Cap. 6 How Do We Compare Biological Sequences

- From Sequence Comparison to Biological Insights
- The Alignment Game and the Longest Common Subsequence
- The Manhattan Tourist Problem
- Dynamic Programming and Backtracking Pointers
- From Manhattan to an Arbitrary Directed Acyclic Graph
- From Global to Local Alignment [Matrici di Score…]
- Penalizing Insertions and Deletions in Sequence Alignment
- Space-Efficient Sequence Alignment
- Multiple Sequence Alignment OGGI "COMPLETIAMO"

1

Multiple Alignment Problem: *Find the highest-scoring alignment between multiple strings.*

- **Input:** A collection of *t* strings (and some way of scoring columns of a multiple alignment).
- **Output:** A multiple alignment of these strings having maximum score.

MSA is used for:

- Detection of conserved domains in a group of genes or proteins
- Construction of a phylogenetic tree
- Prediction of a protein 3D-structure starting with its amino acids sequence (*e.g.*, AlphaFold, RoseTTAFold)
- Determination of a consensus sequence (*e.g.*, transposons)

2

Multiple Alignment Problem: *Find the highest-scoring alignment between multiple strings.*

- **Input:** A collection of *t* strings (and some way of scoring columns of a multiple alignment).
- **Output:** A multiple alignment of these strings having maximum score.

The sum-of-pairs score (sop) of M is the sum of all pair-wise induced alignment scores

```
AAGAA_A
AT_AATG
CTG_G_G
```

```
d/i =
r   =
m   =
```

```
AAGAA_A
_ATAATG
C_TGG_G
```

The optimal MSA for S w.r.t. to sop is the MSA with the highest sop-score over all possible MSA for S

3

Profile Representation of Multiple Alignment

4

Profile Representation of Multiple Alignment

A *profile* is a probability for each letter to occur in each column, i.e., a description of the consensus of a multiple sequence alignment. It uses a position-specific scoring system to capture information about the degree of conservation at various positions in the multiple alignment.

```
  - A G G C T A T C A C C T G
  T A G - C T A C C A - - - G
  C A G - C T A C C A - - - G
  C A G - C T A T C A C - G G
  C A G - C T A T C G C - G G
```

p=1/#

A	0	1	0	0	0	0	1	0	0	.8	0	0	0	0
C	.6	0	0	0	1	0	0	.4	1	0	.6	.2	0	0
G	0	0	1	.2	0	0	0	0	0	.2	0	0	.4	1
T	.2	0	0	0	0	1	0	.6	0	0	0	0	.2	0
-	.2	0	0	.8	0	0	0	0	0	0	.4	.8	.4	0

5

Is the "Profile representation" mentioned above suitable?

To predict an HIV phenotype, we need an *accurate* alignment: a single misalignment at a position influencing the SI phenotype leads to an error (e.g. **11-25 Rule)**.

STOP and Think: Was it a good idea to use the *same* scoring matrix across different columns of an alignment?

We need a statistically solid **problem formulation** for alignment that uses a *different* scoring approach at different columns.

6

Hidden Markov Model (HMM)

7

Hidden Markov Model (HMM)

Σ: an **alphabet** of emitted symbols — H and T

States : a set of **hidden states** — *F* and *B*

Transition = ($transition_{l,k}$): a |*States*| ×|*States*| matrix of **transition probabilities** (of changing from state *l* to state *k*)

	F	*B*
F	0.9	0.1
B	0.1	0.9

Emission= ($emission_k(b)$): a |*States*| × |∑| matrix of **emission probabilities** (of emitting symbol *b* when the HMM is in state *k*)

	H	T
F	0.50	0.50
B	0.75	0.25

Goal: Infer the most likely sequence of hidden states based on the sequence of emitted symbols

8

Hidden Path

Hidden path: the sequence $\pi = \pi_1 \ldots \pi_n$ of states that the HMM passes through.

- $\Pr(x, \pi)$: the probability that an HMM follows the hidden path π *and* emits the string $x = x_1 x_2 \ldots x_n$.

x:	T	H	T	H	H	H	T	H	T	T	H
π:	F	F	F	B	B	B	B	B	F	F	F

$$\sum_{\text{all possible emitted strings } x} \sum_{\text{all possible hidden paths } \pi} \Pr(x, \pi) = 1$$

- $\Pr(x|\pi)$: the **conditional probability** that an HMM emits the string x *after* following the hidden path π.

$$\sum_{\text{all possible emitted strings } x} \Pr(x|\pi) = 1$$

9

$$\Pr(x, \pi) = \Pr(x|\pi) * \Pr(\pi)$$

- $\Pr(x, \pi)$: the probability that an HMM follows the hidden path π and emits the string x.
- $\Pr(x_i|\pi_i)$ – probability that x_i was emitted from the state π_i (equal to $emission_{\pi i}(x_i)$).
- $\Pr(\pi_{i-1} \rightarrow \pi_i)$ – probability that the HMM moved from $\pi_{i-1} \rightarrow \pi_i$ (equal to $transition_{\pi i, \pi i+1}$).

x	T	H	T	H	H	H	T	H	T	T	H
π	F	F	F	B	B	B	B	B	F	F	F
$\Pr(\pi_{i-1} \rightarrow \pi_i)$	.5	.9	.9	.1	.9	.9	.9	.9	.1	.9	.9
$\Pr(x_i|\pi_i)$	½	½	½	¼	¾	¾	¼	¾	½	½	½

$$\Pr(\pi) = \prod_{i=1,n} \Pr(\pi_{i-1} \rightarrow \pi_i) = \prod_{i=1,n} transition_{\pi i-1, \pi i}$$

$$\Pr(x|\pi) = \prod_{i=1,n} \Pr(x_i|\pi_i) = \prod_{i=1,n} emission_{\pi i}(x_i)$$

10

Decoding Problem… Viterbi algorithm

Decoding Problem: *Find an optimal hidden path in an HMM given its emitted string.*

- **Input:** A string $x = x_1 \ldots x_n$ emitted by an HMM $(\sum, States, Transition, Emission)$.
- **Output:** A path π that maximizes the probability $Pr(x,\pi)$ over all possible paths through this HMM.

$$\begin{aligned} Pr(x, \pi) &= Pr(x|\pi) * Pr(\pi) \\ &= \Pi_{i=1,n} Pr(x_i|\pi_i) * Pr(\pi_{i-1} \rightarrow \pi_i) \\ &= \Pi_{i=1,n} emission_{\pi i}(x_i) * transition_{\pi i-1,\pi i} \end{aligned}$$

11

HMM, PROFILE, MSA

12

From Alignment to Profile

	1	2	3	4	5	6	7	8
	A	C	D	E	F	A C	A	D	F
	A	F	D	A	-	- -	C	C	F
Alignment	A	-	-	E	F	D -	F	D	C
	A	C	A	E	F	- -	A	-	C
	A	D	D	E	F	A A	A	D	F

Remove columns if the fraction of space symbols ("-") exceeds θ, **the maximum fraction of insertions threshold**.

13

From Alignment to Profile

	1	2	3	4	5	6	7	8
	A	C	D	E	F	A C	A	D	F
	A	F	D	A	-	- -	C	C	F
Alignment	A	-	-	E	F	D -	F	D	C
	A	C	A	E	F	- -	A	-	C
	A	D	D	E	F	A A	A	D	F
	A	C	D	E	F		A	D	F
	A	F	D	A	-		C	C	F
*Alignment**	A	-	-	E	F		F	D	C
	A	C	A	E	F		A	-	C
	A	D	D	E	F		A	D	F

14

From Alignment to Profile

		1	2	3	4	5	6	7	8
Alignment		A	C	D	E	F	A C	A	D	F
		A	F	D	A	–	– –	C	C	F
		A	–	–	E	F	D –	F	D	C
		A	C	A	E	F	– –	A	–	C
		A	D	D	E	F	A A	A	D	F
*Alignment**		A	C	D	E	F	A	D	F
		A	F	D	A	–	C	C	F
		A	–	–	E	F	F	D	C
		A	C	A	E	F	A	–	C
		A	D	D	E	F	A	D	F
PROFILE(*Alignment**)	A	1	0	1/4	1/5	0	3/5	0	0
	C	0	2/4	0	0	0	1/5	1/4	2/5
	D	0	1/4	3/4	0	0	0	3/4	0
	E	0	0	0	4/5	0	0	0	0
	F	0	1/4	0	0	1	1/5	0	3/5

Profile

15

From Alignment to Profile

		1	2	3	4	5	6	7	8
Alignment		A	C	D	E	F	A C	A	D	F
		A	F	D	A	–	– –	C	C	F
		A	–	–	E	F	D –	F	D	C
		A	C	A	E	F	– –	A	–	C
		A	D	D	E	F	A A	A	D	F
*Alignment**		A	C	D	E	F	A	D	F
		A	F	D	A	–	C	C	F
		A	–	–	E	F	F	D	C
		A	C	A	E	F	A	–	C
		A	D	D	E	F	A	D	F
PROFILE(*Alignment**)	A	1	0	1/4	1/5	0	3/5	0	0
	C	0	2/4	0	0	0	1/5	1/4	2/5
	D	0	1/4	3/4	0	0	0	3/4	0
	E	0	0	0	4/5	0	0	0	0
	F	0	1/4	0	0	1	1/5	0	3/5

GTCt/aGa/c

Vice versa???

16

From Profile to HMM

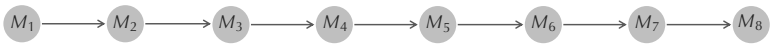
	1	2	3	4	5	6	7	8
Alignment	A	C	D	E	F	A C	A	D	F
	A	F	D	A	–	– –	C	C	F
	A	–	–	E	F	D –	F	D	C
	A	C	A	E	F	– –	A	–	C
	A	D	D	E	F	A A	A	D	F
*Alignment**	A	C	D	E	F		A	D	F
	A	F	D	A	–		C	C	F
	A	–	–	E	F		F	D	C
	A	C	A	E	F		A	–	C
	A	D	D	E	F		A	D	F
PROFILE(*Alignment**)	A	1	0	1/4	1/5	0	3/5	0	0
	C	0	2/4	0	0	0	1/5	1/4	2/5
	D	0	1/4	3/4	0	0	0	3/4	0
	E	0	0	0	4/5	0	0	0	0
	F	0	1/4	0	0	1	1/5	0	3/5

HMM diagram

1	*	.25	*	.75	*	.20	*	1	*	.20	*	.75	*	.60
A		D		D		A		F		F		D		F

17

Toward a Profile HMM

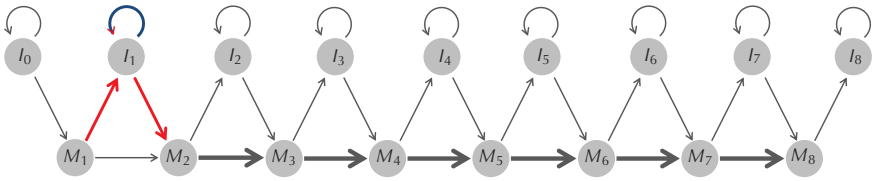


A **F** D D A F F D F

STOP and Think: How do we model insertions?

18

Toward a Profile HMM: Insertions

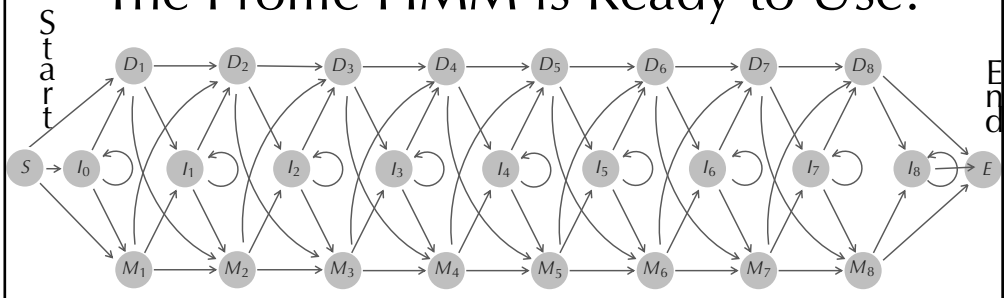


A **F** D D A F F D F

STOP and Think: How do we model deletions?

19

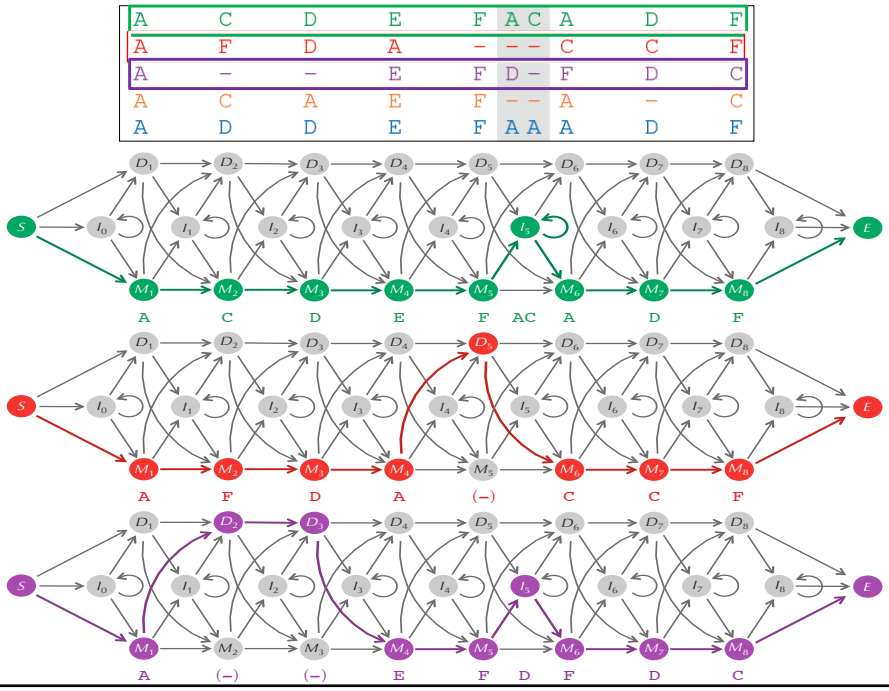
The Profile HMM is Ready to Use!



our «network»

20

Hidden Paths Through Profile HMM



21

Summarizing a Profile HMM

Σ: an **alphabet** of emitted symbols → Amino acids

States : a set of **hidden states** → Start, end, match, insertion, and deletion states

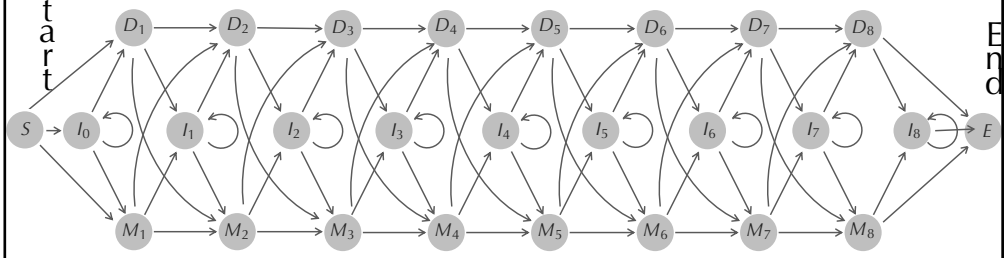
Transition = ($transition_{l,k}$): a |*States*| ×|*States*| matrix of **transition probabilities** (of changing from state *l* to state *k*)

Emission= ($emission_k(b)$): a |*States*| × |∑| matrix of **emission probabilities** (emitting symbol *b* when HMM is in state *k*)

It is not yet clear what the transition and emission probabilities should be!

22

The Profile HMM is Ready to Use!



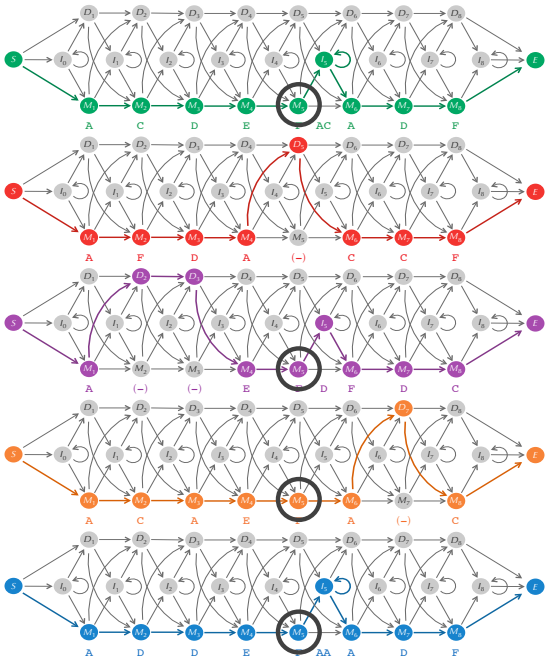
Profile HMM Problem: *Construct a profile HMM from a multiple alignment.*

- **Input:** A multiple alignment *Alignment* and a threshold θ (maximum fraction of insertions per column).
- **Output:** Transition and emission matrices of the profile HMM *HMM*(*Alignment*,θ).

our «network»

23

Transition Probabilities of Profile HMM



consider the outgoing transition

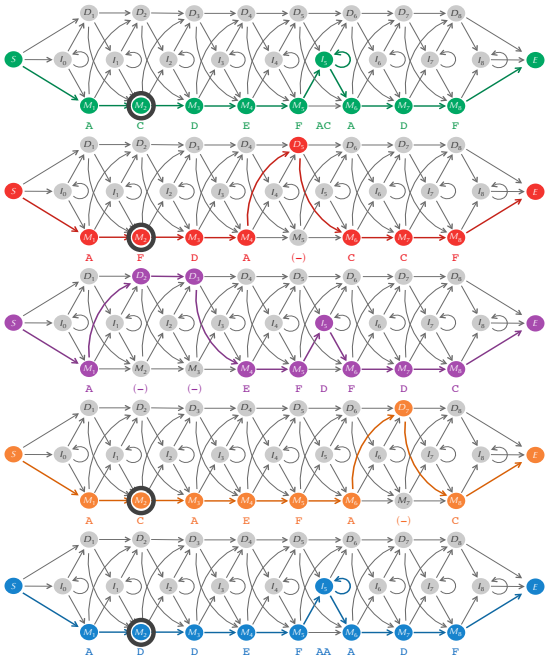
4 transitions from M_5:

1 + 1 + 1 = 3 into I_5
1 into M_6
0 into D_6

$transition_{Match(5),Insertion(5)} = 3/4$
$transition_{Match(5),Match(6)} = 1/4$
$transition_{Match(5),Deletion(6)} = 0$

24

Emission Probabilities of Profile HMM



symbols emitted from M_2:
C, F, C, D

$emission_{Match(2)}(A) = 0$
$emission_{Match(2)}(C) = 2/4$
$emission_{Match(2)}(D) = 1/4$
$emission_{Match(2)}(E) = 0$
$emission_{Match(2)}(F) = 1/4$

25

CLASSIFYING PROTEINS WITH PROFILE HMMS

26

We have HMM of a family of proteins, representing a MSA.

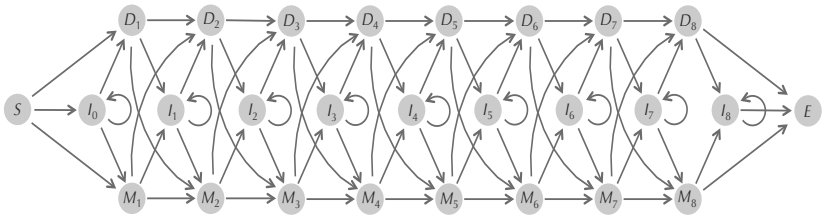
A new protein comes in

We want to decide whether it belongs to the given family or not

CLASSIFYING PROTEINS WITH PROFILE HMMS

27

Aligning a Protein Against a Profile HMM



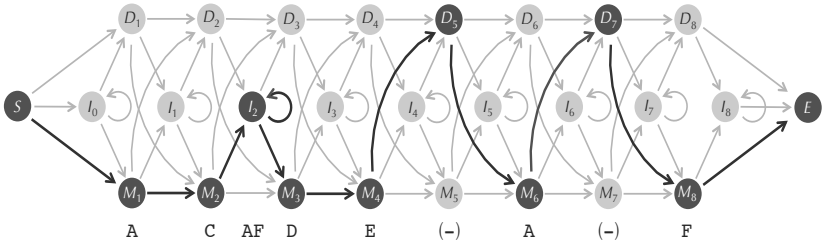
	A	C	– –	D	E	F	AC	A	D	F
	A	F	– –	D	A	–	– –	C	C	F
Alignment	A	–	– –	–	E	F	D–	F	D	C
	A	C	– –	A	E	F	– –	A	–	C
	A	D	– –	D	E	F	AA	A	D	F

Protein **ACAFDEAF**

emitted symbols…

28

Aligning a Protein Against a Profile HMM



Alignment

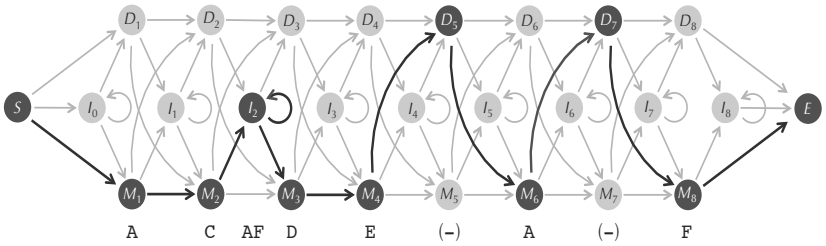
A	C	– –	D	E	F	AC	A	D	F
A	F	– –	D	A	–	– –	C	C	F
A	–	– –	–	E	F	D–	F	D	C
A	C	– –	A	E	F	– –	A	–	C
A	D	– –	D	E	F	AA	A	D	F

Protein **ACAFDEAF**

Apply Viterbi algorithm to find optimal hidden path (i.e., the sequence of states in HMM!)

29

Aligning a Protein Against a Profile HMM



Alignment

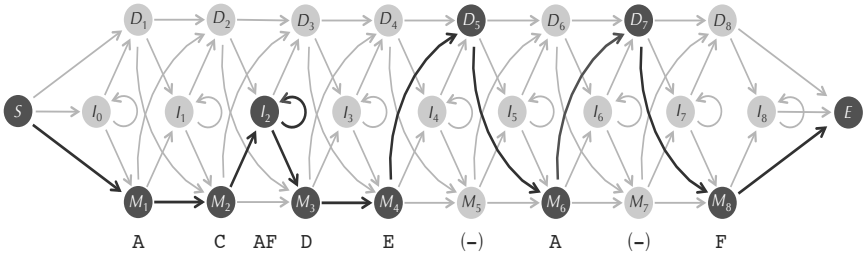
A	C	– –	D	E	F	AC	A	D	F
A	F	– –	D	A	–	– –	C	C	F
A	–	– –	–	E	F	D–	F	D	C
A	C	– –	A	E	F	– –	A	–	C
A	D	– –	D	E	F	AA	A	D	F

Protein **A C AFD E A F**

If the probabilistic threshold is satisfied, the proteins belongs to the family (and extends the alignment)

30

Alignment with a Profile HMM

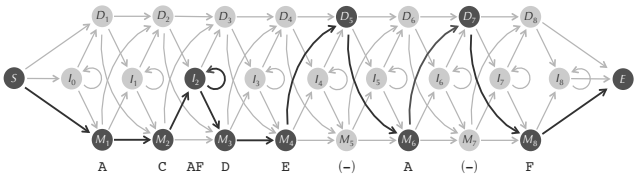


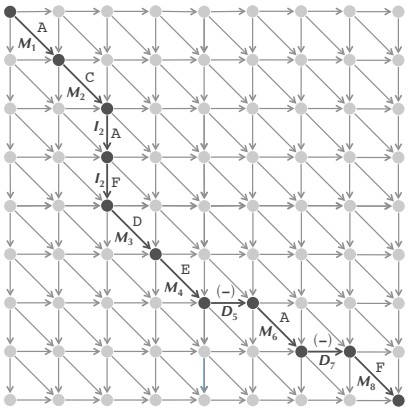
Sequence Alignment with Profile HMM Problem: *Align a new sequence to a family of aligned sequences using a profile HMM.*

- **Input:** A multiple alignment *Alignment*, a string *Text*, a threshold θ (maximum fraction of insertions per column), and a pseudocount σ.
- **Output:** An optimal hidden path emitting *Text* in the profile HMM *HMM*(*Alignment*, θ, σ).

31

Have I Wasted Your Time?

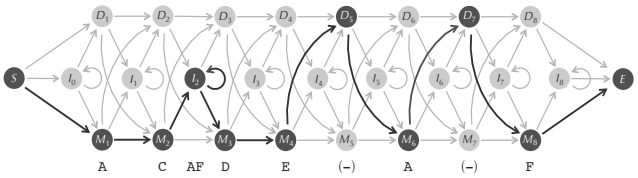


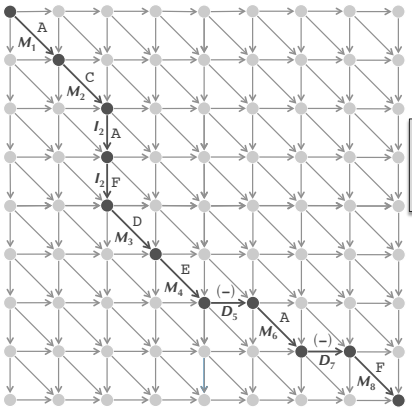


STOP: A path through the profile HMM diagram looks like a lot like a path through an alignment graph! So what is different?

32

I Have Not Wasted Your Time!





$$s_{M(j),i} = \max \begin{cases} s_{I(j-1),i-1} * weight_{i-1}(I(j-1),M(j)) \\ s_{D(j-1),i-1} * weight_{i-1}(D(j-1), M(j)) \\ s_{M(j-1),i-1} * weight_{i-1}(M(j-1), M(j)) \end{cases}$$

The choice of alignment path is now based on varying transition and emission probabilities!

These individual score parameters allow us to detect similarities often missed by traditional sequence alignments

33

Citations

Biological sequence analysis 1998

by R Durbin · Cited by 7614 — **Biological sequence analysis. Probabilistic models of proteins and nucleic acids**. Richard Durbin. Sean R. Eddy. Anders Krogh. Graeme Mitchison… 366 pages

HMMER web server: interactive sequence similarity searching

RD Finn, J Clements, SR Eddy - Nucleic acids research, 2011 - academic.oup.com

… **HMMER** is a software suite for protein sequence similarity searches using probabilistic methods. Previously, **HMMER** has mainly been … A **HMMER** web server (http://**hmmer**.janelia.org) has been designed and implemented such that most protein database searches return …

☆ Save 99 Cite Cited by 3707 Related articles All 17 versions

34

Multiple Alignment: History

1975 Sankoff
Formulated multiple alignment problem and gave dynamic programming solution
1988 Carrillo-Lipman
Branch and Bound approach for MSA
1990 Feng-Doolittle
Progressive alignment
1994 Thompson-Higgins-Gibson-ClustalW
Most popular multiple alignment program
1998 Morgenstern et al.-DIALIGN
Segment-based multiple alignment
2000 Notredame-Higgins-Heringa-T-coffee
Using the library of pairwise alignments
2004 MUSCLE

What's next?

35

Inputs: N sequences

1.Using standard pairwise alignment, calculate a matrix of distances (alignment scores) between each pair of sequences. Consider this as an N-clique G, where edge {i,j} is labeled with the score of an optimal alignment of the i-th and j-th sequences.

2.Use Kruskal's algorithm to find a minimum spanning tree of G. Whenever a minimum spanning tree edge would connect two components, instead add a new root node with directed edges to the roots of the two components. This is the "guide tree" .

3. Do pairwise alignments according to the guide tree, working from the leaves to the root.

36

Multiple Alignment: History

1975 Sankoff
Formulated multiple alignment problem and gave dynamic programming solution
1988 Carrillo-Lipman
Branch and Bound approach for MSA
1990 Feng-Doolittle
Progressive alignment
1994 Thompson-Higgins-Gibson-ClustalW
Most popular multiple alignment program
1998 Morgenstern et al.-DIALIGN
Segment-based multiple alignment
2000 Notredame-Higgins-Heringa-T-coffee
Using the library of pairwise alignments
2004 MUSCLE

uses HMM profile-profile techniques and progressive alignment

Profile HMMs are probabilistic models that encapsulate the evolutionary changes that have occurred in a set of related sequences (i.e. a multiple sequence alignment).

linguaggio C

What's next?

37

ClustalW (Cluster Alignment Weighted)

Iterative progressive alignment

l'idea è di iniziare a raggruppare sequenze simili

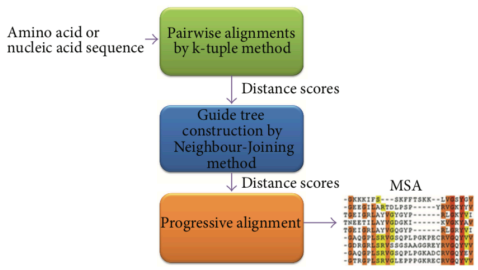


FIGURE 1: ClustalW algorithm, which works by taking an input of amino acid or nucleic acid sequences, completing a pairwise alignment using the k-tuple method, guide tree construction using the Neighbour-Joining method, followed by a progressive alignment to output a multiple sequence alignment.

(Clustal Omega: Iterative progressive alignment using hidden Markov models)

38

Step 1 : Pairwise alignment of all sequences

Example : Alignment of 7 globins (Hbb_human, Hbb_horse, Hba_human, Hba_horse, Myg_phyca, Glb5_petma and Lgb2_lupla)

The alignment can be obtained with:

- global or local methods
- heuristic methods
- ktuple method

```
Hbb_human    1 VHLTPEEKSAVTALWGKVNVDEVGGEALGRLLVVYPWTQRFFESFGDLST ...
                 | |. |||.|| ||| ||| :||||||||||||||||||||:||||||
Hbb_horse    2 VQLSGEEKAAVLALWDKVNEEEVGGEALGRLLVVYPWTQRFFDSFGDLSN ...

Hbb_human    1 LTPEEKSAVTALWGKV..NVDEVGGEALGRLLVVYPWTQRFFESFGDLST ...
                 |.| :|. | | |||  .  | | ||| |: . :| |. :|  | |||
Hba_human    3 LSPADKTNVKAAWGKVGAHAGEYGAEALERMFLSFPTTKTYFPHF.DLS. ...

Hba_human    3 LSPADKTNVKAAWGKVGAHAGEYGAEALERMFLSFPTTKTYFPHF.DLSH ...
                 ||  :|  | | | ||     | | ||| |: . :| |. :|  | |||.
Hbb_horse    2 LSGEEKAAVLALWDKVNEE..EVGGEALGRLLVVYPWTQRFFDSFGDLSN ...
```

Adapted from Julie Thompson, IGBMC

39

Step 2 : Distance matrix construction

Distance between two sequences = 1 - $\frac{\text{No. identical residues}}{\text{No. aligned residues}}$

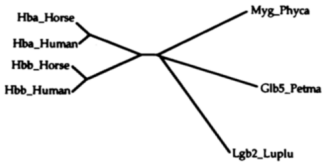
Hbb_human	1	-					
Hbb_horse	2	.17	-				
Hba_human	3	.59	.60	-			
Hba_horse	4	.59	.59	.13	-		
Myg_phyca	5	.77	.77	.75	.75	-	
Glb5_petma	6	.81	.82	.73	.74	.80	-
Lgb2_lupla	7	.87	.86	.86	.88	.93	.90	-
		1	2	3	4	5	6	7

Adapted from Julie Thompson, IGBMC

40

Step 3 : Guide tree construction

Unrooted NJ tree



Hbb_human	1	-						
Hbb_horse	2	.17	-					
Hba_human	3	.59	.60	-				
Hba_horse	4	.59	.59	.13	-			
Myg_phyca	5	.77	.77	.75	.75	-		
Glb5_petma	6	.81	.82	.73	.74	.80	-	
Lgb2_lupla	7	.87	.86	.86	.88	.93	.90	-
		1	2	3	4	5	6	7

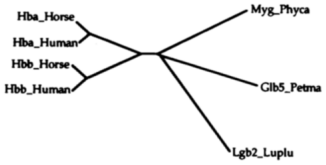
Neighbor-Joining clustering method:

The tree is then built by linking the least distant pair of nodes. When two nodes are linked, their common ancestral node is added to the tree and the terminal nodes with their respective branches are removed from the tree.

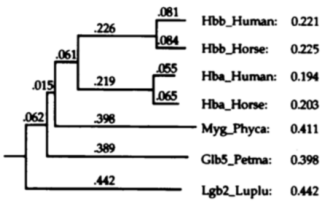
41

Step 3 : Guide tree construction

Unrooted NJ tree



Guide tree



Hbb_human	1	-						
Hbb_horse	2	.17	-					
Hba_human	3	.59	.60	-				
Hba_horse	4	.59	.59	.13	-			
Myg_phyca	5	.77	.77	.75	.75	-		
Glb5_petma	6	.81	.82	.73	.74	.80	-	
Lgb2_lupla	7	.87	.86	.86	.88	.93	.90	-
		1	2	3	4	5	6	7

Neighbor-Joining clustering method:

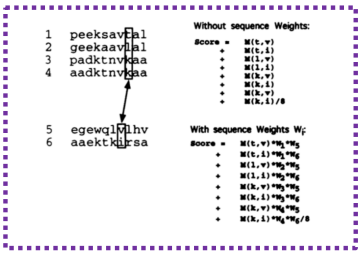
The tree is then built by linking the least distant pair of nodes. When two nodes are linked, their common ancestral node is added to the tree and the terminal nodes with their respective branches are removed from the tree.

42

Step 3 : Guide tree construction

At each step, the pair (X,Y) that diverges least among all is selected. In the matrix, the entry X and Y is replaced with X/Y. Each distance involving X and Y is computed as the arithmetic mean of the distances.

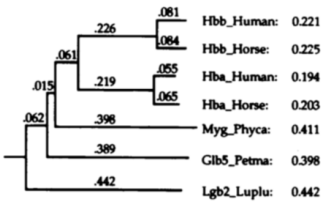
Alignment Weighting
if you align two alignments with 2 and 4 sequences respectively,
the score at each position is the **average** of 8 (2x4) comparisons.



43

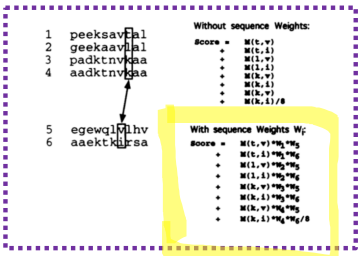
Step 3 : Guide tree construction

At each step, the pair (X,Y) that diverges least among all is selected. In the matrix, the entry X and Y is replaced with X/Y. Each distance involving X and Y is computed as the arithmetic mean of the distances.



The guide tree is then used to calculate WEIGHT for each sequence, which depends on the distance from branch to the root. If a sequence shares a common branch with another sequence, then the two or more sequences will share the weight calculated from the shared branch, and the SEQUENCE LENGTHS will be added together and divided by the number of sequences sharing the same branch.

Alignment Weighting
if you align two alignments with 2 and 4 sequences respectively,
the score at each position is the **average** of 8 (2x4) comparisons.

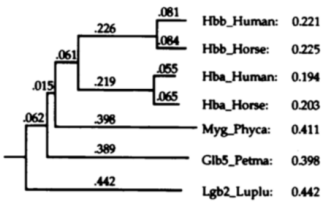


44

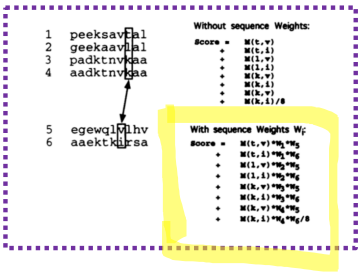
Step 3 : Guide tree construction

At each step, the pair (X,Y) that diverges least among all is selected. In the matrix, the entry X and Y is replaced with X/Y. Each distance involving X and Y is computed as the arithmetic mean of the distances.

Costruct the MSA by seeing the guided tree



Alignment Weighting
if you align two alignments with 2 and 4 sequences respectively,
the score at each position is the **average** of 8 (2x4) comparisons.

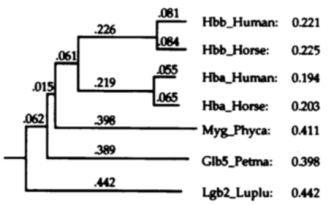


The guide tree is then used to calculate WEIGHT for each sequence, which depends on the distance from branch to the root. If a sequence shares a common branch with another sequence, then the two or more sequences will share the weight calculated from the shared branch, and the SEQUENCE LENGTHS will be added together and divided by the number of sequences sharing the same branch.

45

Step 3 : Guide tree construction

Guide tree



The time complexity for traditional approach is O(N^2). CLUSTER OMEGA uses a different method (mBed) to generate the guide tree, which has a time complexity of O(NlogN) — much faster than ClustalW.

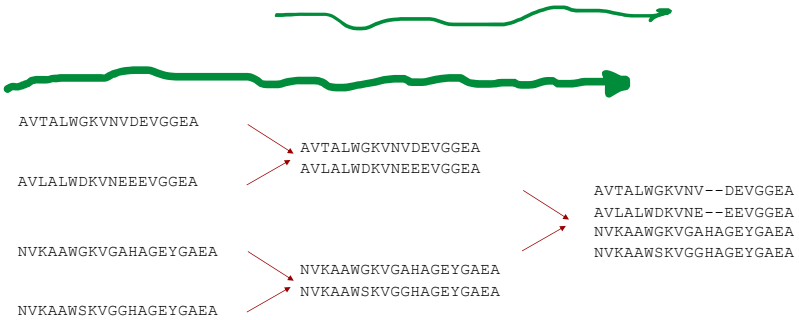
Hbb_human	1	-					
Hbb_horse	2	.17	-				
Hba_human	3	.59	.60	-			
Hba_horse	4	.59	.59	.13	-		
Myg_phyca	5	.77	.77	.75	.75	-	
Glb5_petma	6	.81	.82	.73	.74	.80	-
Lgb2_lupla	7	.87	.86	.86	.88	.93	.90
		1	2	3	4	5	6

Neighbor-Joining clustering method:

The tree is then built by linking the least distant pair of nodes. When two nodes are linked, their common ancestral node is added to the tree and the terminal nodes with their respective branches are removed from the tree.

46

Step 4 : Progressive alignment



The sequences are aligned progressively (global or local algorithm). I can find theses cases, seeing the guided tree:

- alignment of 2 sequences, create profile (consensus)
- alignment of 1 sequence and a profile (group of sequences)
- alignment of 2 profiles (groups of sequences)

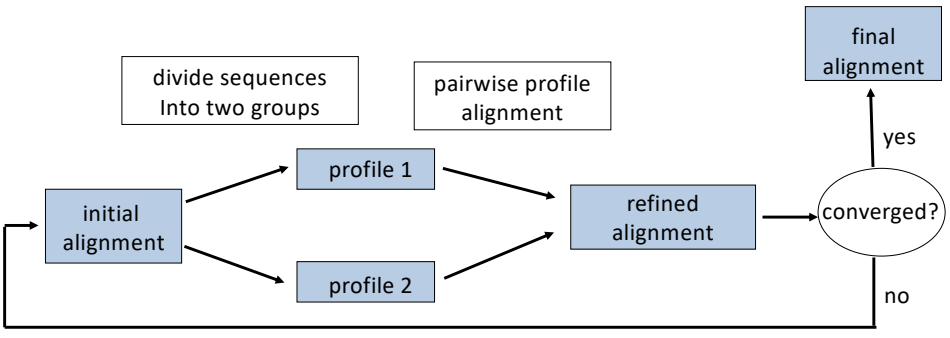
Adapted from Julie Thompson, IGBMC

Cluster Omega uses HHalign to increase accuracy. Sequences alignment is converted to HMM, which then helps the alignment.

47

Iterative alignment

Iterative alignment refines an initial progressive multiple alignment by iteratively dividing the alignment into two profiles and realigning them.



Adapted from Julie Thompson, IGBMC

Cluster Omega does iteration on both the guide tree and HMM alignment.

48

Multiple Alignment: History

1975 Sankoff
Formulated multiple alignment problem and gave dynamic programming solution
1988 Carrillo-Lipman
Branch and Bound approach for MSA
1990 Feng-Doolittle
Progressive alignment
1994 Thompson-Higgins-Gibson-ClustalW
Most popular multiple alignment program
1998 Morgenstern et al.-DIALIGN
Segment-based multiple alignment
2000 Notredame-Higgins-Heringa-T-coffee
Using the library of pairwise alignments
2004 MUSCLE

comparison of whole **segments** of sequences instead of comparison of **single nucleic/amino acids**. The program DiAlign constructs alignments from **gapfree pairs of similar segments** of the sequences.

What's next?

49

Multiple Alignment: History

1975 Sankoff
Formulated multiple alignment problem and gave dynamic programming solution
1988 Carrillo-Lipman
Branch and Bound approach for MSA
1990 Feng-Doolittle
Progressive alignment
1994 Thompson-Higgins-Gibson-ClustalW
Most popular multiple alignment program
1998 Morgenstern et al.-DIALIGN
Segment-based multiple alignment
2000 Notredame-Higgins-Heringa-T-coffee
Using the library of pairwise alignments
2004 MUSCLE

pre-process a data set of all pair-wise alignments between the sequences. This provides us with a library of alignment information that can be used to guide the progressive alignment.

What's next?

50

Multiple Alignment: History

1975 Sankoff
Formulated multiple alignment problem and gave dynamic programming solution
1988 Carrillo-Lipman
Branch and Bound approach for MSA
1990 Feng-Doolittle
Progressive alignment
1994 Thompson-Higgins-Gibson-ClustalW
Most popular multiple alignment program
1998 Morgenstern et al.-DIALIGN
Segment-based multiple alignment
2000 Notredame-Higgins-Heringa-T-coffee
Using the library of pairwise alignments
2004 MUSCLE

What's next?

linguaggio C++

1. Kmer distance between two sequences is defined by first collecting the set of k-mers (subsequences of length k) occuring in the two sequences, then measuring how different the two sets are.
2. re-estimates the tree using the Kimura distance, which is more accurate but requires an alignment.
3. The tree is broken into subtrees, and the sub-alignments refined.

51

Multiple Alignment: History

1975 Sankoff
Formulated multiple alignment problem and gave dynamic programming solution
1988 Carrillo-Lipman
Branch and Bound approach for MSA
1990 Feng-Doolittle
Progressive alignment
1994 Thompson-Higgins-Gibson-ClustalW
Most popular multiple alignment program
1998 Morgenstern et al.-DIALIGN
Segment-based multiple alignment
2000 Notredame-Higgins-Heringa-T-coffee
Using the library of pairwise alignments
2004 MUSCLE

What's next?

Big alignments, do they make sense?

52

Multiple Alignment: History

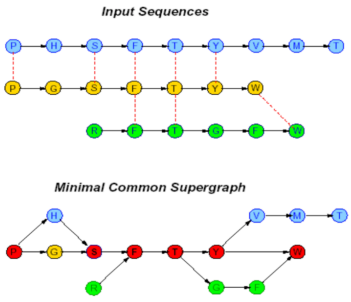
1975 Sankoff
Formulated multiple alignment problem and gave dynamic programming solution
1988 Carrillo-Lipman
Branch and Bound approach for MSA
1990 Feng-Doolittle
Progressive alignment
1994 Thompson-Higgins-Gibson-ClustalW
Most popular multiple alignment program
1998 Morgenstern et al.-DIALIGN
Segment-based multiple alignment
2000 Notredame-Higgins-Heringa-T-coffee
Using the library of pairwise alignments
2004 MUSCLE

What's next?

If k is unknown, the problem is NP-complete.

53

Solution: Representing Sequences as Paths in a Graph



Each protein sequence is represented by a path. Dashed edges connect "equivalent" positions; vertices with identical labels are fused.

54

Partial Order Multiple Alignment

Two objectives:

- Find a graph that represents domain structure
- Find mapping of each sequence to this graph

Solution

- ***PO-MSA (Partial Order Multiple Sequence Alignment)*** *– for a set of sequences S is a graph such that every sequence in S is a path in G.*

55

MAFFT

Introduction

MAFFT (Multiple Alignment using Fast Fourier Transform) is a high speed multiple sequence alignment program which implements the Fast Fourier Transform (FFT) to optimise protein alignments based on the physical properties of the amino acids. The program uses progressive alignment and iterative alignment. MAFFT is useful for hard-to-align sequences such as those containing large gaps (e.g., rRNA sequences containing variable loop regions).

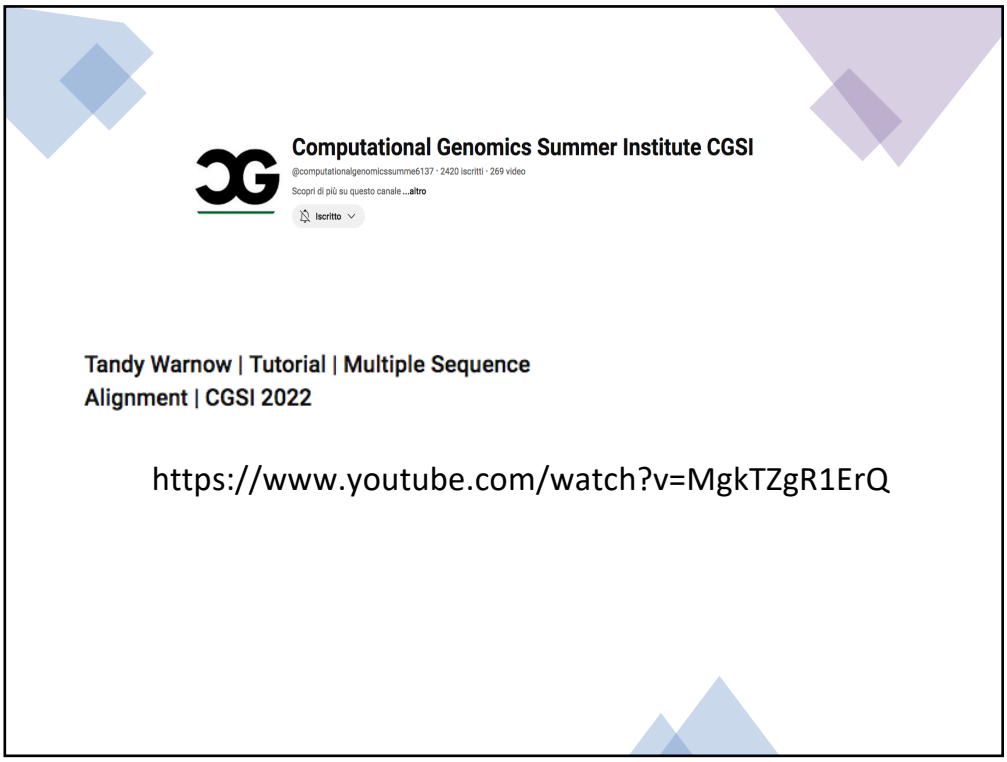
56

To ~~do~~… To see..

BWT-based aligner…

Alignment-free tecniques

57



58

Review Article

An Overview of Multiple Sequence Alignments and Cloud Computing in Bioinformatics

Jurate Daugelaite,[1] Aisling O' Driscoll,[2] and Roy D. Sleator[1]

[1] *Department of Biological Sciences, Cork Institute of Technology, Rossa Avenue, Bishopstown, Cork, Ireland*
[2] *Department of Computing, Cork Institute of Technology, Rossa Avenue, Bishopstown, Cork, Ireland*

Correspondence should be addressed to Roy D. Sleator; roy.sleator@cit.ie

Received 24 May 2013; Accepted 23 June 2013

Academic Editors: M. Glavinovic and X.-Y. Lou

Multiple sequence alignment (MSA) of DNA, RNA, and protein sequences is one of the most essential techniques in the fields of molecular biology, computational biology, and bioinformatics. Next-generation sequencing technologies are changing the biology landscape, flooding the databases with massive amounts of raw sequence data. MSA of ever-increasing sequence data sets is becoming a significant bottleneck. In order to realise the promise of MSA for large-scale sequence data sets, it is necessary for existing MSA algorithms to be run in a parallelised fashion with the sequence data distributed over a computing cluster or server farm. Combining MSA algorithms with cloud computing technologies is therefore likely to improve the speed, quality, and capability for MSA to handle large numbers of sequences. In this review, multiple sequence alignments are discussed, with a specific focus on the ClustalW and Clustal Omega algorithms. Cloud computing technologies and concepts are outlined, and the next generation of cloud base MSA algorithms is introduced.

59

Nucleic Acids Research, 1994, Vol. 22, No. 22 ***4673–4680***

CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice

Julie D.Thompson, Desmond G.Higgins+ and Toby J.Gibson*
European Molecular Biology Laboratory, Postfach 102209, Meyerhofstrasse 1, D-69012 Heidelberg, Germany

Received July 12, 1994; Revised and Accepted September 23, 1994

60

TOOLS FOR PROTEIN SCIENCE

Clustal Omega for making accurate alignments of many protein sequences

Fabian Sievers and Desmond G. Higgins

School of Medicine and Conway Institute of Biomolecular and Biomedical Research, University College Dublin, Belfield, Dublin 4, Ireland

Received 29 June 2017; Accepted 5 September 2017
DOI: 10.1002/pro.3290
Published online 7 September 2017 proteinscience.org

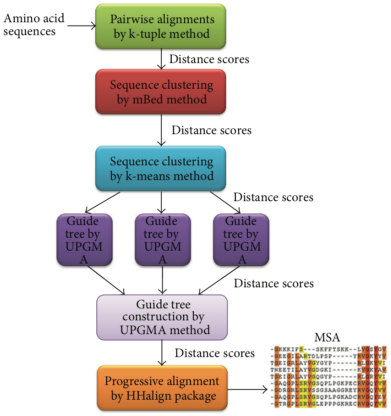


FIGURE 2: Clustal Omega algorithm, which works by taking an input of amino acid sequences, completing a pairwise alignment using the k-tuple method, sequence clustering using mBed method, and k-means method, guide tree construction using the UPGMA method, followed by a progressive alignment using HHalign package to output a multiple sequence alignment.

Unweighted Pair Group Method with Arithmetic mean

61

Why are there no pairwise scores for Clustal Omega?

Clustal Omega uses a different method to calculate the guide tree compared to ClustalW, so we do not output the rough all-against-all pairwise alignment scores used to guide the alignment (as they don't exist). However we do calculate a pairwise identity matrix from the results, which can be downloaded from the Results Summary tab.

How do I see the guide tree for Clustal Omega?

By default Clustal Omega uses a sampling method called mBed to speed up the guide tree calculations so a traditional tree is not shown. To create a phylogenetic tree go to the Phylogenetic Tree tab in the results.

Blackshields *et al. Algorithms for Molecular Biology* 2010, **5**:21
http://www.almob.org/content/5/1/21

AMB ALGORITHMS FOR MOLECULAR BIOLOGY

RESEARCH Open Access

Sequence embedding for fast construction of guide trees for multiple sequence alignment

Gordon Blackshields, Fabian Sievers*, Weifeng Shi, Andreas Wilm and Desmond G Higgins

Abstract

Background: The most widely used multiple sequence alignment methods require sequences to be clustered as an initial step. Most sequence clustering methods require a full distance matrix to be computed between all pairs of sequences. This requires memory and time proportional to N^2 for N sequences. When N grows larger than 10,000 or so, this becomes increasingly prohibitive and can form a significant barrier to carrying out very large multiple alignments.

Results: In this paper, we have tested variations on a class of embedding methods that have been designed for clustering large numbers of complex objects where the individual distance calculations are expensive. These methods involve embedding the sequences in a space where the similarities within a set of sequences can be closely approximated without having to compute all pair-wise distances.

Conclusions: We show how this approach greatly reduces computation time and memory requirements for clustering large numbers of sequences and demonstrate the quality of the clusterings by benchmarking them as guide trees for multiple alignment. Source code is available for download from http://www.clustal.org/mbed.tgz

62

THE EFFECT OF THE GUIDE TREE ON MULTIPLE SEQUENCE ALIGNMENTS AND SUBSEQUENT PHYLOGENETIC ANALYSES

S. NELESEN, K. LIU, D. ZHAO, C. R. LINDER, AND T. WARNOW *
The University of Texas at Austin
Austin, TX 78712
E-mail: {serita,kliu,wzhao,tandy}@cs.utexas.edu, rlinder@mail.utexas.edu

analizza quanto l'albero guidato iniziale possa influenza l'allineamento finale

63

Online Magazine of the European Molecular Biology Laboratory

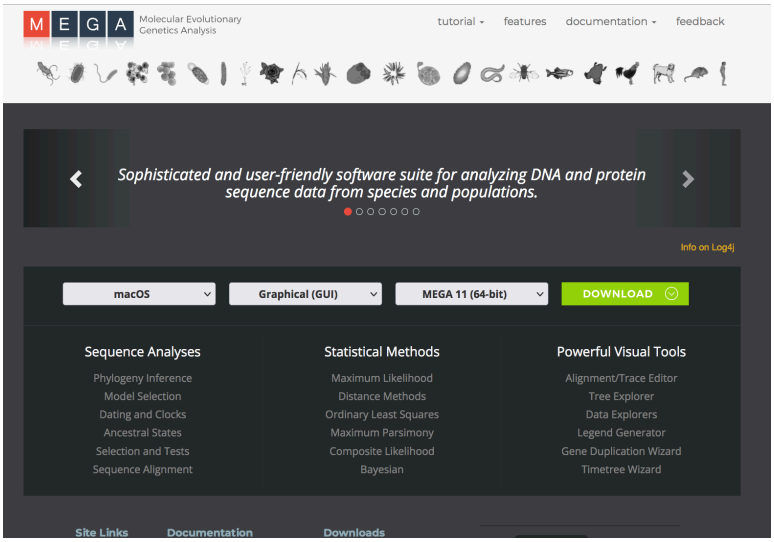
The story of Clustal: democratising sequence alignments

"I figured out how to make multiple alignments work on these tiny little computers," said Higgins. "It meant that now anyone could make their own multiple alignments in their offices."

https://www.embl.org/news/embletc/issue-100/the-story-of-clustal-democratising-sequence-alignments/

"Omega is the last letter of the Greek alphabet," said Higgins, adding that many new programs for multiple alignment have come up in recent years, including MAFFT, also hosted by EMBL-EBI. "Life goes on," he added philosophically.

64



65

EMBL-EBI

https://www.ebi.ac.uk/seqdb/confluence/display/JDSAT/Multiple+Sequence+Alignment

Clustal Omega

Multiple sequence alignment of DNA or protein sequences. Clustal Omega replaces the older ClustalW alignment tools.

HMMER

Fast sensitive protein homology searches using profile hidden Markov models (HMMs) for querying against both sequence and HMM target databases.

abbiamo usato il tool con il file contenente 4 proteine della ferritina (vedi allegato)

66