Chapter 2: Sequence Alignment

2.4 Multiple Sequences Alignment (MSA)

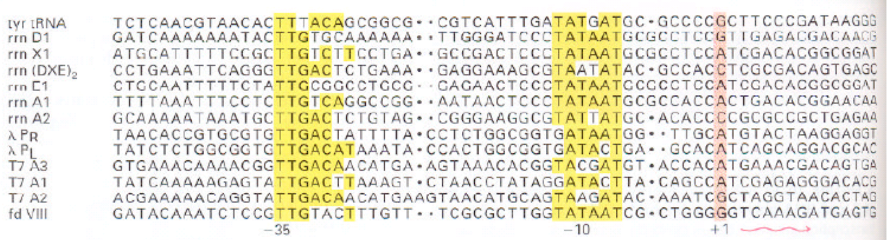
Prof. Yechiam Yemini (YY)
Computer Science Department
Columbia University

Overview

- Introduction to MSA and its applications
- Multiple Sequence Alignments (MSA) techniques
- Progressive alignment, CLUSTAL W
- Iterative alignment
- Emerging techniques

MSA Computes Conserved Patterns

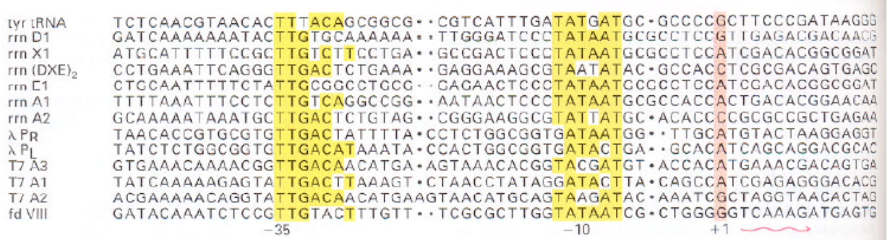
- Challenge: find evolutionary related sequences
- Numerous applications:
  - Phylogenetic analysis: discovering evolutionary relatedness
  - Discovering motifs in DNA and proteins
  - Predicting protein secondary structure through homology
  - Designing oligonucleotide probes for microarrays
  - Designing restriction enzymes for gene cloning



```
tyr tRNA     TCTCAACGTAACACTTTACAGCGGCG••CGTCATTTGATATGATGC•GCCCCGCTTCCCGATAAGGG
rrn D1       GATCAAAAAAATACTTGTGCAAAAAA••TTGGGATCCCTATAATGCGCCTCCGTTGAGACGACAACG
rrn X1       ATGCATTTTTCCGCTTGTCTTCCTGA••GCCGACTCCCTATAATGCGCCTCCATCGACACGGCGGAT
rrn (DXE)_2  CCTGAAATTCAGGGTTGACTCTGAAA••GAGGAAAGCGTAATATAC•GCCACCTCGCGACAGTGAGC
rrn E1       CTGCAATTTTTCTATTGCGGCCTGCG••GAGAACTCCCTATAATGCCCCTCCATCGACACGGCGGAT
rrn A1       TTTTAAATTTCCTCTTGTCAGGCCGG••AATAACTCCCTATAATGCGCCACCACTGACACGGAACAA
rrn A2       GCAAAAATAAATGCTTGACTCTGTAG••CGGGAAGGCGTATTATGC•ACACCCCGCGCCGCTGAGAA
λ P_R        TAACACCGTGCGTGTTGACTATTTTA•CCTCTGGCGGTGATAATGG••TTGCATGTACTAAGGAGGT
λ P_L        TATCTCTGGCGGTGTTGACATAAATA•CCACTGGCGGTGATACTGA••GCACATCAGCAGGACGCAC
T7 A3        GTGAAACAAAACGGTTGACAACATGA•AGTAAACACGGTACGATGT•ACCACATGAAACGACAGTGA
T7 A1        TATCAAAAAGAGTATTGACTTAAAGT•CTAACCTATAGGATACTTA•CAGCCATCGAGAGGGACACG
T7 A2        ACGAAAAACAGGTATTGACAACATGAAGTAACATGCAGTAAGATAC•AAATCGCTAGGTAACACTAG
fd VIII      GATACAAATCTCCGTTGTACTTTGTT••TCGCGCTTGGTATAATCG•CTGGGGGTCAAAGATGAGTG
                           −35                        −10          +1
```

Grand Challenge: How Does Regulation Work?

- Regulatory motifs provide an important starting point
- How do we discover and identify regulatory motifs?
  - Align upstream regions of related genes
  - Identify conserved patterns
- How do we correlate this with transcription factors?



```
tyr tRNA     TCTCAACGTAACACTTTACAGCGGCG••CGTCATTTGATATGATGC•GCCCCGCTTCCCGATAAGGG
rrn D1       GATCAAAAAAATACTTGTGCAAAAAA••TTGGGATCCCTATAATGCGCCTCCGTTGAGACGACAACG
rrn X1       ATGCATTTTTCCGCTTGTCTTCCTGA••GCCGACTCCCTATAATGCGCCTCCATCGACACGGCGGAT
rrn (DXE)_2  CCTGAAATTCAGGGTTGACTCTGAAA••GAGGAAAGCGTAATATAC•GCCACCTCGCGACAGTGAGC
rrn E1       CTGCAATTTTTCTATTGCGGCCTGCG••GAGAACTCCCTATAATGCCCCTCCATCGACACGGCGGAT
rrn A1       TTTTAAATTTCCTCTTGTCAGGCCGG••AATAACTCCCTATAATGCGCCACCACTGACACGGAACAA
rrn A2       GCAAAAATAAATGCTTGACTCTGTAG••CGGGAAGGCGTATTATGC•ACACCCCGCGCCGCTGAGAA
λ P_R        TAACACCGTGCGTGTTGACTATTTTA•CCTCTGGCGGTGATAATGG••TTGCATGTACTAAGGAGGT
λ P_L        TATCTCTGGCGGTGTTGACATAAATA•CCACTGGCGGTGATACTGA••GCACATCAGCAGGACGCAC
T7 A3        GTGAAACAAAACGGTTGACAACATGA•AGTAAACACGGTACGATGT•ACCACATGAAACGACAGTGA
T7 A1        TATCAAAAAGAGTATTGACTTAAAGT•CTAACCTATAGGATACTTA•CAGCCATCGAGAGGGACACG
T7 A2        ACGAAAAACAGGTATTGACAACATGAAGTAACATGCAGTAAGATAC•AAATCGCTAGGTAACACTAG
fd VIII      GATACAAATCTCCGTTGTACTTTGTT••TCGCGCTTGGTATAATCG•CTGGGGGTCAAAGATGAGTG
                           −35                        −10          +1
```

Discovering Motifs

- Motif: characteristic pattern of a family
  - Regulatory motif
  - Protein motif; e.g., active site
- Using consensus sequences to describe motifs: map position ➔ likely letter
  - E.g., TATA box

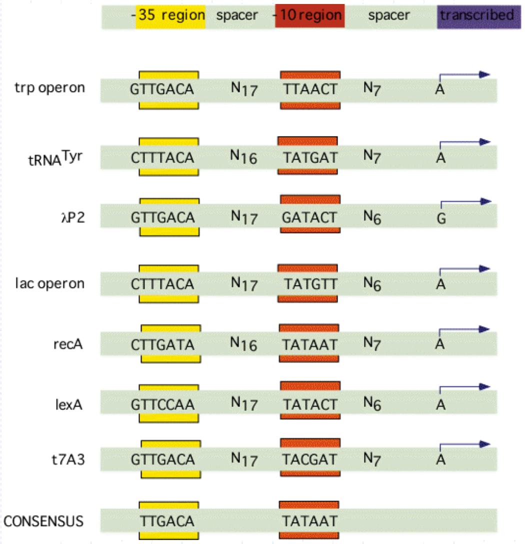
Segments at -10

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

A	0	10	1	5	9	0
C	0	0	1	0	1	0
G	2	0	0	2	0	0
T	8	0	8	3	0	10

Consensus ➔ T A T A A T

Example



trp operon	GTTGACA	N_{17}	TTAACT	N_7	A
tRNA^{Tyr}	CTTTACA	N_{16}	TATGAT	N_7	A
λP2	GTTGACA	N_{17}	GATACT	N_6	G
lac operon	CTTTACA	N_{17}	TATGTT	N_6	A
recA	CTTGATA	N_{16}	TATAAT	N_7	A
lexA	GTTCCAA	N_{17}	TATACT	N_6	A
t7A3	GTTGACA	N_{17}	TACGAT	N_7	A
CONSENSUS	TTGACA		TATAAT		

Example: Cellulose Binding Domain (CBD-CBH1)

```
                                 1           2         3
                          45678901...23456789012345678901 2
GUX1_TRIRE/481-509        HYGQCGGI...GYSGPTVCASGTTCQVLNPYY
GUN1_TRIRE/427-455        HWGQCGGI...GYSGCKTCTSGTTCQYSNDYY
GUX1_PHACH/484-512        QWGQCGGI...GYTGSTTCASPYTCHVLNPYY
GUN2_TRIRE/25-53          VWGQCGGI...GWSGPTNCAPGSACSTLNPYY
GUX2_TRIRE/30-58          VWGQCGGQ...NWSGPTCCASGSTCVYSNDYY
GUN5_TRIRE/209-237        LYGQCGGA...GWTGPTTCQAPGTCKVQNQWY
GUNF_FUSOX/21-49          IWGQCGGN...GWTGATTCASGLKCEKINDWY
GUX3_AGABI/24-52          VWGQCGGN...GWTGPTTCASGSTCVKQNDFY
GUX1_PENJA/505-533        DWAQCGGN...GWTGPTTCVSPYTCTKQNDWY
GUXC_FUSOX/482-510        QWGQCGGQ...NYSGPTTCKSPFTCKKINDFY
GUX1_HUMGR/493-521        RWQQCGGI...GFTGPTQCEEPYICTKLNDWY
GUX1_NEUCR/484-512        HWAQCGGI...GFSGPTTCPEPYTCAKDHDIY
PSBP_PORPU/26-54          LYEQCGGI...GFDGVTCCSEGLMCMKMGPYY
GUNB_FUSOX/29-57          VWAQCGGQ...NWSGTPCCTSGNKCVKLNDFY
PSBP_PORPU/69-97          PYGQCGGM...NYSGKTMCSPGFKCVELNEFF
GUNK_FUSOX/339-370        AYYQCGGSKSAYPNGNLACATGSKCVKQNEYY
PSBP_PORPU/172-200        RYAQCGGM...GYMGSTMCVGGYKCMAISEGS
 PSBP_PORPU/128-156       EYAACGGE...MFMGAKCCKFGLVCYETSGKW

Consensus (14/18)         ...QCGG.......G...C.....C.......
```

The MSA Problem

- How do we best align multiple sequences?
- Define the problem:
  - Given sequences $X_1,X_2…X_k$
  - Find extensions $X'_1,X'_2…X'_k$; (X' is X augmented with indels)
  - Such that some measure of "relatedness" $S(X'_1,X'_2…X'_k)$ is optimized

```
VTISCTGSSSNIGAG-NHVKWYQQLPG
VTISCTGTSSNIGS--ITVNWYQQLPG
LRLSCSSSGFIFSS--YAMYWVRQAPG
LSLTCTVSGTSFDD--YYSTWVRQPPG
PEVTCVVVDVSHEDPQVKFNWYVDG--
ATLVCLISDFYPGA--VTVAWKADS--
AALGCLVKDYFPEP--VTVSWNSG---
VSLTCLVKGFYPSD--IAVEWESNG--
```

How do we know it is not D--G?

Immunoglobins fragments

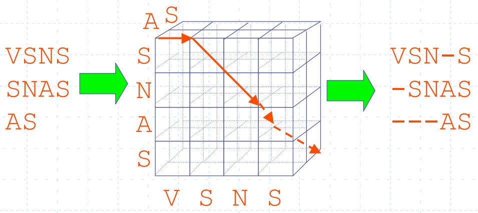
Distant sequences may have great impact

Can Pair Alignment Be Generalized To MSA?

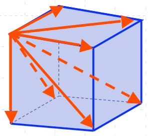
- Generalizing DP: modeling evolutionary edits as a grid path
- Generalizing scoring: measures of evolutionary relatedness
- Generalizing FASTA/BLAST: grow alignment from seeds
  - Finding good high-scoring diagonal seeds
  - Using seeds to grow an alignment
- ????

Problem 1: Can DP Be Generalized?

- Consider k=3 sequences; alignment may represented in terms of a 3-D grid:



2^3-1=7 neighbors can provide optimal paths

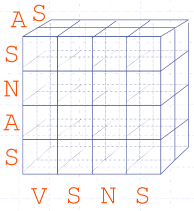


- The DP algorithm generalizes to:

F(i,j,k)=MAX [F(i-1,j-1,k-1)+S(X_i,Y_j,Z_k), F(i-1,j-1,k)+S(Xi,Yj,-),
F(i-1,j,k-1)+S(Xi,-,Zk), F(i-1,j,k)+S(Xi,-,-),
F(i,j-1,k-1)+S(-,Yj,Zk), F(i,j,k-1)+ S(-,-,Zk),
F(i,j-1,k)+S(-,Yj,-)]

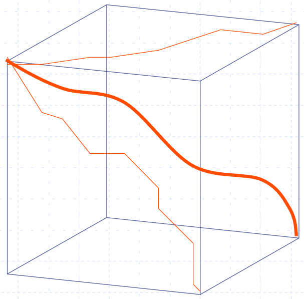
Complexity of Multi Dimensional DP

- Consider n sequences of length L
- How many cells need to be traversed by DP?
  - Space complexity: L^n
- How many computations occur at a cell?
  - Need to evaluate 2^n-1 neighbors
- Time complexity is $O(2^nL^n)$
- Is there an efficient algorithm to find optimal routes on a hypercube?
  - NP Complete problem



Note: Can BLAST/FASTA Be Generalized

- Generalize the filtering of low-scoring diagonals to reduce DP search [Carillo-Lipman 88+]
- Key idea: filter pairwise alignments by bounding their score
  - Every MSA projects into pair-wise alignments
  - Can bound the score of these projections
- Works well for small sequences



Problem 2: How To Generalize Scoring?

- Scoring pair alignment s($\underline{X}$,$\underline{Y}$) is based on assumptions:

```
LSLTCTVSGTSFDD--YYSTWVRQPPG
PEVTCVVVDVSHEDPQVKFNWYVDG--
```

- Column independence
  - Evolution changes sequence positions (columns) independently
  - Therefore, the score is sum of column scores
- Markovian edits
  - A given position is edited by a Markovian process
  - The score represents the log-likelihood of edits
- How good are these assumptions?
- How do we score MSA?

```
VTISCTGSSSNIGAG-NHVKWYQQLPG
VTISCTGTSSNIGS--ITVNWYQQLPG
LRLSCSSSGFIFSS--YAMYWVRQAPG
LSLTCTVSGTSFDD--YYSTWVRQPPG
PEVTCVVVDVSHEDPQVKFNWYVDG--
ATLVCLISDFYPGA--VTVAWKADS--
AALGCLVKDYFPEP--VTVSWNSG---
VSLTCLVKGFYPSD--IAVEWESNG--
```

Generalizing Scoring

- Score measures evolutionary "relatedness"

```
LSLTCTVSGTSFDD--YYSTWVRQPPG
PEVTCVVVDVSHEDPQVKFNWYVDG--
```

- Generalize scoring to columns
  - Sum of Pairs (SP): For a column **C**, S(**C**)=$\Sigma_{x,y\varepsilon \mathbf{C}}$ s(x,y)
  - E.g., s(x,y)= 1 for match; -1 for substitution/gap: S(C6)=-3-4-5-4-3+2+1=-16 S(C15)=-7-1-1-1-3=-13
- Finding max-SP alignment is NP complete
  - Are there good heuristics?
  - Other column scoring ideas?

```
VTISCTGSSSNIGAG-NHVKWYQQLPG
VTISCTGTSSNIGS--ITVNWYQQLPG
LRLSCSSSGFIFSS--YAMYWVRQAPG
LSLTCTVSGTSFDD--YYSTWVRQPPG
PEVTCVVVDVSHEDPQVKFNWYVDG--
ATLVCLISDFYPGA--VTVAWKADS--
AALGCLVKDYFPEP--VTVSWNSG---
VSLTCLVKGFYPSD--IAVEWESNG--
```

SP=-16 SP=-7

- Generalize scoring to evolutionary tree (phylogeny)

Progressive & Iterative Heuristics

- Progressive MSA [Doolittle & Feng 87]

  ==> CLUSTAL [Gibson, Higgins, Thompson 94+]

  - Given sequences $X_1 \ldots X_n$ to be aligned
  - Compute the pair-wise alignment of each pair X_k, X_m
  - Merge pair-wise alignments to create MSA

- Iterative MSA [Barton Sternberg 87]
  - Select the highest scoring pair-wise alignment to compute initial profile
  - Find a sequence that is most similar to the profile and align with profile. Repeat this until all sequences are included in MSA.
  - Iterate the following process until convergence: select a sequence X_k and align it against the profile of the other sequences.

Progressive Alignment

- Step 1: find all pair alignments
  - E.g., consider

```
ACG CGA GAC
```

  - Compute all PAs:

```
ACG-  -ACG  CGA-
-CGA  GAC-  -GAC
 1     2     3
```

- Step 2: merge alignments by adding gaps
  - E.g., merge 1⇔2, 1⇔3

```
-ACG-
--CGA
GAC--
```

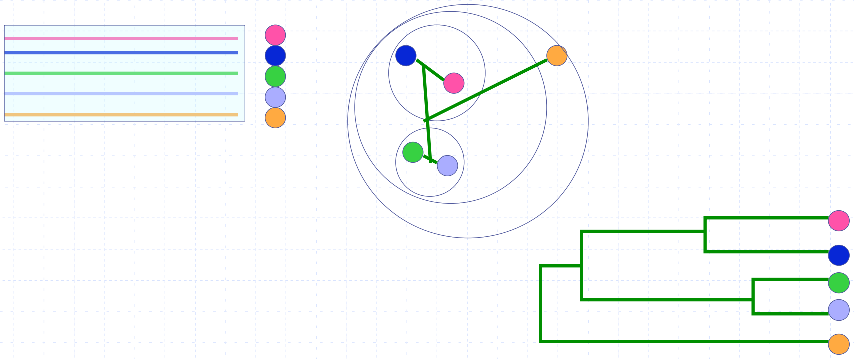
  - E.g merge 2⇔3, 1⇔2

```
ACG--
-CGA-
--GAC
```

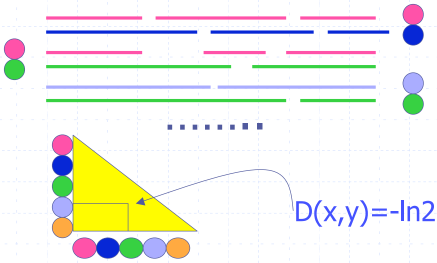
- Ambiguity: MSA depends on merging order

Using A Guide Tree To Order Alignments

- Key idea: merging order should reflect similarity
- Merging order may be described as a tree to cluster sequences by similarity



ClustalW: Order Merging by Similarity



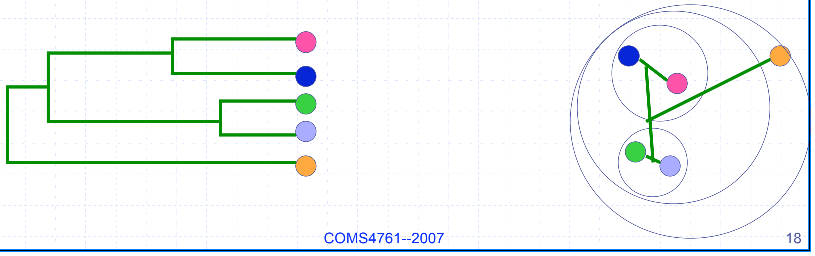
1. Align pairs (full DP, or faster)

2. Convert scores to distances

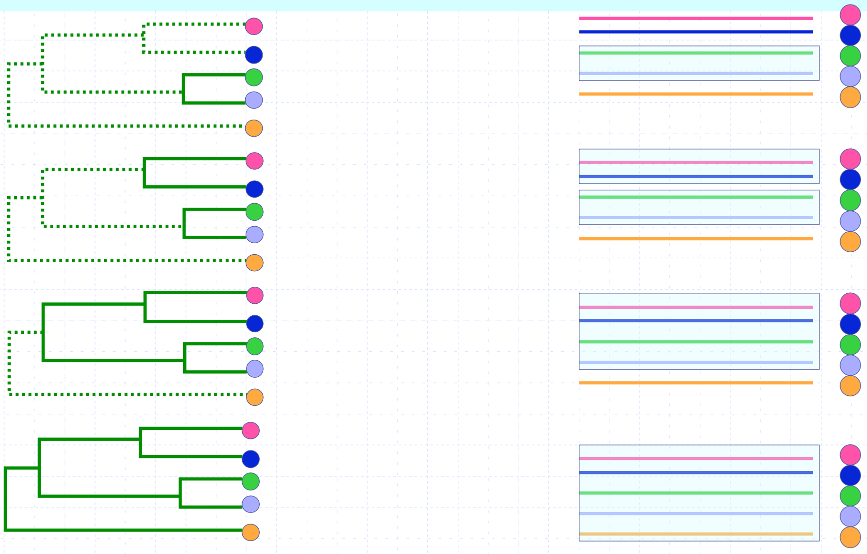
$$D(x,y)=-\ln200*\left(\frac{S(x,y)-Sr(x,y)}{S(x,x)+S(y,y)-2Sr(x,y)}\right)$$

Sr(x,y) is the average score of random shuffling

3. Construct a clustering tree



Step 4: Merge by Similarity Order



Introduction To Clustering Trees

- Guide tree approximates phylogenetic tree
  - We will study phylogeny in later classes
  - Here we pursue a basic introduction to trees
- Pair Group Method using Arithmetic Mean (PGMA)
  - Idea: create a parent of two closest nodes
  - If w is parent of u,v; D(w,x) is computed from D(u,x), D(v,x)
- Unweighted PGMA (UPGMA)

  D(w,x)= a(u)D(u,x)+b(v)D(v,x)
  a(u)=m(u)/[m(u)+m(v)] where m(u)=#leaves under u
- Weighted PGMA (WPGMA)

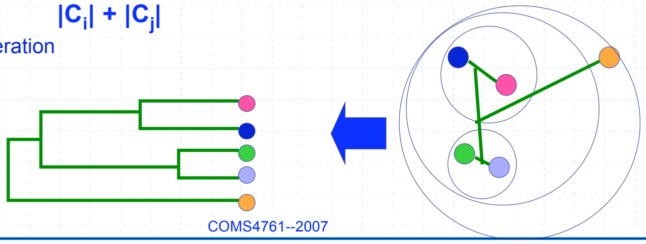
  D(w,x)= 0.5*D(u,x)+0.5*D(v,x)

UPGMA Algorithm

- Initialization
  - Initialize n clusters $C_i=\{S_i\}$
  - Initialize T with leaves for each cluster Ci
- Iteration
  - Find C_i, C_j with smallest distance D_{ij}
  - Create new cluster $C_k = C_i \cup C_j$
  - Add a new node to T, for C_k, and connect it to C_i,C_j
  - If all nodes are connected to a tree exit; otherwise, assign $D_{ki}=D_{kj}=D_{ij}/2$ and compute the distances D_{kl} to all clusters C_l

$$D_{kl} = \frac{D_{il}\,|C_i| + D_{jl}\,|C_j|}{|C_i| + |C_j|}$$

  - Repeat the iteration



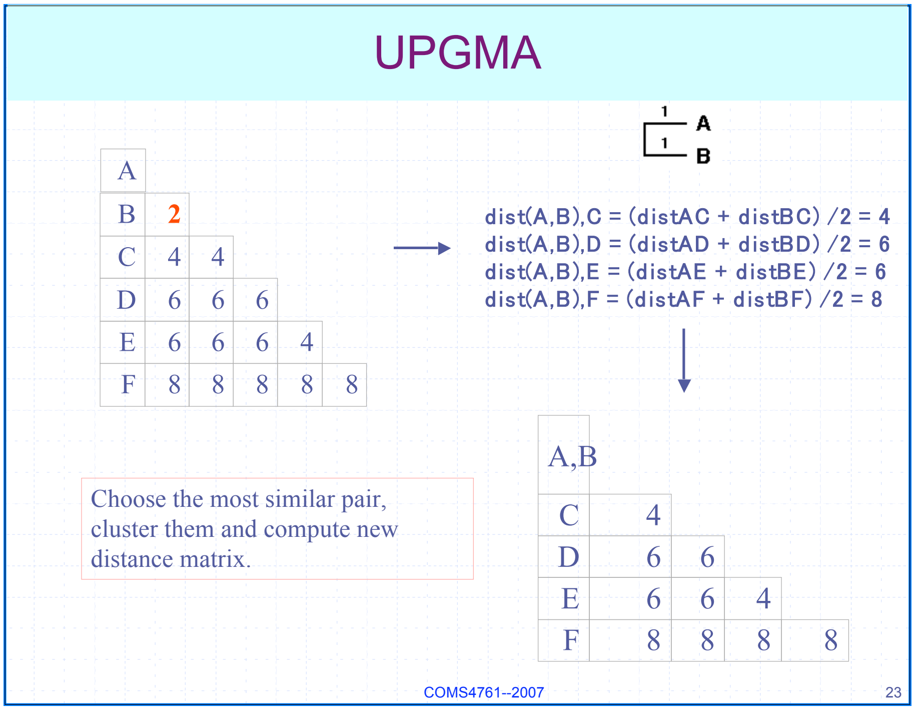
Example: Constructing A UPGMA Tree

```
A - GCTTGTCCGTTACGAT
B - ACTTGTCTGTTACGAT
C - ACTTGTCCGAAACGAT
D - ACTTGACCGTTTCCTT
E - AGATGACCGTTTCGAT
F - ACTACACCCTTATGAG
```

→

A					
B	2				
C	4	4			
D	6	6	6		
E	6	6	6	4	
F	8	8	8	8	8

From http://www.icp.ucl.ac.be/~opperd/private/upgma.html



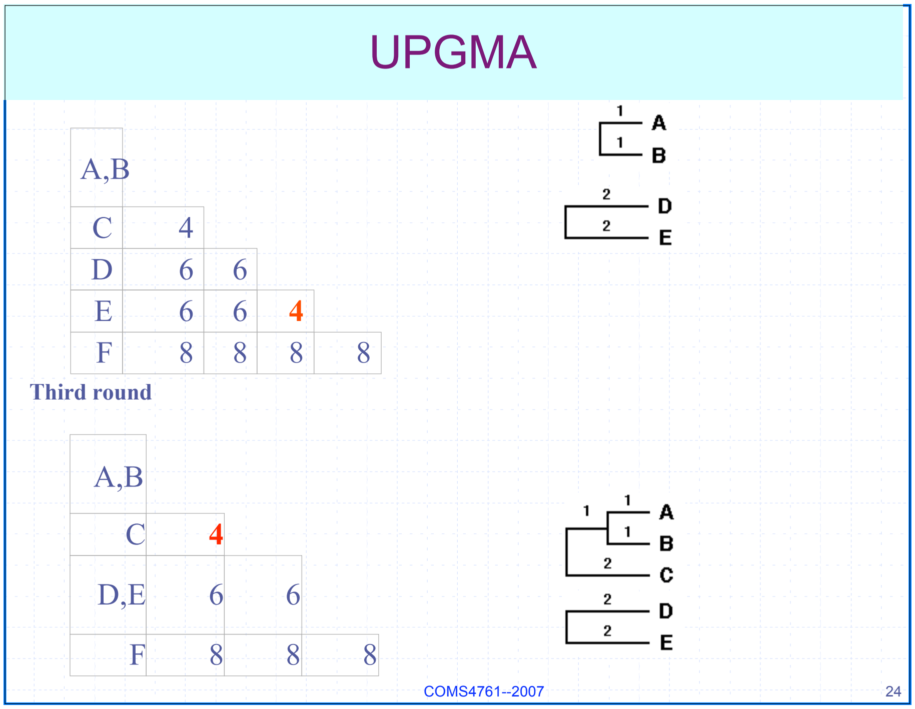
UPGMA

A					
B	2				
C	4	4			
D	6	6	6		
E	6	6	6	4	
F	8	8	8	8	8

dist(A,B),C = (distAC + distBC) /2 = 4
dist(A,B),D = (distAD + distBD) /2 = 6
dist(A,B),E = (distAE + distBE) /2 = 6
dist(A,B),F = (distAF + distBF) /2 = 8

A,B				
C	4			
D	6	6		
E	6	6	4	
F	8	8	8	8

Choose the most similar pair, cluster them and compute new distance matrix.



UPGMA

A,B				
C	4			
D	6	6		
E	6	6	4	
F	8	8	8	8

1
1
A
B
2
2
D
E

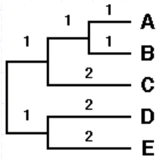
Third round

A,B			
C	4		
D,E	6	6	
F	8	8	8

1
1
1
A
B
2
C
2
2
D
E

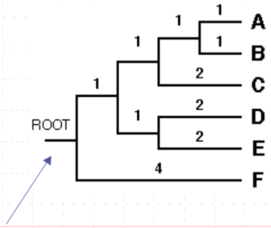
UPGMA

AB,C		
D,E	**6**	
F	8	8



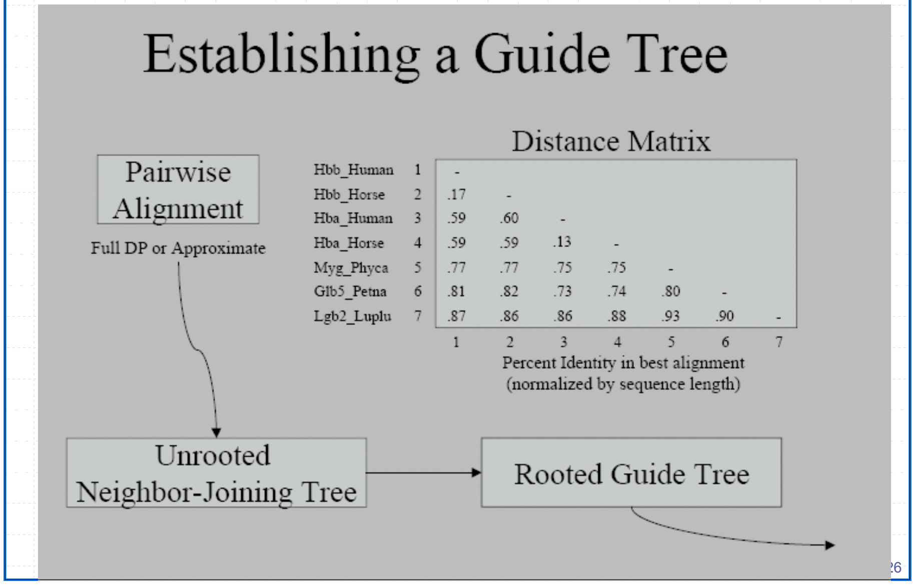
Fifth round

ABC,DE	
F	**8**



Note the this method identifies the root of the tree.

Example: MSA Of Globins



Establishing a Guide Tree

Pairwise Alignment

Full DP or Approximate

Distance Matrix

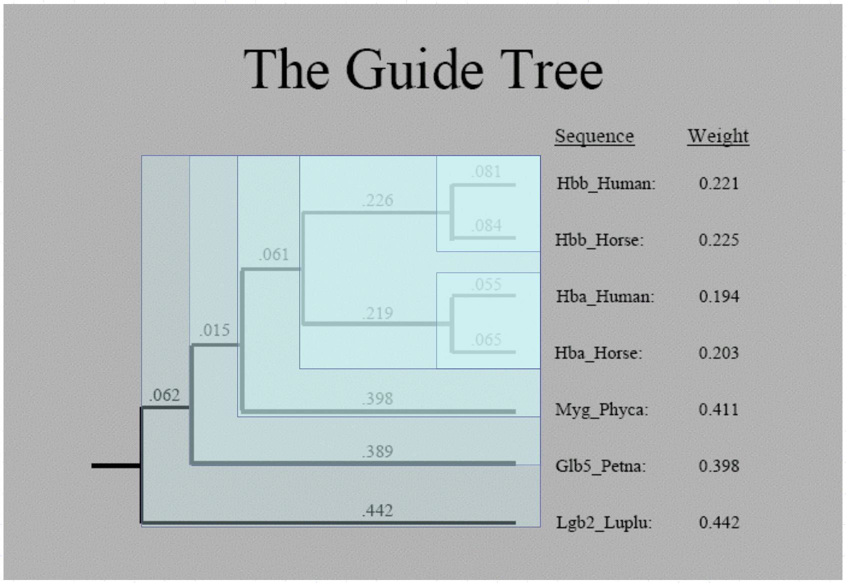
		1	2	3	4	5	6	7
Hbb_Human	1	-						
Hbb_Horse	2	.17	-					
Hba_Human	3	.59	.60	-				
Hba_Horse	4	.59	.59	.13	-			
Myg_Phyca	5	.77	.77	.75	.75	-		
Glb5_Petna	6	.81	.82	.73	.74	.80	-	
Lgb2_Luplu	7	.87	.86	.86	.88	.93	.90	-

Percent Identity in best alignment (normalized by sequence length)

Unrooted Neighbor-Joining Tree

Rooted Guide Tree

Clustering Into a Similarity Tree



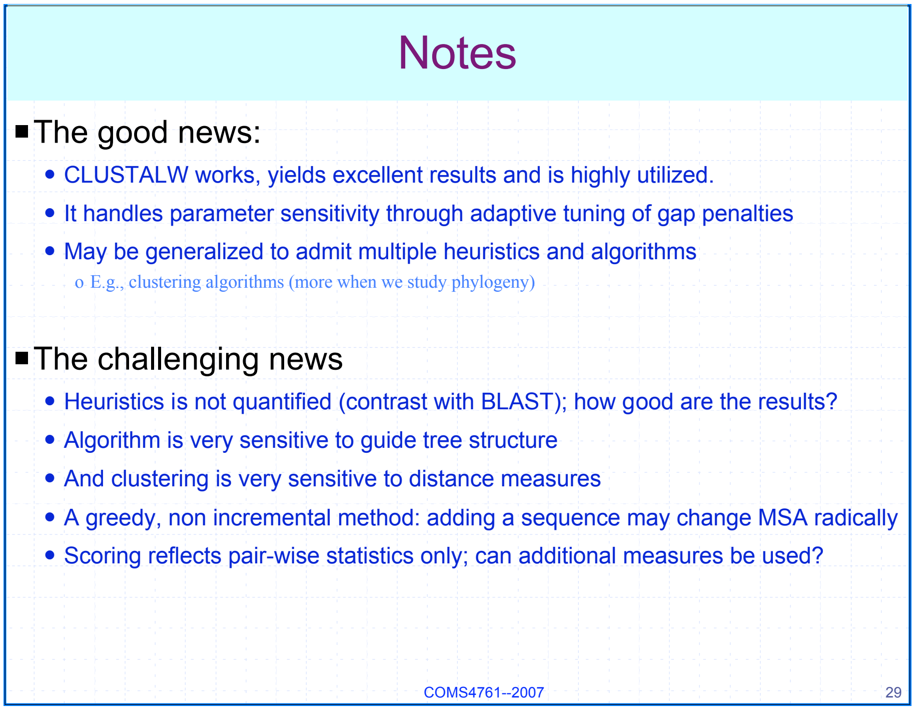
The Guide Tree

Sequence	Weight
Hbb_Human:	0.221
Hbb_Horse:	0.225
Hba_Human:	0.194
Hba_Horse:	0.203
Myg_Phyca:	0.411
Glb5_Petna:	0.398
Lgb2_Luplu:	0.442

Notes

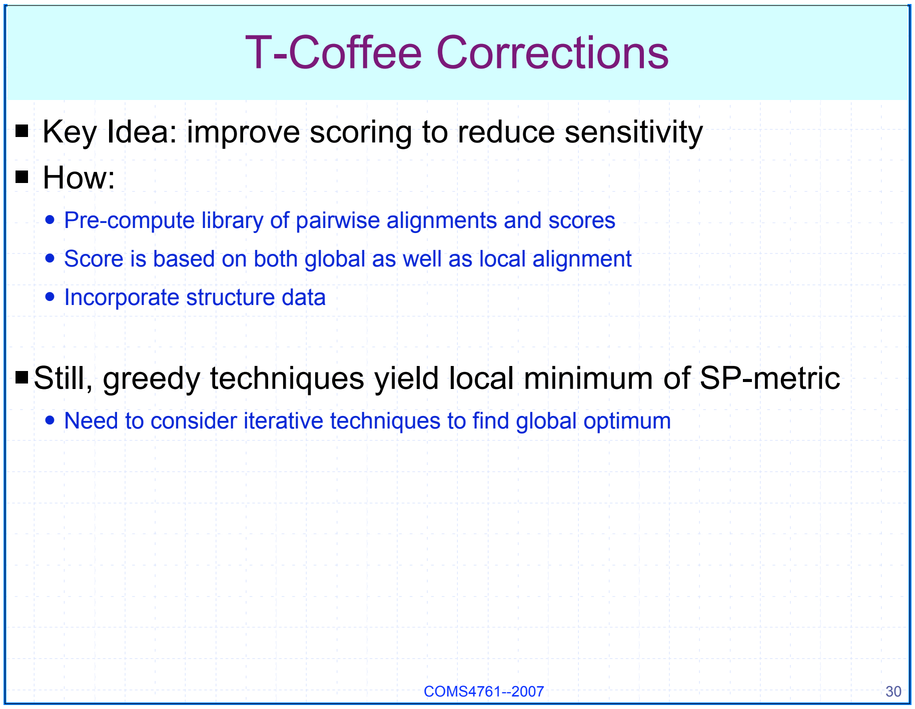
- Guide tree seeks to approximate evolutionary order
  - Merge alignments according to reflect evolutionary proximity
  - (We will consider alternatives when we study phylogenetics)
- There are two fundamental problems:
  - Local Minimum: this greedy algorithm may not optimize sum-of-pairs metric
  - Sensitivity: the result can be very sensitive to scoring; particularly to gap penalties

```
VTISCTGSSSNIGAG-NHVKWYQQLPG
VTISCTGTSSNIGS--ITVNWYQQLPG
LRLSCSSSGFIFSS--YAMYWVRQAPG
LSLTCTVSGTSFDD--YYSTWVRQPPG
PEVTCVVVDVSHEDPQVKFNWYVDG--
ATLVCLISDFYPGA--VTVAWKADS--
AALGCLVKDYFPEP--VTVSWNSG---
VSLTCLVKGFYPSD--IAVEWESNG--
```



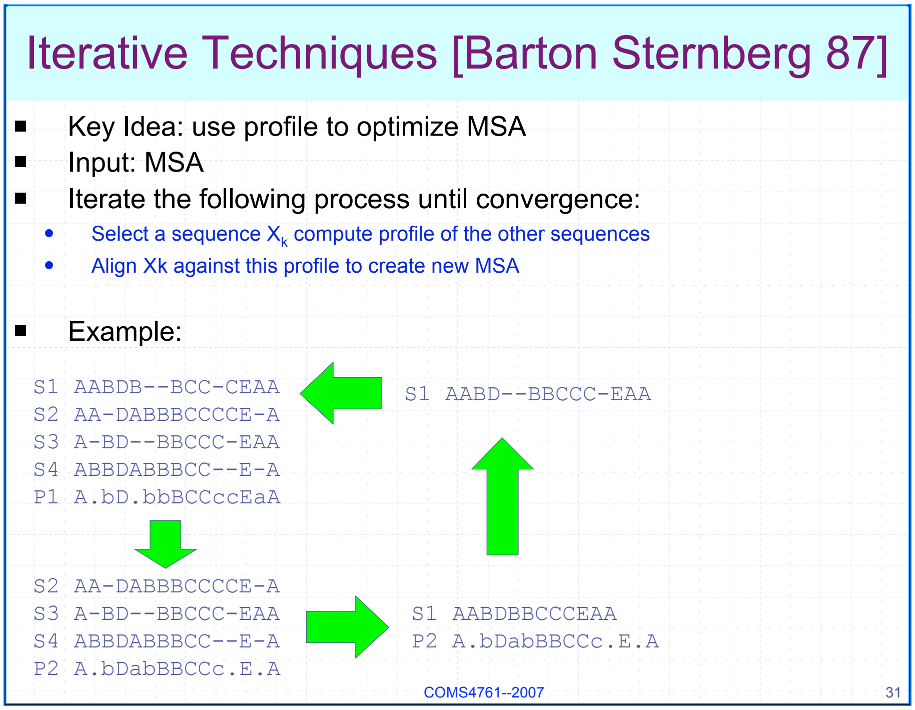
Notes

- The good news:
  - CLUSTALW works, yields excellent results and is highly utilized.
  - It handles parameter sensitivity through adaptive tuning of gap penalties
  - May be generalized to admit multiple heuristics and algorithms
    - E.g., clustering algorithms (more when we study phylogeny)
- The challenging news
  - Heuristics is not quantified (contrast with BLAST); how good are the results?
  - Algorithm is very sensitive to guide tree structure
  - And clustering is very sensitive to distance measures
  - A greedy, non incremental method: adding a sequence may change MSA radically
  - Scoring reflects pair-wise statistics only; can additional measures be used?



T-Coffee Corrections

- Key Idea: improve scoring to reduce sensitivity
- How:
  - Pre-compute library of pairwise alignments and scores
  - Score is based on both global as well as local alignment
  - Incorporate structure data
- Still, greedy techniques yield local minimum of SP-metric
  - Need to consider iterative techniques to find global optimum



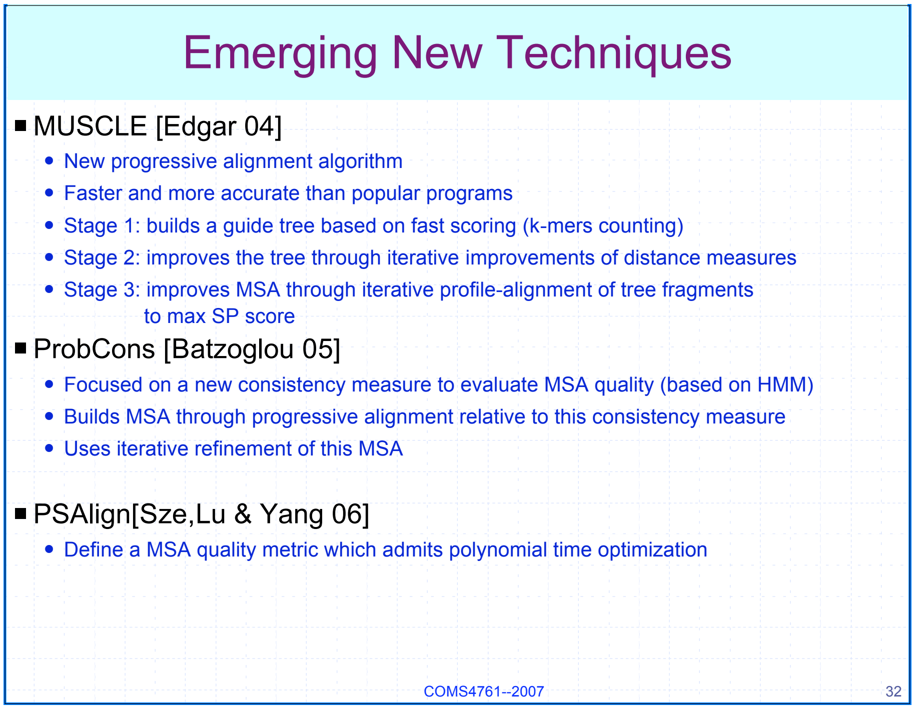
Iterative Techniques [Barton Sternberg 87]

- Key Idea: use profile to optimize MSA
- Input: MSA
- Iterate the following process until convergence:
  - Select a sequence X_k compute profile of the other sequences
  - Align Xk against this profile to create new MSA
- Example:

```
S1 AABDB--BCC-CEAA          S1 AABD--BBCCC-EAA
S2 AA-DABBBCCCCE-A
S3 A-BD--BBCCC-EAA
S4 ABBDABBBCC--E-A
P1 A.bD.bbBCCccEaA

S2 AA-DABBBCCCCE-A
S3 A-BD--BBCCC-EAA           S1 AABDBBCCCEAA
S4 ABBDABBBCC--E-A           P2 A.bDabBBCCc.E.A
P2 A.bDabBBCCc.E.A
```

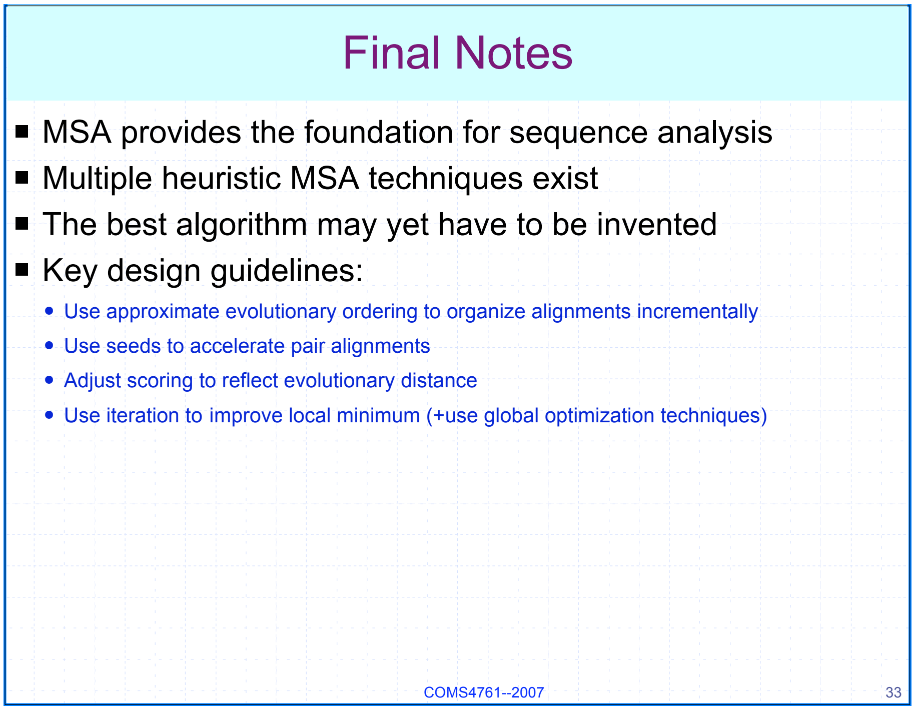
COMS4761--2007 31



Emerging New Techniques

- MUSCLE [Edgar 04]
  - New progressive alignment algorithm
  - Faster and more accurate than popular programs
  - Stage 1: builds a guide tree based on fast scoring (k-mers counting)
  - Stage 2: improves the tree through iterative improvements of distance measures
  - Stage 3: improves MSA through iterative profile-alignment of tree fragments to max SP score
- ProbCons [Batzoglou 05]
  - Focused on a new consistency measure to evaluate MSA quality (based on HMM)
  - Builds MSA through progressive alignment relative to this consistency measure
  - Uses iterative refinement of this MSA
- PSAlign[Sze,Lu & Yang 06]
  - Define a MSA quality metric which admits polynomial time optimization

COMS4761--2007 32



Final Notes

- MSA provides the foundation for sequence analysis
- Multiple heuristic MSA techniques exist
- The best algorithm may yet have to be invented
- Key design guidelines:
    - Use approximate evolutionary ordering to organize alignments incrementally
    - Use seeds to accelerate pair alignments
    - Adjust scoring to reflect evolutionary distance
    - Use iteration to improve local minimum (+use global optimization techniques)