Rohit Jhawer

Digitally signed by Rohit Jhawer
DN: cn=Rohit Jhawer, o, ou, email=rohit_jhawer@hotmail.com, c=IN
Date: 2007.03.09 14:02:20 +05'30'

Lecture 3:
Pairwise Sequence Alignment
Multiple Sequence Alignment

Pairwise Sequence Alignment

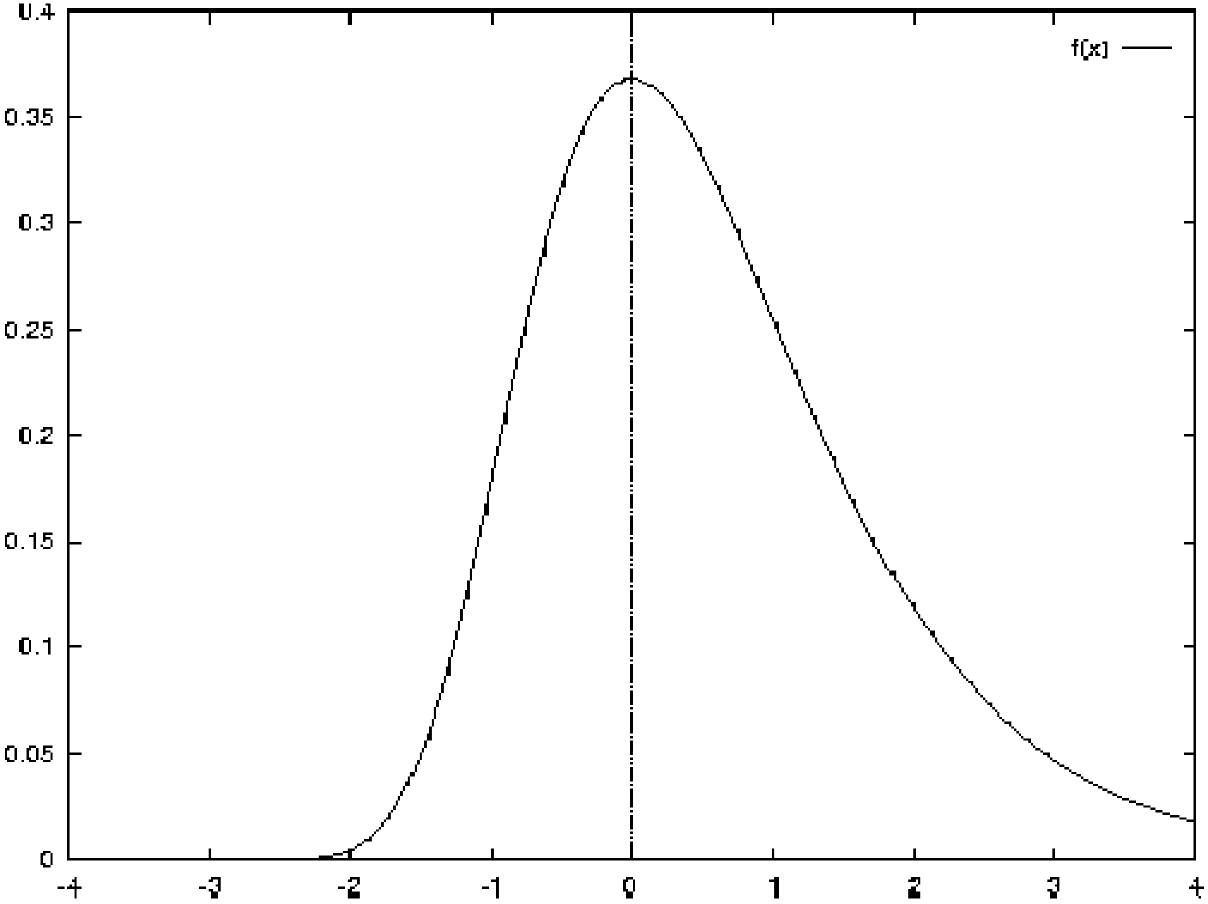
- Determining Significance of an Alignment

Significance of Alignment

- Determine probability of alignment occurring at random
  - Sequence 1: length m
  - Sequence 2: length n

- Random sequences:
  - Alignment follows Gumbel Extreme Value Distribution

Gumbel Extreme Value Distribution



- http://roso.epfl.ch/mbi/papers/discretechoice/node11.html

Probability of Alignment Score

- Expected # of alignments with score at least S (E-value):

$$E = Kmn\ e^{-\lambda S}$$

  - m,n: Lengths of sequences
  - *K* ,λ: natural scales
    - Search space size
    - Scoring system

Converting to Bit Scores

A raw score can be normalized to a bit score using the formula:

$$S' = \frac{\lambda S - \ln K}{\ln 2}$$

- The E-value corresponding to a given bit score can then be calculated as:

$$E = mn\, 2^{-S'}$$

P-Value

- P-Value: probability of obtaining a given score at random

$$P = 1 – e^{-E}$$

Which is approximately e^{-E}

Significance of Ungapped Alignments

- PAM matrices are 10 * $\log_{10}x$
- Converting to $\log_2 x$ gives bits of information
- Converting to $\log_e x$ gives nats of information

Quick Calculation

- If bit scoring system is used, significance cutoff is:

$$\log_2(mn)$$

Example (p110)

- 2 Sequences, each 250 amino acids long
- Significance:
  - $\log_2(250 * 250) = 16$ bits

Example (p110)

- Using PAM250, the following alignment is found:

- F W L E V E G N S M T A P T G
- F W L D V Q G D S M T A P A G

Example (p110)

- Using PAM250 (p82), the score is calculated:

- F W L E V E G N S M T A P T G
- F W L D V Q G D S M T A P A G

- S = 9 + 17 + 6 + 3 + 4 + 2 + 5 + 2 + 2 + 6 + 3 + 2 + 6 + 1 + 5 = 73

Significance Example

- S is in 10 * $\log_{10}x$ -- convert to a bit score:
-
- S = 10 $\log_{10}x$
- S/10 = $\log_{10}x$
- S/10 = $\log_{10}x$ * ($\log_2 10/\log_2 10$)
- S/10 * $\log_2 10$ = $\log_{10}x$ / $\log_2 10$
- S/10 * $\log_2 10$ = $\log_2 x$
- 1/3 S ~ $\log_2 x$
-
- S’ ~ 1/3S

Significance Example

- S’ = 1/3S = 1/3 * 73 = 24.333 bits
- Significance cutoff = 16 bits
- Therefore, this alignment is significant

Estimation of E and P

- For PAM250, K = 0.09; λ = 0.229
- Using equations 30 and 31 (normalize to mean of 0; λ = 1):
  - S’ = 0.229 * 73 – ln 0.09 * 250 * 250
  - S’ = 16.72 – 8.63 = 8.09 bits
- $P(S' >= 8.09) = 1 - e^{(-e^{-8.09})} = 3.1 * 10^{-4}$

Significance of Gapped Alignments

- Gapped alignments use same statistics
- λ and K cannot be easily estimated
- Empirical estimations and gap scores determined by looking at random alignments

Pairwise Sequence Alignment Programs

- needle
  - Global Needleman/Wunsch alignment
- water
  - Local Smith/Waterman alignment
- Blast 2 Sequences
  - NCBI
  - word based sequence alignment
- LALIGN
  - FASTA package
  - Mult. Local alignments

Various Sequence Alignments

Wise2 -- Genomic to protein

Sim4 -- Aligns expressed DNA to genomic sequence

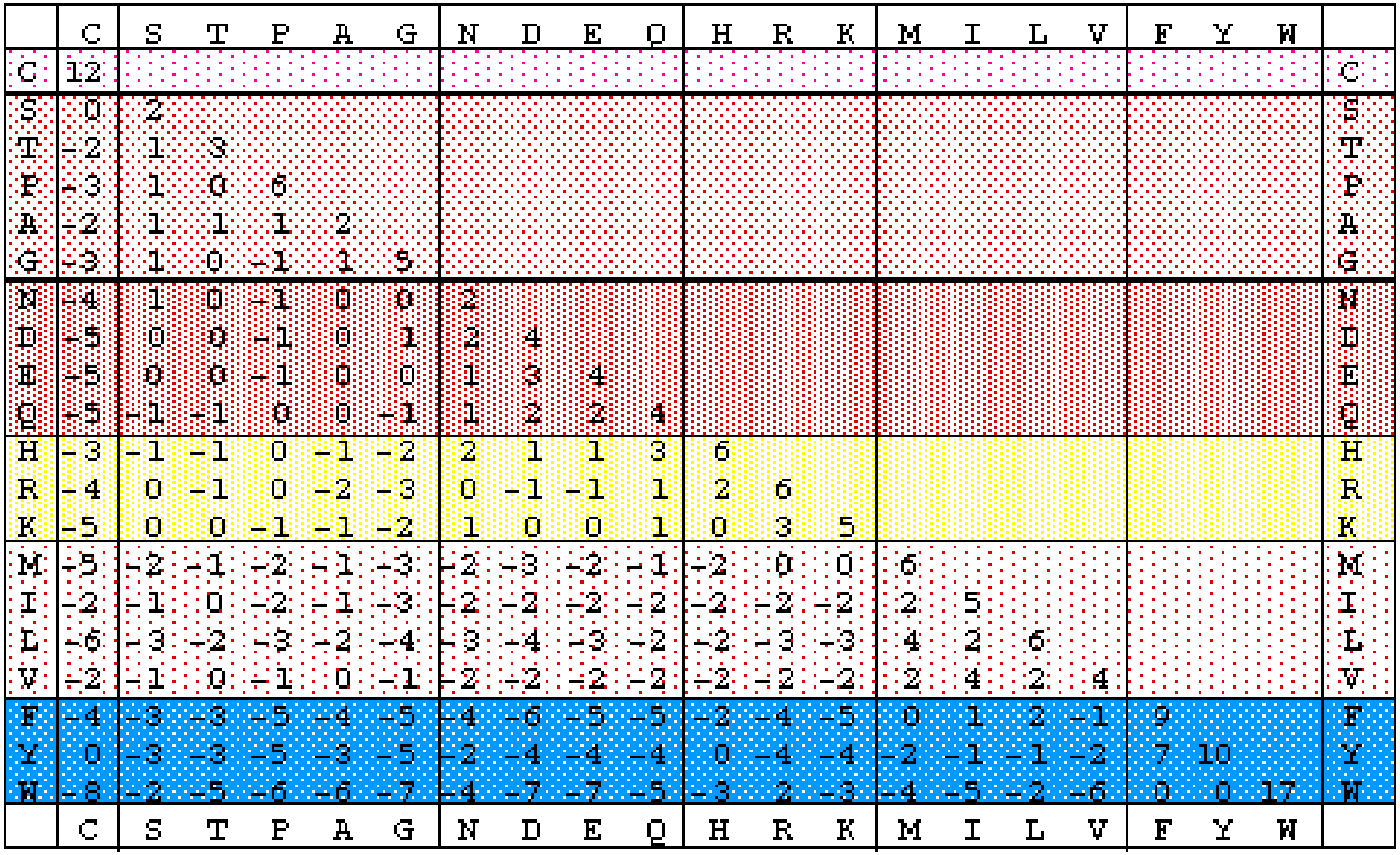
spidey -- aligns mRNAs to genomic sequence

est2genome -- aligns ESTs to genomic sequence

Amino Acid Sequence Alignment

- No exact match/mismatch scores
- Match state score calculated by table lookup
- Lookup table is mutation matrix

PAM250 Lookup



	C	S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W	
C	12																				C
S	0	2																			S
T	-2	1	3																		T
P	-3	1	0	6																	P
A	-2	1	1	1	2																A
G	-3	1	0	-1	1	5															G
N	-4	1	0	-1	0	0	2														N
D	-5	0	0	-1	0	1	2	4													D
E	-5	0	0	-1	0	0	1	3	4												E
Q	-5	-1	-1	0	0	-1	1	2	2	4											Q
H	-3	-1	-1	0	-1	-2	2	1	1	3	6										H
R	-4	0	-1	0	-2	-3	0	-1	-1	1	2	6									R
K	-5	0	0	-1	-1	-2	1	0	0	1	0	3	5								K
M	-5	-2	-1	-2	-1	-3	-2	-3	-2	-1	-2	0	0	6							M
I	-2	-1	0	-2	-1	-3	-2	-2	-2	-2	-2	-2	-2	2	5						I
L	-6	-3	-2	-3	-2	-4	-3	-4	-3	-2	-2	-3	-3	4	2	6					L
V	-2	-1	0	-1	0	-1	-2	-2	-2	-2	-2	-2	-2	2	4	2	4				V
F	-4	-3	-3	-5	-4	-5	-4	-6	-5	-5	-2	-4	-5	0	1	2	-1	9			F
Y	0	-3	-3	-5	-3	-5	-2	-4	-4	-4	0	-4	-4	-2	-1	-1	-2	7	10		Y
W	-8	-2	-5	-6	-6	-7	-4	-7	-7	-5	-3	2	-3	-4	-5	-2	-6	0	0	17	W
	C	S	T	P	A	G	N	D	E	Q	H	R	K	M	I	L	V	F	Y	W	

Affine Gap Penalties

- Gap Open
- Gap Extension
- Maximum score matrix determined by maximum of three matrices:
  - Match matrix (match residues in A & B)
  - Insertion matrix (gap in sequence A)
  - Deletion matrix (gap in sequence B)

Dynamic Programming with Affine Gap

$M_{i,j}$ = MAX{ $M_{i-1,\, j-1} + s(x_i, y_i)$,
$I_{i-1,\, j-1} + s(x_i, y_i)$,
$D_{i-1,\, j-1} + s(x_i, y_i)$ }

$I_{i,j}$ = MAX{ $M_{i-1,\, j} - g$, // Opening new gap, g = gap open penalty;
$I_{i-1,\, j} - r$} // Extending existing gap, r = gap extend penalty

$D_{i,j}$ = MAX{$M_{i,j-1} - g$, // Opening new gap;
$D_{i,j-1} - r$} // Extending existing gap

$V_{i,j}$ = MAX {$M_{i,j}$, $I_{i,j}$, $D_{i,j}$}

Sequence File Formats

- We have been using DNA and amino acid sequences already
- What is the typical format for these?
- ANSWER: Many different options
  - Consider two most popular for now

Standard Codes (IUPAC)

A = adenine

C = cytosine

G = guanine

T = thymine

U = uracil

R = G A (purine)

Y = T C (pyrimidine)

K = G T (keto)

M = A C (amino)

S = G C

W = A T

B = G T C

D = G A T

H = A C T

V = G C A

N = A G C T (any)

Standard IUPAC Codes

A	Ala	Alanine
R	Arg	Arginine
N	Asn	Asparagine
D	Asp	Aspartic acid
C	Cys	Cysteine
Q	Gln	Glutamine
E	Glu	Glutamic acid
G	Gly	Glycine
H	His	Histidine
I	Ile	Isoleucine
L	Leu	Leucine
K	Lys	Lysine
M	Met	Methionine
F	Phe	Phenylalanine
P	Pro	Proline
S	Ser	Serine
T	Thr	Threonine
W	Trp	Tryptophan
Y	Tyr	Tyrosine
V	Val	Valine
B	Asx	Aspartic acid or Asparagine
Z	Glx	Glutamine or Glutamic acid
X	Xaa or Xxx	Any amino acid

Fasta File Format

- most basic and widespread sequence format
- first line descriptor begins with a '>' character
- proceeding lines contain sequence
- useful for sequence analysis programs

Fasta File Format

- Example Fasta Sequence:

```
>gi|27819608|ref|NP_776342.1| hemoglobin, beta [beta globin] [Bos taurus]
MLTAEEKAAVTAFWGKVKVDEVGGEALGRLLVVYPWTQRFFESFGDLSTADAVMNNPKVKAHGKKVLDSF
SNGMKHLDDLKGTFAALSELHCDKLHVDPENFKLLGNVLVVVLARNFGKEFTPVLQADFQKVVAGVANAL
AHRYH
```

- first line begins with ‘>’, followed by gi, -- next field surrounded by ‘|’ is GenBank identifier
- the keyword ‘ref’ -- field will be the reference for the version of this sequence.
- final field is the description

Fasta File Format

- Example Fasta Sequence:

```
>gi|27819608|ref|NP_776342.1| hemoglobin, beta [beta globin] [Bos taurus]
MLTAEEKAAVTAFWGKVKVDEVGGEALGRLLVVYPWTQRFFESFGDLSTADAVMNNPKVKAHGKKVLDSF
SNGMKHLDDLKGTFAALSELHCDKLHVDPENFKLLGNVLVVVLARNFGKEFTPVLQADFQKVVAGVANAL
AHRYH
```

- nearly all sequence based programs treat anything following the ‘>’ as a comment

- a few sequence analysis programs expect sequences to be in a strict fasta format

GenBank

- GenBank: National Center for Biotechnology Information’s (NCBI) nucleic acid and protein sequence database
- widely used source of biological sequence data
- format contains information about the sequence: literature references, functions, features, etc.

GenBank

- information organized into fields, each with an identifier, justified to the farthest left column.
- Some identifiers have additional subfields.
- sequence data lies between the identifier ORIGIN and the ‘//’ which signals the end of a GenBank record.

GenBank Record

- View NCBI GenBank Record:
  - NP_776342

CECS 694-02

Introduction to Bioinformatics

Multiple Sequence Alignment

Eric C. Rouchka, D.Sc.
eric.rouchka@uofl.edu

http://kbrin.a-bldg.louisville.edu/CECS694/

Multiple Sequence Alignment

- Similar regions conserved across organisms
  - Same or similar function
  - Same or similar structure

Multiple Sequence Alignment

- Simultaneous alignment of similar regions yields:
  - regions subject to mutation
  - regions of conservation
  - mutations or rearrangements causing change in conformation or function

Multiple Sequence Alignment

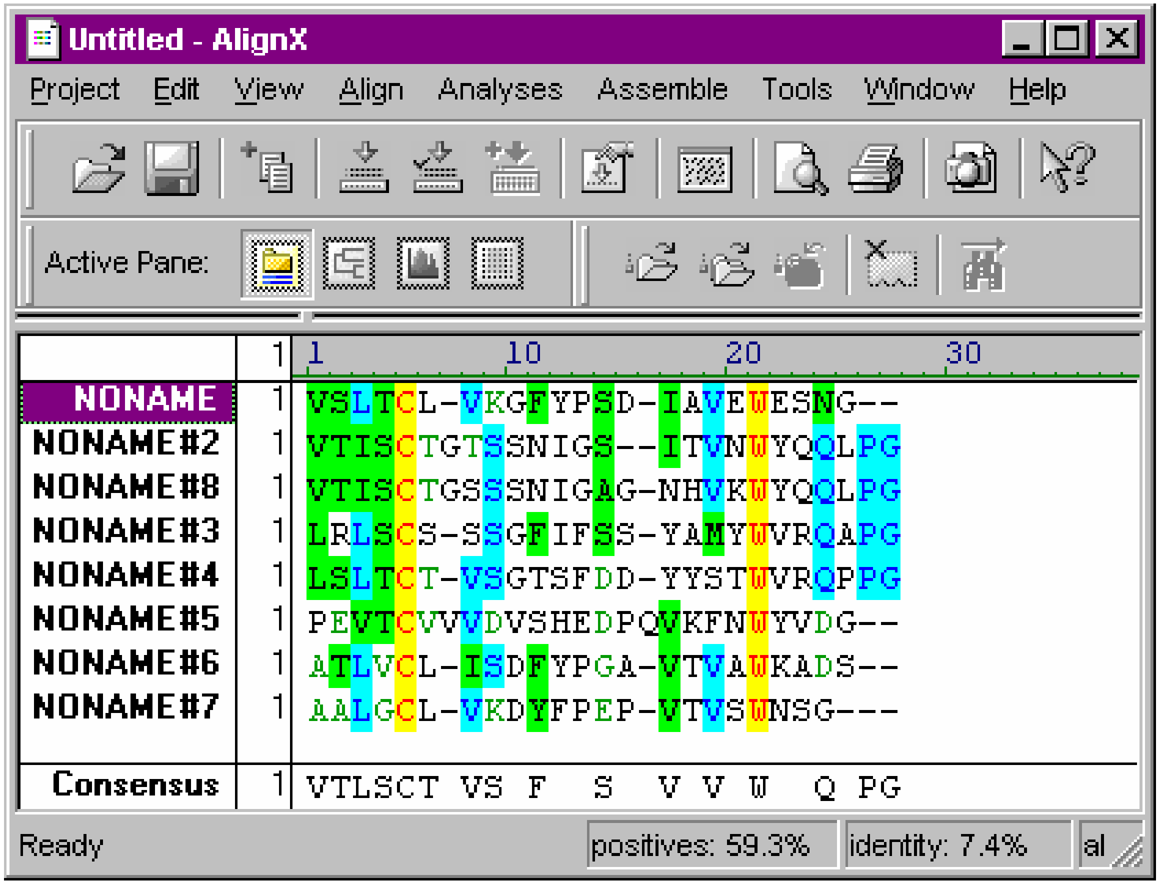
- New sequence can be aligned with known sequences
  - Yields insight into structure and function
- Multiple alignment can detect important features or motifs

Multiple Sequence Alignment

- GOAL: Take 3 or more sequences, align so greatest number of characters are in the same column

- Difficulty: introduction of multiple sequences increases combination of matches, mismatches, gaps

Example Multiple Alignment



```
                 1  1        10        20        30
NONAME           1  VSLTCL-VKGFYPSD-IAVEWESNG--
NONAME#2         1  VTISCTGTSSNIGS--ITVNWYQQLPG
NONAME#8         1  VTISCTGSSSNIGAG-NHVKWYQQLPG
NONAME#3         1  LRLSCS-SSGFIFSS-YAMYWVRQAPG
NONAME#4         1  LSLTCT-VSGTSFDD-YYSTWVRQPPG
NONAME#5         1  PEVTCVVVDVSHEDPQVKFNWYVDG--
NONAME#6         1  ATLVCL-ISDFYPGA-VTVAWKADS--
NONAME#7         1  AALGCL-VKDYFPEP-VTVSWNSG---

Consensus        1  VTLSCT VS F  S  V V W  Q PG

Ready                     positives: 59.3%   identity: 7.4%   al
```

- Example alignment of 8 IG sequences.

Approaches to Multiple Alignment

- Dynamic Programming
- Progressive Alignment
- Iterative Alignment
- Statistical Modeling

Dynamic Programming Approach

- Dynamic programming with two sequences
  - Relatively easy to code
  - Guaranteed to obtain optimal alignment

- Can this be extended to multiple sequences?

Dynamic Programming With 3 Sequences

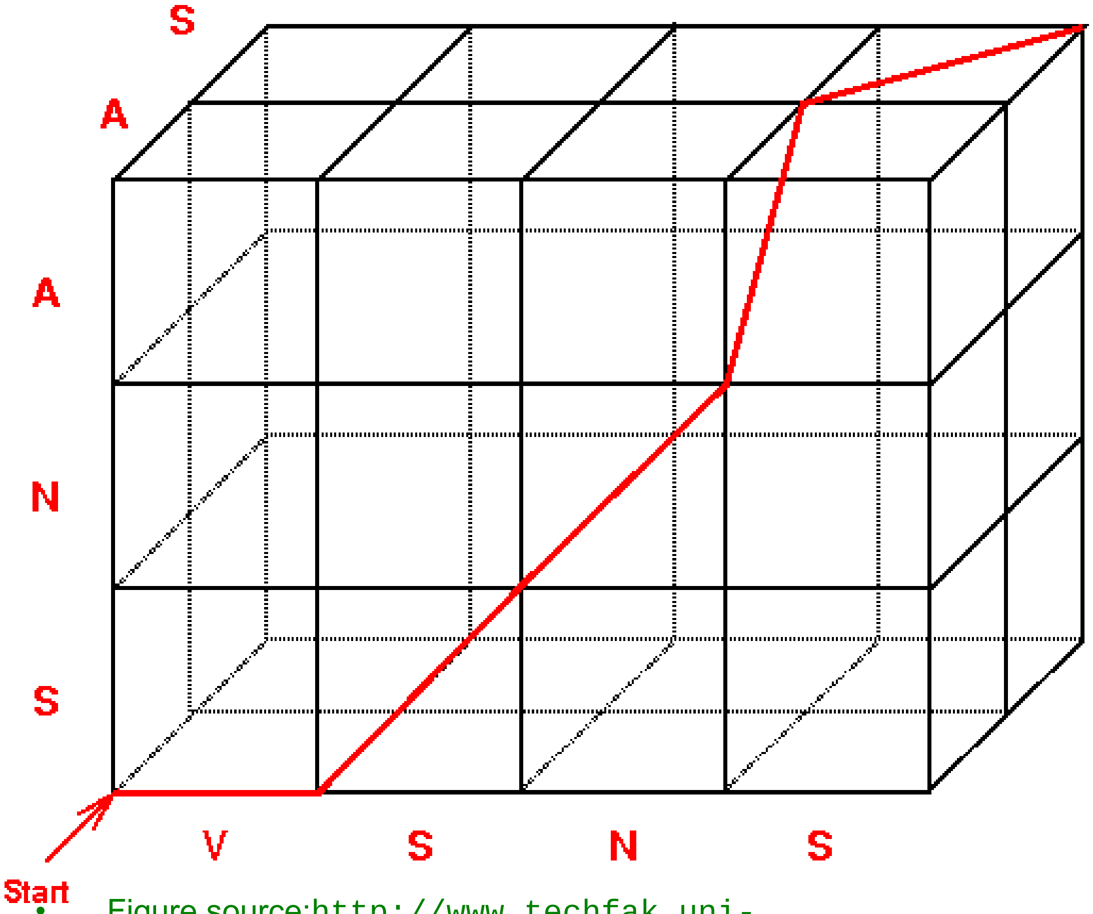
- Consider the amino acid sequences VSNS, SNA, AS
- Put one sequence per axis (x, y, z)
- Three dimensional structure results

Dynamic Programming With 3 Sequences

Possibilities:

- All three match;
- A & B match with gap in C
- A & C match with gap in B
- B & C match with gap in A
- A with gap in B & C
- B with gap in A & C
- C with gap in A & B

Dynamic Programming With 3 Sequences



V S N _ S
_ S N A _
_ _ _ A S

- Figure source:`http://www.techfak.uni-bielefeld.de/bcd/Curric/MulAli/node2.html#SECTION00020000000000000000`

Multiple Dynamic Programming complexity

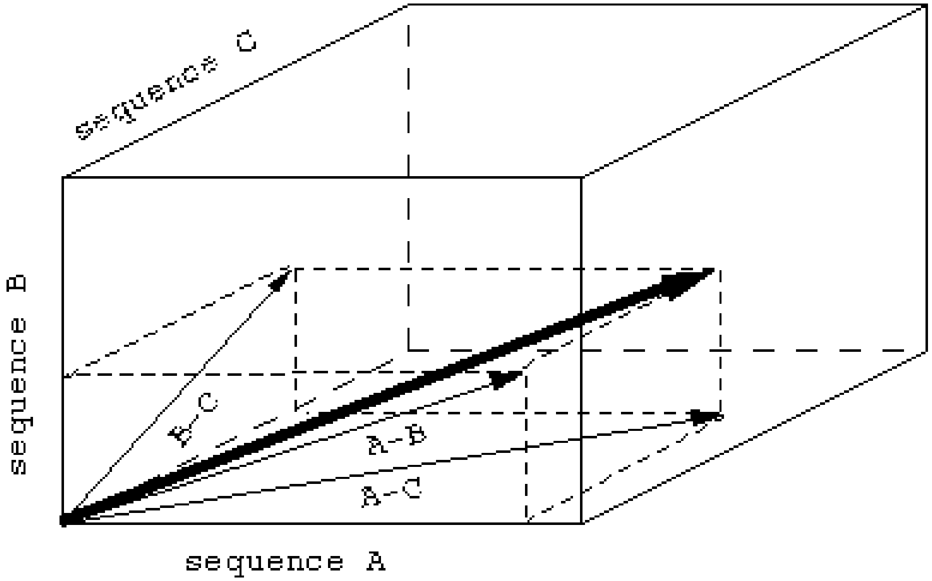
- Each sequence has length of n
  - 2 sequences: $O(n^2)$
  - 3 sequences: $O(n^3)$
  - 4 sequence: $O(n^4)$
  - N sequences: $O(n^N)$

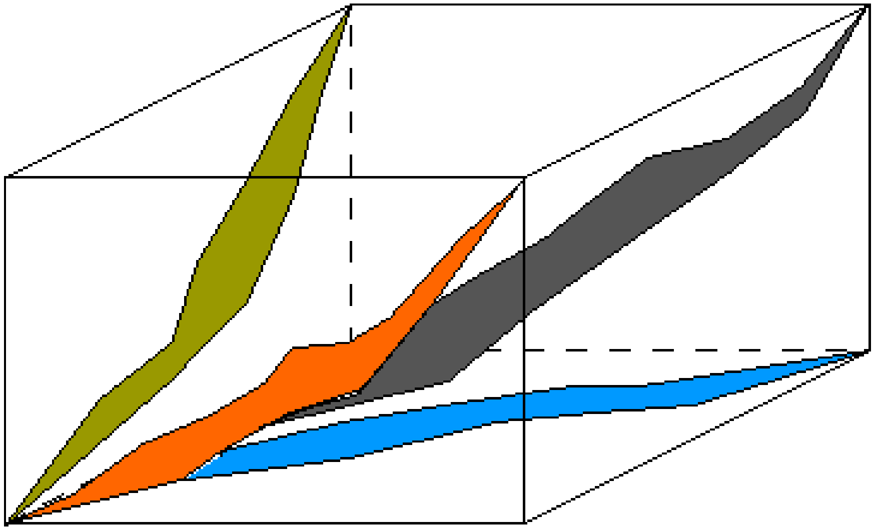
- Quickly becomes impractical

Reduction of space and time

- Carrillo and Lipman: multiple sequence alignment space bounded by pairwise alignments
- Projections of these alignments lead to a bounded

Volume Limits





Reduction of space and time

- Step 1: Find pairwise alignment for sequences.
- Step 2: Trial msa produced by predicting a phylogenetic tree for the sequences
- Step 3: Sequences multiply aligned in the order of their relationship on the tree

Reduction of space and time

- Heuristic alignment – not guaranteed to be optimal

- Alignment provides a limit to the volume within which optimal alignments are likely to be found

MSA

- MSA: Developed by Lipman, 1989
- Incorporates extended dynamic programming

Scoring of msa's

- MSA uses Sum of Pairs (SP)
  - Scores of pair-wise alignments in each column added together
  - Columns can be weighted to reduce influence of closely related sequences
  - Weight is determined by distance in phylogenetic tree

Sum of Pairs Method

- Given: 4 sequences

  ECSQ

  SNSG

  SWKN

  SCSN

- There are 6 pairwise alignments:
- 1-2; 1-3; 1-4; 2-3; 2-4; 3-4

Sum of Pairs Method

- ECSQ
  SNSG
  SWKN
  SCSN

1-2	E-S	0	C-N	-4	S-S	2	Q-G	-1
1-3	E-S	0	C-W	-8	S-K	0	Q-N	1
1-4	E-S	0	C-C	12	S-S	2	Q-N	1
2-3	S-S	2	N-W	-4	S-K	0	G-N	0
2-4	S-S	2	N-C	-4	S-S	2	G-N	0
3-4	S-S	2	W-C	-8	K-S	0	N-N	2
		6		-16		6		3

Summary of MSA

1. Calculate all pairwise alignment scores
2. Use the scores to predict tree
3. Calcuate pair weights based on the tree
4. Produce a heuristic msa based on the tree
5. Calculate the maximum weight for each sequence pair
6. Determine the spatial positions that must be calculated to obtain the optimal alignment
7. Perform the optimal alignment

- Report the weight found compared to the maximum weight previously found

Progressive Alignments

- MSA program is limited in size
- Progressive alignments take advantage of Dynamic Programming

Progressive Alignments

- Align most related sequences
- Add on less related sequences to initial alignment

CLUSTALW

- Perform pairwise alignments of all sequences
- Use alignment scores to produce phylogenetic tree
- Align sequences sequentially, guided by the tree

CLUSTALW

- Enhanced Dynamic Programming used to align sequences

- Genetic distance determined by number of mismatches divided by number of matches

CLUSTALW

- Gaps are added to an existing profile in progressive methods
- CLUSTALW incorporates a statistical model in order to place gaps where they are most likely to occur

CLUSTALW

- http://www.ebi.ac.uk/clustalw/

PILEUP

- Part of GCG package
- Sequences initially aligned using Needleman-Wunsch
- Scores used to produce tree using unweighted pair group method (UPGMA)

Shortcoming of Progressive Approach

- Dependence upon initial alignments
  - Ok if sequences are similar
  - Errors in alignment propagated if not similar

- Choosing scoring systems that fits all sequences simultaneously

Iterative Methods

- Begin by using an initial alignment

- Alignment is repeatedly refined

MultAlign

- Pairwise scores recalculated during progressive alignment
- Tree is recalculated
- Alignment is refined

PRRP

- Initial pairwise alignment predicts tree
- Tree produces weights
- Locally aligned regions considered to produce new alignment and tree
- Continue until alignments converge

DIALIGN

- Pairs of sequences aligned to locate ungapped aligned regions
- Diagonals of various lengths identified
- Collection of weighted diagonals provide alignment

Genetic Algorithms

- Generate as many different msas by rearrangements simulating gaps and recombination events
- SAGA (Serial Alignment by Genetic Algorithm) is one approach

Genetic Algorithm Approach

- 1) Sequences (up to 20) written in row, allowing for overlaps of random length – ends padded with gaps (100 or so alignments)

```
XXXXXXXXXX-----
---------XXXXXXXX
--XXXXXXXXX-----
```

Genetic Algorithm Approach

- 2) initial alignments scored using sum of pairs
  - Standard amino acid scoring matrices
  - gap open, gap extension penalties
- 3) Initial alignments are replaced
  - Half are chosen to proceed unchanged (Natural selection)
  - Half proceed with introduction of mutations
  - Chosen by best scoring alignments

Genetic Algorithm Approach

- 4) MUTATION: gaps inserted sequences and rearranged

- sequences subject to mutation split into two sets based on estimated phylogenetic tree

- gaps of random lengths inserted into random positions in the alignment

Genetic Algorithm Approach

- Mutations:

- XXXXXXXX XXX---XXX—XX
- XXXXXXXX XXX---XXX—XX
- XXXXXXXX X—XXX---XXXX
- XXXXXXXX X—XXX---XXXX
- XXXXXXXX X—XXX---XXXX

Genetic Algorithm Approach

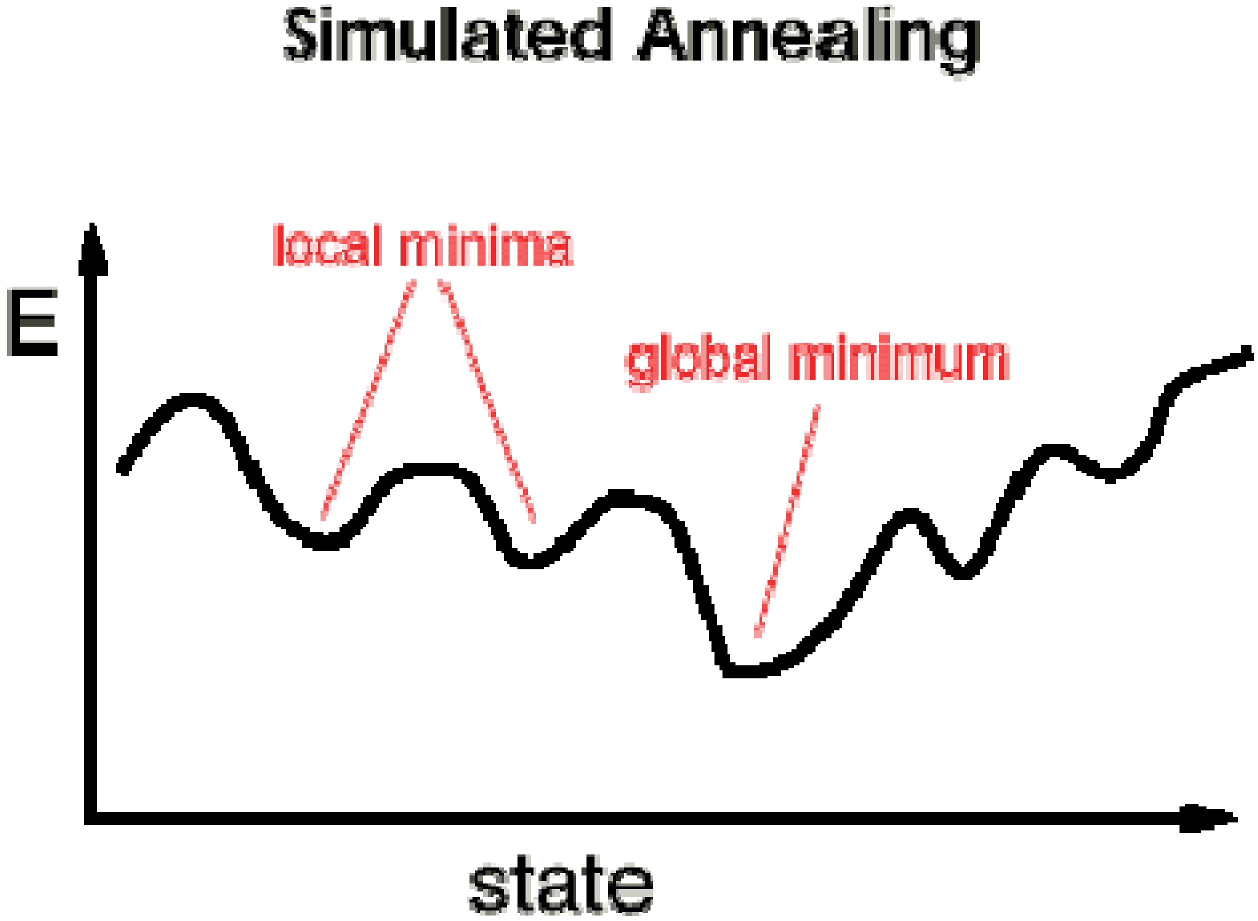
- 5) Recombination of two parents to produce next generation alignment
- 6) Next generation alignment evaluated – 100 to 1000 generations simulated (steps 2-5)
- 7) Begin again with initial alignment

Simulated Annealing

- Obtain a higher-scoring multiple alignment

- Rearranges current alignment using probabalistic approach to identify changes that increase alignment score

Simulated Annealing



http://www.cs.berkeley.edu/~amd/CS294S97/notes/day15/day15.html

Simulated Annealing

- Drawback: can get caught up in locally, but not globally optimal solutions
- MSASA: Multiple Sequence Alignment by Simulated Annealing
- Gibbs Sampling

Group Approach

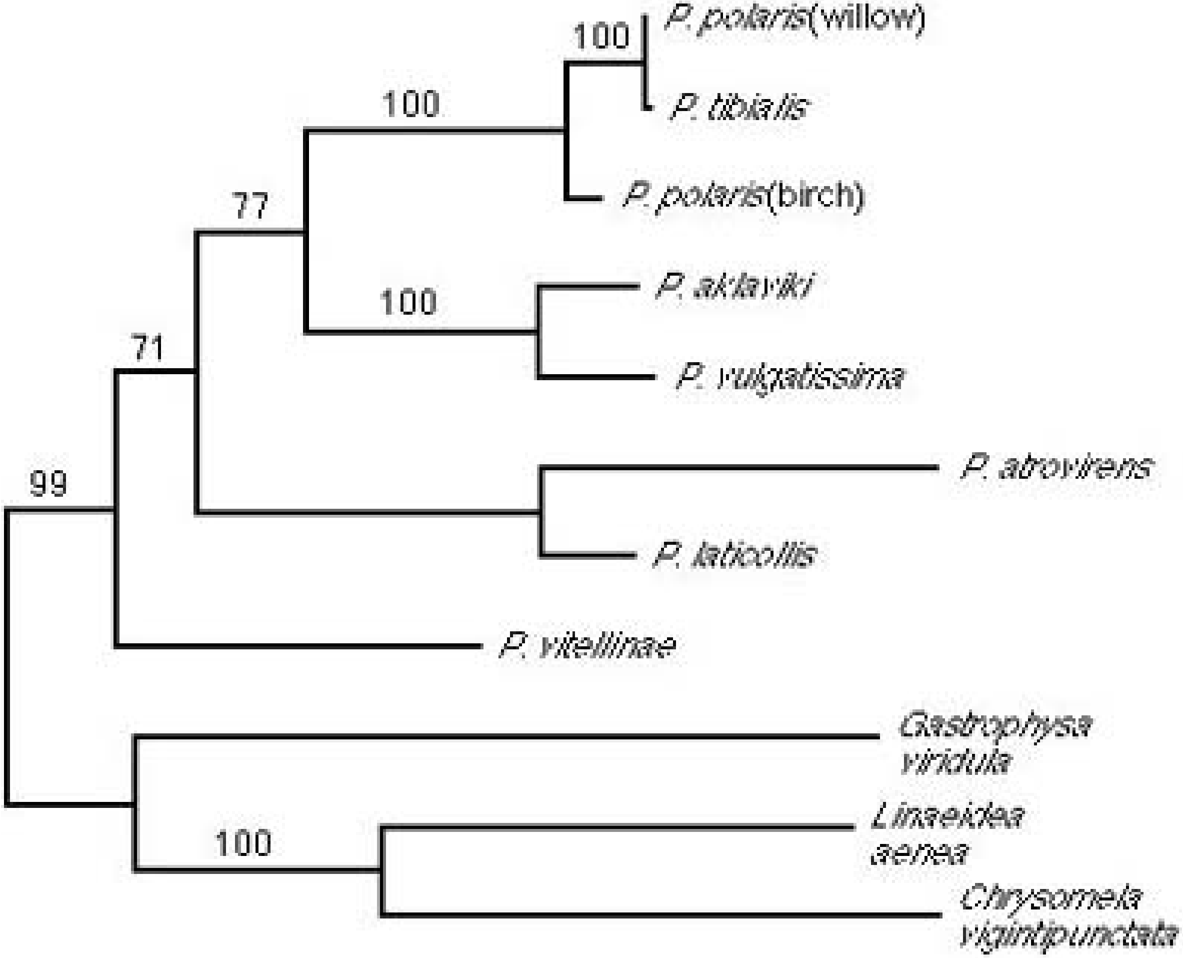
- Sequences aligned into similar groups
- Consensus of group is created
- Alignments between groups is formed

- EXAMPLES: PIMA, MULTAL

Tree Approach

- Tree created
- Two closest sequences aligned
- Consensus aligned with next best sequence or group of sequences
- Proceed until all sequences are aligned

Tree Approach to msa



- www.sonoma.edu/users/r/rank/ research/evolhost3.html

Tree Approach to msa

- PILEUP, CLUSTALW and ALIGN
- TREEALIGN rearranges the tree as sequences are added, to produce a maximum parsimony tree (fewest evolutionary changes)

Profile Analysis

- Create multiple sequence alignment
- Select conserved regions
- Create a matrix to store information about alignment
  - One row for each position in alignment
  - one column for each residue; gap open; gap extend

Profile Analysis

- Profile can be used to search target sequence or database for occurrence
- Drawback: profile is skewed towards training data