

CRC-CGF BIOINFORMATICS COURSE 2002

Sep 9–13 2002

Sequence Alignment

Paul Pallaghy

p.pallaghy@unimelb.edu.au

Biochemistry Dept.

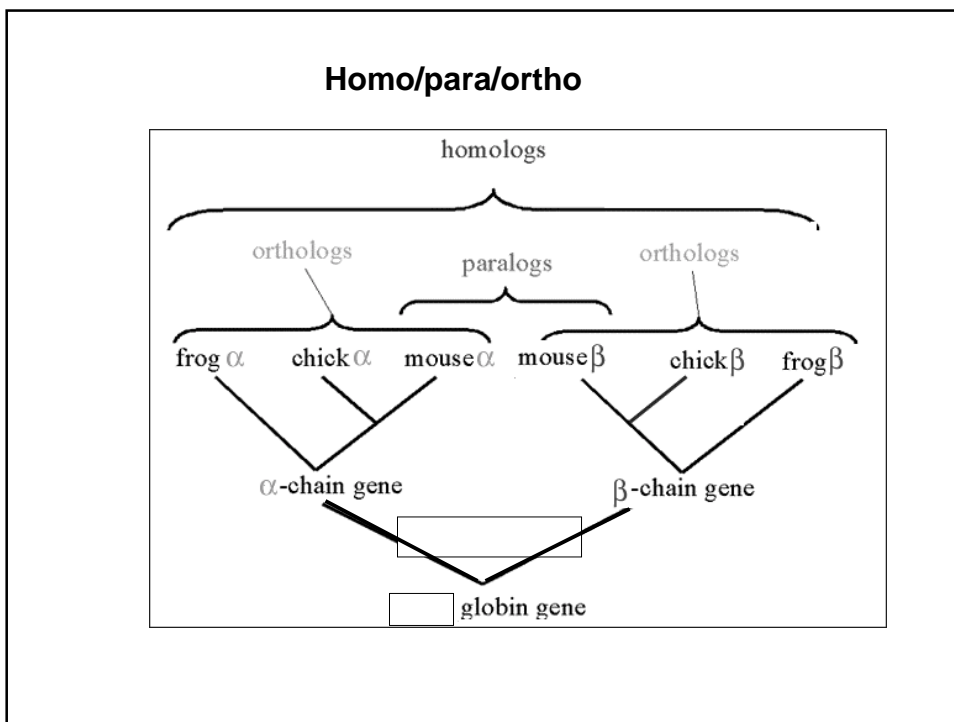
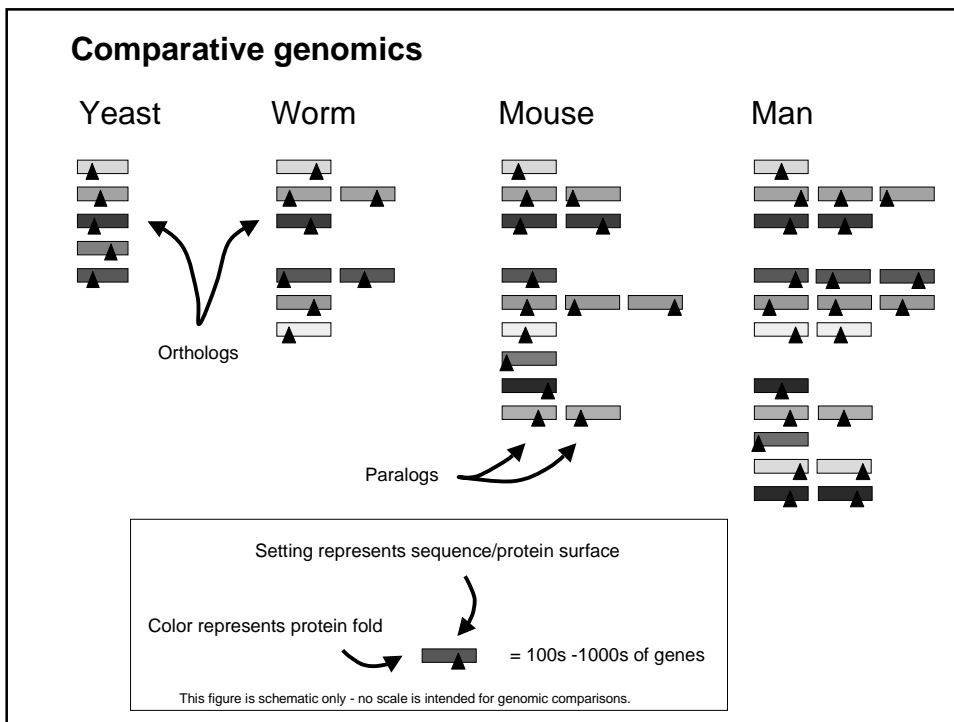
University of Melbourne

- **Why all these alignments?**
- **Types of sequence alignment**
- **Dot plots**
- **Examples**
- **Scoring**
- **Optimal vs. heuristic**
- **Global vs. local**
- **Significance**

Why all these alignments? Genomes are:

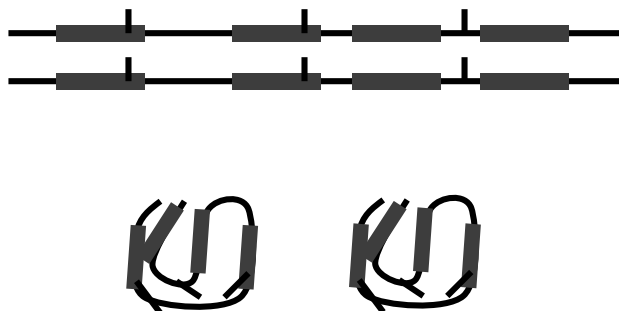
- **Chemo-physically special**
 - Genomic sequences are special – not random: fold, stable, lots of secondary structure & are functional -> Chemo-physically special.
 - Protein Superfamilies share little sequence similarity but have similar 3D structures and may be detectable by sophisticated alignment methods such as 'threading'.
- **Taxonomically/evolutionarily related**
 - Genomes consist of a finite set of distinct protein families.
 - Protein families are defined to display close sequence similarity due to a taxonomic/evolutionary relationship indicating shared:
 - Basic Function
 - Structure
 - Folding

So we can suggest a basic function for our 'query' sequence if there is a good match ('hit' or 'target') in the databases with functional 'annotation' and/or a 3D structure. At the very least good matches across taxa suggest functional importance of the query gene.



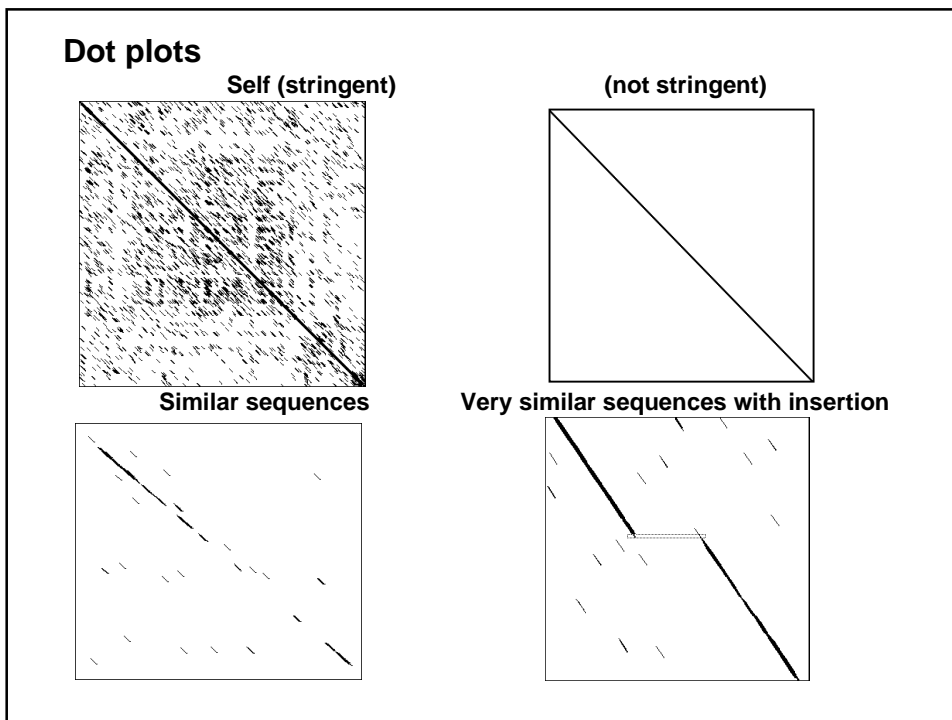
Structurally relevant alignments

- The desire is for **STRUCTURALLY** relevant alignments
- If structurally relevant then, if the function is indeed the same, the crucial residues will usually align in a functionally relevant manner



Types of alignments

- **Sequence to sequence alignments**
 - Optimal alignments
 - 'Dynamic programming' algorithms
 - 'Heuristic' alignments – close enough to optimal
 - BLAST
 - FASTA
 - Gapped/ungapped
 - Global/local
- **Profile alignments: (sequence alignment to 'profile' to sequence)**
 - Build up chemo-physical position specific pattern
 - PSI-BLAST
 - Hidden Markov Models (HMMer)
- **Motif alignments (motif to sequence)**
 - eg: Glutamate mutase motif: DHXXG
 - PROSITE
 - PHI-BLAST
- **Sequence to structure alignments (manifestly)**
 - Threading



Alignments: Example #1

| | | |
|-------|---|----|
| | : . : : . : | |
| seq1 | F A E V L S S S P W I T C E L I L V T G | 21 |
| seq2 | F A G F N R S S S G I M Y A Q K I D N S | 21 |
| seq3 | Y N G D L I A T G P W S M Q N Q L I C V T | 21 |
| ruler | 1 10 20 . | |

1. Identities

| | |
|-----|---|
| 1-2 | 6 |
| 1-3 | 4 |
| 2-3 | 3 |

Example #2

```

      : .   : : : : :   :
seq1  FAAEVLSSSPWTTCELILVTG      21
seq2  FAGFNRSSSSSGTMYAOKIDNS    21
seq3  YNGDLIATGPWSMCNQLICVT     21
ruler 1.....10.....20.
  
```

2. Identities + Similarities

1-2 6 + 1 = 7

2-3 4 + 3 = 7

1-3 3 + 8 = 11

Same "value" for identities and similarities ?

All similarities equally valid ?

Gaps ??

Example #3

```

      : .   : : : : :   . : * :
seq1  FAAEVLSSSPWTT-C-ELILVTG    21
seq2  FAGFNRSSSSSGTMYA-QKIDNS-   21
seq3  YNGDLIATGPWSM-CNQLICVT-    21
ruler 1.....10.....20...
  
```

Global multiple alignment of the three sequences.

Example #4

| | | | | | | | | | | | |
|---------------------------|----------------|------------|----------|----|---|---|---|---|---|----|----|
| | **. | ***. | * | .: | * | : | | | | | |
| seq1 | FAAEVLSSSPWTT | - | CELILVTG | | | | I | S | s | iG | tG |
| seq2 | FAGFNRSSSSGTM | YAQKIDNS | - | | | | 7 | 2 | 3 | 1 | 1 |
| : * : :. . : * * : | | | | | | | | | | | |
| seq2 | FAG-FNRSSSSGTM | YAQKIDNS | | | | | I | S | s | iG | tG |
| seq3 | YNGDLI-ATGPW | SMCNQLICVT | | | | | 4 | 5 | 2 | 2 | 0 |
| : . : : : : . : * : * * * | | | | | | | | | | | |
| seq1 | FAAEVLSSSPWTT | C- | ELILVTG | | | | I | S | s | iG | tG |
| seq3 | YNGDLIATGPW | SMCNQLICVT | - | | | | 7 | 8 | 2 | 1 | 1 |

Same "value" internal and terminal gaps ?

All similarities equally valid ?

Global alignment or local alignments ?

Example #5

| | | | | | | | | | | | |
|---------------------------|----------------|------------|----------|----|---|---|---|---|---|----|----|
| | **. | ***. | * | .: | * | : | | | | | |
| seq1 | FAAEVLSSSPWTT | - | CELILVTG | | | | I | S | s | iG | tG |
| seq2 | FAGFNRSSSSGTM | YAQKIDNS | - | | | | 7 | 2 | 3 | 1 | 1 |
| score = 16 | | | | | | | | | | | |
| : * : :. . : * * : | | | | | | | | | | | |
| seq2 | FAG-FNRSSSSGTM | YAQKIDNS | | | | | I | S | s | iG | tG |
| seq3 | YNGDLI-ATGPW | SMCNQLICVT | | | | | 4 | 5 | 2 | 2 | 0 |
| score = 7 | | | | | | | | | | | |
| : . : : : : . : * : * * * | | | | | | | | | | | |
| seq1 | FAAEVLSSSPWTT | C- | ELILVTG | | | | I | S | s | iG | tG |
| seq3 | YNGDLIATGPW | SMCNQLICVT | - | | | | 7 | 8 | 2 | 1 | 1 |
| score = 41 | | | | | | | | | | | |

7 11 9

Substitutions & gaps

Substitution matrix elements

Log-odds ratio is the log of the probability that a substitution would occur relative to random (i.e. normalized by frequencies of occurrence of amino-acids).

$$s(a,b) = \log \left(\frac{P_{ab}}{q_a q_b} \right)$$

Substitution matrix element for a to b

Frequency that a -> b

Background amounts of a and b

Log so that the substitution values can be added rather than multiplied across the sequence length

Gap penalties

- Linear

$$\text{penalty} = -gd \quad \text{where} \quad \begin{array}{l} g = \text{gap size} \\ d = \text{gap penalty (eg: 8)} \end{array}$$

- Affine

$$\text{penalty} = -d - (g - 1)e \quad \text{where} \quad \begin{array}{l} g = \text{gap size} \\ d = \text{gap opening penalty (eg: 12)} \\ e = \text{gap extension penalty (eg: 2)} \end{array}$$

BLOSUM

The BLOSUM family is now the most commonly used and has replaced the PAM matrices. It is Based on a database of 1,572 substitutions in 71 groups of closely related proteins

Multiple alignments of conserved blocks of distantly related proteins are used directly, rather than extrapolate the evolutionary process as an accumulation / repetition of the changes observed in very closely related sequences. Assumes protein divergence is the result of multiple accumulated, uncorrelated mutations. Alignment to one sequence can be based only on "averages" and "independency", as there is no additional information

+: fits better the data, are superior to PAM in detecting relationships in database searches.

-: less related to a model of evolution. Problem: Sampling bias (mainly small globular proteins)

Local Alignment Scores

Range of Alignment

ATTGTCAAAGACTTGAGCTGATGCAT
GGCAGACATGA-CTGACAAGGGTATCG

Mismatch **Gap**

$S = \sum(\text{identities, mismatches}) - \sum(\text{gap penalties})$

Score = **Max(S)** => extended if score(ext) > 0

| | | |
|-----------------------|-----------|-----------------------|
| BLOSUM 80 | BLOSUM 62 | BLOSUM 45 |
| PAM 1 | PAM 120 | PAM 250 |
| <i>Less divergent</i> | ←—————→ | <i>More divergent</i> |

Scores and Significance I

Gumbel distribution for 2 sequences => expected number of local alignments with score S or greater

m , n = sequence lengths; K = scale parameter for database size
 lambda = scale parameter for the scores

standardised
score scale

↓

$$E = Kmn e^{-\lambda S} \quad S \Rightarrow E = f(S) \quad (1)$$

$$S' = \frac{\lambda S - \ln K}{\ln 2} \quad S' = \text{"bit score"} \quad (2)$$

$$E = mn 2^{-S'} \quad (3)$$

Scores and Significance II

a times value S, exp E: approx. Poisson distribution:

$$e^{-E} \frac{E^a}{a!} \quad S \Rightarrow E = f(S) \quad (4)$$

$$P = 1 - e^{-E} \quad (5)$$

P = estimated probability for NOT finding any match with score S or higher if there are no homologs in the database.

So: high S => small E, small P => is probably a homolog (ortholog or paralog)

Low gap penalty

```

LGB1_PEA.pep  --GFTDKQE-ALVNSSSEFKQNLPGYSILFYTIVLEKAPAAKGLF-SF--LKDTAGVEDS
HBHU.pep      MVHLTPEEKSAVTALWGKVVNDEVGGEALGRLLVVY--PWTQRFFESFGDLSTPDAVMGN
                * . * . * * * * * * * * * * * * * * * * * * * * * * * *
LGB1_PEA.pep  PKLQAHAEQVFLVRDSAAQLR-TKGEVVLGNATLGAIHVQKGVVNP-HFVVVKEALLQT
HBHU.pep      PKVKAHGKKVLGAFSDGLAHLNLRKGTG----ATLSELHCDKLHVDPENFRLLGNVLCV
                **..** . * * * * * * * * * * * * * * * * * * * * * * *
LGB1_PEA.pep  IKKASGNNWSEELNTAWEVAYDGLATAIKKAMKTA
HBHU.pep      LAHHFGKEFTPPVQAAQKVVAGVANAL--AHKYH
                . . * . . . . * . * . * . * *
    
```

High gap penalty

```

LGB1_PEA.pep  ---GFTDKQEALVNSSEFKQNLPGYSILFYTIVLEKAPAAKGLFSFLKDTAGVEDSPK
HBHU.pep      MVHLTPEEKSAVTALWGKVNVDVGGGEALGRLLVVYPWTQRFFESFGDLSTPDAVMGNPK
                . *                . * . .                . * .                * *                * *                **

LGB1_PEA.pep  LQAHAEQVFGLVRDSAAQLRRTKGEVVLGNATLGAIHVQKGVTPHFFVVKEALLQTIKKA
HBHU.pep      VKAHGKKVLGAFSDGLAHLNLTGTFATLSELHCDKLVDPEN--FRLGNVLCVLAHH
                ..** . * *                * * . *                . * . . . .                * * . .                * . . .

LGB1_PEA.pep  SGNNWSEELNTAWEVAYDGLATAIKKAMKTA
HBHU.pep      FGKEFTPPVQAAYQKVVAGVANALAHKYH--
                * . . . . * .                * . * . * . . .
    
```

Pairwise alignment by computer program

Use various **ALGORITHMS** which are sets of rules that the computer program **implements**. The types of rules that people have found give **optimal alignments** use '**DYNAMIC PROGRAMMING**' - an optimization formalism (that isn't necessarily connected to computer programming).

- Breaks the problem into sub-problems
- Trace back the best sequence
- Can be proven to give the optimal sequence
- Varieties dependent on local/global/gap penalty types

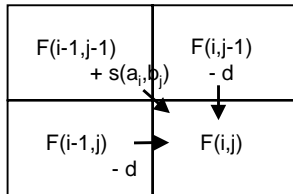
Global alignment: Needleman-Wunsch algorithm

Put together by Needleman & Wunsch (1970)

- Create a matrix $F(i,j)$ for the two sequences
- Start at N-terminus and create a cumulative score as you go from point to point. Going diagonally means no gaps, going down/right means a gap is introduced.
- So just add substitution scores = s (diagonally) or gap penalties = d (down/right) but only for the best local path

• Mathematically

$$F(i,j) = \max \begin{cases} F(i-1,j-1) + s(a_i,b_j) \\ F(i-1,j) - d \\ F(i,j-1) - d \end{cases}$$



with $F(i,0) = -id$ $F(0,j) = -jd$

Eg: Needleman-Wunsch global alignment

Sequences:

HEAGAWGHEE

PAWHEAE

| | | | | | | | | | | | | | | | | | | | | | | | | |
|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| A | R | N | D | C | Q | E | G | H | I | L | K | M | F | P | S | T | W | Y | V | B | Z | X | * | |
| A | 5 | -2 | -1 | -2 | -1 | -1 | 0 | -2 | -1 | -2 | -1 | -1 | -3 | -1 | 1 | 0 | -3 | -2 | 0 | -2 | -1 | -1 | -5 | |
| R | -2 | 7 | -1 | -2 | -4 | 1 | 0 | -3 | 0 | -4 | -3 | 3 | -2 | -3 | -3 | -1 | -1 | -3 | -1 | -3 | -1 | 0 | -1 | -5 |
| N | -1 | -1 | 7 | 2 | -2 | 0 | 0 | 0 | 1 | -3 | -4 | 0 | -2 | -4 | -2 | 1 | 0 | -4 | -2 | -3 | 4 | 0 | -1 | -5 |
| D | -2 | -2 | 2 | 8 | -4 | 0 | 2 | -1 | -1 | -4 | -4 | -1 | -4 | -5 | -1 | 0 | -1 | -5 | -3 | -4 | 5 | 1 | -1 | -5 |
| C | -1 | -4 | -2 | -4 | 13 | -3 | -3 | -3 | -2 | -2 | -2 | -3 | -2 | -2 | -4 | -1 | -1 | -5 | -3 | -1 | -3 | -2 | -5 | |
| Q | -1 | 1 | 0 | 0 | -3 | 7 | 2 | -2 | 1 | -3 | -2 | 2 | 0 | -4 | -1 | 0 | -1 | -1 | -1 | -3 | 0 | 4 | -1 | -5 |
| E | -1 | 0 | 0 | 2 | -3 | 2 | 6 | -3 | 0 | -4 | -3 | 1 | -2 | -3 | -1 | -1 | -1 | -3 | -2 | -3 | 1 | 5 | -1 | -5 |
| G | 0 | -3 | 0 | -1 | -3 | -2 | -3 | 8 | -2 | -4 | -4 | -2 | -3 | -4 | -2 | 0 | -2 | -3 | -3 | -4 | -1 | -2 | -2 | -5 |
| H | -2 | 0 | 1 | -1 | -3 | 1 | 0 | -2 | 10 | -4 | -3 | 0 | -1 | -1 | -2 | -1 | -2 | -3 | 2 | 4 | 0 | 0 | -1 | -5 |
| I | -1 | -4 | -3 | -4 | -2 | -3 | -4 | -4 | 5 | 2 | -3 | 2 | 0 | -3 | -3 | -1 | -3 | -1 | 4 | -4 | -3 | -1 | -5 | |
| L | -2 | -3 | -4 | -4 | -2 | -2 | -3 | -4 | -3 | 2 | 5 | -3 | 3 | 1 | -4 | -3 | -1 | -2 | -1 | 1 | -4 | -3 | -1 | -5 |
| K | -1 | 3 | 0 | -1 | -3 | 2 | 1 | -2 | 0 | -3 | -3 | 6 | -2 | -4 | -1 | 0 | -1 | -3 | -2 | -3 | 0 | 1 | -1 | -5 |
| M | -1 | -2 | -2 | -4 | -2 | 0 | -2 | -3 | -1 | 2 | 3 | -2 | 7 | 0 | -3 | -2 | -1 | -1 | 0 | 1 | -3 | -1 | -1 | -5 |
| F | -3 | -3 | -4 | -5 | -2 | -4 | -3 | -4 | -1 | 0 | 1 | -4 | 0 | 8 | -4 | -3 | -2 | 1 | 4 | -1 | -4 | -4 | -2 | -5 |
| P | -1 | -3 | -2 | -1 | -4 | -1 | -1 | -2 | -2 | -3 | -4 | -1 | -3 | -4 | 10 | -1 | -1 | -4 | -3 | -2 | -1 | -2 | -5 | |
| S | -1 | -1 | 1 | 0 | -1 | 0 | -1 | 0 | -1 | -3 | -3 | 0 | -2 | -3 | -1 | 5 | 2 | -4 | -2 | -2 | 0 | 0 | -1 | -5 |
| T | 0 | -1 | 0 | -1 | -1 | -1 | -2 | -2 | -1 | -1 | -1 | -1 | -2 | -1 | 2 | 5 | -3 | -2 | 0 | 0 | -1 | 0 | -5 | |
| W | -3 | -3 | -4 | -5 | -5 | -1 | -3 | -3 | -3 | -2 | -3 | -1 | 1 | -4 | -4 | -3 | 15 | 2 | -3 | -5 | -2 | -3 | -5 | |
| Y | -2 | -1 | -2 | -3 | -3 | -1 | -2 | -3 | 2 | -1 | -1 | -2 | 0 | 4 | -3 | -2 | -2 | 2 | 8 | -1 | -3 | -2 | -1 | -5 |
| V | 0 | -3 | -3 | -4 | -1 | -3 | -3 | -4 | -4 | 4 | 1 | -3 | 1 | -1 | -3 | -2 | 0 | -3 | -1 | 5 | -4 | -3 | -1 | -5 |
| B | -2 | -1 | 4 | 5 | -3 | 0 | 1 | -1 | 0 | -4 | -4 | 0 | -3 | -4 | -2 | 0 | 0 | -5 | -3 | -4 | 5 | 2 | -1 | -5 |
| Z | -1 | 0 | 0 | 1 | -3 | 4 | 5 | -2 | 0 | -3 | -3 | 1 | -1 | -4 | -1 | 0 | -1 | -2 | -2 | -3 | 2 | 5 | -1 | -5 |
| X | -1 | -1 | -1 | -2 | -1 | -1 | -2 | -1 | -1 | -1 | -1 | -1 | -1 | -2 | -2 | -1 | 0 | -3 | -1 | -1 | -1 | -1 | -1 | -5 |
| * | -5 | -5 | -5 | -5 | -5 | -5 | -5 | -5 | -5 | -5 | -5 | -5 | -5 | -5 | -5 | -5 | -5 | -5 | -5 | -5 | -5 | -5 | -5 | -5 |

Dynamic programming matrix $F(i,j)$

| | | | | | | | | | | | |
|---|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| | H | E | A | G | A | W | G | H | E | E | |
| 0 | -8 | -16 | -24 | -32 | -40 | -48 | -56 | -64 | -72 | -80 | |
| P | -8 | -2 | -9 | -17 | -25 | -33 | -42 | -49 | -57 | -65 | -73 |
| A | -16 | -10 | -3 | -4 | -12 | -20 | -28 | -36 | -44 | -52 | -60 |
| W | -24 | -18 | -11 | -6 | -7 | -15 | -5 | -13 | -21 | -29 | -37 |
| H | -32 | -14 | -18 | -13 | -8 | -9 | -13 | -7 | -3 | -11 | -19 |
| E | -40 | -22 | -8 | -16 | -16 | -9 | -12 | -15 | -7 | 3 | -5 |
| A | -48 | -30 | -16 | -3 | -11 | -11 | -12 | -12 | -15 | -5 | 2 |
| E | -56 | -38 | -24 | -11 | -6 | -12 | -14 | -5 | -12 | -9 | 1 |

BLOSUM50

Eg: Needleman-Wunsch global alignment

TRACING BACK Dynamic programming matrix F(i,j)

| | | | | | | | | | | | |
|---|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| | | H | E | A | G | A | W | G | H | E | E |
| 0 | | -8 | -16 | -24 | -32 | -40 | -48 | -56 | -64 | -72 | -80 |
| P | -8 | -2 | -9 | -17 | -25 | -33 | -42 | -49 | -57 | -65 | -73 |
| A | -16 | -10 | -3 | -4 | -12 | -20 | -28 | -36 | -44 | -52 | -60 |
| W | -24 | -18 | -11 | -6 | -7 | -15 | -5 | -13 | -21 | -29 | -37 |
| H | -32 | -14 | -18 | -13 | -8 | -9 | -13 | -7 | -3 | -11 | -19 |
| E | -40 | -22 | -8 | -16 | -16 | -9 | -12 | -15 | -7 | 3 | -5 |
| A | -48 | -30 | -16 | -3 | -11 | -11 | -12 | -12 | -15 | -5 | 2 |
| E | -56 | -38 | -24 | -11 | -6 | -12 | -14 | -5 | -12 | -9 | 1 |

ALIGNMENT

HEAGAWGHE-E

--P-AW-HEAE

Heuristic algorithms

For searching whole databases and genomes dynamics programming is too slow. Database size = 10^8 (100 million residues) and the query might be 10^3 (1000 amino=acids). So 10^{11} comparisons. Even at 10^7 per second that takes 1000s = 17 minutes.

'Heuristic' approximations have been developed that usually give almost the same results. These methods do not guarantee optimal alignments.

- BLAST

| | |
|------------------------|----------|
| Altschul et al (1990) | Ungapped |
| Altschul & Gish (1996) | Gapped |

The algorithm looks for **identities** that **seeds** further alignment based on similarities.

BLAST

| Program | Query | Database | Search level |
|---------|-------|----------|--------------|
| blastp | aa | aa | aa |
| blastn | nt | nt | nt |
| blastx | nt | aa | aa |
| tblastn | aa | nt | aa |
| tblastx | nt | nt | aa |

BLAST

Can search:

- 1) nr p Non-redundant updated protein database PDB+SwissProt+PIR
- 2) pdb p Brookhaven Protein Data Bank, current release
- 3) swissprot p SWISS-PROT current release
- 4) pir p PIR, current release
- 5) spupdate p SWISS-PROT cumulative weekly update
- 6) genpept p translations of coding sequences in GenBank
- 7) gpupdate p cumulative daily updates of GenPept
- 8) kabatpro p Kabat Proteins of immunological interest
- 9) tfd p TFD transcription factor
- 10) palu p Six-frame translations of representative human ALU repeats
- 11) nr n Non-redundant updated nucleotide database PDB+GenBank+EMBL
- 12) pdb n Brookhaven Protein Data Bank
- 13) genbank n GenBank, current release
- 14) gbupdate n GenBank cumulative daily update
- 15) embl n EMBL current release
- 16) emblu n EMBL cumulative daily update
- 17) vector n Vector subset of GenBank
- 18) repbase n Primate Alu repeats
- 19) kabatnuc n Kabat nucleotide sequences of immunological interest
- 20) epd n Eukaryotic Promotor Database 21) dbest n Expressed sequence tags

Conclusions

- Goal - to identify optimal relative positions of two sequences being compared (or, to identify optimal numbers and positions of gaps to introduce)
- This implies a choice of
 - scoring matrix
 - gap penalties
- Remember: when you make an alignment, the result is conditional upon an implicit model

Caveats

```
THESEALGORITHMSARETRYINGTOFINDTHEBESTWAYTMATCHTWOSEQUENCES  
TH S+          + T Y FIND          YT SE  
THISDESN'TMEANT-HATTHEYWILLFINDAN-----YTHING---SEFL-----
```

- Any two sequences can be forced into alignment
- If the underlying model (*i.e.*, scoring matrix and gap penalties) is inappropriate, the alignment will be useless

Acknowledgment

- Thank-you to Mauro Delorenzi for about half of the slides from last year!