Homework 01: Number of Possible Alignments

Let f(N,M) be equal to the number of possible alignments of sequences S and T, where |S| = N, and |T| = M. Then we have:

$$f(i,0) = f(0,j) = 1 \qquad \text{,for } 0 \le i \le N \text{ and } 0 \le j \le M$$
$$f(i,j) = f(i-1,j) + f(i,j-1) + f(i-1,j-1)$$
$$\text{,for } 1 \le i \le N \text{ and } 1 \le j \le M$$

		T1	T2	T3	T4	T5	T6	T7	T8	T9
	1	1	1	1	1	1	1	1	1	1
S1	1	3	5	7	9	11	13	15	17	19
S2	1	5	13	25	41	61	85	113	145	181
S3	1	7	25	63	129	231	377	575	833	1159
S4	1	9	41	129	321	681	1289	2241	3649	5641
S5	1	11	61	231	681	1683	3653	7183	13073	22363
S6	1	13	85	377	1289	3653	8989	19825	40081	75517
S7	1	15	113	575	2241	7183	19825	48639	108545	224143
S8	1	17	145	833	3649	13073	40081	108545	265729	598417
S9	1	19	181	1159	5641	22363	75517	224143	598417	1462563
S10	1	21	221	1561	8361	36365	134245	433905	1256465	3317445

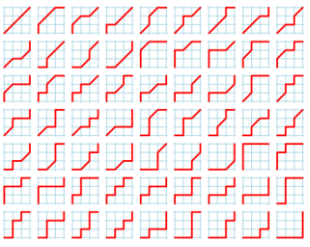
A. Torres, A. Cabada, J.J. Nieto, An Exact Formula for the Number of Alignments Between Two DNA Sequences. International Journal of Biomathematics, Vol. 09, No. 04, 1650053 (2016) Research Articles. (Also in DNA Sequence, 2003.)

Homework 01: Number of Possible Alignments

Delannoy Numbers D(N,M)

https://en.wikipedia.org/wiki/Delannoy_number

Definition: Delannoy number D describes the number of paths from the southwest corner (0, 0) of a rectangular grid to the northeast corner (m, n), using only single steps north, northeast, or east.



D(3,3)=63

F(N,M) = D(N+1,M+1)

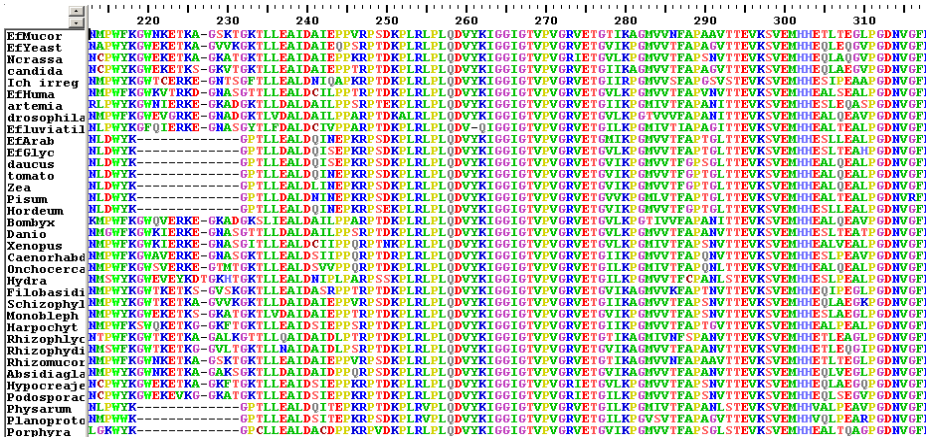
$$D(n,m) = \sum_{k=0}^{\min(m,n)} \binom{n}{k}\binom{m}{k} 2^k = \sum_{k=0}^{\min(n,m)} \binom{m+n-k}{m}\binom{m}{k}$$

HW 01: Speed Optimal Global Alignment $|S|=|T|=10^5$

Rank	Time	Language
1	19.02	.cpp
2	34.00	.cpp
3	63.82	.cpp
4	70.16	.cpp
5	83.00	.cpp
6	86.00	.java
7	115.00	.py
8	118.02	.cc
9	118.96	.cpp
10	119.05	.java
11	157.29	.java
12	280.20	.java
13	303.68	.cpp
14	2700.00	.py
15	3000.00	.cpp
16	4129.23	.py
17	13975.00	.py
18	18761.96	.py
19	19460.27	.py

Length 10000

Multiple Sequence Alignment



Multiple Sequence Alignment

Shows multiple similarities:

- Common structure of protein product
- Common function
- Common evolutionary process

- Protein Structure Prediction
- Protein Family Identification
- Protein Characterization: signatures of protein families
- Phylogeny estimation

5

Many Different Alignment Methods

- Aligning a String to a Profile (HMMs)
- Iterative Pair-wise Alignment
- Progressive Multiple Alignment
  - Feng-Doolittle (1987)[2]
  - CLUSTALW, CLUSTALX, and other versions
  - State of the Art Parallel MSA (2018) [15]
    - Coffee, MAFT, MSAProbs, M2Align
- PAGAN Phylogeny Aware MSA (2015) [13]
- Etc.

6

Example: Signature Profiles

Helicases

- A protein to unwound DNA for further read for duplication, transcription, recombination or repair.
- Werner's syndrome an aging disease is believed to be due to a gene WRN that codes for a helicase protein.

A Signature Profile for Helicases

- Conserved sequence signatures or motifs
- Some of these motifs are unique identifiers for helicases
- Maybe functional units

Multiple Alignment Profile

a b a
a b –
– b a
c a –

	Col 1	Col 2	Col 3
a	50%	25%	50%
b	0 %	75%	0 %
c	25%	0 %	0 %
-	25%	0 %	50%

Multiple Alignment Profile:

- Character frequencies given per column
- $p_i(a)$ is the fraction of a's in column i
- $p(a)$ is the fraction of a's overall
- Log likelihood ratio $\log(p_i(a)/p(a))$ can be used.

Aligning a String to a Profile

Definition 4.16 For an alignment S' of length l, a *profile* is an $l \times |\Sigma \cup \{-\}|$ matrix, whose columns are probability vectors denoting the frequencies of each symbol in the corresponding alignment column.

Average Profile Score (total) = -5
Average Profile Score (no gaps) = -6

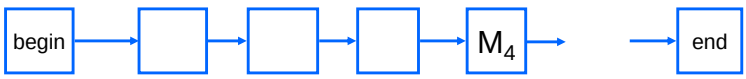
Cons	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
E	-2	0	-9	1	5	-10	-8	-5	-8	0	1	-7	-3	-1	0	-5	2	-2
D	-2	0	-9	1	1	-10	-2	-4	-9	0	-1	-10	-6	-1	0	-1	0	-3
E	-2	0	-15	3	11	-16	-10	-5	-14	0	1	-14	-9	-2	0	-2	3	-3
G	-2	-2	-9	-1	0	-8	-1	-3	-9	0	-4	-9	-6	-1	0	-3	-2	-6
E	-9	7	-26	13	26	-24	-15	-4	-25	0	7	-24	-15	0	0	-4	9	-3
E	-8	3	-24	7	23	-20	-12	-1	-24	0	3	-22	-13	0	0	-10	12	0
E	-7	-6	-17	-2	8	-8	-15	-7	-6	0	-5	-10	-6	-9	0	-12	-1	-11
Y	-15	-18	-15	-17	-11	18	-20	6	-10	0	-11	-8	-6	-13	0	-20	-6	-12
V	2	-11	-10	-8	1	-12	-14	-11	2	0	-6	-4	-2	-11	0	-10	-4	-11
V	4	-32	-8	-29	-21	-11	-27	-27	25	0	-20	9	6	-29	0	-15	-20	-28
E	-16	16	-52	28	75	-43	-28	0	-48	0	15	-48	-30	0	0	-13	30	0
K	-2	-4	-14	-9	3	-20	-12	-5	-21	0	19	-15	-8	-2	0	-9	4	14
I	-9	-36	-10	-32	-31	-3	-41	-30	42	0	-30	21	10	-31	0	-28	-30	-32
L	-8	-28	-6	-25	-20	-3	-29	-20	17	0	-16	21	11	-21	0	-21	-15	-16
D	-4	13	-13	21	8	-22	-8	-9	-22	0	-4	-26	-17	0	0	-5	0	-11
H	-3	-2	-7	-4	0	-8	-8	4	-11	0	2	-10	-6	0	0	-7	0	1
R	-7	-8	-18	-12	1	-17	-16	-4	-18	0	13	-12	-5	-3	0	-13	5	18

consensus

Profile for Classical Chromo Domains [10].

9

Hidden Markov Models (HMM) profile alignment (no gaps)



Assume a given profile set:

```
12345678
VGAHAGEY
VTGNVDEV
VEADVAGH
VKSNDVAD
VYSTVETS
FNANIPKH
IAGNGAGV
```

=> Emission probability distribution function at state 4

No gaps
transition probabilities: 1
trivial alignment HMM to sequence

profile HMM $\mathcal{P}$ 'dedicated topology'

$$X = x_1 x_2 \ldots x_L$$

Let $e_i(b)$ be equal to the probability of observing symbol b at pos i, then:

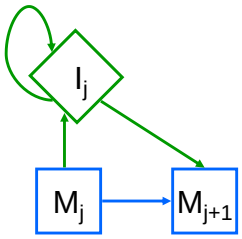
$$P(X|\mathcal{P}) = \prod_{i=1}^{L} e_i(x_i)$$

10

HMMs profile alignment with gaps

Given profile sequences:

```
VGA--HAGEY
VNA--NVDEV
VEA--DVAGH
VKG--NYDED
VYS--TYETS
FNA--NIPKH
IAGADNGAGV
123__45678
```



insert state

match states

Emission probability distribution based on:
- background probabilities: $e_i(b) = p(b)$
- or based on alignment (match)

affine model

$$t_{M_j I_j} \cdot t_{I_j M_{j+1}} \cdot t_{I_j I_j}^{h-1}$$

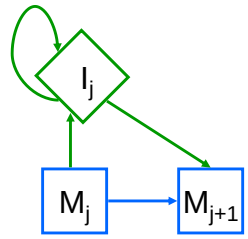
open gap extension

11

HMMs profile alignment with gaps and deletes

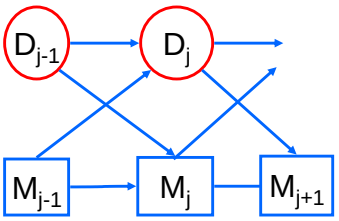
Given profile Sequences:

```
VGA--HAGEY
V----NVDEV
VEA--DVAGH
VKG-----D
VYS--TYETS
FNA--NIPKH
IAGADNGAGV
123__45678
```



insert state

match states



delete state (*silent*)

adapt Viterbi =>

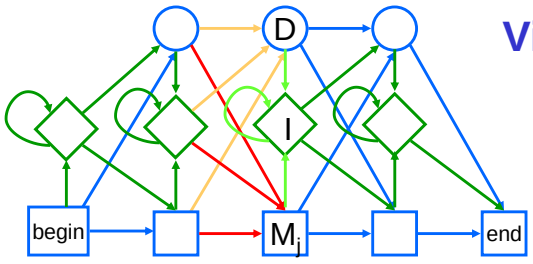
12

HMMs for profiles / multiple alignment

Deletion (D)

Insertion (I)

Match (M)



Viterbi

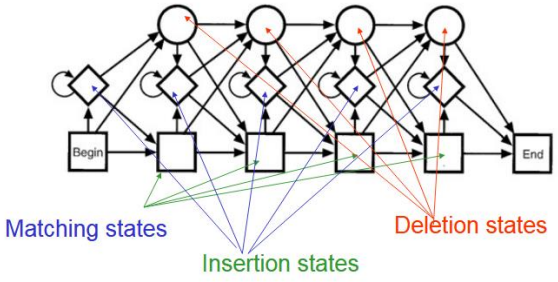
$$v_j^M(i) = e_{M_j}(x_i).\max_{Y=M,I,D} v_{j-1}^Y(i-1).t_{Y_{j-1}M_j}$$

$$v_j^I(i) = p(x_i).\max_{Y=M,I,D} v_j^Y(i-1).t_{Y_{j-1}I_j}$$ same level

$$v_j^D(i) = \max_{Y=M,I,D} v_{j-1}^Y(i).t_{Y_{j-1}D_j}$$ same position

13

Multiple Sequence Alignment: HMM known



Multiple Sequence Alignment Problem:

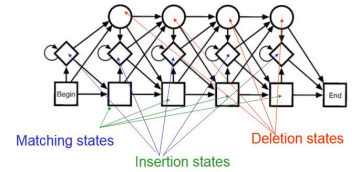
Given sequence $S^1,\ldots, S^n$, how can they be optimally aligned?

Assume a profile HMM *P* is known, then:

- Align each sequence S^i to the profile separately
- Accumulate the obtained alignments to a multiple alignment
- Hereby insert runs are not aligned. (Just left-justify insert regions.)

14

Multiple Sequence Alignment: HMM unknown



Multiple Sequence Alignment Problem:
Given sequence $S^1,\ldots, S^n$, how can they be optimally aligned?

Assume a profile HMM *P* is not known, then obtain an HMM profile P from $S^1,\ldots, S^n$ as follows:

- Choose a length L for the profile HMM and intialize the transition and emission probabilities.
- Train the HMM using Baum- Welch on all sequences $S^1,\ldots, S^n$.

Now obtain the multiple alignment using this HMM P as in the previous case:

- Align each sequence S^i to the profile separately
- Accumulate the obtained alignments to a multiple alignment
- Hereby insert runs are not aligned. (Just left-justify insert regions.)

15

Iterative Pair-wise Alignment

Algorithm

1. Align some pair
2. While (not done)
   (a) Pick an unaligned string which is ”near” some aligned one(s).
   (b) Align with the *profile* of the previously aligned group. Resulting new spaces are inserted into all strings in the group.

This approach uses pair-wise alignment scores to iteratively add one additional string to a growing multiple alignment.

1. We start by aligning the two strings whose edit distance is the minimum over all pairs of strings.
2. Then we iteratively consider the string with the smallest distance to any of the strings already in the multiple alignment.

16

Progressive Multiple Alignment

- Again a heuristic => not guaranteed to be optimal
- Progressive alignment of the sequences

Problems

- What are the initial sequences?
- What is the order in which the sequences are aligned?

17

Progressive Multiple Alignment

Sketch

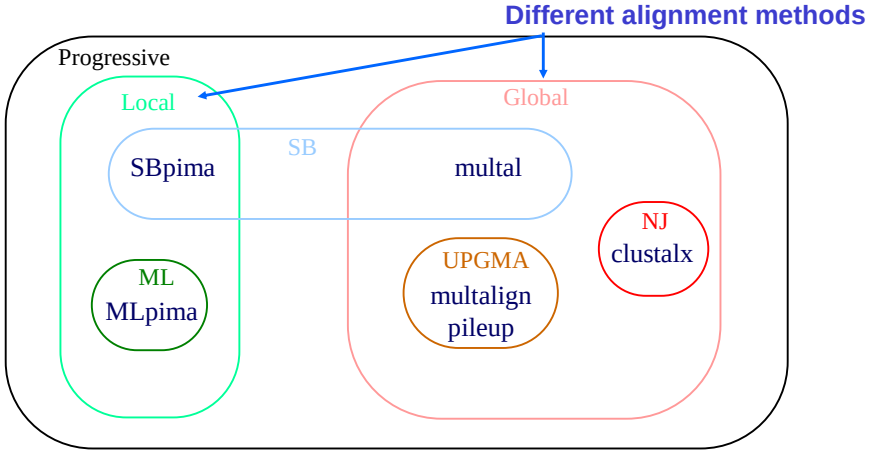
- Align all pairs of sequences
- Determine distance matrix
- Construct a guide tree from the distance matrix
- Progressive multiple alignment following the guide tree.

For example:

CLUSTAL, …, CLUSTALW, CLUSTALX, CLUSTALΩ, CLUSTAL2

18

Progressive multiple alignment methods



Different Guide Tree Construction Methods:

SB - Sequential Branching
UPGMA - Unweighted Pair Grouping Method
ML - Maximum Likelihood
NJ - Neighbor-Joining

From: O.Poch, Ecole Phylogénomique, Carry le Rouet 2006

Many many MSA methods: which one is the best?

Fast and inaccurate

- DIALIGN
  - Distinction of alignable vs non-alignable
  - Less accurate than ClustalW on some benchmarks
- MAFFT, MUSCLE
  - Faster, more accurate than CLUSTALW, but still accuracy trade-off

Slow but accurate

- T-COFFEE
  - High accuracy, uses heterogenous information
  - Computational and space intensive (can be a limiting factor)

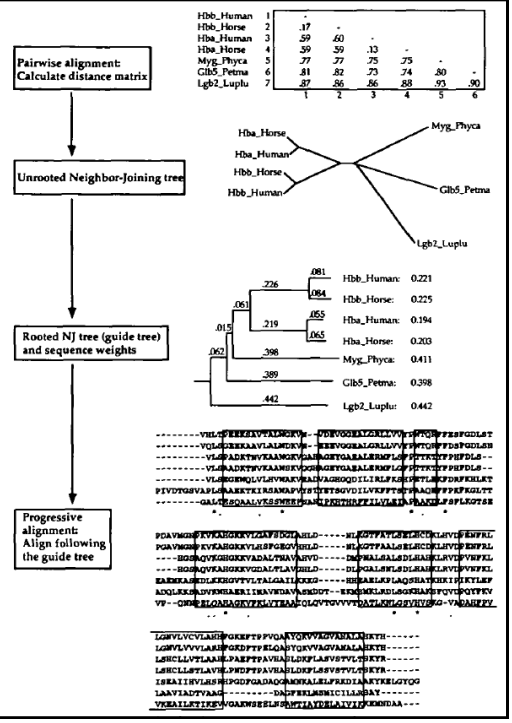
Slow and inaccurate

- ClustalW
  - Needs less memory
  - Is less accurate, and less scalable
  - New improved versions: ClustalΩ, Clustal2

20

ClustalW
Thompson et al. 1994

- Pairwise alignment scores
- Distance matrix
  scores => distances
- Clustering by Neighbor-Joining Tree (UPGMA, NJ-Tree)
- Guide Tree: UPGMA, NJ-Tree
- Progressive Alignment following the guide tree.



Hierarchical Clustering of Sequences

Unweighted Pair Group Method with Arithmetic Averages (UPGMA)

- Iteratively clustering the sequences
- Assume two clusters i and j with the shortest distance d_{ij}, then node u is formed between i and j.
- In general the distance between a cluster k and the new node u is calculated as:

$$d_{ku} = \frac{clustersize(i) \cdot d_{ki} + clustersize(j) \cdot d_{kj}}{clustersize(i) \cdot clustersize(j)}$$

(Sokal and Michener '58)

	a	b	c	d	e
a	0	17	21	31	23
b	17	0	30	34	21
c	21	30	0	28	39
d	31	34	28	0	43
e	23	21	39	43	0

Example: Unweighted Pair Group Method with Arithmetic Averages (UPMGA)

- Start with the distance matrix.
- Group the closest elements (a,b) => insert node u

	(a,b)	c	d	e
(a,b)	0	**25.5**	**32.5**	**22**
c	**25.5**	0	*28*	*39*
d	**32.5**	*28*	0	*43*
e	**22**	*39*	*43*	0

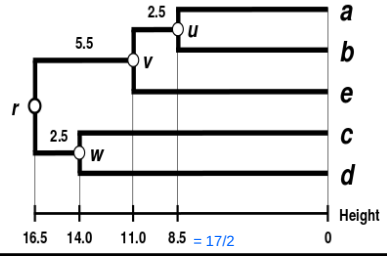
- Update the distance matrix.
- Calculate average distance of the nodes to u = (a,b)
- d(c,(a,b)) = (21+30)/2 = 25.5
- Now d((a,b),e) is smallest => insert new node v

	((a,b),e)	c	d
((a,b),e)	0	**30**	**36**
c	**30**	0	*28*
d	**36**	*28*	0

- Update matrix, take into account the sizes of subtrees.
- d(c,v) = (2*25.5+1*39)/(2+1) = 90/3 = 30
- Now d(c,d) is smallest => insert node w

	((a,b),e)	(c,d)
((a,b),e)	0	**33**
(c,d)	**33**	0

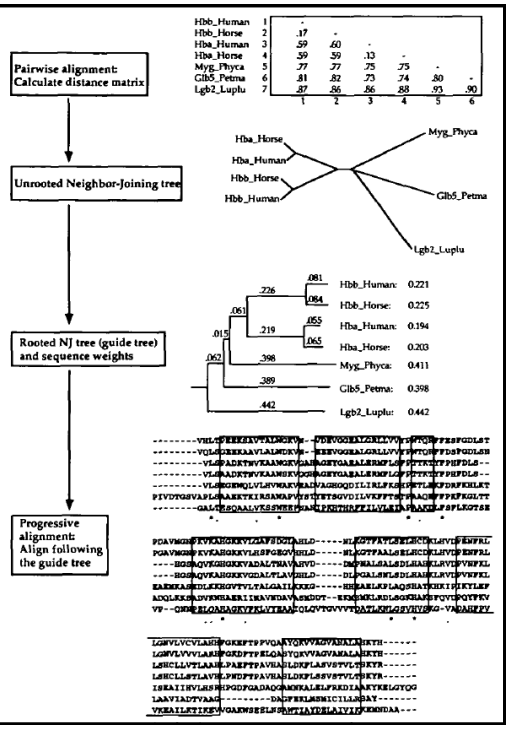
- Update matrix



(Sokal and Michener '58; Wikipedia)

ClustalW
Thompson et al. 1994

- Pairwise alignment scores
- Distance matrix
  scores => distances
- Clustering by Neighbor-Joining Tree (NJ-Tree)
- Guide Tree: NJ-Tree
- Progressive Alignment following the guide tree.

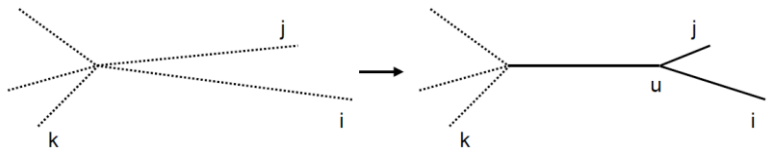


Neighbor Joining Algorithm (NJ-Algorithm)

- Iteratively building a NJ-Tree
- Let clusters *i* and *j* with the shortest modified distance M_{ij}, then a new node *u* is formed between them.

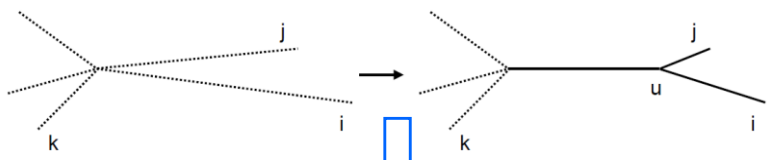
$M_{ij} = d_{ij} - \frac{r_i + r_j}{N-2}$, where $r_i = \sum_k d_{ik}$ (divergence of cluster *i*)

- The distance between a cluster *k* and the new node *u* is calculated as:



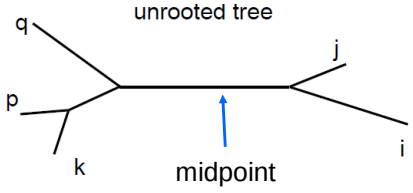
$$d_{ku} = \frac{d_{ik} + d_{jk} - d_{ij}}{2} \qquad d_{uj} = \frac{d_{ij} + r_i - r_j}{2}$$

Neighbor Joining Algorithm (NJ-Algorithm)



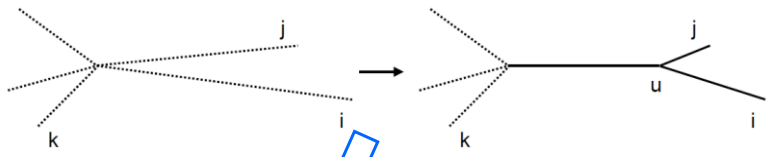
$$d_{ku} = \frac{d_{ik} + d_{jk} - d_{ij}}{2} \qquad d_{uj} = \frac{d_{ij} + r_i - r_j}{2}$$

unrooted tree



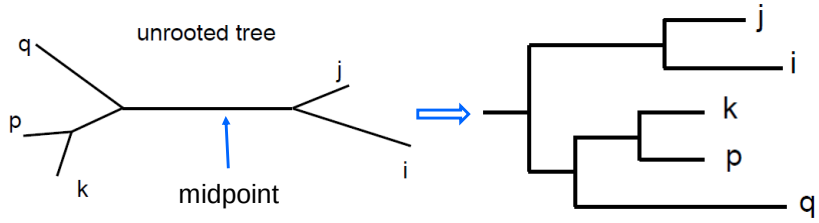
A midpoint can be defined to give equal average branch lengths on either sides => rooted NJ-tree

Neighbor Joining Algorithm (NJ-Algorithm)



$$d_{ku} = \frac{d_{ik}+d_{jk}-d_{ij}}{2}$$

$$d_{uj} = \frac{d_{ij}+r_i-r_j}{2}$$



rooted NJ-tree

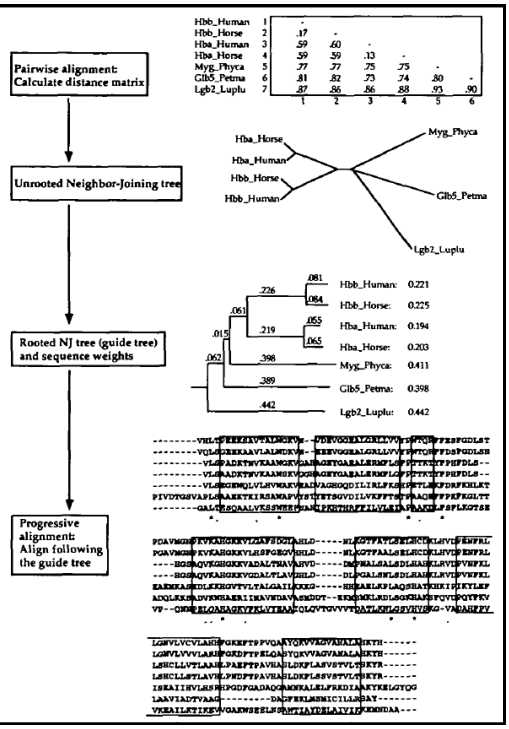
A midpoint can be defined to give equal average branch lengths on either sides => rooted NJ-tree

Homework 02

27

ClustalW
Thompson et al. 1994

- Pairwise alignment scores
- Distance matrix
  scores => distances
- Clustering by Neighbor-Joining Tree (NJ-Tree)
- Guide Tree: NJ-Tree
- Progressive Alignment following the guide tree.



Clustal Omega: handling of large sequence datasets

- Progressive multiple alignment using all pairwise distances is not possible for large numbers of sequences.
- Clustering of sequences by calculation of distances only to selected seed sequences makes the procedure faster.
- Clustal Omega algorithm can use both distances from pairwise alignments (socalled Kimura distances) and k-mer distances (*k-mers: next slide*)
  - constructs the profiles of subclusters, which define probabilities of substitutions, deletions and insertions.
  - profiles can be aligned to each other (also HMMs can be used here)
- Similar approaches are used in some other algorithms.

29

k-mers / *k*-tuples / *k*-words / *n*-mers / *n*-grams

A G C C G C T T A G T C A A C T

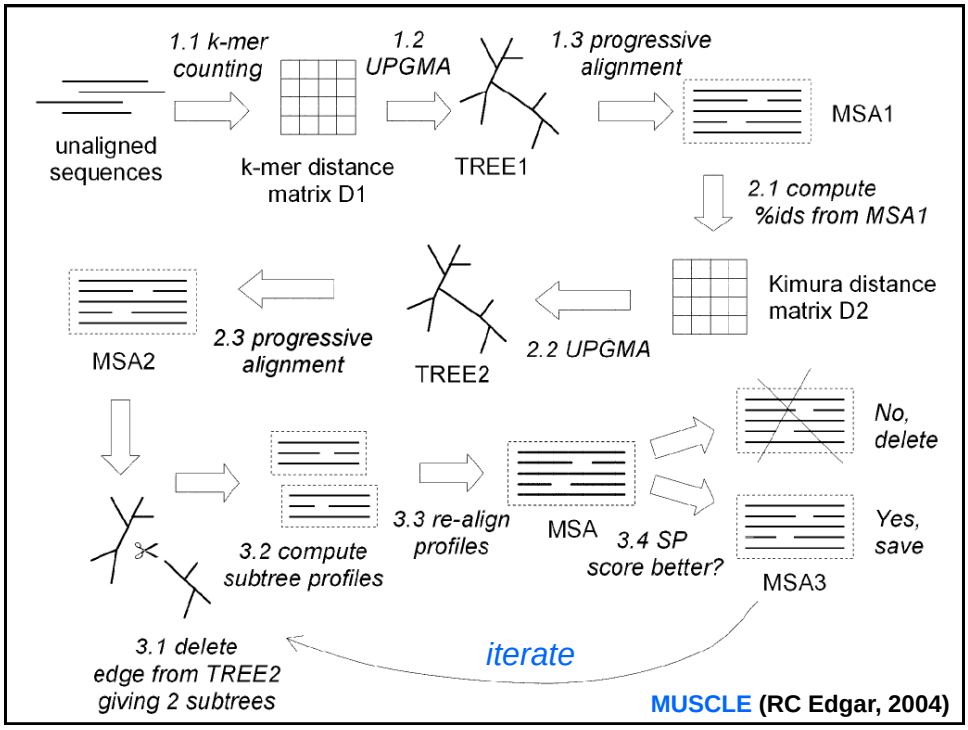
Here, $k = 6$

AGCCGC
GCCGCT
CCGCTT
(…)
TCAACT

For a sequence of length S, there are $S - k + 1$ k-mers, not all of which are necessarily unique

There's also a parallel world of ***patterns***

Höhl, Rigoutsos & Ragan, *Evol Bioinf* 2:357-373 (2006)



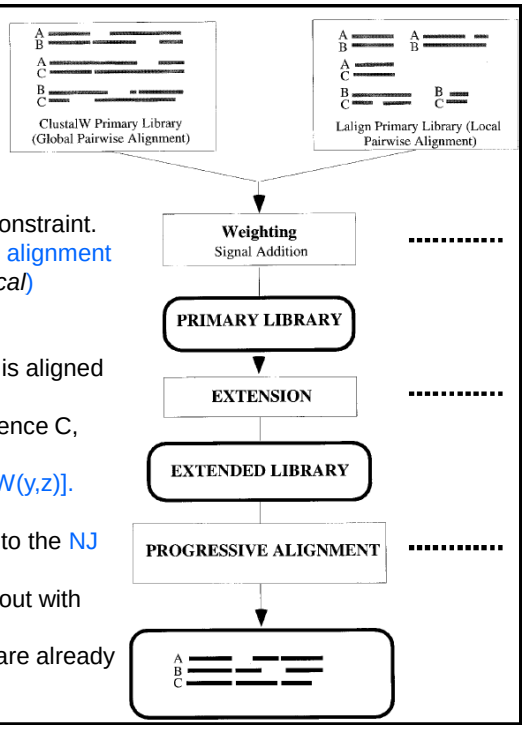
T-Coffee (Notredame et al. 2004)
(very rough sketch)

- Lalign local: top 10 scoring
- non-intersecting local alignments

- Monomers *x(A)* aligned to *y(B):* constraint.
  W(constraint) = % similarity in the alignment
- W(x,y) = W(x,y,*global*) + W(x,y,*local*)
  W=0 if x and y are not aligned

- If x(A) is aligned to z(C) and y(B) is aligned to z(C) as well:
- x and y are aligned through sequence C, thus additional constraint weight:
  W(x,y) = W(x,y) + min [W(x,z) + W(y,z)].

- Progressive alignment according to the NJ tree from pairwise alignments.
- Dynamic programming is carried out with account of weights.
- No gap penalties (indirectly they are already taken into account).



Selected slides from Mark Ragan's presentation:

Phylogenetics without multiple sequence alignment

Mark Ragan
Institute for Molecular Bioscience
and
School of Information Technology & Electrical Engineering
The University of Queensland, Brisbane, Australia

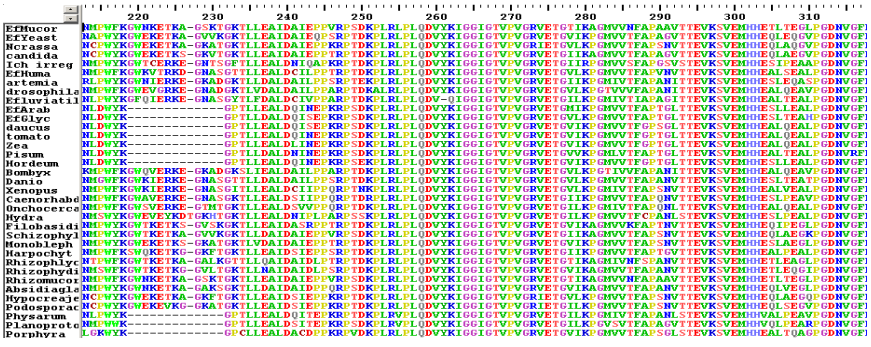
IPAM Workshop on Multiple Sequence Alignment
UCLA, 13 January 2015

See also: http://www.ipam.ucla.edu/programs/workshops/multiple-sequence-alignment/?tab=schedule

THE UNIVERSITY OF QUEENSLAND AUSTRALIA | IMB

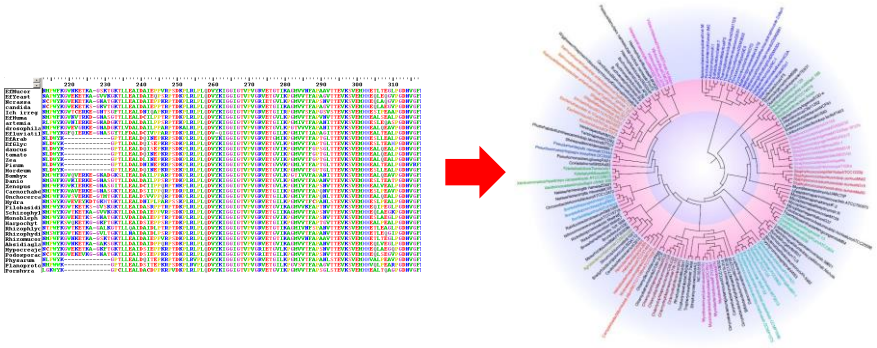
An MSA* gives us (visual) access to…

- Patterns within columns
- Local adjacency relationships within rows (across columns): context
- Global architecture



** MSA = multiple sequence alignment*

Here, we focus on MSA as input into a tree-inference program



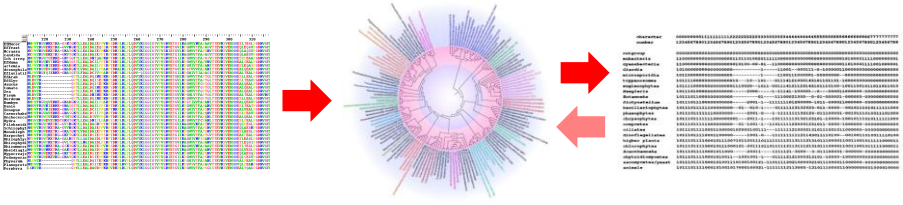
For this application to **phylogeny** , we interpret the MSA as a **position-by-position** (*i.e.* column-by-column) **hypothesis of homology** among these sequences

35

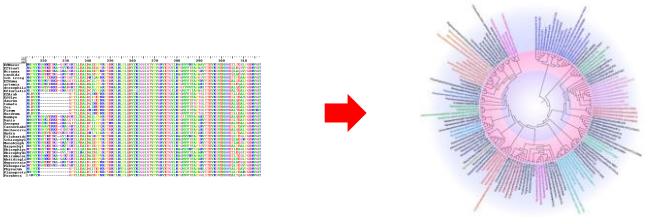
MSA by Mark Ragan; tree by Cheong Xin Chan

Tree inference from MSA: a few comments

- The sequences must be *homologous* for tree inference to make sense
- Trees and matrices are related complementary *data structures*
- Trees show inferred relationships; MSAs show conserved regions



Homology signal



We use the **homology signal** inherent in the sequences to make an inference about treelike relationships

Homology signal inheres in the sequences, not in their MSA

MSA can make it easier to see*, but doesn't create it

** and easier for existing computer programs to work with*

Homology signal (continued)

We shouldn't assume that MSA captures it all, or uses it optimally

MSA gives us access to

- Patterns within columns
- Local adjacency relationships
- Global architecture

} Let's consider these to be *components* of the homology signal

Here we'll focus on the first two of these components

Pattern and adjacency

The column component needs to capture
"sameness" of a character across sequences

For application in phylogenetics, "sameness" has to mean *homology* (or *orthology*).

It's difficult to build a statistical case that a particular single character in one sequence is homologous with a particular one in a second sequence.

MSA uses adjacency (and sometimes global) information to build this support.

Alternatively we might compare sets of adjacent characters (strings), which are less likely to occur by chance.

The adjacency component doesn't just provide
statistical support for the column component

Because conserved function arises in part from chemical properties of adjacent residues (*e.g.* in making that part of the molecule an active site or α-helix), we expect homology signal to have an adjacency component in its own right.

MSA: potential (and real) problems

Genomes are dynamic, data can be dirty, and MSA is hard

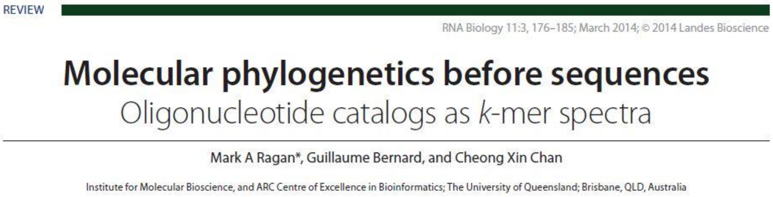
Within some but not all members of a gene set…

- Homologous regions may be inserted / deleted
- Homologous regions may be rearranged / duplicated
- Regions may have different evolutionary histories (LGT)
- Transcriptional variation → similar issues for protein sets

Sequences may be **mis-assembled** (or not assembled in the first place) and/or truncated

MSA is computationally difficult and/or heuristic

Can we extract enough/most/all of the homology signal without MSA ?



REVIEW

RNA Biology 11:3, 176–185; March 2014; © 2014 Landes Bioscience

Molecular phylogenetics before sequences

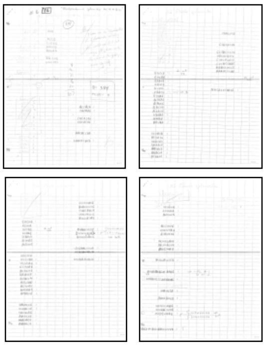
Oligonucleotide catalogs as *k*-mer spectra

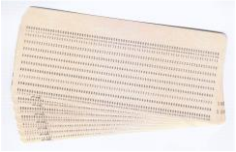
Mark A Ragan*, Guillaume Bernard, and Cheong Xin Chan

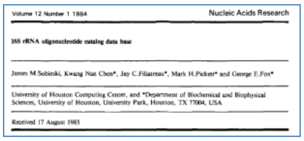
Institute for Molecular Bioscience, and ARC Centre of Excellence in Bioinformatics; The University of Queensland; Brisbane, QLD, Australia

Carl Woese – photo by Ken Luehrsen

Ribosomal RNA (1964)







Volume 12 Number 1 1984 Nucleic Acids Research

16S rRNA oligonucleotide catalog data base

James M.Sobieski, Kwang Nan Chen*, Jay C.Filiatreau*, Mark H.Pickett* and George E.Fox*

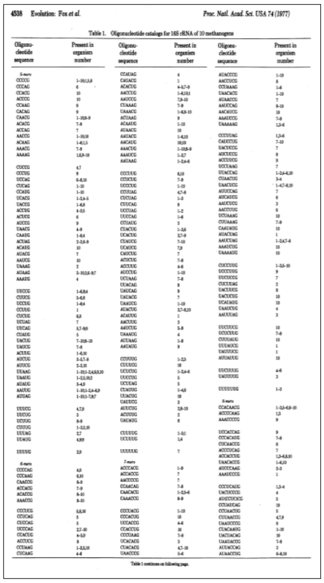
University of Houston Computing Center, and *Department of Biochemical and Biophysical Sciences, University of Houston, University Park, Houston, TX 77004, USA

Received 17 August 1983

Courtesy of George Fox 2013

Electrophoresis of Ribosomal RNA segments out of a lot of different species.

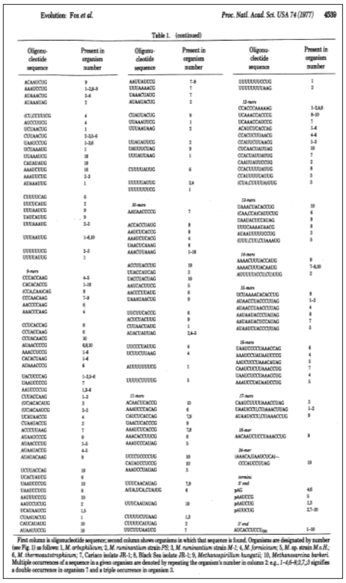
Oligonucleotide catalogs



4538 Evolution: Fox et al. Proc. Natl. Acad. Sci. USA 74 (1977)

Table 1. Oligonucleotide catalogs for 16S rRNA of 10 methanogens

Table 1 continues on following page.



Evolution: Fox et al. Proc. Natl. Acad. Sci. USA 74 (1977) 4539

Table 1. (continued)

First column is oligonucleotide sequence; second column shows organisms in which that sequence is found. Organisms are designated by number (see Fig. 1) as follows: 1, *M. arbophilicum*; 2, *M. ruminantium strain PS*; 3, *M. ruminantium* strain *M-1*; 4, *M. formicicum*; 5, *M.* sp. strain *M.o.H.*; 6, *M. thermoautotrophicum*; 7, Cariaco isolate JR-1; 8, Black Sea isolate JR-1; 9, *Methanospirillum hungatii*; 10, *Methanosarcina barkeri*. Multiple occurrences of a sequence in a given organism are denoted by repeating the organism's number in column 2: e.g., *1–4,6–8;3,7,;3* signifies a double occurrence in organism 7 and a triple occurrence in organism 3.

Similarity coefficient:

$S_{AB} = 2\,N_{AB} / (N_A + N_B)$

where

N_A = number of oligomers of at least length *L for RNA of organism A*, and

N_{AB} = total number of coincident oligomers between catalogs A and B

See (Fox *et al. IJSB* 1977) for a detailed definition.

Table 2 Relatedness in pairwise comparisons

Similarity coefficient, S_{ij}

	E. coli	*B. subtilis*	*R. spheroides*	6301	6714	6701	6308	*Porphyridium*	*Euglena*
E. coli		33	37	31	31	28	28	27	27
B. subtilis	86		35	37	34	32	32	34	26
R. spheroides	88	87		32	33	30	29	30	23
6301	85	88	86		49	44	41	36	35
6714	85	87	86	92		58	52	47	32
6701	84	86	85	90	94		83	43	33
6308	84	86	84	89	92	98		42	32
Porphyridium	83	87	85	88	91	90	89		36
Euglena	83	82	80	87	86	86	86	87	

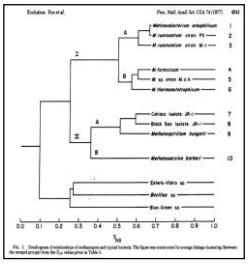
% Homology, C_{ij}

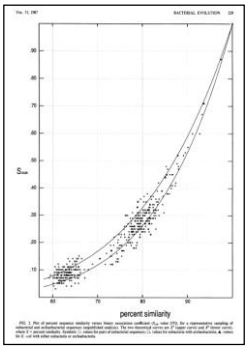
There are several ways in which the data of Table 1 can be used to derive single values (similarity coefficients S_{ij}) which measure the relatedness of any pair of RNAs *i* and *j*. That which seems most reasonable to us defines S_{ij} as

$$\left[200\sum_{N=L}^{\infty} k_{N_{ij}} N\right] / \sum_{N=L}^{\infty} (t_{N_i} + t_{N_j}) N$$

where *N* is oligonucleotide length (in residues), $k_{N_{ij}}$ is the number of *N*-mers common to the two RNAs, and t_{N_i} and t_{N_j} are the total number of *N*-mers obtained from RNAs *i* and *j*, respectively. S_{ij} is thus equivalent to 200 times the number of individual nucleotide residues present in sequences (pentamers and larger) which are common to two rRNAs divided by the number of individual residues present in all sequences (pentamers and larger) found in both rRNAs. It will range in value from 100 for identical molecules down to 10–15 for totally unrelated species. (Chance sequence coincidences prevent S_{ij} from reaching zero in such cases.) Table 2 shows (above the diagonal) values of S_{ij} computed for the 36 possible pairwise comparisons involving the blue-green and chloroplast catalogues and, in addition, representative bacterial 16S rRNA catalogues obtained by Woese and his collaborators for *Escherichia coli*[11-14], *Bacillus subtilis*[9], and *Rhodopseudomonas spheroides*[10] (a photosynthetic bacterium). For certain purposes it is useful to convert values of S_{ij} to values (C_{ij}) for the probability of any single nucleotide residue having remained unchanged during the evolutionary divergence of RNAs *i* and *j*, since the latter has more obvious physical significance (and can be equated with "% homology"). Such conversion was made with

Bonen & Doolittle, *Nature* 1976





Fox *et al.*, *PNAS* 1977 (top)
Woese, *Microbiol. Rev*. 1987 (bottom)

The three kingdoms (domains) of life

Proc. Natl. Acad. Sci. USA
Vol. 74, No. 11, pp. 5088–5090, November 1977
Evolution

Phylogenetic structure of the prokaryotic domain: The primary kingdoms

(archaebacteria/eubacteria/urkaryote/16S ribosomal RNA/molecular phylogeny)

CARL R. WOESE AND GEORGE E. FOX*

Department of Genetics and Development, University of Illinois, Urbana, Illinois 61801

Communicated by T. M. Sonneborn, August 18, 1977

ABSTRACT A phylogenetic analysis based upon ribosomal RNA sequence characterization reveals that living systems represent one of three aboriginal lines of descent: (*i*) the eubacteria, comprising all typical bacteria; (*ii*) the archaebacteria, containing methanogenic bacteria; and (*iii*) the urkaryotes, now represented in the cytoplasmic component of eukaryotic cells.

The biologist has customarily structured his world in terms of certain basic dichotomies. Classically, what was not plant was animal. The discovery that bacteria, which initially had been considered plants, resembled both plants and animals less than plants and animals resembled one another led to a reformulation of the issue in terms of a yet more basic dichotomy, that of eukaryote versus prokaryote. The striking differences between eukaryotic and prokaryotic cells have now been documented in endless molecular detail. As a result, it is generally taken for granted that all extant life must be of these two basic types.

Thus, it appears that the biologist has solved the problem of the primary phylogenetic groupings. However, this is not the case. Dividing the living world into *Prokaryotae* and *Eukaryotae* has served, if anything, to obscure the problem of what extant groupings represent the various primeval branches from the common line of descent. The reason is that eukaryote/prokaryote is not primarily a phylogenetic distinction, although to construct phylogenetic classifications between domains: Prokaryotic kingdoms are not comparable to eukaryotic ones. This should be recognized by an appropriate terminology. The highest phylogenetic unit in the prokaryotic domain we think should be called an "urkingdom"—or perhaps "primary kingdom." This would recognize the qualitative distinction between prokaryotic and eukaryotic kingdoms and emphasize that the former have primary evolutionary status.

The passage from one domain to a higher one then becomes a central problem. Initially one would like to know whether this is a frequent or a rare (unique) evolutionary event. It is traditionally assumed—without evidence—that the eukaryotic domain has arisen but once; all extant eukaryotes stem from a common ancestor, itself eukaryotic (2). A similar prejudice holds for the prokaryotic domain (2). [We elsewhere argue (6) that a hypothetical domain of lower complexity, that of "progenotes," may have preceded and given rise to the prokaryotes.] The present communication is a discussion of recent findings that relate to the urkingdom structure of the prokaryotic domain and the question of its unique as opposed to multiple origin.

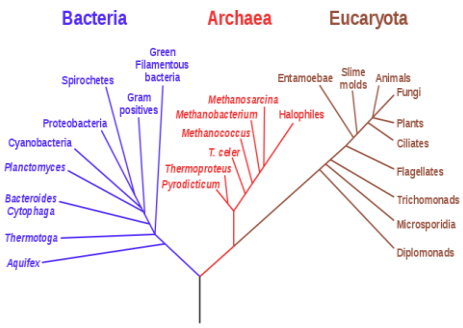
Phylogenetic relationships cannot be reliably established in terms of noncomparable properties (7). A comparative approach that can measure degree of difference in comparable

Table 1. Association coefficients (S_{AB}) between representative members of the three primary kingdoms

	1	2	3	4	5	6	7	8	9	10	11	12	13
1. *Saccharomyces cerevisiae*, 18S	—	0.29	0.33	0.05	0.06	0.08	0.09	0.11	0.08	0.11	0.11	0.08	0.08
2. *Lemna minor*, 18S	0.29	—	0.36	0.10	0.05	0.06	0.10	0.09	0.11	0.10	0.10	0.13	0.07
3. L cell, 18S	0.33	0.36	—	0.06	0.06	0.07	0.07	0.09	0.06	0.10	0.10	0.09	0.07
4. *Escherichia coli*	0.05	0.10	0.06	—	0.24	0.25	0.28	0.26	0.21	0.11	0.12	0.07	0.12
5. *Chlorobium vibrioforme*	0.06	0.05	0.06	0.24	—	0.22	0.22	0.20	0.19	0.06	0.07	0.06	0.09
6. *Bacillus firmus*	0.08	0.06	0.07	0.25	0.22	—	0.34	0.26	0.20	0.11	0.13	0.06	0.12
7. *Corynebacterium diphtheriae*	0.09	0.10	0.07	0.28	0.22	0.34	—	0.23	0.21	0.12	0.12	0.09	0.10
8. *Aphanocapsa* 6714	0.11	0.09	0.09	0.26	0.20	0.26	0.23	—	0.31	0.11	0.11	0.10	0.10
9. Chloroplast (*Lemna*)	0.08	0.11	0.06	0.21	0.19	0.20	0.21	0.31	—	0.14	0.12	0.10	0.12
10. *Methanobacterium thermoautotrophicum*	0.11	0.10	0.10	0.11	0.06	0.11	0.12	0.11	0.14	—	0.51	0.25	0.30
11. *M. ruminantium* strain M-1	0.11	0.10	0.10	0.12	0.07	0.13	0.12	0.11	0.12	0.51	—	0.25	0.24
12. *Methanobacterium* sp., Cariaco-isolate JR-1	0.08	0.13	0.09	0.07	0.06	0.06	0.09	0.10	0.10	0.25	0.25	—	0.32
13. *Methanosarcina barkeri*	0.08	0.07	0.07	0.12	0.09	0.12	0.10	0.10	0.12	0.30	0.24	0.32	—

The 16S (18S) ribosomal RNA from the organisms (organelles) listed were digested with T1 RNase and the resulting digests were subjected to two-dimensional electrophoretic separation to produce an oligonucleotide fingerprint. The individual oligonucleotides on each fingerprint were then sequenced by established procedures (13, 14) to produce an oligonucleotide catalog characteristic of the given organism (3, 4, 13–17, 22, 23; unpublished data). Comparisons of all possible pairs of such catalogs defines a set of association coefficients (S_{AB}) given by: $S_{AB} = 2N_{AB}/(N_A + N_B)$, in which N_A, N_B, and N_{AB} are the total numbers of nucleotides in sequences of hexamers or larger in the catalog for organism A, in that for organism B, and in the intersection of the two catalogs, respectively (13, 23).

Phylogenetic Tree of Life



From Wikimedia Commons
after Carl Woese and colleagues (~1972 ff.)

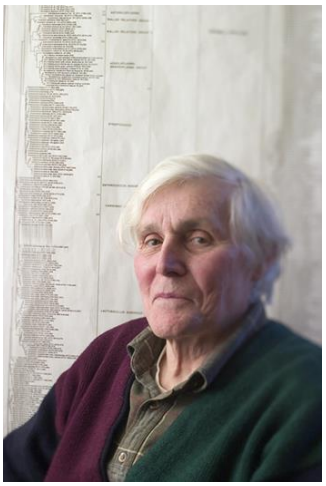
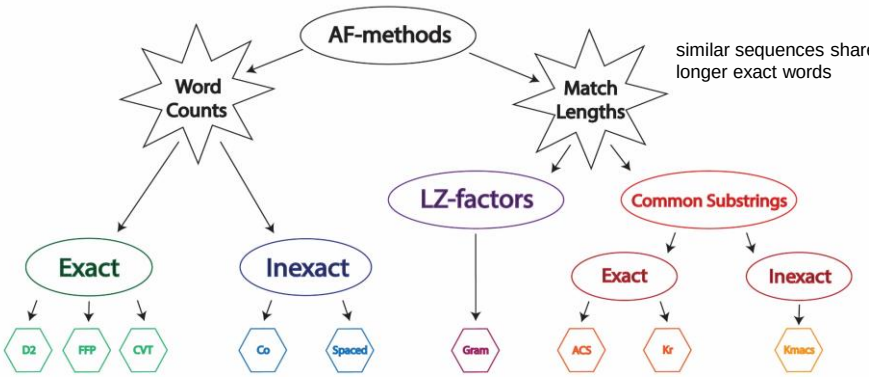


Image courtesy of Institute for Genomic Biology
University of Illinois

Alignment-free methods



similar sequences share longer exact words

Guillaume Bernard, after Haubold, *Briefings in Bioinformatics* (2013)

k-mers / *k*-tuples / *k*-words / *n*-mers / *n*-grams

A G C C G C T T A G T C A A C T

Here, $k = 6$

AGCCGC
GCCGCT
CCGCTT
(...)
TCAACT

For a sequence of length S, there are $S - k + 1$ k-mers, not all of which are necessarily unique

There's also a parallel world of ***patterns***

Höhl, Rigoutsos & Ragan, *Evol Bioinf* 2:357-373 (2006)

D_2 statistics: a brief overview

The D_2 statistic is the count of exact word matches of length k between two sequences

For alphabet A, there are $|A|^k$ possible words w of length k. Given sequences X and Y,

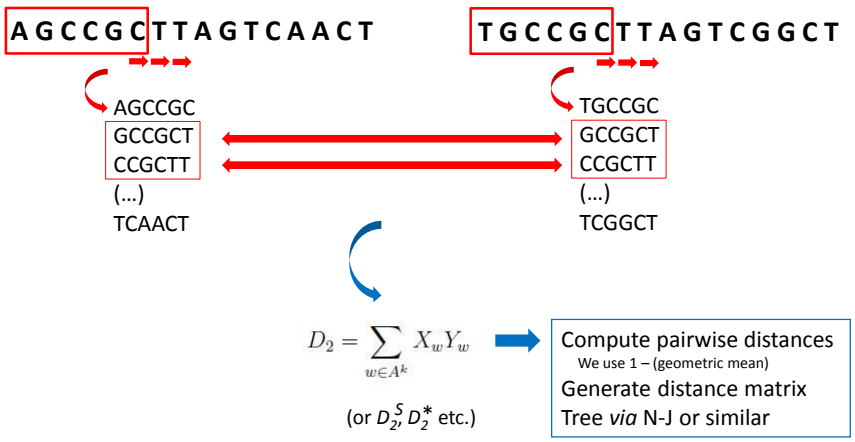
$$D_2 = \sum_{w \in A^k} X_w Y_w \qquad , where\ X_w = count\ of\ w\ in\ X$$

Because D_2 is sensitive to sequence length, the statistic is often normalised by the probability of occurrence of specific words (D_2^S), or by assuming a Poisson distribution of word occurrence (D_2^*) for long words

Although defined for exact word matches, D_2 can be easily extended to n mismatches (neighbourhood of order n): D_2^n

Chor *et al.*, *Genome Biol* 10:R108 (2009); Reinert *et al.*, *J Comput Biol* 16:1615-1634 (2009); Reinert *et al.*, *J Comput Biol* 17:1349-1372 (2010); Burden *et al.*, *J Comput Biol* 21:41-63 (2014)

D_2-based distance



Other Alignment Free methods based on word counts

Feature frequency profile *Sims & Kim, PNAS 2011*	Compares *k*-mer frequency profiles (Jensen-Shannon divergence) & computes a pairwise distance
Composition vector *Wang & Hao, JME 2004*	using word frequencies normalised by probability of chance of occurrence
Word context **Co-phylog:** *Yi & Jin, NAR 2013*	Pairwise distances based on proportions of *k*-mers that differ in a certain position; more-realistic branch lengths
Spaced word frequencies *Leimeister, Bioinformatics 2014*	Considers word mismatches as well as matches; less statistical dependency between neighbouring matches

Alignment Free (AF) methods based on match length

In general, similar sequences share longer exact words

Grammar-based distance ***d*-gram:** Russell, *BMC Bioinf* 2010	The concatenate of two sequences is more compressible (*e.g.* by Lempel-Ziv) if the sequences are similar
Average common substring Ulitsky, *J Comp Biol* 2006	Mean of longest matches between sequences, starting from each position; unlike L-Z, word overlap is allowed
Shortest unique substring Haubold, *J Comp Biol* 2009	Longest common substring + 1, corrected for random matches: "AF version of Jukes-Cantor distance"
Underlying subwords Comin, *Algorith Mol Biol* 2012	Like ACS, but discards common subwords that are covered by longer (more-significant) ones
***k*-Mismatch ACS (kmacs)** Leimester, *Bioinformatics* 2014	ACS with *k* (in our notation, *n*) mismatches

Can we compute accurate trees using AF-based distances ?

Simulated data

- Generate replicate data on a known tree, varying data size, substitution model, tree shape, branch lengths etc.
- Extract *k*-mers & compute a tree; sweep over relevant parameters
- **Compare topologies (Robinson-Foulds metric)**
- Measure performance (precision, recall, sensitivity…)

Advantages/disadvantages

- We can study effects of different factors & scenarios individually
- Sequence models may be too simplistic

Empirical data

- Identify empirical datasets for which someone has ventured a phylogenetic tree
- Extract *k*-mers & compute a tree; sweep over *k*
- **Compare topologies (Robinson-Foulds metric)**
- Count congruent/incongruent edges & try to interpret

Advantages/disadvantages

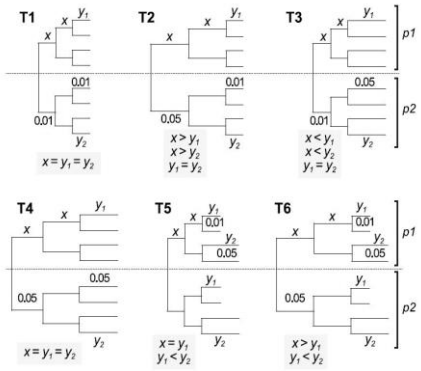
- Sequences are (by definition) real
- We can't study effects of different factors & scenarios individually
- The true tree remains unknown

First we simulated sequence data on a tree

Simulation software ranges from simplistic to maddeningly complex

Using *evolver* (PAML) we simulated DNA and protein sequence sets on trees of different size (8 / 32 / 128 taxa), symmetry, and absolute and relative branch lengths

We also simulated DNA sequences on trees generated under a coalescent model (not shown)



Chan *et al.*, *Scientific Reports* 4:6504 (2014)

We extracted *k*-mers at different *k*, computed distances under different variants of the D_2 statistic, and generated a N-J tree

Synthetic data
Related by a tree of known topology

→

***k*-mer lists**
One list per sequence at different k

→

***k*-mer distances**
Matrix of pairwise distances

→

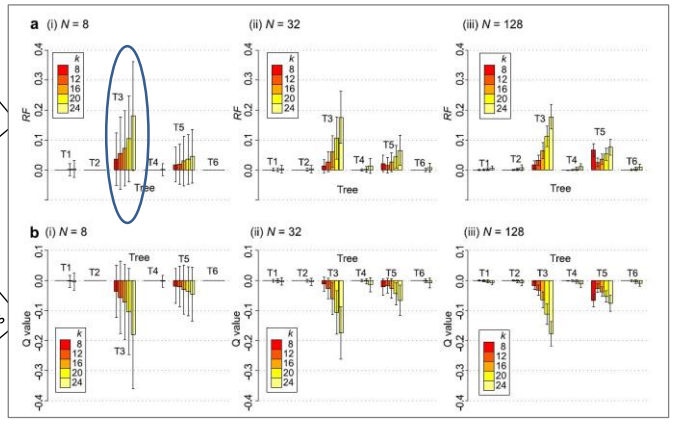
Neighbour-Joining
Or another distance approach

No method for confidence estimation is currently available, but one can imagine using a variant of the nonparametric bootstrap, or by jackknifing

Then we compared the D_2 + NJ tree with the known true topology, and with the topologies inferred using MSA + MrBayes

RF > 0 means D2n1 tree topology is incongruent

Q < 0 means D2n1 is less-accurate than MSA + MrBayes



DNA alphabet, L = 1500 nt, 100 replicates

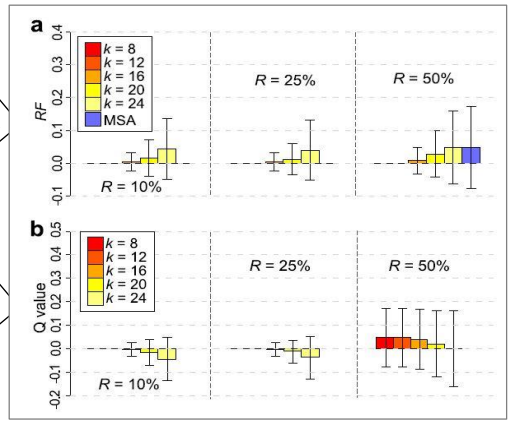
RF=Robinson Foulds metric

Chan *et al.*, *Scientific Reports* 2014

D_2 + NJ performs well with rearranged sequences

RF > 0 means D2n1 tree topology is incongruent

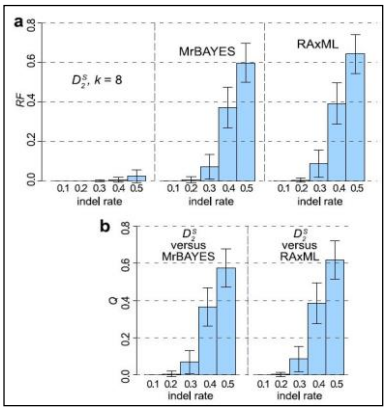
Q < 0 means D2n1 is less-accurate than MSA + MrBayes



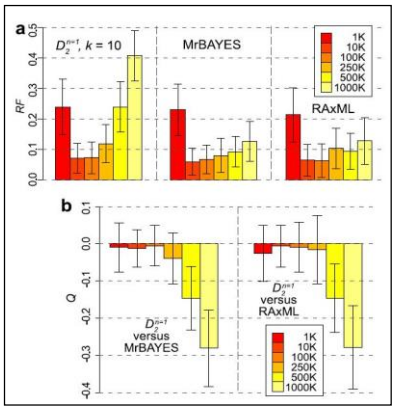
Non-overlapping rearrangement of R% of a DNA sequence set, N= 8, L= 5000.
MSA = MUSCLE + MrBayes. MAFFT performs slightly worse than MUSCLE.

Chan *et al.*, *Scientific Reports* 2014

D_2^S + NJ is more-robust to indels than leading MSA methods



With data simulated under a coalescent model, D_2^{n1} + NJ results are similar to MSA except at high/low sequence divergence



Numbers in box are N_e = effective population size
Smaller N_e implies shorter branch lengths on the tree

Chan *et al.*, *Scientific Reports* 2014

Summary: trees computed from *k*-mer distances

Aspect	👍
Sequence length	**D_2**
Recent sequence divergence	MSA
Ancient sequence divergence	**D_2**
Among-site rate heterogeneity	**D_2** or MSA
Compositional bias	**D_2** or MSA
Genetic rearrangement	**D_2**
Incomplete sequence data	MSA
Insertions/deletions	**D_2**
Computational scalability	**D_2**
Memory consumption	MSA

Accuracy of D_2 methods increases with L

D_2 methods are more **robust** to ancient sequence divergence, to rearrangement and to indel frequency

D_2 methods are more **sensitive** to recent sequence divergence and to the presence of incomplete (truncated) data

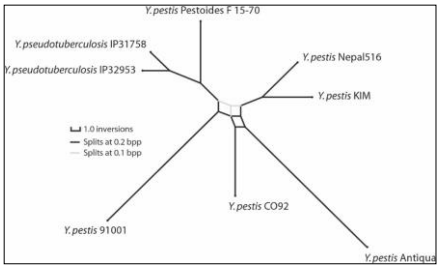
Optimal *k* is negatively correlated with alphabet size, and is not greatly affected by N or L in a biologically relevant range

D_2 methods are more **scalable** to large data than are MSA-based approaches, but usually require more **memory**

Eight *Yersinia* genomes: AF *versus* inversion phylogeny

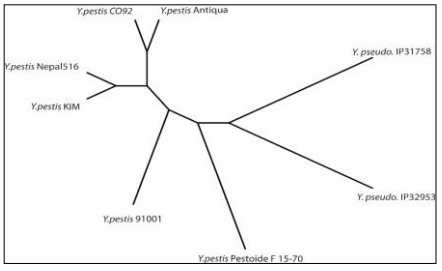
Consensus phylogenetic network based on inversions. Mauve (78 locally collinear blocks) then BADGER (Larget, *MBE* 2005). Requires extensive parameter estimation, with each run 500K MCMC generations. Took ~ 2 weeks.

Darling, Miklós & Ragan, *PLoS Genetics* (2008)

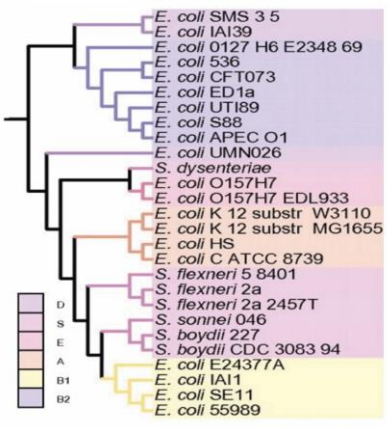


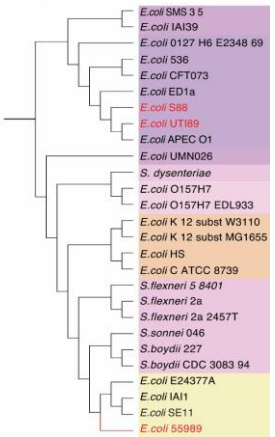
Kr (Haubold, *BMC Bioinformatics* 2005) yields a congruent phylogeny; no parameter optimisation, runtime 1 minute on laptop.

Bernard, Chan & Ragan, unpublished



27 *Escherichia coli* + *Shigella* genomes





ProgressiveMauve alignment (17 hours on HPC), extract 5282 single-copy gene sets $N \geq 4$, GBlocks, MrBayes (5M MCMC generations, 10 models) followed by MRP

Skippington & Ragan, *BMC Genomics* (2011)

Co-phylog (Yi & Jin, *NAR* 2013) with k=8, 113sec on laptop

Bernard, Chan & Ragan, unpublished

Conclusions & outlook

AF methods hold considerable potential in phylogenetics & phylogenomics

But MSA-based approaches have a six-decade head start

With synthetic data, AF methods perform better than MSA-based approaches under some evolutionarily relevant scenarios, but worse under others

With empirical data, the jury is still out

(Some) AF methods could likely be subsumed under a rigorous model, although probably at the cost of speed & scalability

i.e. what makes them attractive in the first place

Efficient data structures & precomputation have much to offer

Other application areas include LGT analysis, and trees directly from NGS data

Song *et al.*, *J Comp Biol* 2013; Yi & Jin, *NAR* 2013

Bibliography

[1] H. Carrillo and D. Lipmann. The multiple sequence alignment problem in biology. *SIAM J. Appl. Math*, 48:1073–1082, 1988.

[2] D. Feng and R. F. Doolittle. Progressive sequence alignment as a prerequisite to correct phylogenetic trees. *J. Mol. Evol.*, 25:351–360, 1987.

[3] W. M. Fitch and E. Margoliash. Construction of phylogenetic trees. *science*, 15:279–284, 1967.

[4] D. Gusfield. *Algorithms on Strings, Trees and Sequences*. Cambridge University Press, New York, 1997.

[5] T. Jiang, L. Wang, and E. L. Lawler. Approximation algorithms for tree alignment with a given phylogeny. *Algorithmica*, 16:302–315, 1996.

[6] D. J. Lipman, S. Altshul, and J. Kececiogly. A tool for multiple sequence alignment. *Proc. Natl. Academy Science*, 86:4412–4415, 1989.

[7] M. Murata, J.S. Richardson, and J.L. Sussman. Three protein alignment. *Medical Information Sciences*, 231:9, 1999.

[8] J. D. Thompson, D. G. Higgins, and T. J. Gibson. Clustal w: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res*, 22:4673–80, 1994.

[9] L. Wang and T. Jiang. On the complexity of multiple sequence alignment. *Computational Biology*, 1:337–348, 1994.

[10] http://www.uib.no/aasland/chromo/chromoCC.html.

[11] http://www.uib.no/aasland/chromo/chromo-tree.gif

[12] L. Sterck, ProCoGen Training Workshop, Umea, 31-1 2013.

[13] Mark Ragan. Phylogenetics without multiple sequence alignment, IPAM Workshop on Multiple Sequence Alignment, UCLA, 13 January 2015.

[14] T. Lengauer, C. Hartmann, in Comprehensive Medicinal Chemistry II, 2007