

Multiple Sequence Alignment

BMI/CS 576

www.biostat.wisc.edu/bmi576.html

Colin Dewey

cdewey@biostat.wisc.edu

Fall 2011

Multiple Sequence Alignment: Task Definition

- Given
 - a set of more than 2 sequences
 - a method for scoring an alignment
- Do:
 - determine the correspondences between the sequences such that the alignment score is maximized

Multiple Alignment of SH3 Domain

```
GGWWRGdy.ggkkqLWFP SN YV
IGWLNgyne.tgkerGGDFP GT YV
PNWWEgql..nnrrrGIFP SN YV
DEWWQAr r..deqqiGIVP SK --
GEWWKAqs...tgqqeGFI P FN FV
GDWWLARs...sgqqrGGYIP SN YV
GDWWDAel...kgrrrGKVP SN YL
-DWWEArsls.sghrrGYVP SN YV
GDWWYArslitnseGYIP ST YV
GEWWKArslatrkeGYIP SN YV
GDWWLARsylvtgreGYVP SN FV
GEWWKAksls.skreGFI P SN YV
GEWCEAqt.kngq.GWVP SN YI
SDWWRVvnl.ttrqqeGLIPLN FV
LPWWRARd.kngqqeGYIP SN YI
RDWWEFRskt.vytpGGYI ES GYV
EHWWKVkd.a.lgnvGGYIP SN YV
IHWWRVqd.r.nghheGYVP SS YL
KDWWKVe v..ndrqqGFP AA YV
VGWMPGln.e.rtrqrGGDFP GT YV
PDWWEGel...ngqrrGVFP AS YV
ENWWNGEci...gnrkGIFP AT YV
EEWLEGEc...kgtkvGIFP KV FV
GGWWKGdy.g.triqQYFP SN YV
DGWWRGsy...ngqvGWFP SN YV
QGWWRGeli...ygrvGWFP AN YV
GRWWKAr r..angetGIIP SN YV
GGWTOGeli.k.sgqkGWAPT N YL
GDWWEAr sn.tgenGYIP SN YV
NDWWTGrt..n.gkeGIFP AN YV
```

Figure from A. Krogh, An Introduction to Hidden Markov Models for Biological Sequences

Motivation for MSA

- establish input data for phylogenetic analyses
- determine evolutionary history of a set of sequences
 - At what point in history did certain mutations occur?
- discovering a common motif in a set of sequences (e.g. DNA sequences that bind the same protein)
- characterizing a set of sequences (e.g. a protein family)
- building *profiles* for sequence-database searching
 - PSI-BLAST generalizes a query sequence into a profile to search for remote relatives

Scoring a Multiple Alignment

- key issue: how do we assess the quality of a multiple sequence alignment?
- usually, the assumption is made that the individual *columns* of an alignment are independent

$$\text{Score}(m) = G + \sum_i S(m_i)$$

gap function score of i^{th} column

- we'll discuss two methods
 - sum of pairs (SP)
 - minimum entropy

Scoring an Alignment: Sum of Pairs

- compute the sum of the pairwise scores

$$S(m_i) = \sum_{k < l} s(m_i^k, m_i^l)$$

m_i^k = character of the k th sequence in the i th column

S = substitution matrix

Scoring an Alignment: Minimum Entropy

- basic idea: try to minimize the *entropy* of each column
- another way of thinking about it: columns that can be communicated using few bits are good
- information theory tells us that an optimal code uses $-\log_2 p$ bits to encode a message of probability p
 - Frequently sent messages require few bits
 - Rarely sent messages require many bits

Scoring an Alignment: Minimum Entropy

- the messages in this case are the characters in a given column
- the entropy of a column is given by:

$$S(m_i) = - \sum_a c_{ia} \log_2 p_{ia}$$

m_i = the i th column of an alignment m

c_{ia} = count of character a in column i

p_{ia} = probability of character a in column i

Dynamic Programming Approach

- can find optimal alignments using dynamic programming
- generalization of methods for pairwise alignment
 - consider k -dimension matrix for k sequences (instead of 2-dimensional matrix)
 - each matrix element represents alignment score for k subsequences (instead of 2 subsequences)
- given k sequences of length n
 - space complexity is

$$O(n^k)$$

Dynamic Programming Approach

$$\alpha_{i_1, i_2, \dots, i_k} = \max \left\{ \begin{array}{l} \alpha_{i_1-1, i_2-1, \dots, i_k-1} + S(x_{i_1}^1, x_{i_2}^2, \dots, x_{i_k}^k) \\ \alpha_{i_1, i_2-1, \dots, i_k-1} + S(-, x_{i_2}^2, \dots, x_{i_k}^k) \\ \alpha_{i_1-1, i_2, \dots, i_k-1} + S(x_{i_1}^1, -, \dots, x_{i_k}^k) \\ \vdots \\ \alpha_{i_1, i_2, \dots, i_k-1} + S(-, -, \dots, x_{i_k}^k) \\ \vdots \end{array} \right.$$

max score of alignment
for subsequences
 $x_{i_1}^1, x_{i_2}^2, \dots, x_{i_k}^k$

Dynamic Programming Approach

- given k sequences of length n
 - time complexity is

$$O(k^2 2^k n^k)$$

if we use sum of pairs

$$O(k 2^k n^k)$$

if column scores can be
computed in $O(k)$,
as with entropy

Heuristic Alignment Methods

- since time complexity of DP approach is exponential in the number of sequences, heuristic methods are usually used
- *progressive alignment*: construct a succession of pairwise alignments
 - star approach
 - tree approaches, like CLUSTALW
 - etc.
- iterative refinement
 - given a multiple alignment (say from a progressive method)
 - remove a sequence, realign it to profile of other sequences
 - repeat until convergence

Star Alignment Approach

- given: k sequences to be aligned
 x_1, \dots, x_k
 - pick one sequence x_c as the “center”
 - for each $x_i \neq x_c$ determine an optimal alignment between x_i and x_c
 - merge pairwise alignments
- return: multiple alignment resulting from aggregate

Star Alignment Example

Given:

ATTGCCATT
ATGGCCATT
ATCCAATTTT
ATCTTCTT
ATTGCCGATT

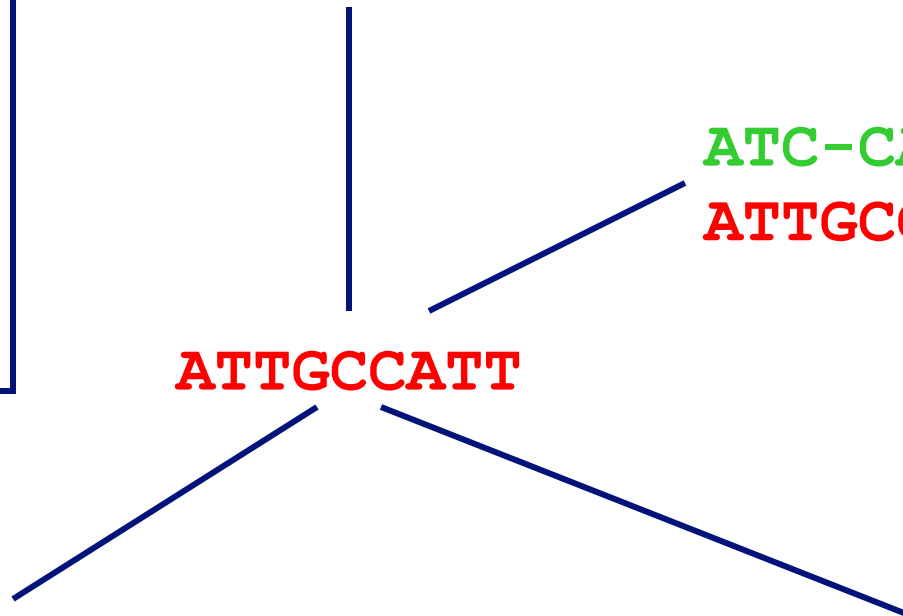
ATGGCCATT
ATTGCCATT

ATC-CAATTTT
ATTGCCATT--

ATTGCCATT

ATTGCCGATT
ATTGCC-ATT

ATCTTCTT
ATTGCCATT



Star Alignment Example

- merging pairwise alignments

	present pair	alignment
1.	ATGGCCATT ATTGCCATT	ATTGCCATT ATGGCCATT
2.	ATC-CAATTTT ATTGCCATT--	ATTGCCATT-- ATGGCCATT-- ATC-CAATTTT

Star Alignment Example

present pair

alignment

3.

ATCTTC-TT
ATTGCCATT

ATTGCCATT--
ATGGCCATT--
ATC-CAATTTT
ATCTTC-TT--

4.

ATTGCCGATT
ATTGCC-ATT

ATTGCC- A TT--
ATGGCC- A TT--
ATC-CA- A TTTT
ATCTTC- - TT--
ATTGCCG A TT--

shift entire columns
when incorporating a gap



Star Alignments: Aggregating Pairwise Alignments

- “once a gap, always a gap”
- shift entire columns when incorporating gaps

Star Alignments: Approaches to Picking the Center

Two possible approaches:

1. try each sequence as the center, return the best multiple alignment
2. compute all pairwise alignments and select the string x_c that maximizes:

$$\sum_{i \neq c} \text{sim}(x_i, x_c)$$

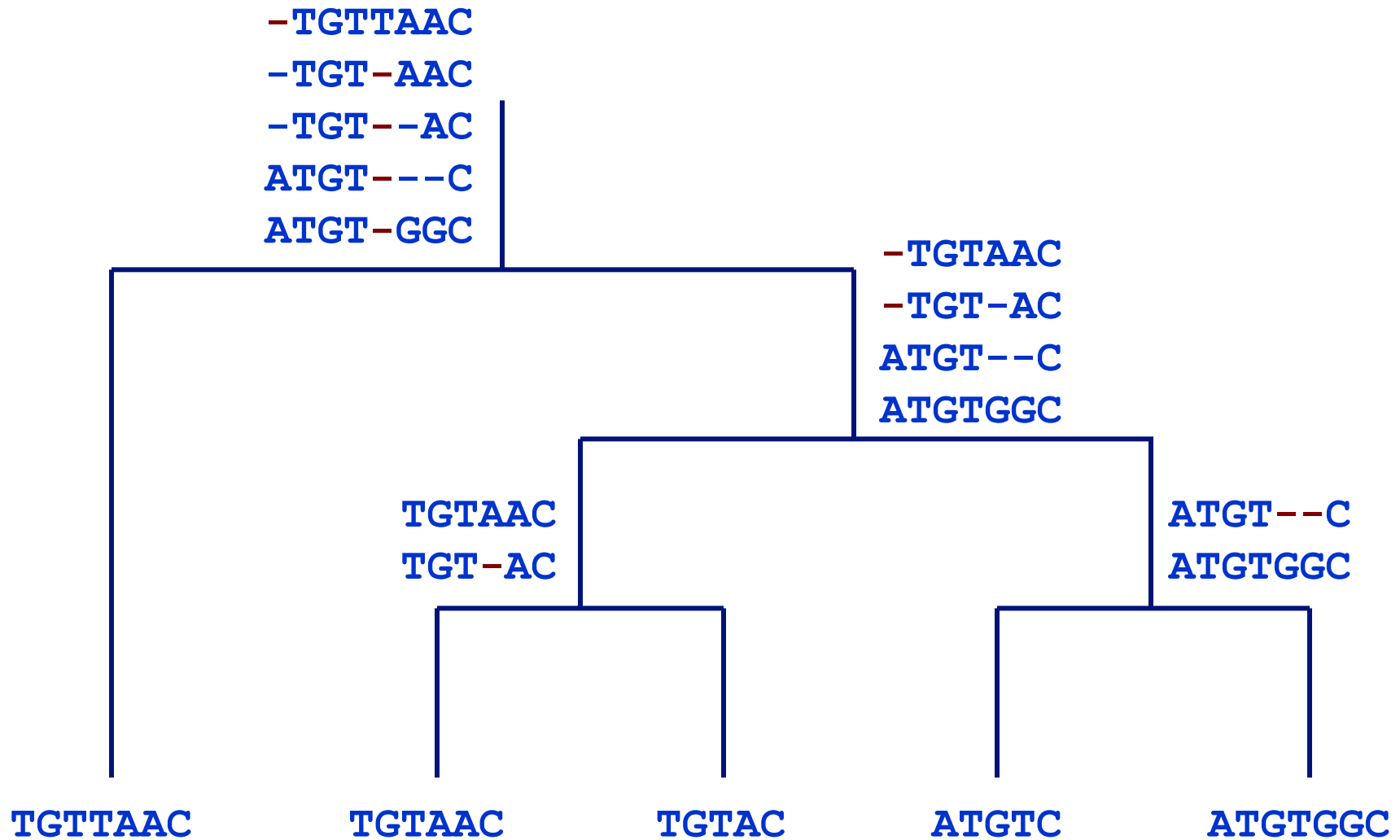
Tree Alignments

- basic idea: organize multiple sequence alignment using a *guide tree*
 - leaves represent sequences
 - internal nodes represent alignments
- determine alignments from bottom of tree upward
 - return multiple alignment represented at the root of the tree
- one common variant: the CLUSTALW algorithm [Thompson et al. 1994]

Doing the Progressive Alignment in CLUSTALW

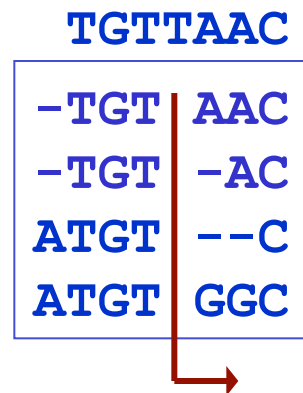
- depending on the internal node in the tree, we may have to align a
 - a sequence with a sequence
 - a sequence with a *profile* (partial alignment)
 - a *profile* with a *profile*
- in all cases we can use dynamic programming
 - for the profile cases, use SP scoring

Tree Alignment Example



Aligning Profiles

- aligning sequences/profiles to profiles is essentially pairwise alignment
 - shift entire columns when incorporating gaps



-TGTTAAC
-TGT-AAC
-TGT--AC
ATGT---C
ATGT-GGC

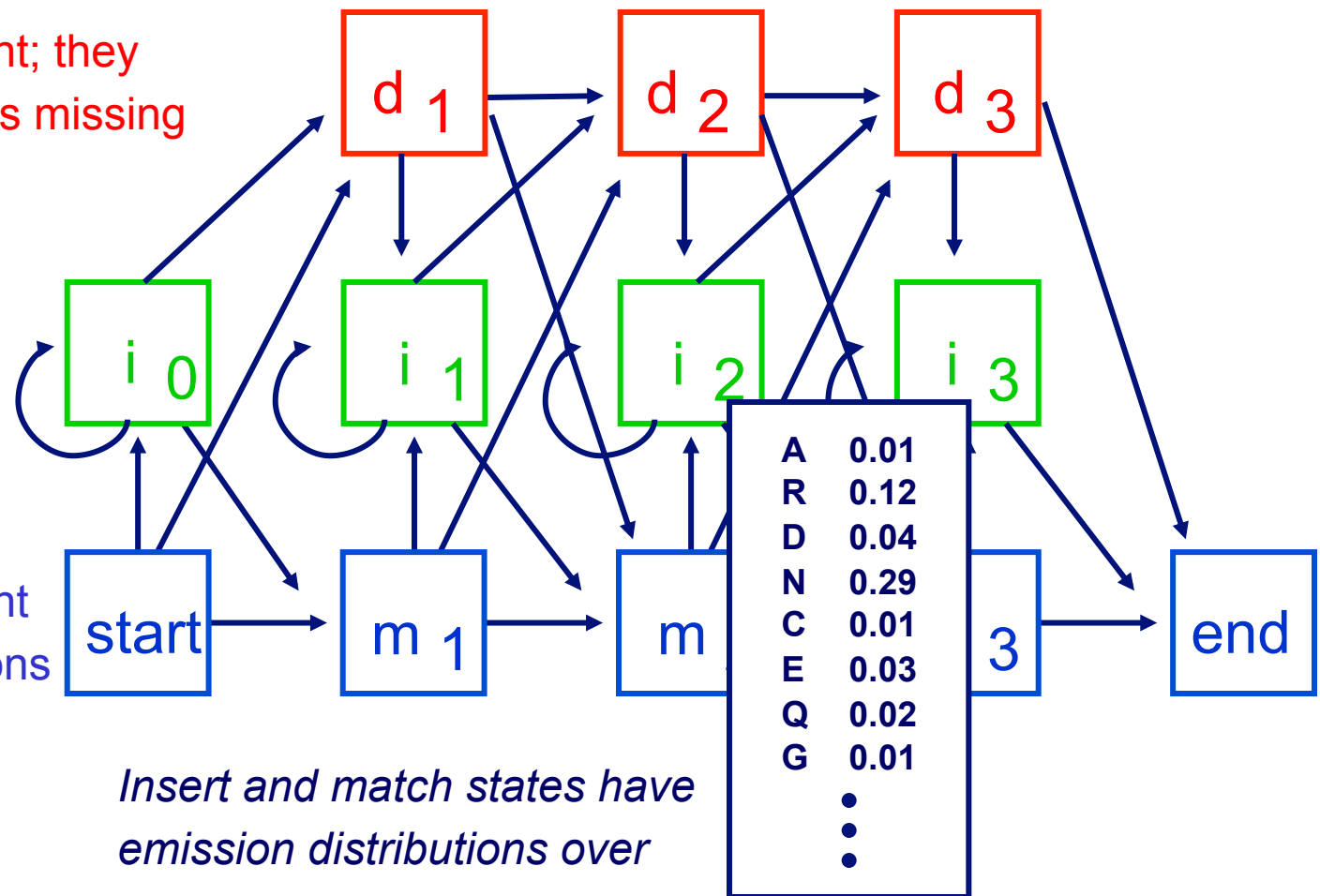
Profile HMMs

- profile HMMs are commonly used to model families of sequences

Delete states are silent; they account for characters missing in some sequences

Insert states account for extra characters in some sequences

Match states represent key conserved positions



Insert and match states have emission distributions over sequence characters

Multiple Alignment with Profile HMMs

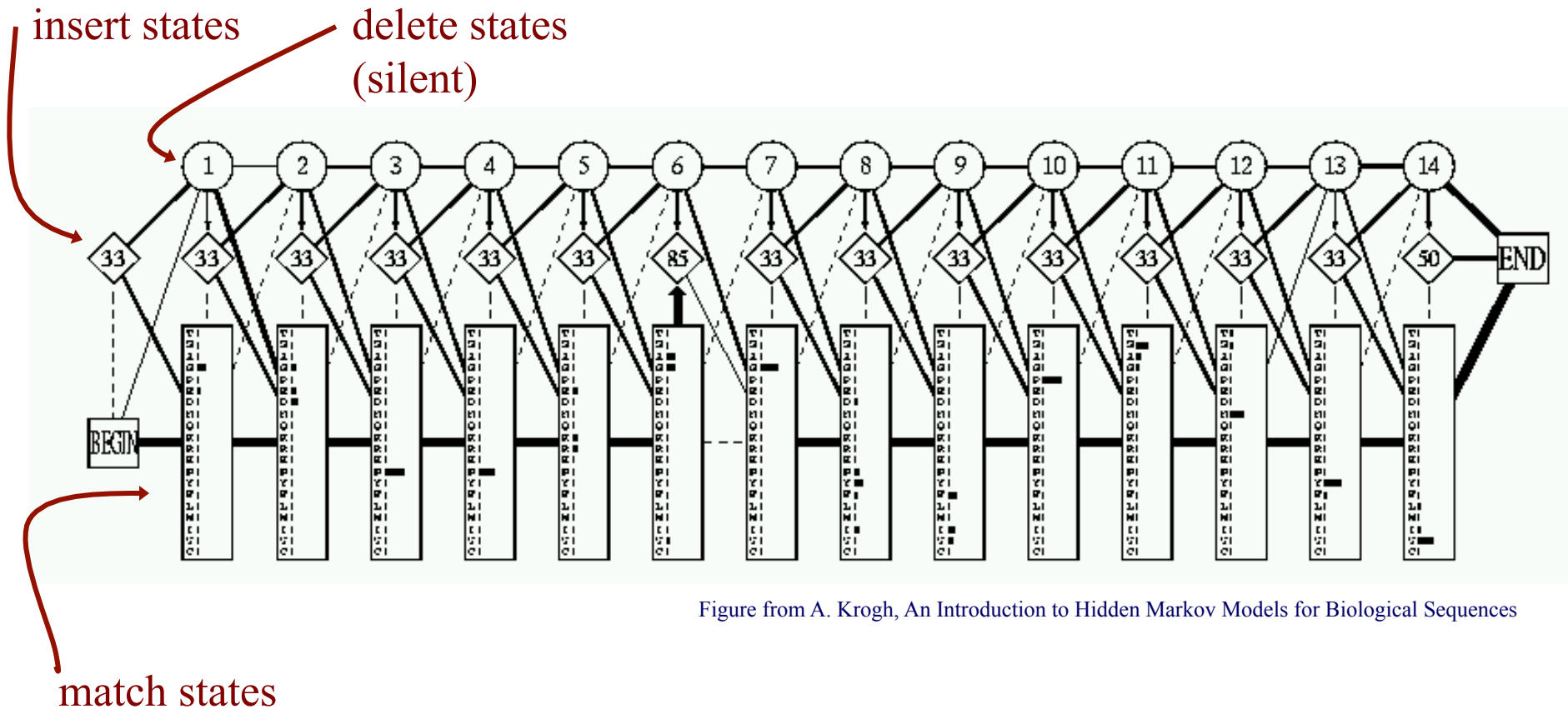
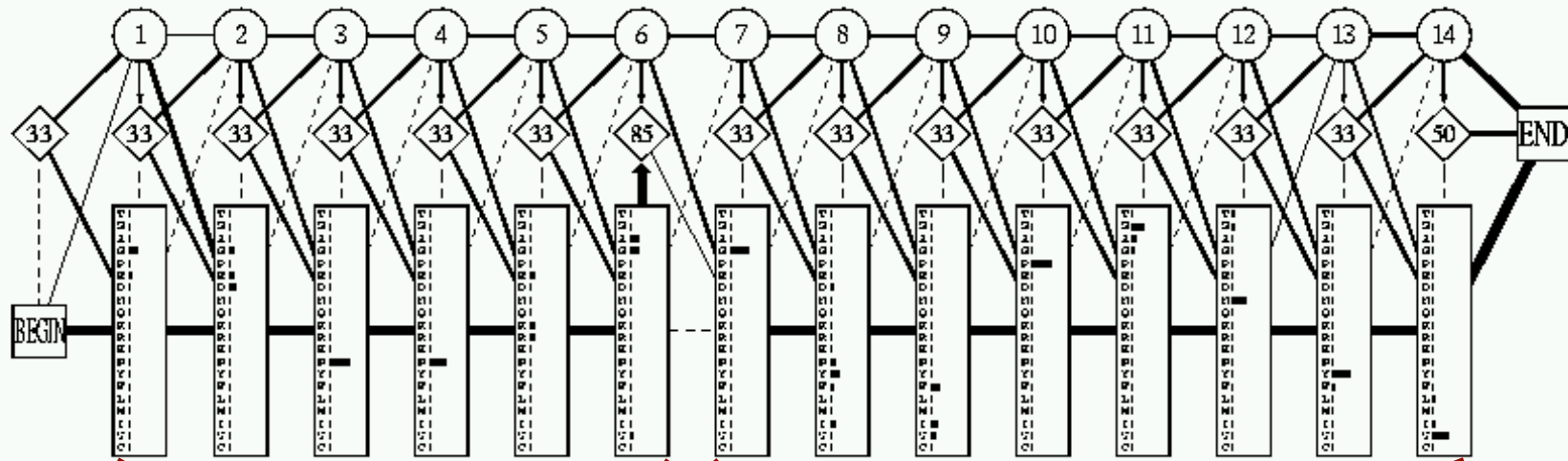


Figure from A. Krogh, An Introduction to Hidden Markov Models for Biological Sequences

Multiple Alignment with Profile HMMs

- given a set of sequences to be aligned
 - use Baum-Welch to learn parameters of model
 - may also adjust length of profile HMM during training
- to compute a multiple alignment given the profile HMM
 - run the Viterbi algorithm on each sequence
 - Viterbi paths indicate correspondences among sequences

Multiple Alignment with Profile HMMs



GGWWRGdy.ggkkqLWFPSPNYV
IGWLNNGynehettgkerLGDFPSTYV
PNWWEgql.nnrrrGIFPSTNYV
DEWWRQAr.r.dedqqiGIVPSK--V
GEWWRKAqs..tqqqegGFIPFNfV
GDWWRLARs..sqqqtGYIPSTNYV
GDWWRDAel..kqqrGKVPSTNYL
-DWWEARs.lssghrGYVPSSTNYV
GDWWRAYars.litnseGYIPSTNYV
GEWWRKArslatrkEGYIPSTNYV
GDWWRLARslvtgreGYVPSSTNYV
GEWWRKAKslsskreGFIPSTNYV
GEWWRCAqt.kngq.GWVPSSTNYI
SDWWRVRvnltrqqeGLIPLNfV
LPWWRARd.kngqegGYIPSTNYI
RDWWRERsktvytpGGYIESGYV
EHWWRVKd.algnvGGYIPSTNYV
IHWWRVq.d.rnqhGYVPSSTNYL

Multiple Alignment Case Study: The Cystic Fibrosis Gene

- cystic fibrosis (CF)
 - recessive genetic disease caused by a defect in a single-gene
 - causes the body to produce abnormally thick mucus that clogs the lungs and the pancreas
- the cystic fibrosis conductance regulator (CFTR) gene
 - gene and its role in CF identified in 1989 [Riordan et al., *Science*]
 - most common mutation is called $\Delta F508$; a deletion of a phenylalanine (F) at position 508 in the CFTR protein
 - the CFTR protein controls the movement of salt and water into and out of cells; mutations in CFTR block this movement, causing mucus problem

Multiple Alignment Case Study: the Cystic Fibrosis Gene

- two key features of the protein made apparent in multiple sequence alignment (and other analyses)
 - membrane-spanning domains
 - ATP-binding motifs
- these features indicated that CFTR is likely to be involved in transporting ions across the cell membrane

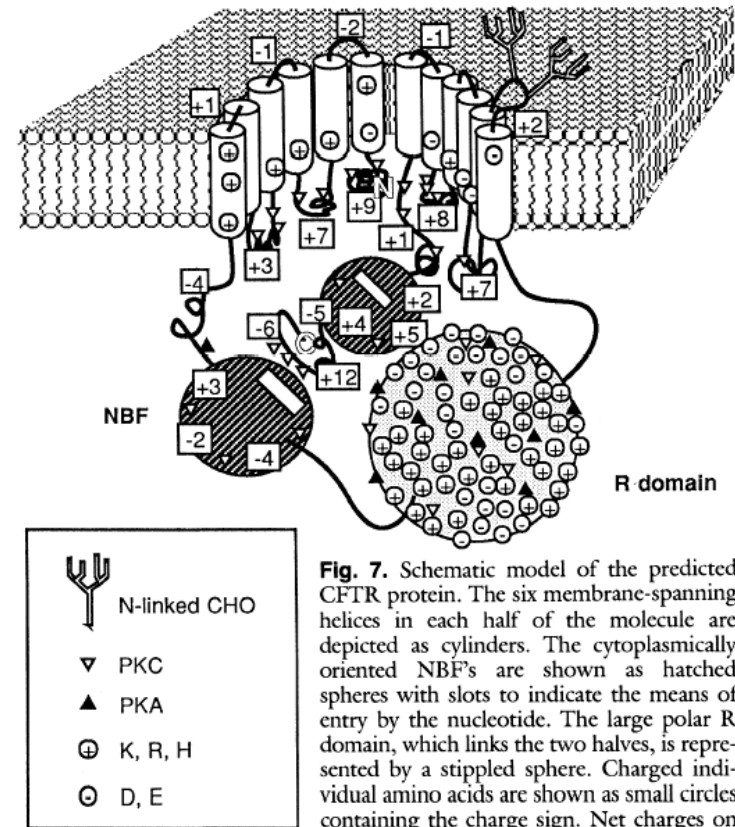


Fig. 7. Schematic model of the predicted CFTR protein. The six membrane-spanning helices in each half of the molecule are depicted as cylinders. The cytoplasmically oriented NBF's are shown as hatched spheres with slots to indicate the means of entry by the nucleotide. The large polar R domain, which links the two halves, is represented by a stippled sphere. Charged individual amino acids are shown as small circles containing the charge sign. Net charges on the internal and external loops joining the membrane cylinders and on regions of the NBF's are contained in open squares. Potential sites for phosphorylation by protein kinases A or C (PKA or PKC) and N-glycosylation (N-linked CHO) are as indicated. K, Lys; R, Arg; H, His; D, Asp; and E, Glu.

Figure from Riordan et al, *Science* 245:1066-1073, 1989.

Notes on Multiple Alignment

- as with pairwise alignment, can compute *local* and *global* multiple alignments
- dynamic programming is not feasible for most cases -- heuristic methods usually used instead

Summary: Some Methods for Multiple Sequence Alignment

method	alignment types	search
multi-dimensional dynamic programming	global/local	dynamic programming
Star	global	greedy via pairwise alignments
CLUSTALW (tree)	global	greedy via pairwise alignment
profile HMMs	global/local	Baum-Welch (EM) to learn model, Viterbi to recover alignments
EM/MEME Gibbs sampling Random projections etc.	local	EM Gibbs sampling random projections