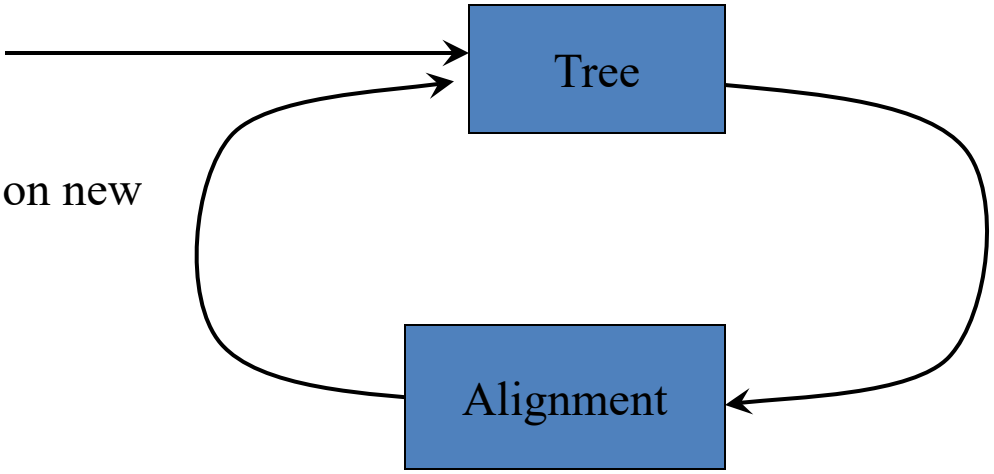
- Divide-and-conquer “meta-methods” for large numbers of sequences and high evolutionary rates:
  - SATé, PASTA, and MAGUS

Re-aligning on a tree



SATé, PASTA, and MAGUS Algorithms

Obtain initial alignment and estimated ML tree

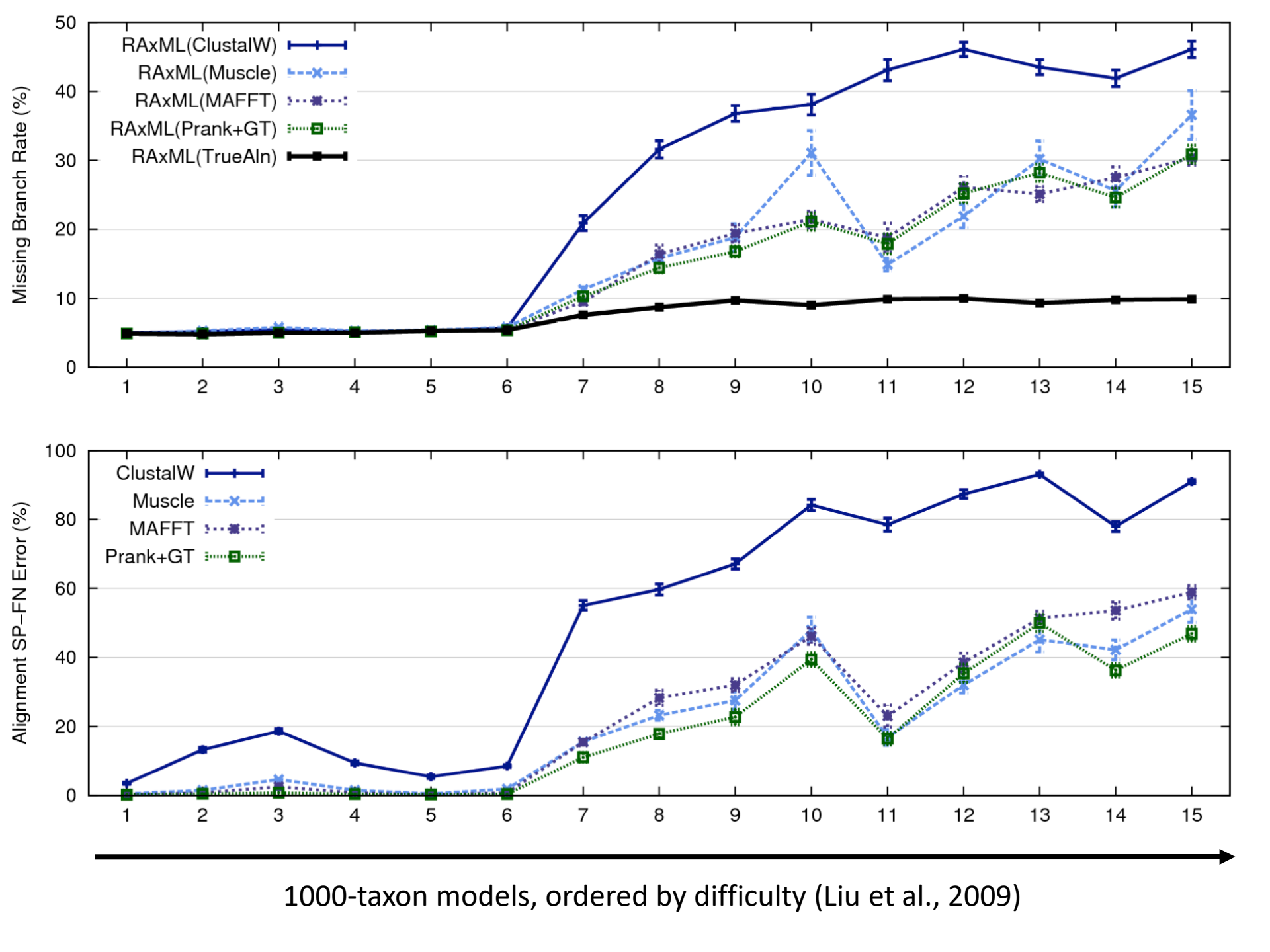


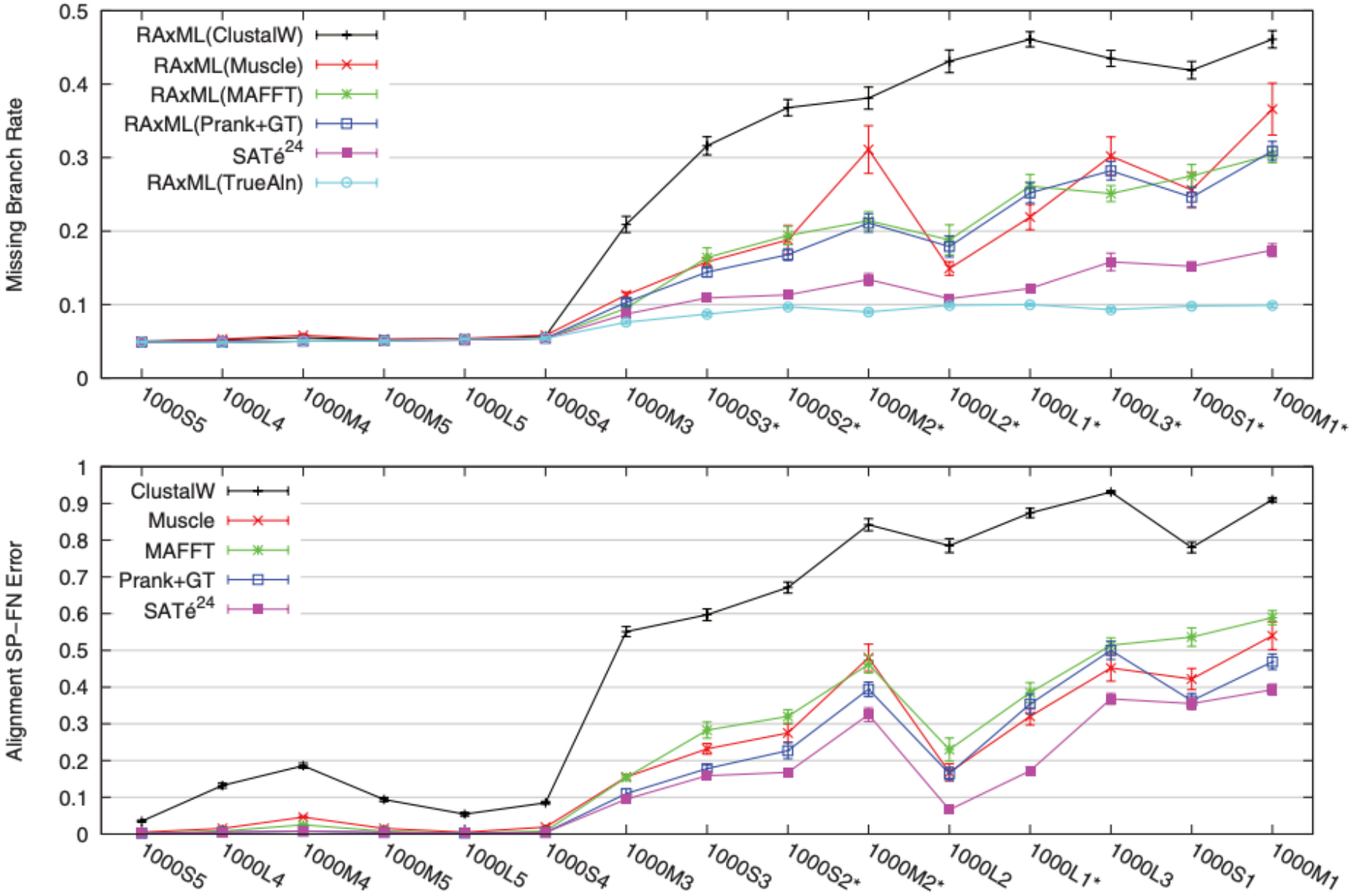
Estimate ML tree on new alignment

Use tree to compute new alignment

Repeat until termination condition, and

return the alignment/tree pair with the best ML score


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)


Missing Branch Rate
0.5
0.4
0.3
0.2
0.1
0
RAxML(ClustalW)
RAxML(Muscle)
RAxML(MAFFT)
RAxML(Prank+GT)
SATé24
RAxML(TrueAln)
1000S5
1000L4
1000M4
1000M5
1000L5
1000S4
1000M3
1000S3*
1000S2*
1000M2*
1000L2*
1000L1*
1000L3*
1000S1*
1000M1*
Alignment SP−FN Error
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
ClustalW
Muscle
MAFFT
Prank+GT
SATé24
1000S5
1000L4
1000M4
1000M5
1000L5
1000S4
1000M3
1000S3
1000S2*
1000M2*
1000L2
1000L1*
1000L3
1000S1
1000M1

Improvement over time

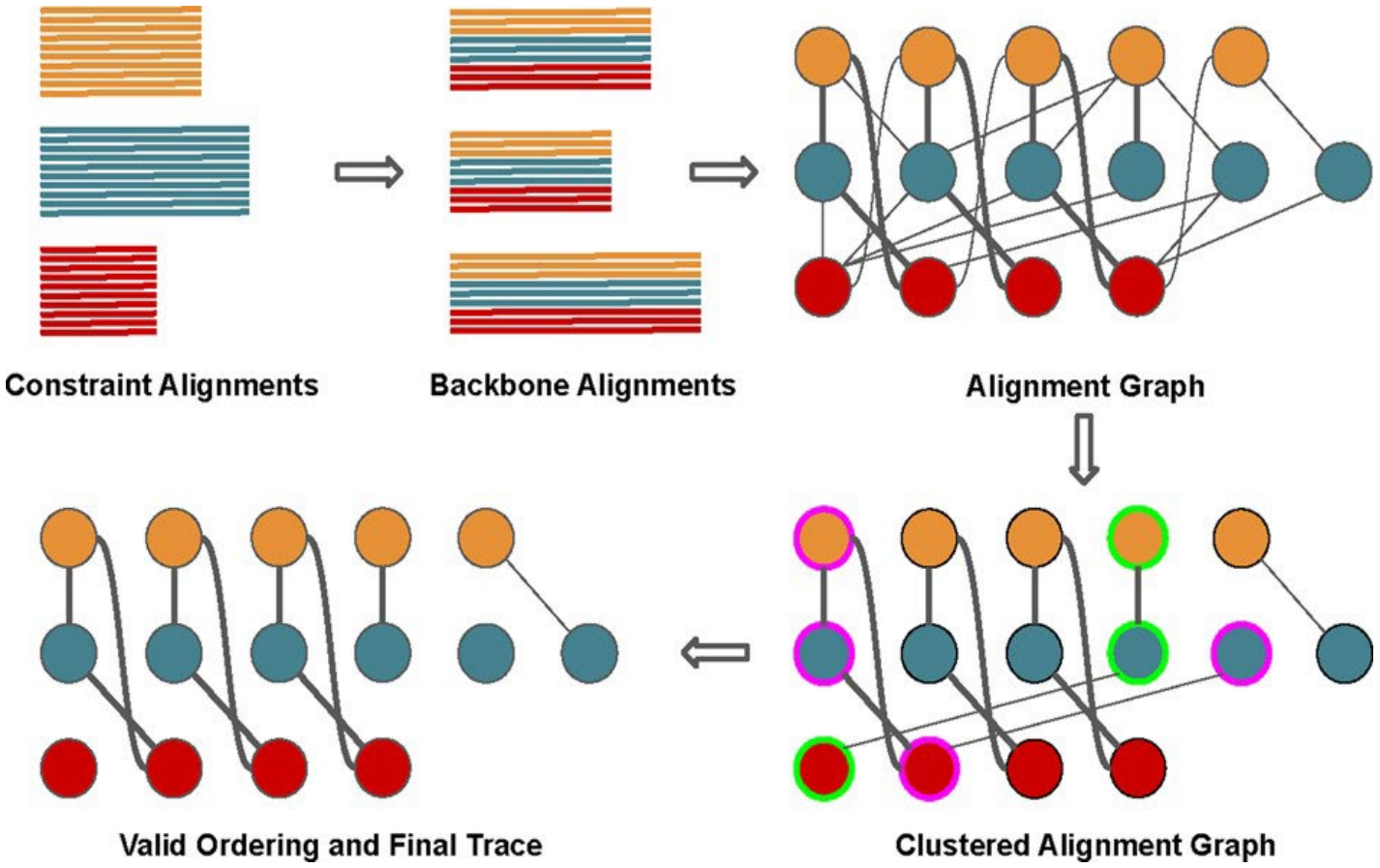
- SATé-1 (Science 2009): up to about 8,000
- SATé-2 (Syst Biol 2012): up to 50,000
- PASTA (J Comp Biol 2014): up to 1,000,000
- MAGUS (Bioinformatics 2021): more accurate than PASTA (and one iteration suffices) – up to 1,000,000

Each method improved on the previous with respect to MSA and Tree accuracy, speed, and scalability to large datasets

SATé-II vs PASTA vs MAGUS

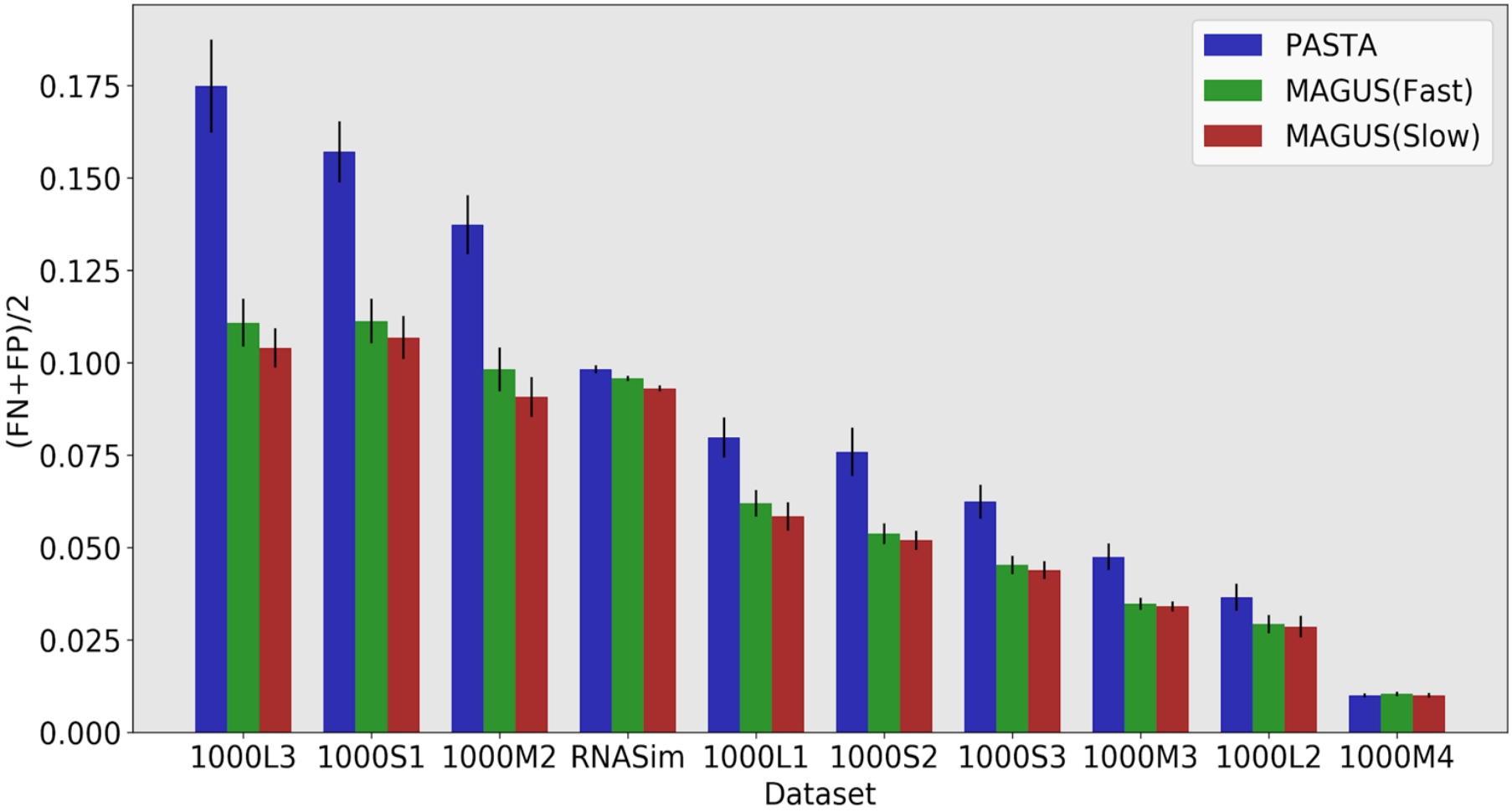
- Decomposition: the same technique (delete centroid edges)
- Subset alignments: the same (all computed MAFFT-linsi alignments)
- Merging:
  - SATé-II uses a guide tree to merge the subset alignments up the tree
  - PASTA aligns all "adjacent pairs" of alignments, and then finishes with transitivity
  - MAGUS aligns all subset alignments *at once* (using a complex pipeline involving Markov Clustering)

The Graph Clustering Merger (GCM) in MAGUS.

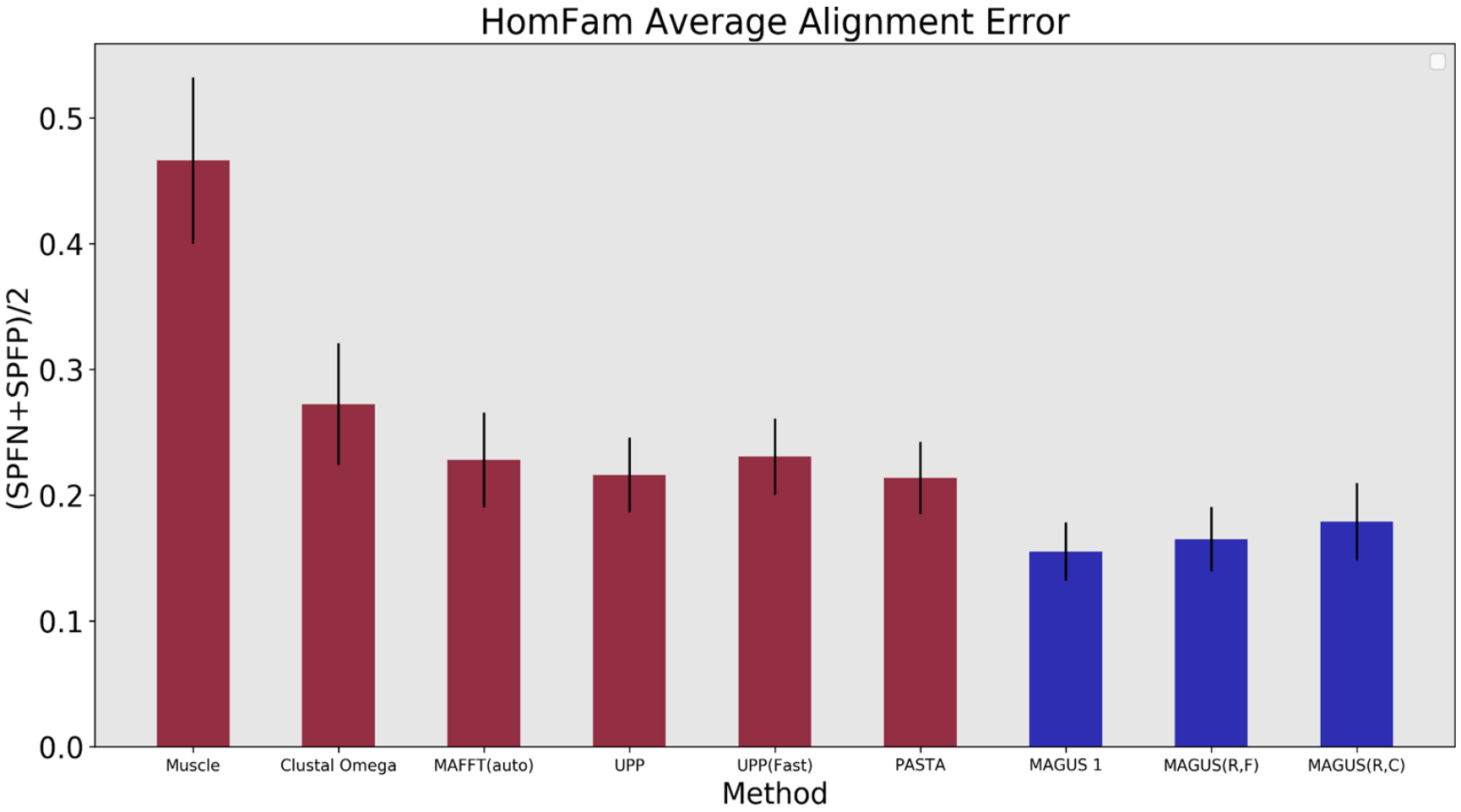


Bioinformatics, Volume 37, Issue 12, June 2021, Pages 1666–1672, https://doi.org/10.1093/bioinformatics/btaa992

MAGUS: More Accurate Alignments than PASTA



MAGUS: excellent on protein benchmarks too



Summary for Divide-and-Conquer

- Can be used with any base MSA method (we showed results with MAFFT-lins)
- Iteration can help
- Merging alignments “all at once” promising; related to John Kececioglu’s “Maximum Weight Trace” problem

Part 2: Adding Sequences into MSA

- Input: MSA on set S of sequences, and additional sequences S’
- Output: Extension of MSA to include S’

Application:

- Growing large alignment as new sequences are found
- MSA on datasets with sequence length heterogeneity

1kp: Thousand Transcriptome Project

G. Ka-Shu Wong
U Alberta

J. Leebens-Mack
U Georgia

N. Wickett
Northwestern

N. Matasci
iPlant

T. Warnow,
UIUC

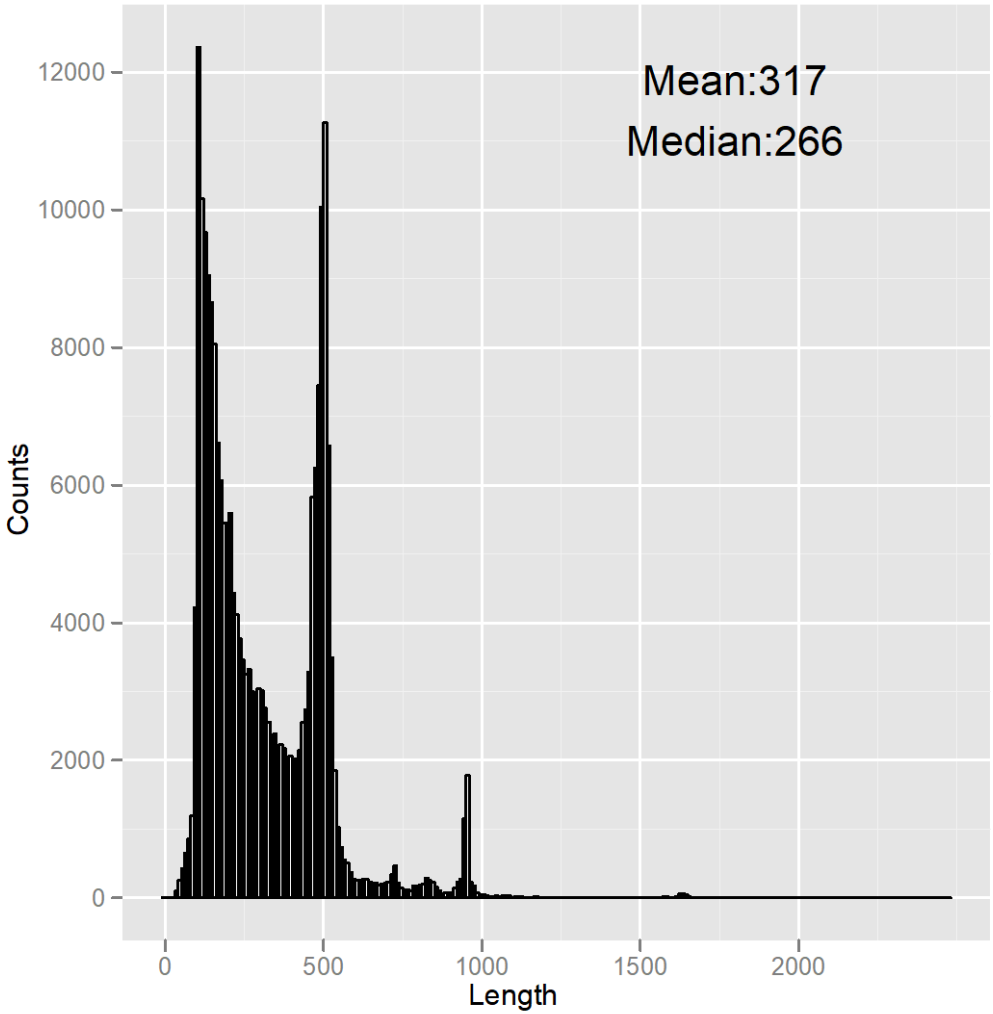
S. Mirarab,
UT-Austin

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)

Challenge:
Alignment of datasets with > 100,000 sequences with many very short sequences



1KP dataset: more than 100,000 p450 amino-acid sequences, many fragmentary

All standard multiple sequence alignment methods we tested performed poorly on datasets with fragments.

Solution: Two-phase approach

- Phase 1: Select a collection of “full-length” sequences, and compute a “backbone” alignment on them.
- Phase 2: Add the remaining sequences into the backbone alignment.

Note: Each stage matters!

- Depends on which sequences are in the backbone, and how the backbone alignment is computed (but can use expensive methods)
- Depends on how the remaining sequences are added to the backbone (can use “local alignment” techniques)

UPP

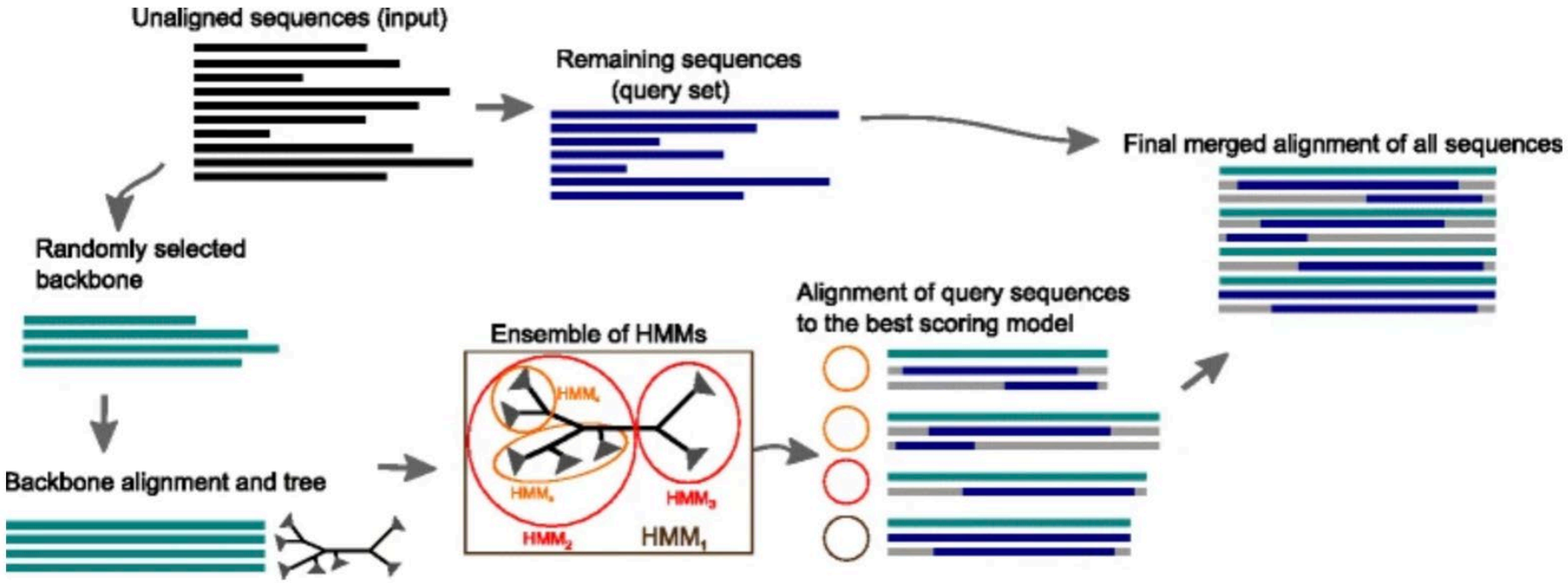
UPP = “Ultra-large multiple sequence alignment using Phylogeny-aware Profiles”

Nguyen, Mirarab, and Warnow. Genome Biology, 2015

Purpose: highly accurate large-scale multiple sequence alignments, even in the presence of fragmentary sequences.

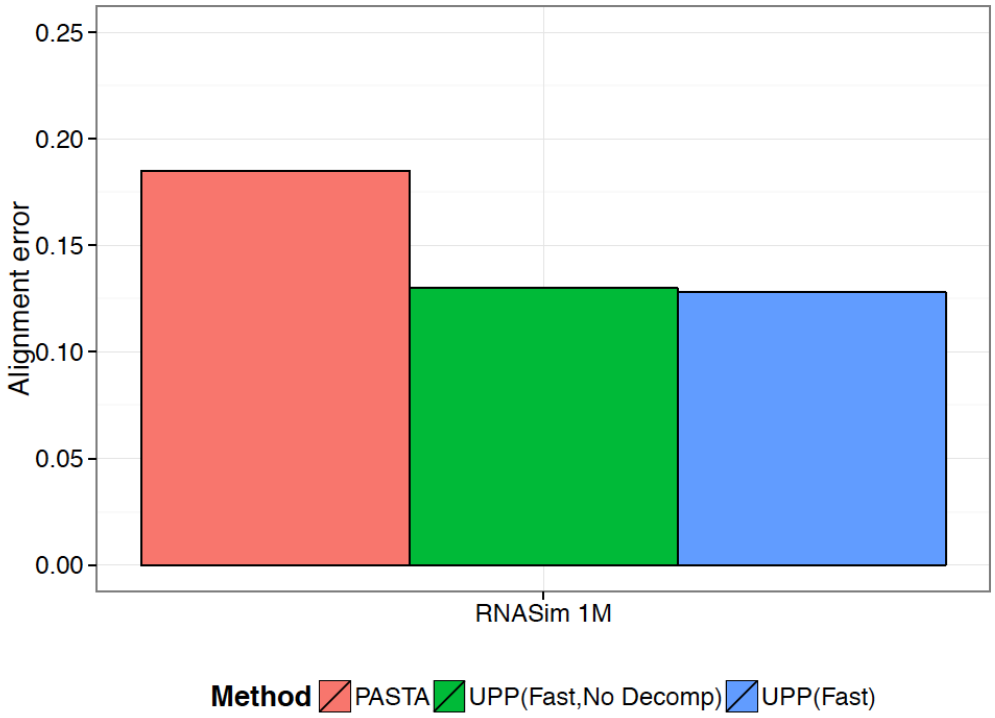
Uses an ensemble of HMMs

UPP (Nguyen et al. 2015)



The Ensemble of Hidden Markov Models is a “model” for the backbone alignment. The HMMs are built on subset alignments, may not be clades in the backbone tree.

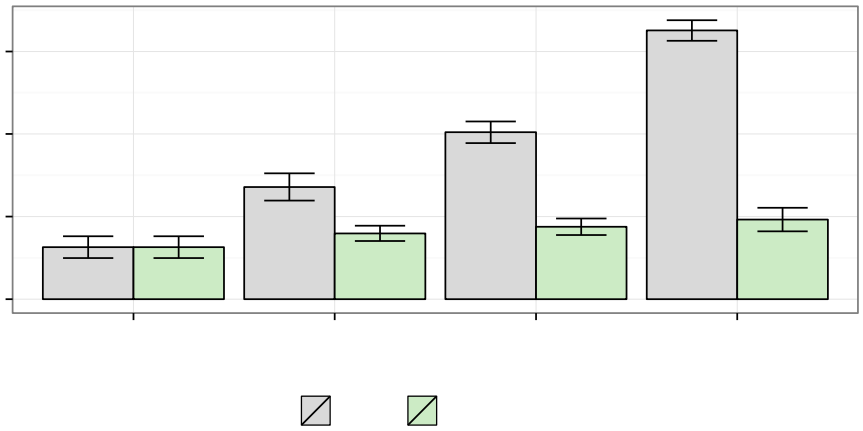
RNASim Million Sequences: alignment error



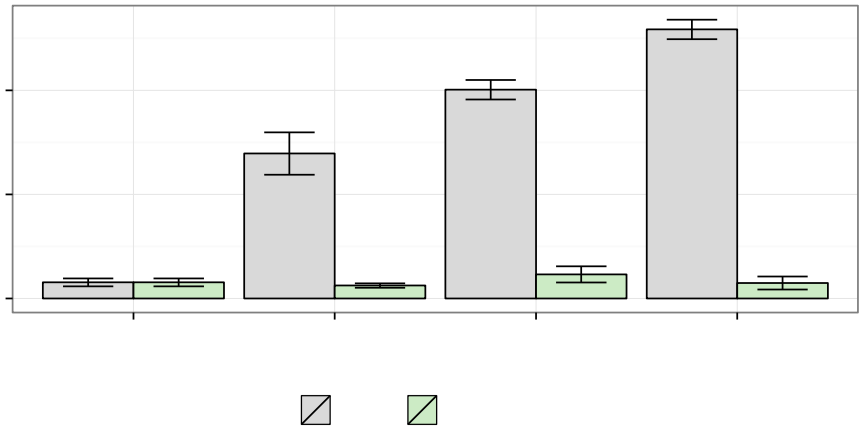
Notes:

- We show alignment error using average of SP-FN and SP-FP.
- UPP variants have better alignment scores than PASTA.
- No other methods tested could complete on these data

UPP vs. PASTA: impact of fragmentation



(a) Average alignment error



(b) Average tree error

Under high rates of evolution, PASTA is badly impacted by fragmentary sequences (the same is true for other methods).

Under low rates of evolution, PASTA can still be highly accurate (data not shown).

UPP continues to have good accuracy even on datasets with many fragments under all rates of evolution.

Performance on fragmentary datasets of the 1000M2 model condition

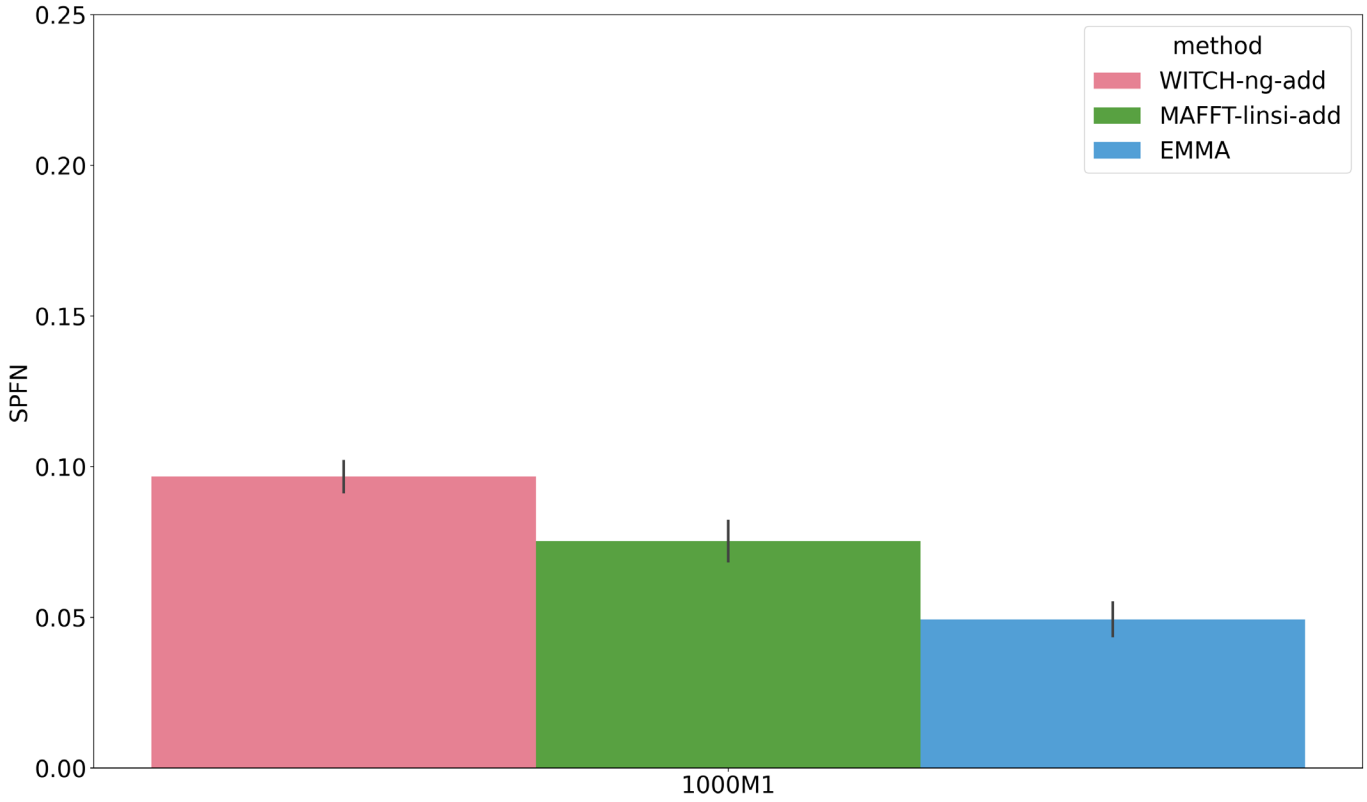
Other two-phase methods

These methods start the same as UPP (extract backbone alignment, build ensemble of HMMs on backbone), but then do things differently to add the query sequences to the backbone

- **WITCH** (Chengze Shen et al, J. Comp Biol 2022.) and WITCH-ng (Baqiao Liu and T. Warnow, Bioinformatics Advances 2022.): weights the HMMs, computes extended alignment for each HMM, merges the extended alignments using "consensus alignment" technique
- HMMerge (Minhyuk Park and T. Warnow, Bioinformatics Advances.): weights the HMMs, combines them into a new Hidden Markov Model (not profile HMM), and uses that new HMM to add query sequences
- EMMA (Shen et al., Algorithms for Molecular Biology 2023): Adds sequences into backbone alignment using MAFFT-linsi-add within a divide-and-conquer framework

All are more accurate than UPP

Comparison of EMMA-add, WITCH-ng-add, and MAFFT-linsi-add: The benefit of ***not*** using HMMs to align query sequences



SPFN of different methods (fraction of missing true pairwise homologies)

Dataset: 1000M1 with a high rate of evolution; all sequences are full-length
Backbone: 250 randomly selected sequences from full set.

Part III: Statistical Alignment

- The previous methods computed alignments using a variety of techniques
- Statistical alignment estimate an alignment with respect to a statistical model of sequence evolution that directly addresses insertions and deletions (indels)
- Bali-Phy (Redelings and Suchard) is a Bayesian method that co-estimates the alignment and the tree, and is the leading such method. However, it is very computationally intensive.

But: BAli-Phy is limited to small datasets

From www.bali-phy.org/README.html, **5.2.1. Too many taxa?**

"BAli-Phy is quite CPU intensive, and so we recommend using 50 or fewer taxa in order to limit the time required to accumulate enough MCMC samples. (Despite this recommendation, data sets with more than 100 taxa have occasionally been known to converge.) We recommend initially pruning as many taxa as possible from your data set, then adding some back if the MCMC is not too slow."

BAli-Phy

- Uses MCMC to sample MSAs and trees from the posterior
- Recommendation is to run Bali-Phy until it seems to have converged
- This can require months on moderate-sized datasets, infeasible for use on datasets with hundreds of sequences
- Question: Can we use it within divide-and-conquer (e.g., PASTA) or two-phase methods (e.g., UPP)?

The Author(s) *BMC Genomics* 2016, **17**(Suppl 10):764
DOI 10.1186/s12864-016-3101-8

BMC Genomics

RESEARCH Open Access

Scaling statistical multiple sequence alignment to large datasets

Michael Nute[1] and Tandy Warnow[2,3,4,5*]

From 14th Annual Research in Computational Molecular Biology (RECOMB) Comparative Genomics Satellite Workshop
Montreal, Canada. 11-14 October 2016

Objective: Scale BALi-Phy to large datasets by (a) using BAli-Phy as the base method within PASTA or (b) using PASTA(BAli-Phy) as the backbone within UPP.

PASTA-default vs PASTA(BAli-Phy): subset size 100

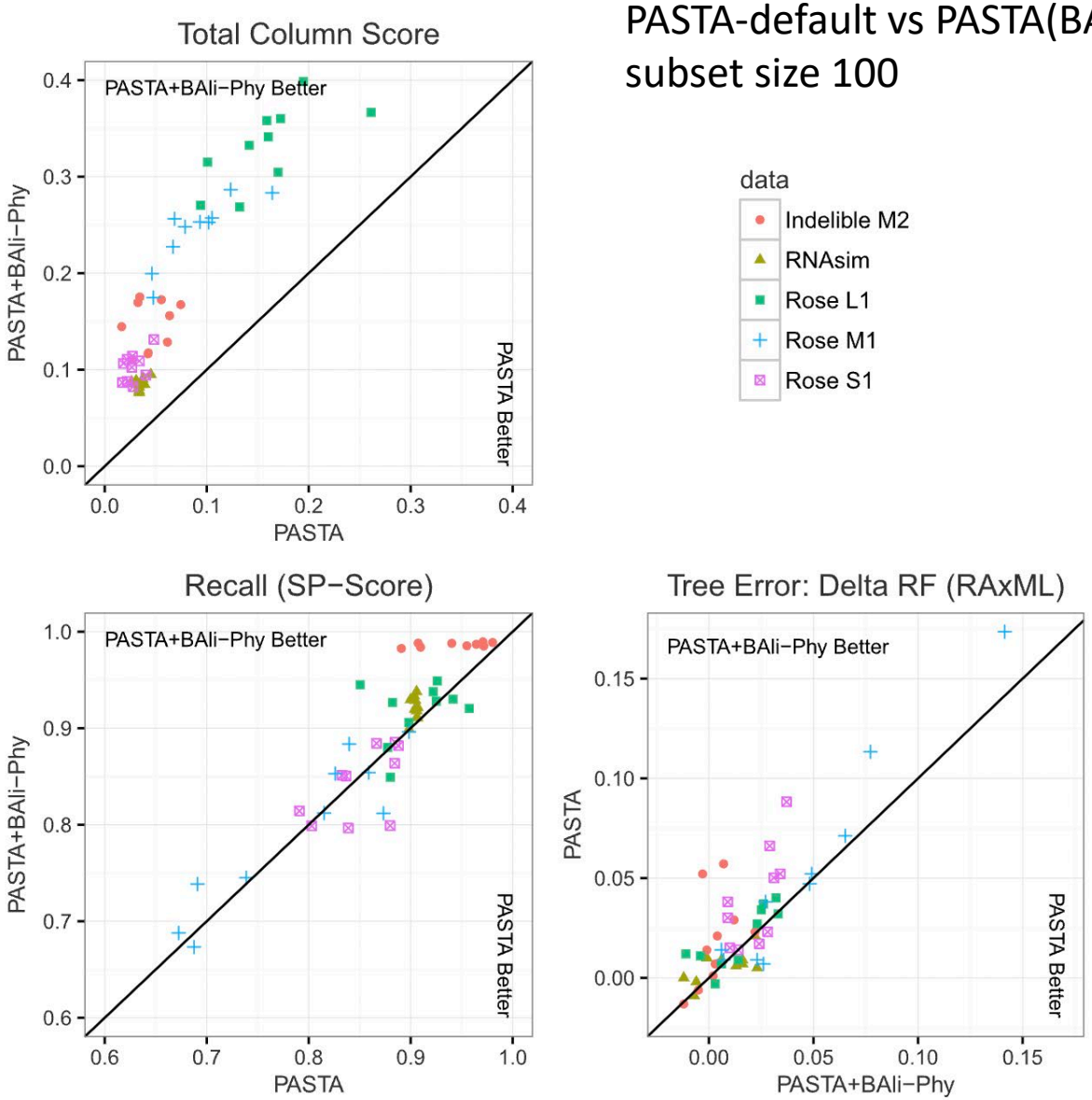


Fig. 1 Results on 1000-sequence datasets, comparing default PASTA and PASTA+BAliPhy. Each point represents one replicate. PASTA denotes the alignment from PASTA under default settings (referred to as "PASTA(default)" in the text), and PASTA+BAli-Phy denotes the alignment after an

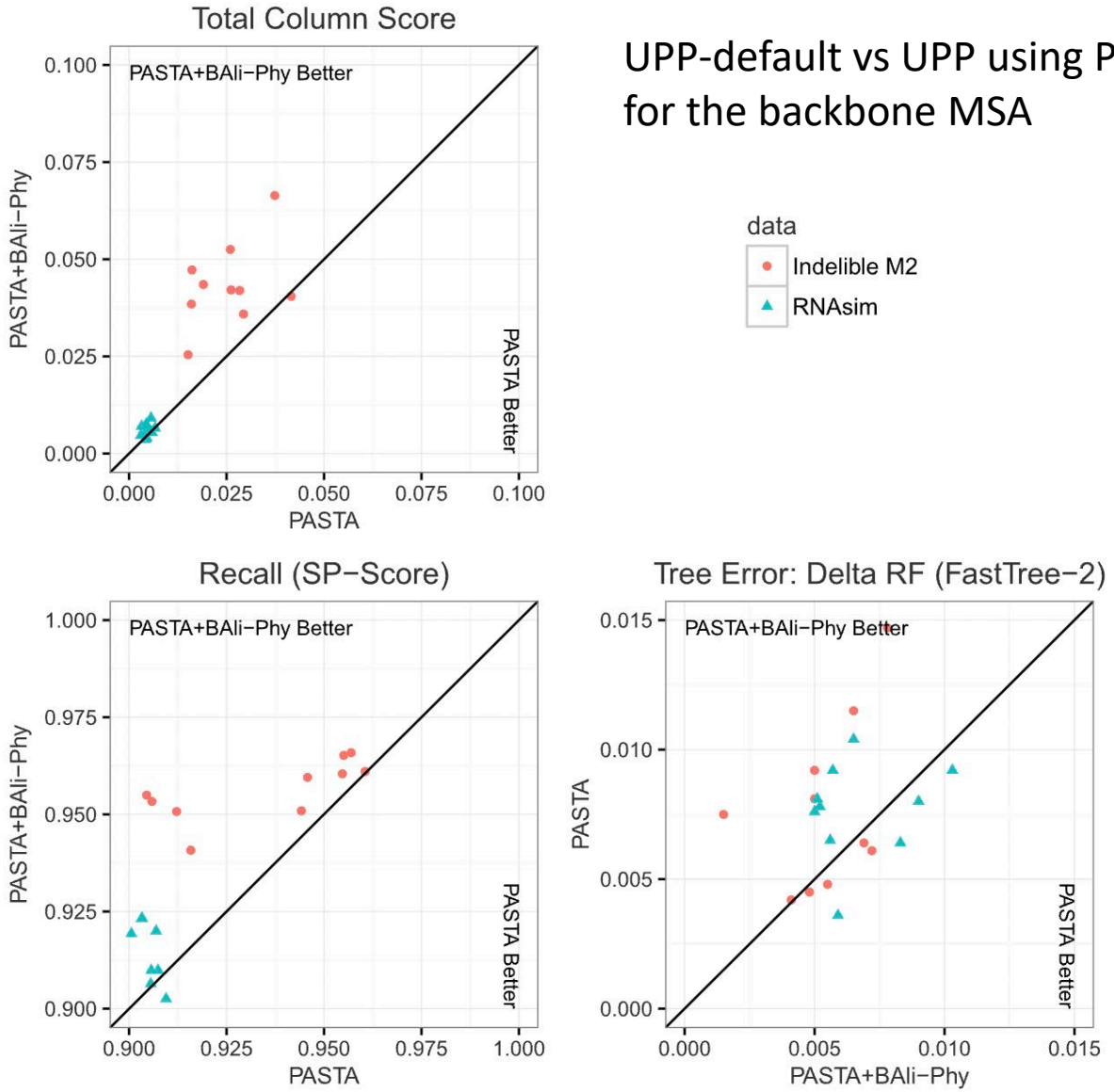
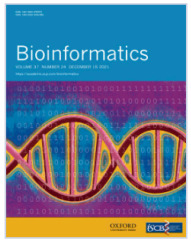


Fig. 2 Results on 10,000 sequences. Using UPP on two different backbones: one computed using default PASTA and the other computed using PASTA+BAliPhy (i.e., one iteration of PASTA using BAli-Phy as the subset aligner after default PASTA completes). Each point represents one replicate.

Observations

- Using BAli-Phy within PASTA (and subsequently within UPP) improves scalability.



Volume 37, Issue 24

JOURNAL ARTICLE

Accurate large-scale phylogeny-aware alignment using BAli-Phy FREE

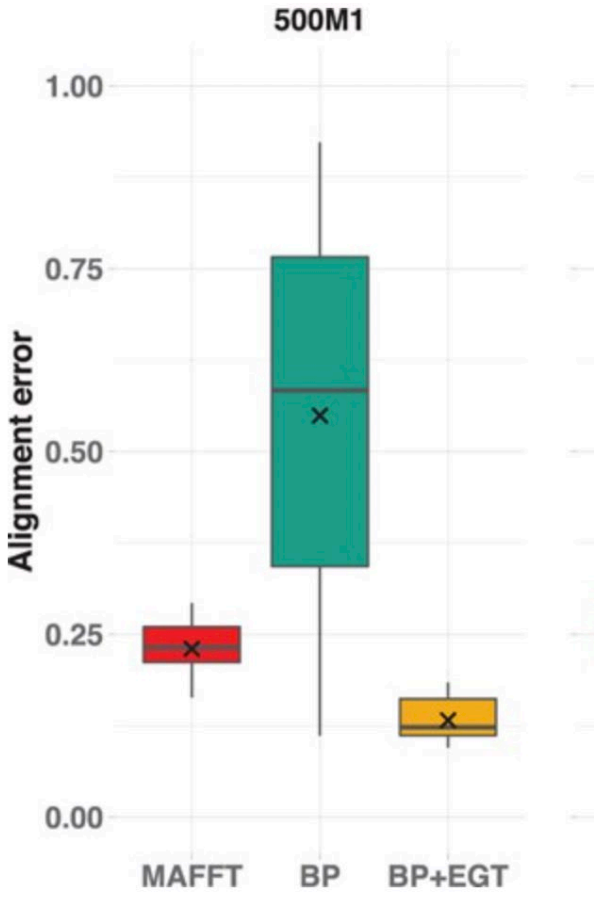
Maya Gupta , Paul Zaharias , Tandy Warnow ✉

Bioinformatics, Volume 37, Issue 24, December 2021, Pages 4677–4683,
https://doi.org/10.1093/bioinformatics/btab555
Published: 28 July 2021 **Article history** ▾

BAli-Phy co-estimates the MSA and tree. Suppose we estimate the tree and then apply BAli-Phy on this fixed tree.

1. What is the impact on alignment accuracy of using an estimated guide tree when using BAli-Phy?
2. What happens if we do not require that Bali-Phy converges?

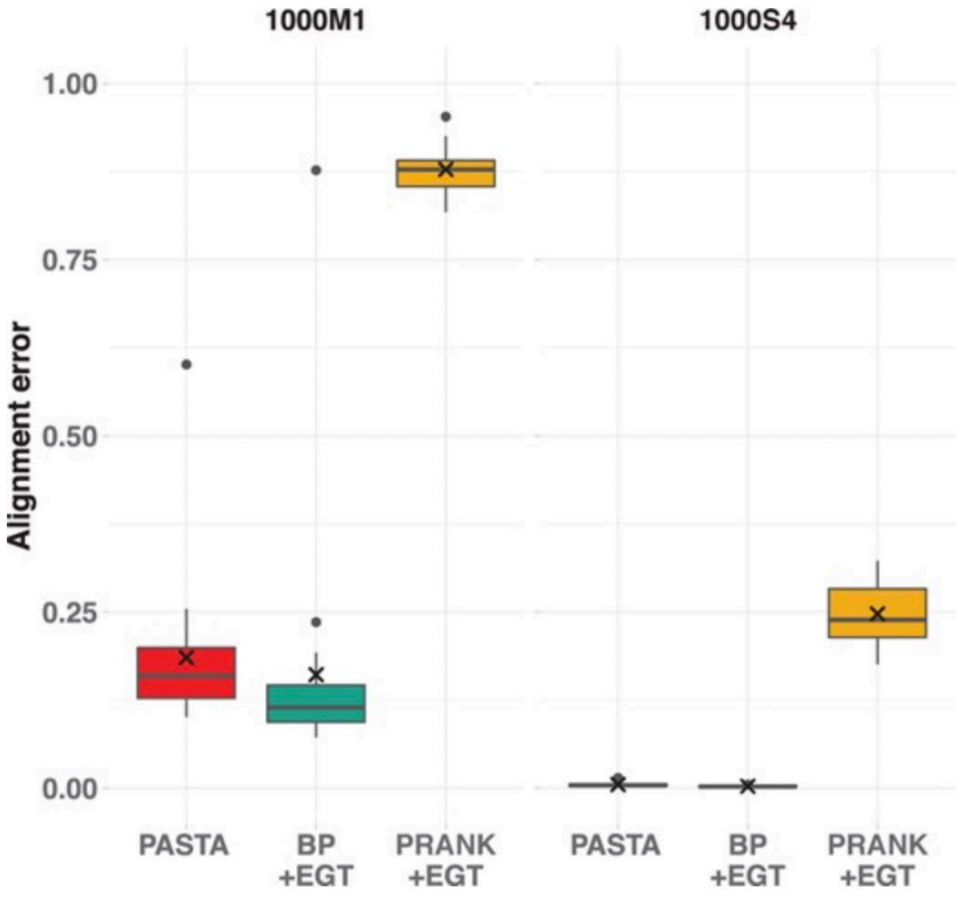
BP+EGT is best!



BP: Bali-Phy
BP+EGT: Bali-Phy with an estimated guide tree

MAFFT: MAFFT-lins-i (best way of running MAFFT)

BP+EGT is best!



PRANK+EGT: PRANK with an estimated guide tree

CONVERGENCE?

- All analyses were far from converging, according to ESS values, even when given 500 hours!
  - The ESS values remained low even for the 500-hour analyses, for both the RNASim1 and 1000M1 (high rate of evolution) datasets, and were only acceptable on the 1000M3 dataset (low rate of evolution).
  - The ESS values for these 4-hour BAli-Phy analyses were very low, clearly indicating that BAli-Phy is far from converging.

Comments

- Bali-Phy is computationally intensive (especially if you want to get it to converge), but these “tricks” may make it feasible:
  - On moderate-sized datasets,
    - Compute and then use an estimated guide tree
    - Don’t worry too much about convergence
  - On very large datasets (1000+ sequences)
    - Use within a divide-and-conquer method (e.g., PASTA, MAGUS) so that it is only run on smallish datasets
    - Use only for backbone within a two-phase method (e.g., UPP or WITCH)

Summary

- MSA is challenging, but algorithmic techniques can improve accuracy and scalability:
  - Dataset size can be addressed using good divide-and-conquer approaches.
  - Heterogeneity in sequence length can be addressed using “local alignment” approaches, such as profile HMMs, with ensembles of profile HMMs providing improved accuracy.

Algorithmic challenges

- How can we assess alignment uncertainty and use it in downstream analyses?
- Can we use a set of MSAs to advantage, instead of a single MSA? For example, can we develop effective and efficient “ensemble” methods?
- What are the best ways to merge disjoint alignments?
- How can we efficiently perform statistical alignment?

Some Recommendations

- For datasets with at most 1000 sequences and low sequence length heterogeneity:
  - MAFFT (especially –l-insi or –g-insi)
  - Promals, Contralign, and other methods for proteins
  - BAli-Phy (statistical alignment)
  - T-Coffee (combines different MSA methods)

- Large number of sequences:
  - w/o sequence length heterogeneity: MAGUS, TWILIGHT, FAMSA, and others (e.g., Clustal-Omega)
  - with sequence length heterogeneity:
    - UPP, WITCH (-ng), EMMA (recent improvement of UPP)
- For genome-scale datasets: different and harder problem

Summary

- Multiple sequence alignment (MSA) has large downstream consequences in bioinformatics analyses.
- MSA is far from solved – esp. (but not only) on large datasets with high rates of evolution, sequence length heterogeneity, and streaming data.
- New techniques show promise
- Not discussed: multiple whole genome alignment, MSA with rearrangements

Acknowledgments

PASTA and UPP: Siavash Mirarab and Nam-phuong Nguyen
MAGUS: Vlad Smirnov
WITCH: Chengze Shen and Minhyuk Park
EMMA: Chengze Shen and Baqiao Liu

NSF grant: 2006069
Grainger Foundation (at UIUC)
UIUC campus cluster

PASTA and UPP available on github at
https://github.com/smirarab/

MAGUS: at
https://github.com/vlasmirnov/MAGUS

WITCH: at
https://github.com/c5shen/WITCH

EMMA at
https://github.com/c5shen/EMMA

Papers available at
http://tandy.cs.illinois.edu/papers.html