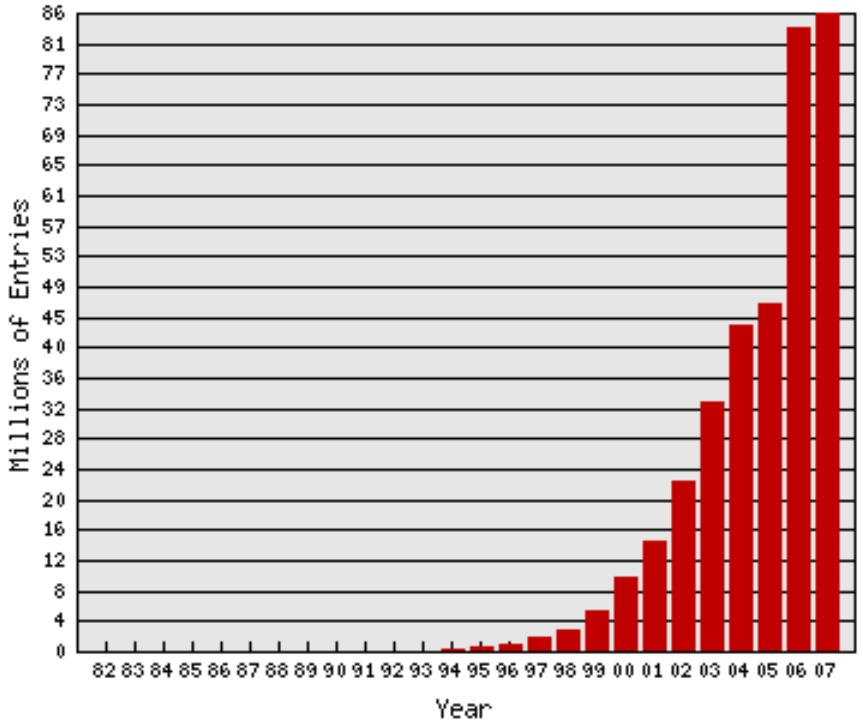
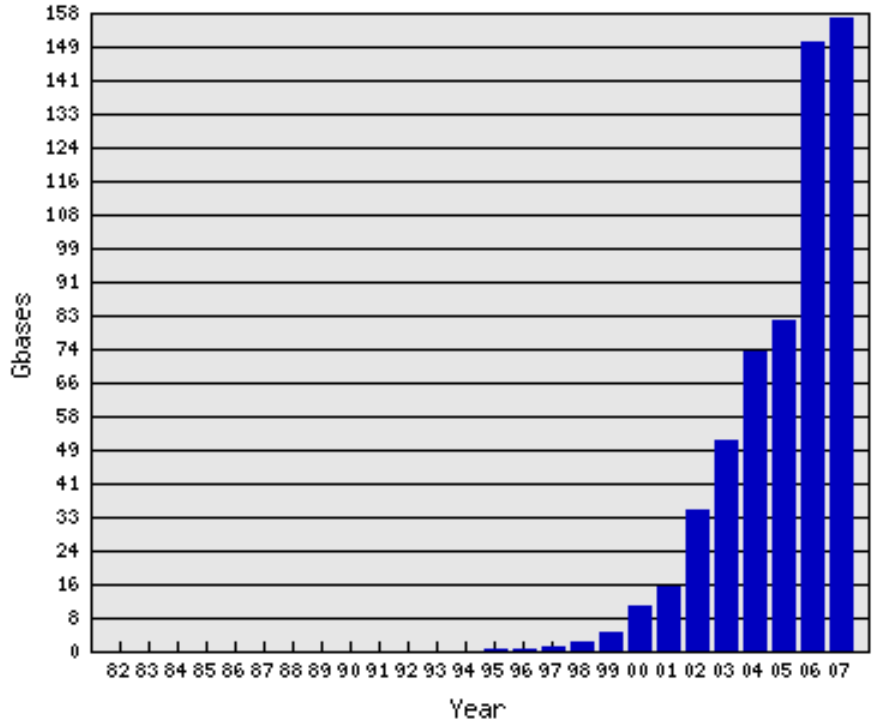
Database searches

DNA and protein databases

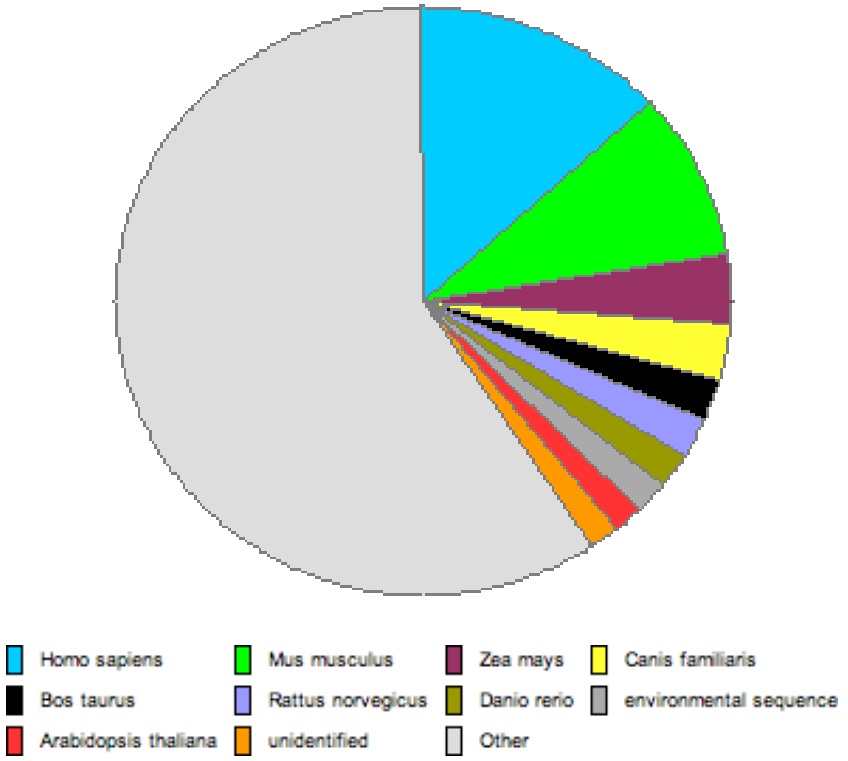
- EMBL/GenBank/DDBJ database of nucleic acids

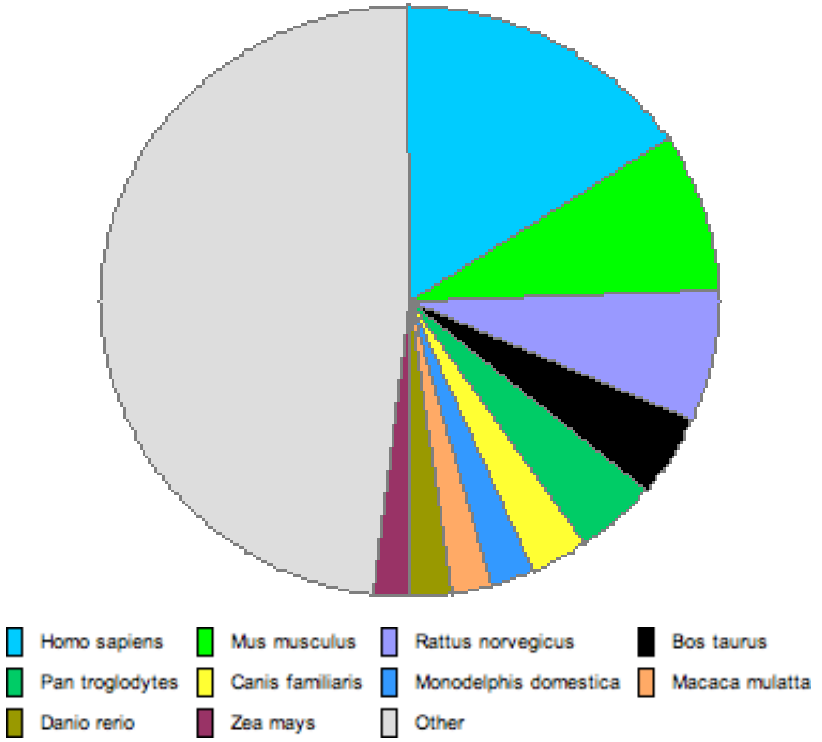




DNA and protein databases

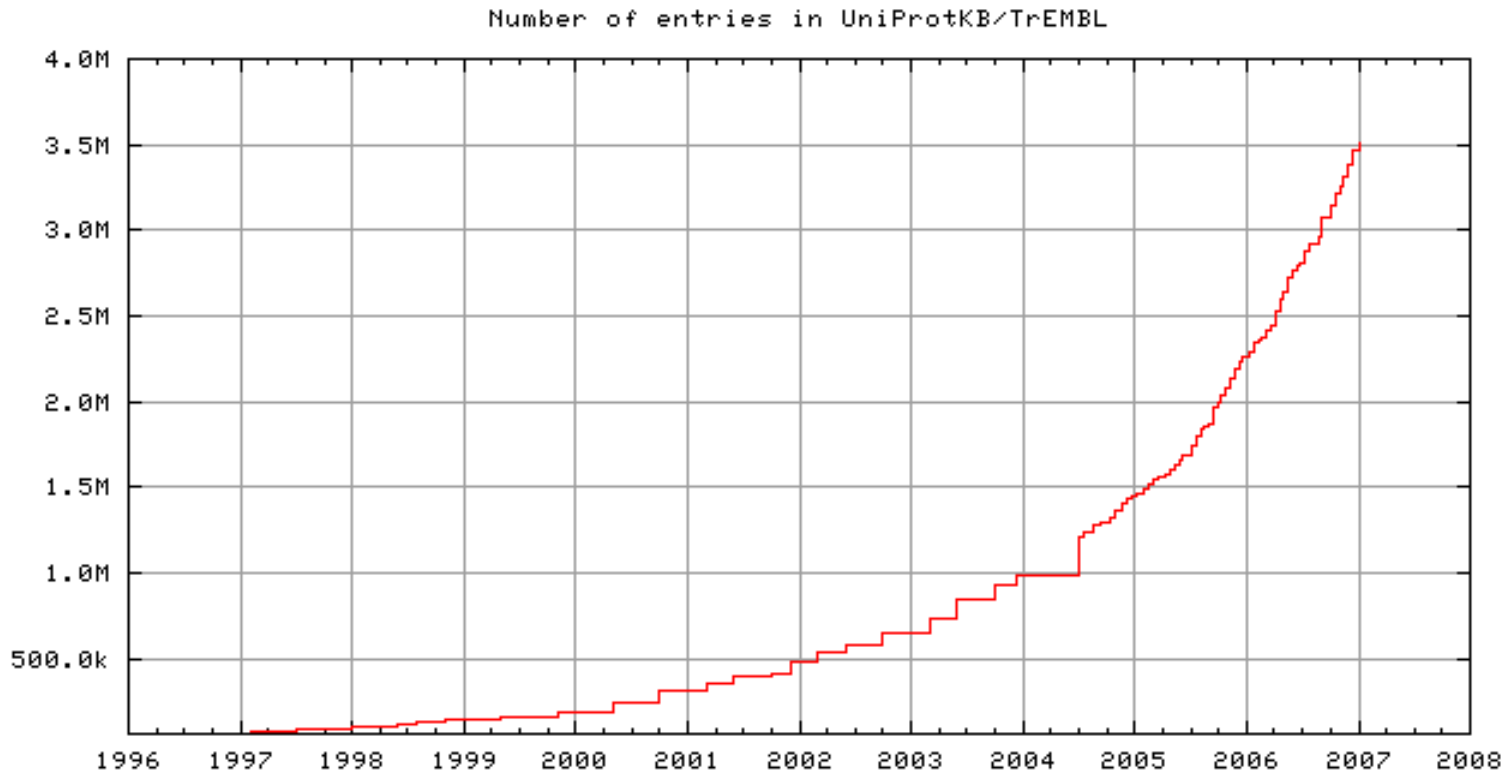
- EMBL/GenBank/DDBJ database of nucleic acids (cntd)





DNA and protein databases

- SWISS-PROT & TrEMBL database of proteins



Searching databases: motivation

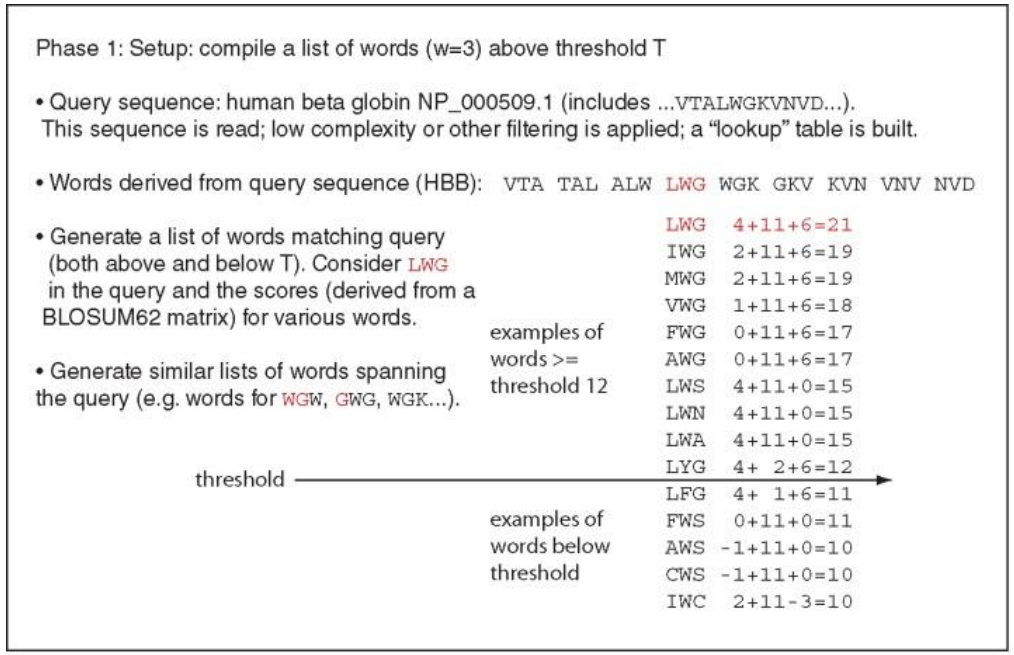
- Determine orthologs and paralogs for a protein of interest, assign putative function
  - A new bacterial genome is sequenced, how many genes have related genes in other species
- Determine if a genome contains specific types of proteins
- Determine the identity of a DNA or protein sequence
  - What is the identity of a clinical pathogen?
- Determine if particular variant has been described before
  - Many pathogens, especially viruses, mutate rapidly. We should like to know if we have a new strain.

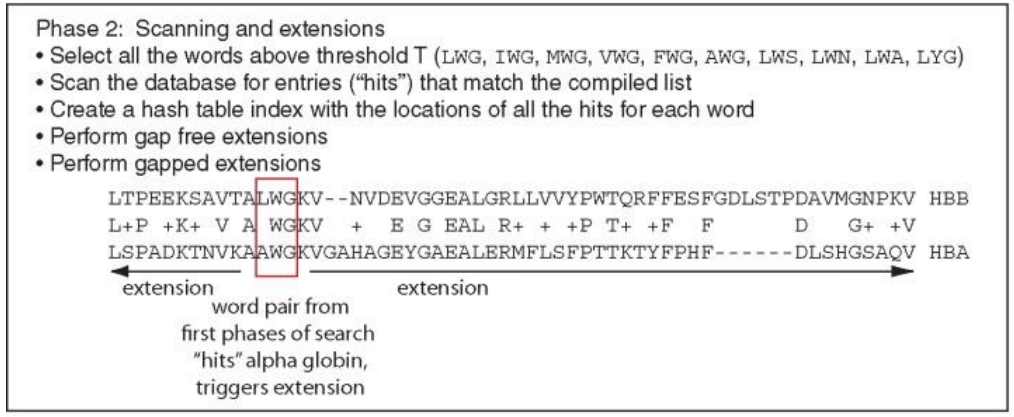
BLAST: Basic Local Alignment Search Tool

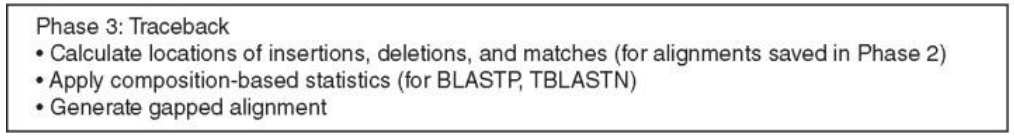
- Performs many alignments at once
- Heuristic algorithms are used instead of DP. *Why?*
  - Size of SWISS-PROT + TrEMBL (Rel. 9.5):
    3.9M entries or 1,276M residues.
  - Exact algorithms are O(*NM*) fast.
- Heuristic methods can look at a small fraction of the searching space that will include all (or most) of the high scoring pairs.
- Web interface and standalone program

BLASTP algorithm

- Idea: search space is reduced by looking only at small exact (or near exact matches)--**words**
- List – compile the list of words using the query
- Scan –scan the reference for matches
  - HSP-high scoring segment pairs
  - Location is not stored at this point
- Extend –
  - Ungapped extension-simple
  - Gapped extension using dynamic programing
- Report hits above threshold







Compositional adjustment

- Recall the log odds score
- Background frequencies should be the the marginal frequencies of q_{ij}
- Compositional adjustment
  - Use empirical p_i p_j
  - Adjust q_{ij} accordingly

$$s_{ij} = \frac{1}{\lambda} \ln\left(\frac{q_{ij}}{p_i p_j}\right),$$

$$p_i = \sum_j q_{ij}; \quad p'_j = \sum_i q_{ij}.$$

Yu et al. The compositional adjustment of amino acid substitution matrices

Compositional adjustment

Sequence pairs	Organisms compared	No. of sequence pairs	Mean BLOSUM-62 bit score*	Background frequencies specified	Median change in bit score* with respect to BLOSUM-62		Cases improved (%)	Cases (%) with statistical significance improved/worsened by a factor >10†
					Absolute	Relative (%)		
Related	*C. tetani* and *M. tuberculosis*	40	68.3	Organism	+1.6	+2.7	58	20/8
				Sequence‡	**+2.3**	**+3.3**	**85**	**38/3**
	B. subtilis and *L. lactis*	37	59.8	Organism	+1.1	+1.8	84	16/3
				Sequence‡	**+2.1**	**+3.6**	**95**	**11/3**
	M. tuberculosis and *S. coelicolor*	34	58.6	Organism	+1.4	+2.6	76	24/3
				Sequence‡	**+2.7**	**+4.1**	**100**	**32/0**
Unrelated (negative control)	*C. tetani* and *M. tuberculosis*	1,560	16.7	Organism	−0.02	−0.1	49	0.4/0.1
				Sequence‡	−0.05	−0.3	47	0.6/0.4
	B. subtilis and *L. lactis*	1,332	15.7	Organism	+0.00	+0.0	50	0.0/0.0
				Sequence‡	+0.04	+0.3	52	0.2/0.4
	M. tuberculosis and *S. coelicolor*	1,122	16.4	Organism	+0.05	+0.3	53	0.0/0.1
				Sequence‡	+0.06	+0.4	53	0.6/0.2
Structural	Various	32	50.4	**Sequence‡**	**+1.3**	**+3.2**	**72**	**22/0**

Yu et al. The compositional adjustment of amino acid substitution matrices

(b) Query: human insulin NP_000198
Program: blastp
Database: *C. elegans* RefSeq
Option: No compositional adjustment

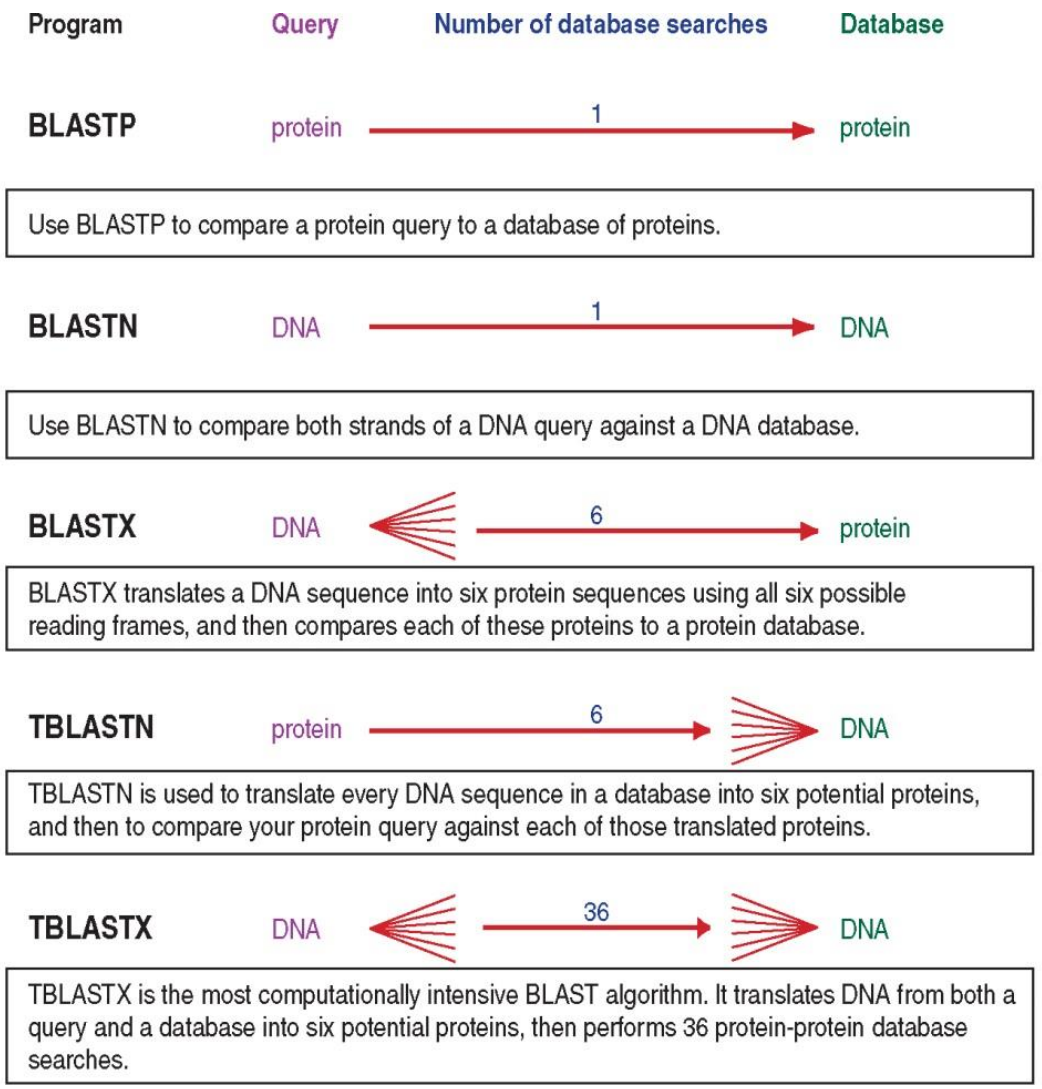
```
>ref|NP_501926.1|  INSulin related family member (ins-1) [Caenorhabditis elegans]
Length=109

 Score = 34.7 bits (78),  Expect = 0.009
 Identities = 30/100 (30%), Positives = 41/100 (41%), Gaps = 14/100 (14%)

Query  11   LALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGERGFFYTPKTRREAEDLQVGQVELGG  70
            LA+L L   P P+ A +   LCGS L    L  VC  +         +R A+
Sbjct  17   LAILLLSSPTPSDASIR--LCGSRLTTTLLAVCRNQLCTGLTAFKRSADQSY--------  66

Query  71   GPGAGSLQPLALEGSLQKRG-IVEQCCTSICSLYQLENYC  109
               A + + L      QKRG I  +CC    CS    L+ +C
Sbjct  67   ---APTTRDLFHIHHQQKRGGIATECCEKRCSFAYLKTFC  103
```

(c) Query: human insulin NP_000198
Program: blastp
Database: *C. elegans* RefSeq
Option: conditional compositional score matrix adjustment

```
>ref|NP_501926.1|  INSulin related family member (ins-1) [Caenorhabditis elegans]
Length=109

 Score = 33.5 bits (75),  Expect = 0.020, Method: Compositional matrix adjust.
 Identities = 27/100 (27%), Positives = 39/100 (39%), Gaps = 12/100 (12%)

Query  10   LLALLALWGPDPAAAFVNQHLCGSHLVEALYLVCGERGFFYTPKTRREAEDLQVGQVELG  69
            LA+L L   P P+ A +   LCGS L   L  VC  +        +R A+
Sbjct  16   FLAILLLSSPTPSDASIR--LCGSRLTTTLLAVCRNQLCTGLTAFKRSADQ--------S  65

Query  70   GGPGAGSLQPLALEGSLQKRGIVEQCCTSICSLYQLENYC  109
              P    L    +    ++ GI  +CC   CS   L+ +C
Sbjct  66   YAPTTRDL--FHIHHQQKRGGIATECCEKRCSFAYLKTFC  103
```

BLAST flavors

- BLASTN-requires exact word matched
  - Length 7-15 (11 default)
  - Requires two word pairs within some distance
    - Returns more hits but saves time in the extension phase

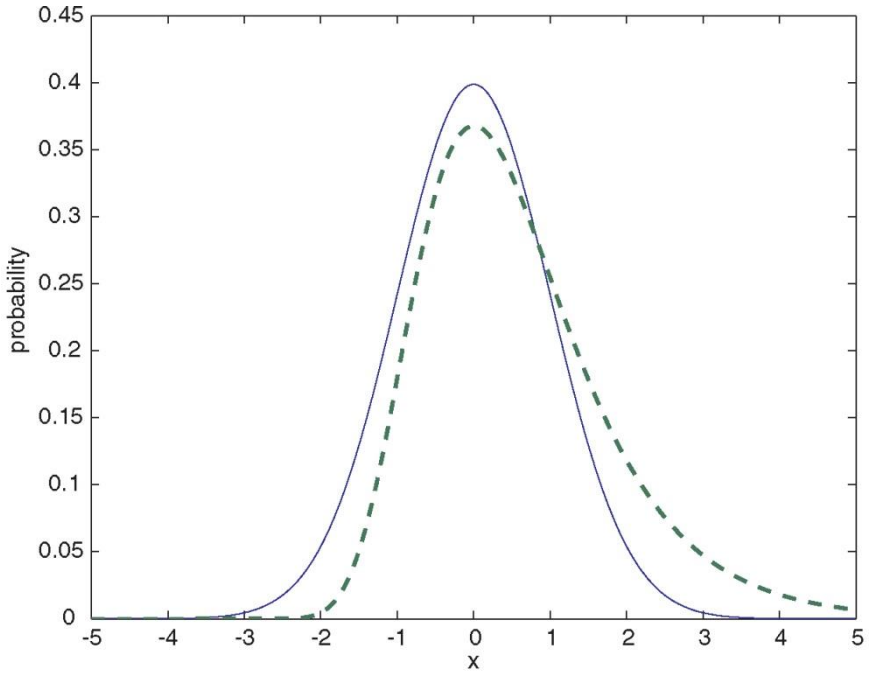


BLAST statistics

- Parameters to consider
  - Length of query – longer queries will generate more matches
  - Database size
- Raw score
- Bit score –parameter normalized score comparable across searches
- E value-expected number of sequences
- P-value-probability of a chance alignment occurring with this score or better

How to interpret a BLAST search: expect value

- It is important to assess the statistical significance of search results
- For global alignments, the statistics are poorly understood.
- For local alignments (including BLAST search results), the statistics are well understood.
- The scores follow an extreme value distributio (EVD)
  - Theoretically for ungapped alignments
  - Empirically for gapped alignments



Calculating E

- $\boldsymbol{E = Kmn\ e^{-\lambda S}}$
- This equation is derived from a description
- of the extreme value distribution
- S = the score
- m, n = the length of two sequences
- λ, K = Karlin Altschul statistics

E vs P

- Very small *E* values are very similar to *p* values.
- *E* values of about 1 to 10 are far easier to interpret
- than corresponding *p* values.

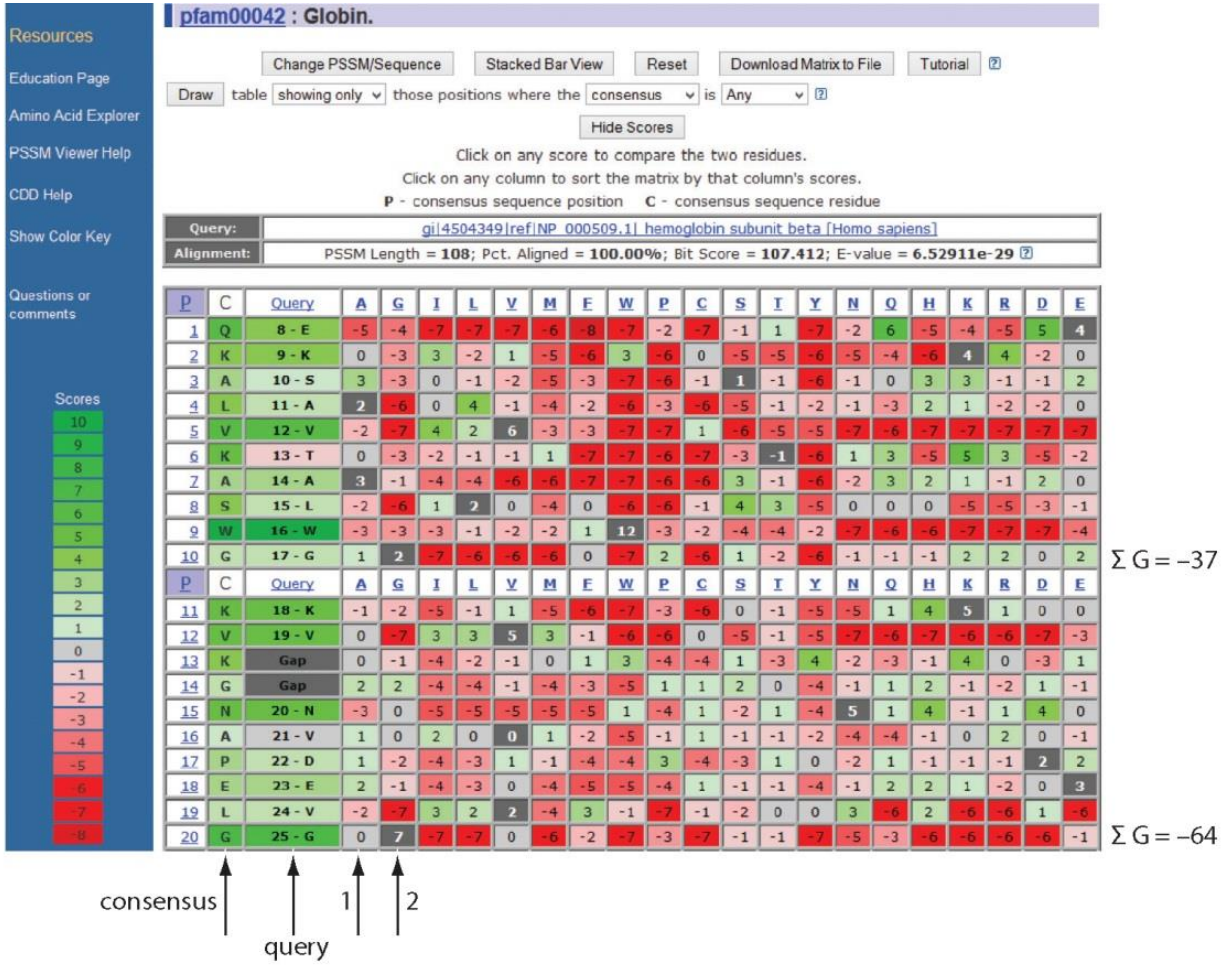
E	*p*
10	0.99995460
5	0.99326205
2	0.86466472
1	0.63212056
0.1	0.09516258 (about 0.1)
0.05	0.04877058 (about 0.05)
0.001	0.00099950 (about 0.001)
0.0001	0.0001000

Problems that BLAST can’t solve

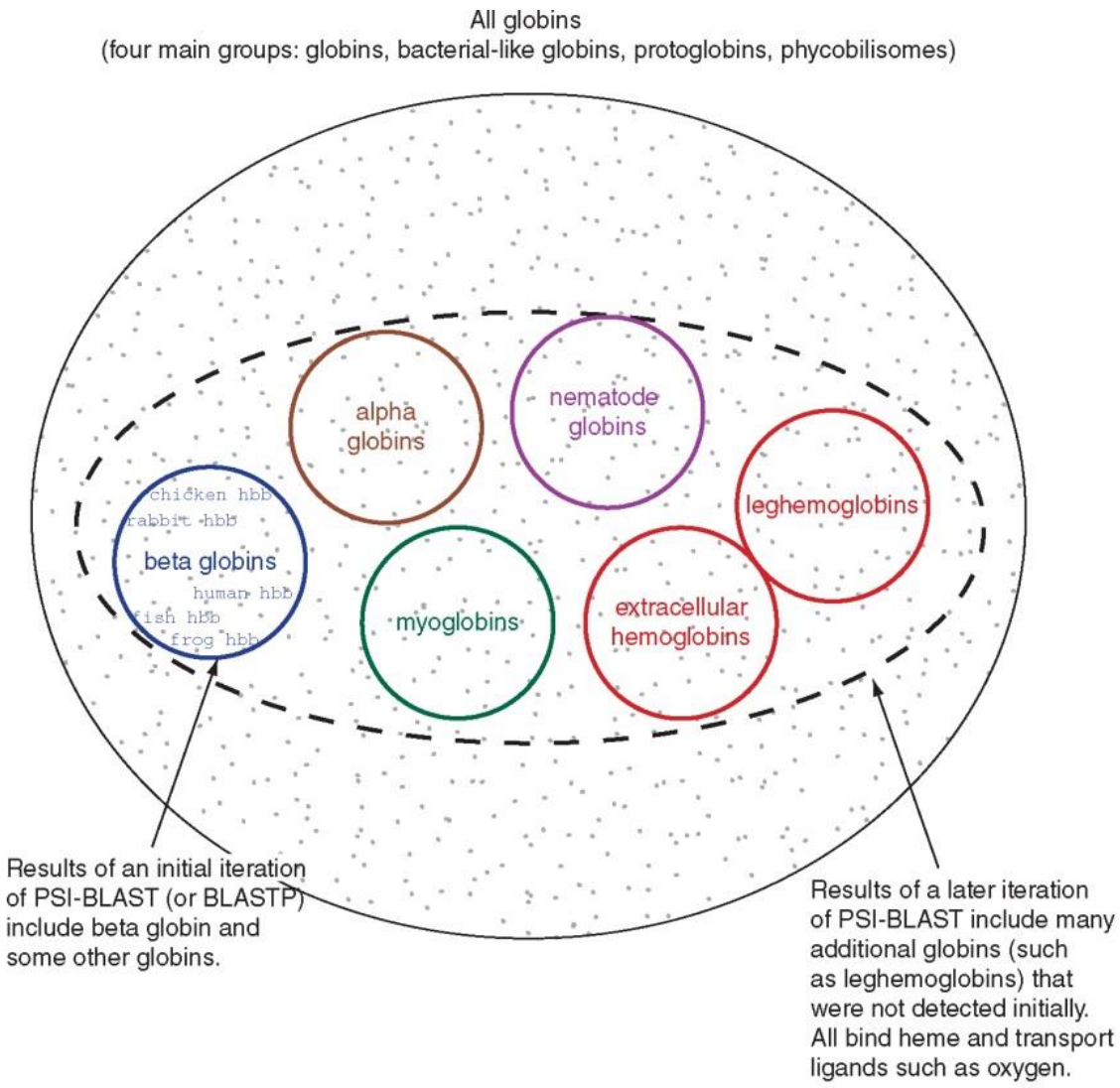
- Finding very distant homologs
  - Human myoglobin does not come up as a hit when searching with beta-globin even though they share the same structure
  - To little similarity
- Aligning large genomic segments
  - Align large chromosomal regions between mouse and human genome
  - Some regions have high similarity while others do not
- Aligning nextgen sequencing data
  - Millions of 100bp reads to 3 billion bp of human gnome – time to run BLASTN –weeks!

Distant homologs

- Position specific iterated blast PSI-BLAST
  - Perform initial search with BLASTP
  - Perform a multiple sequence alignment with results
  - Define a position-specific scoring matrix PSSM
    - Reference position if fixed
    - One dimension represents position along reference instead of amino acids
  - Use the PSSM to search the database
  - Repeat



PSI-BLAST globin family

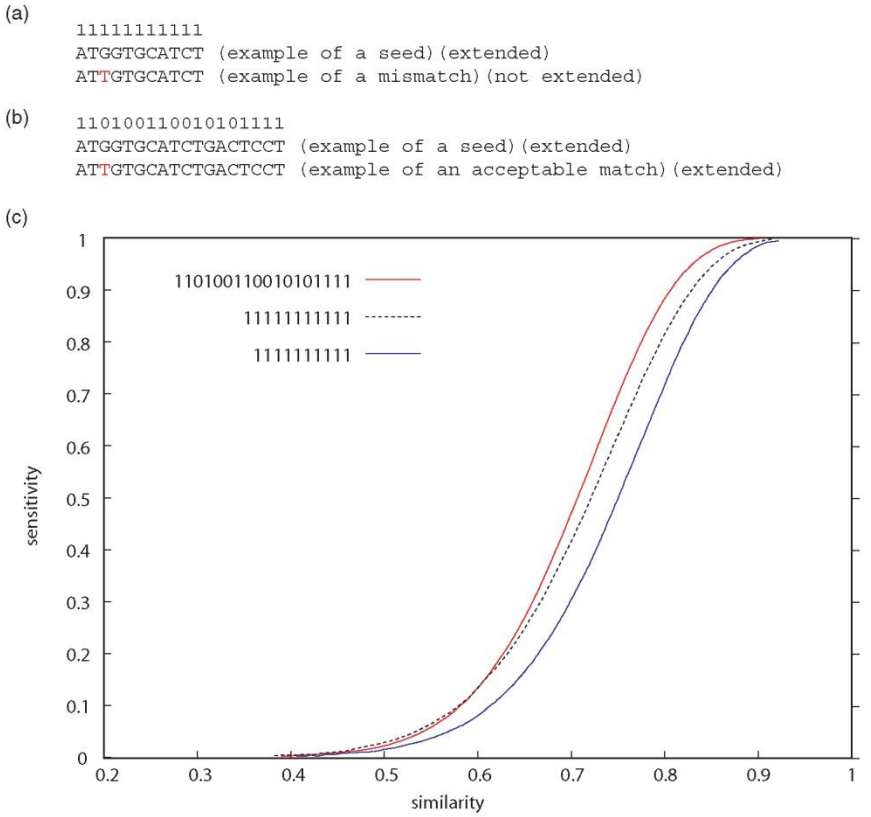


PSI-BLAST problems and pitfalls

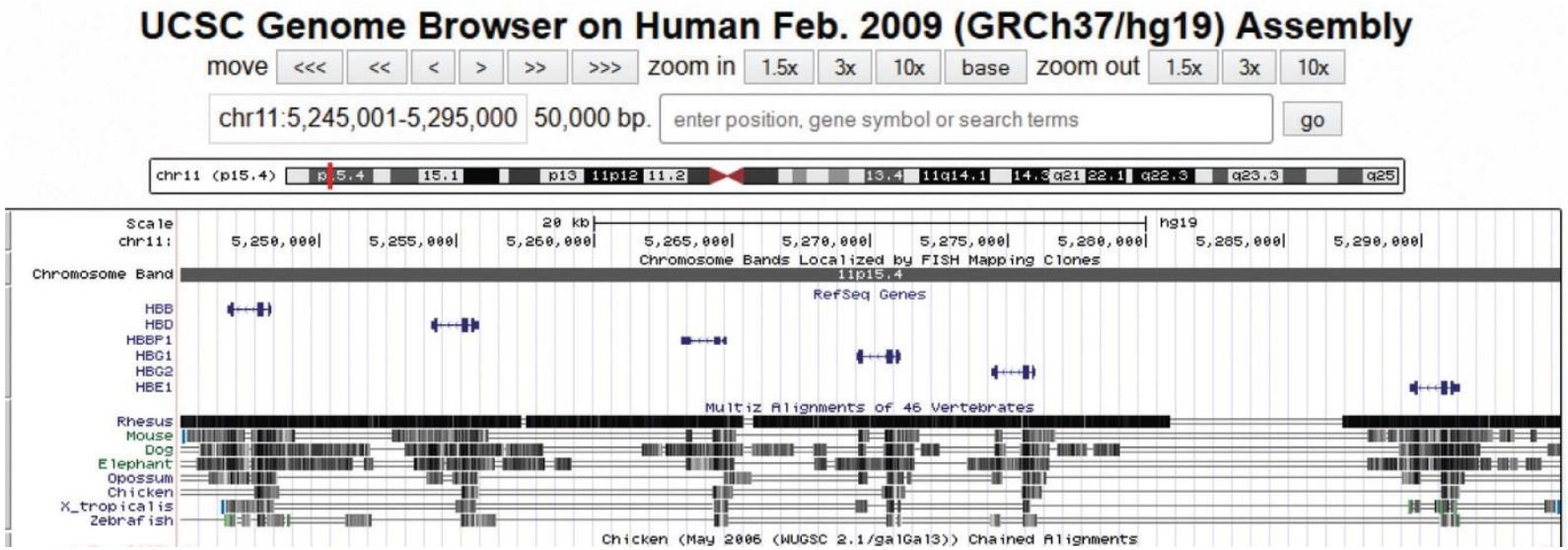
- Iterative algorithm – errors propagate
- Low entropy regions –regions with biased a.a composition can corrupt the PSSM
- Iterations can be adjusted by hand to remove suspicious sequences from the alignment

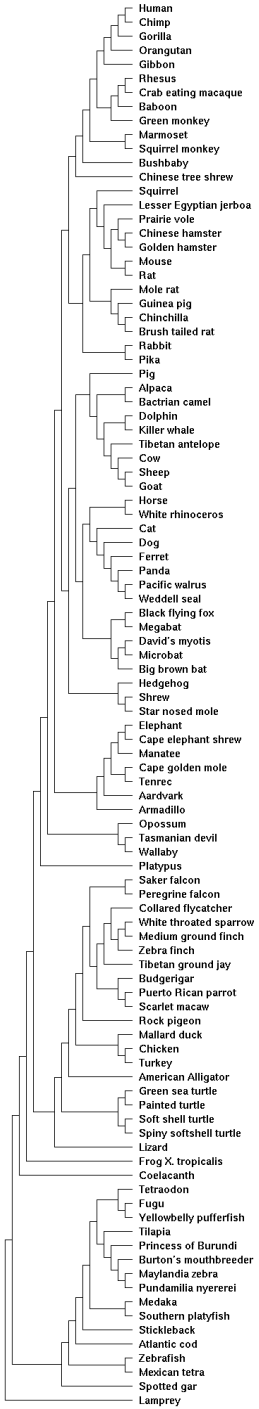
Genomic alignment BLASTZ

- PatternHunter
  - Nonconsecutive seed boost sensitivity
  - Need 11 matches spanning 18 nucleotides
  - At specific positions
  - 110100110010101111
  - 64 nucleotides with 70% identity
    - Probability of BLASTN match 0.3
    - Probability of PatternHunter match 0.47
- Other considerations
  - Removing lineage specific repeats
    - Gene duplications
    - Retrotransposons
  - Masking already aligned regions



Genome wide alignments

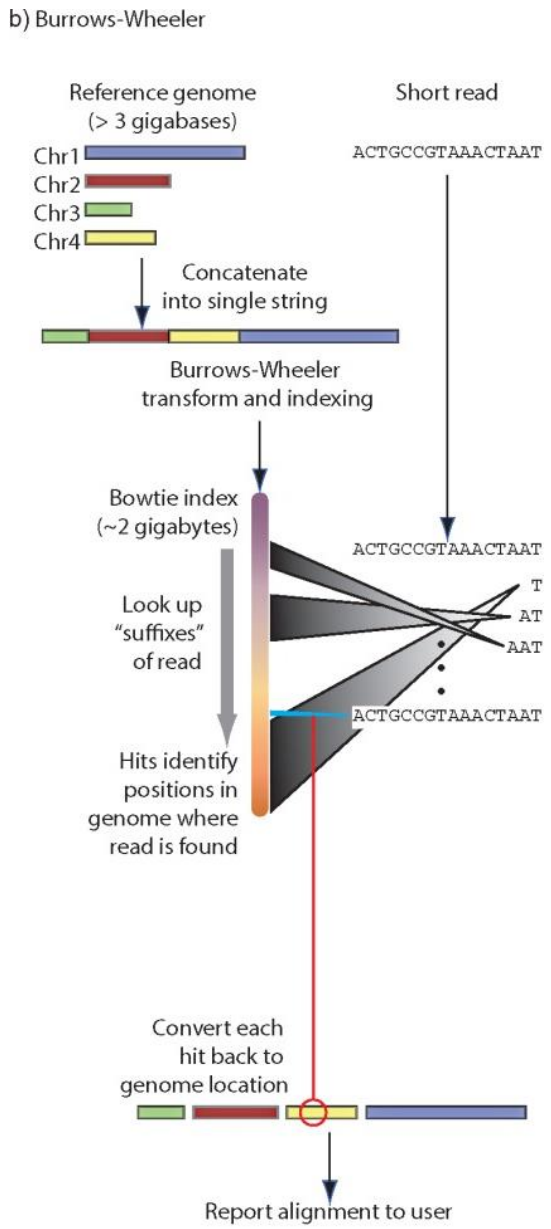




- Alignments can be viewed in UCSC
- Download alignment files and extract regions of interest
- If a regions(gene) is missing from an organism doesn’t mean it is not there
  - Incomplete alignment
  - Incomplete genome assembly
  - Use BLAST!

Short read alignment

- Bowtie -ultrafast, memory-efficient short read aligner
- Basic strategy
  - Index the genome
  - Use Burrows-Wheeler Transform BWT
  - Human genome fits entirely into RAM



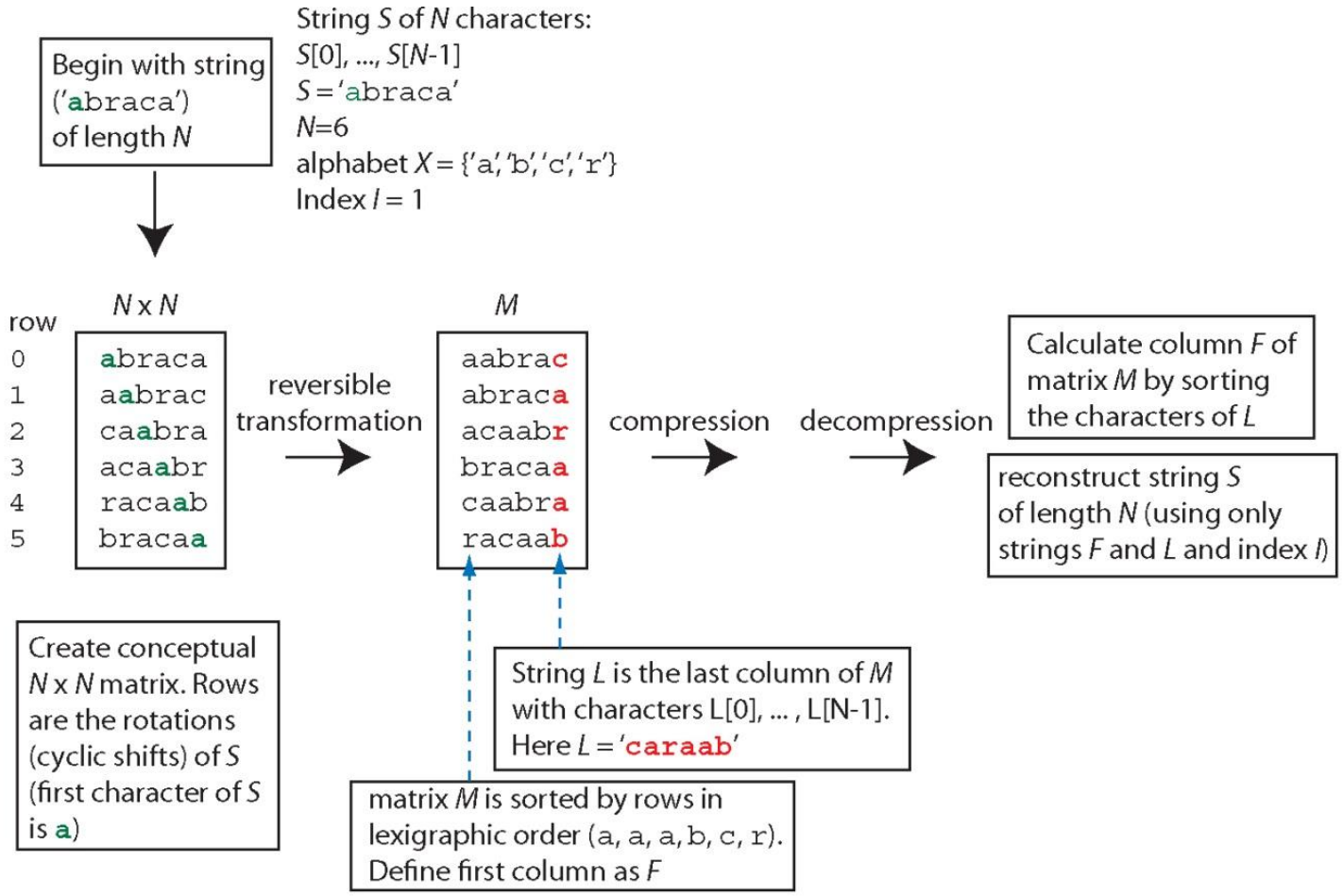
Why Burrows-Wheeler?

BWT very compact:

- Approximately ½ byte per base
- As large as the original text, plus a few “extras”
- Can fit onto a standard computer with 2GB of memory

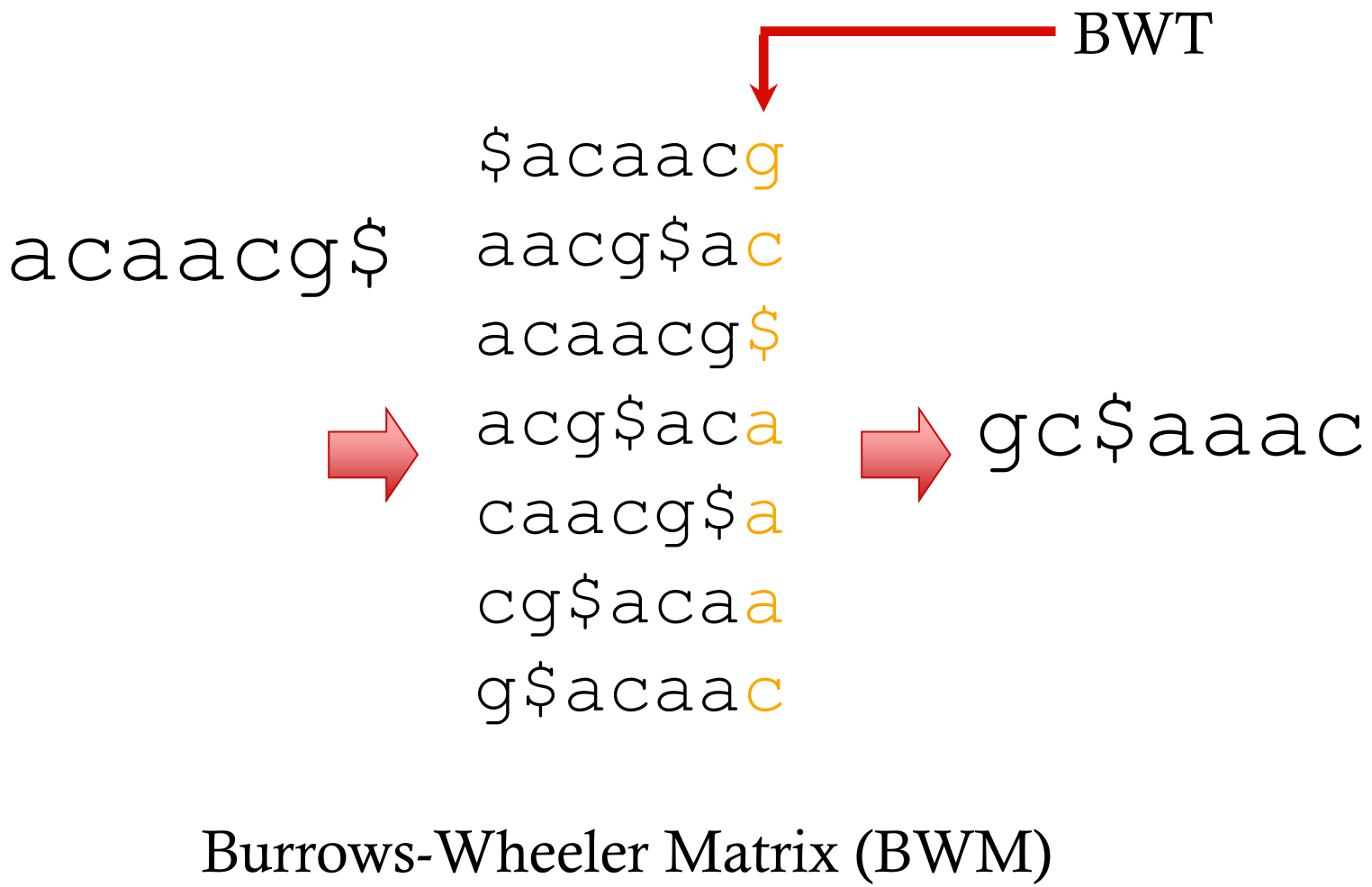
Linear-time search algorithm

- proportional to length of query for exact matches



Burrows-Wheeler Transform (BWT)

- Generate all circular permutations
- Sort by first letter
- Lest column is the BWT
- Everything else is discarded
- First column can be recovered from BWT
  - It has all the same characters sorted



Burrows-Wheeler Matrix (BWM)

Exact match

BWT(agcagcagact) = tgcc$ggaaaac

Search for pattern: gca

gca	gca	gca	gca
$agcagcagact	$agcagcagact	$agcagcagact	$agcagcagact
act$agcagcag	act$agcagcag	act$agcagcag	act$agcagcag
agact$agcagc	agact$agcagc	agact$agcagc	agact$agcagc
agcagact$agc	agcagact$agc	agcagact$agc	agcagact$agc
agcagcagact$	agcagcagact$	agcagcagact$	agcagcagact$
cagact$agcag	cagact$agcag	cagact$agcag	cagact$agcag
cagcagact$ag	cagcagact$ag	cagcagact$ag	cagcagact$ag
ct$agcagcaga	ct$agcagcaga	ct$agcagcaga	ct$agcagcaga
gact$agcagca	gact$agcagca	gact$agcagca	gact$agcagca
gcagact$agca	gcagact$agca	gcagact$agca	gcagact$agca
gcagcagact$a	gcagcagact$a	gcagcagact$a	gcagcagact$a
t$agcagcagac	t$agcagcagac	t$agcagcagac	t$agcagcagac

Inexact matching

- If match cannot be extended try mismatches
- A->C->G X
- A->T->G X
- A->G->G ✓

Multiple Sequence Alignment

- Goal: Given several sequences bring the greatest number of similar characters into the same column of the alignment
- Why?
- Correspondence. Find out which parts “do the same thing”
  - Similar genes are conserved across widely divergent species, often performing similar functions
- Structure prediction
  - Use knowledge of structure of one or more members of a protein MSA to predict structure of other members
  - Structure is more conserved than sequence
  - Predict if mutations are deleterious by looking at cross species conservation
- Create “profiles” for protein families
  - Allow us to search for other members of the family– PSI-BLAST
- MSA is the starting point for evolutionary analysis

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

Multiple alignment problem

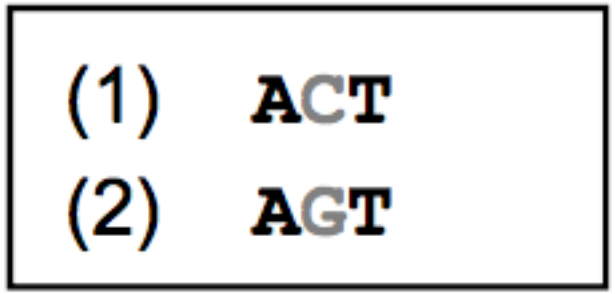
- Ribosome: an RNA/protein complex rpS14: a ribosomal protein in yeast
- Goal: Determine residues responsible for binding rpS14 to ribosomal RNA
- Known:
  - Sequence of rpS14
  - Structure of homolog in bacteria
  - Sequences in many species
- Find the MSA
- Find conserved residues
- Use structure to check for binding function

Slide credit: Dannie Durand

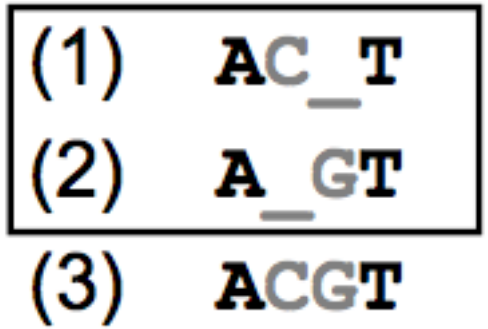
Multiple vs pairwise

Alignments should put together bases/amino acids that are related by evolution—roughly corresponds to being in the same structural and functional position

Better Score



Correct evolutionary history



Multiple Sequence Alignment: Approaches

- **Optimal Global Alignments** -Dynamic programming
  - Generalization of Needleman-Wunsch
  - Find alignment that maximizes a score function
  - Computationally expensive: Time grows as product of sequence lengths
- **Global Progressive Alignments** - Match closely-related sequences first using a guide tree

What is an optimal multiple alignment

- Sum of pairs (SOP)
- Score of multiple alignment

$$= \sum_{i<j} \text{score}(S_i, S_j)$$

- score(S_i,S_j) = score of induced pairwise alignment
- The alignment of si with sj induced by M is generated as follows
  - Remove from M all rows except i and j
  - Remove all columns that contain only blanks

Can be solved by dynamic programming

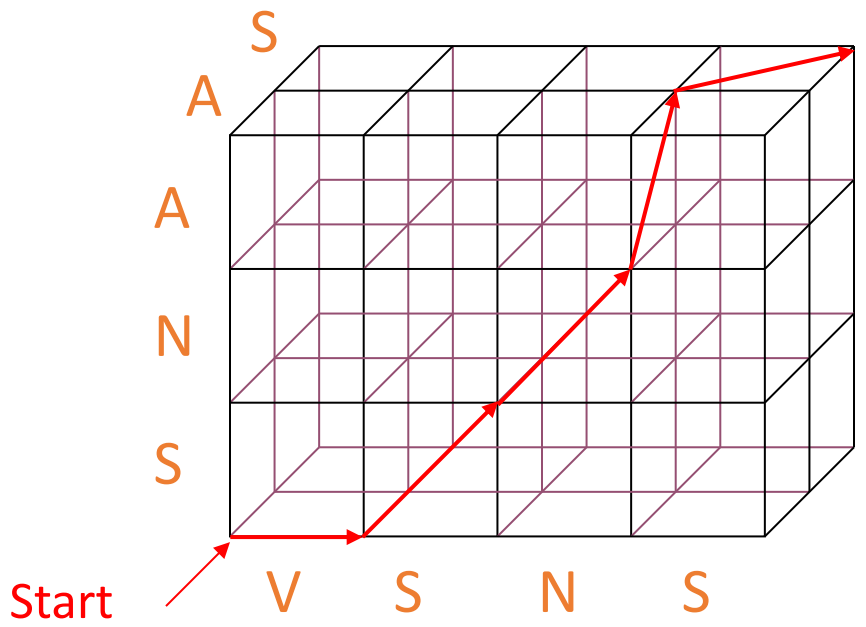
- The two-sequence alignment algorithm can be generalized to any number of sequences.
- As for two sequences, divide possible alignments into different classes, depending on how they end.
- E.g., for three sequences X, Y, W define

  C[i,j,k] = score of optimum alignment
  among X[1..i], Y[1..j], W[1..k]

Dynamic programming for MSA with k sequences

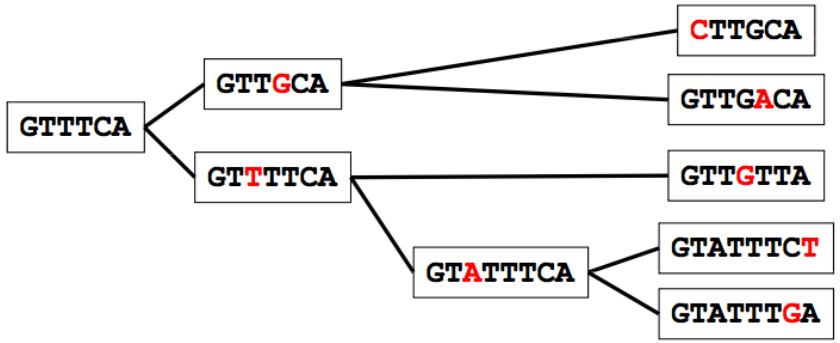
- K-dimensional “matrix”
- There are n^k cell corners in the cube
- For each corner, we need to look at 2^k-1 other corners – Together: $O(2^k n^k)$ computations
- Example: 6 sequences of length 100 require $6.4X10^{13}$ calculations
- Implementations (e.g., WashU MSA 2.1) use tricks and only search subset of dynamic programming table
  - Even this is expensive. E.g., Baylor CM Search launcher limits MSA to 8 sequences of 800 characters and 10 minutes processing time

V S N — S
— S N A —
— — — A S



Problem with sum of pairs

- Alignment should reflect the evolutionary process
- Alignment score should be related to the number of evolutionary evens
- Sum of pairs overcounts alignments
- Too much weight for evolutionary distant pairs

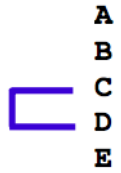



```
CT_TGC_A
GT_TGACA
GT_TGTTA
GTATTTCT
GTATTTGA
```

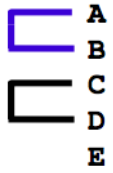
Progressive multiple alignment

- Align most closely related sequences first, how to decide the order
  - Ideally we would follow the evolutionary history--phyolgeny
  - Need and MSA to infer evolutionary history
  - Compute phylogeny that close enough

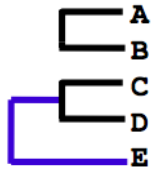
	A	B	C	D	E
A		17	59	59	77
B			37	61	53
C				13	41
D					21

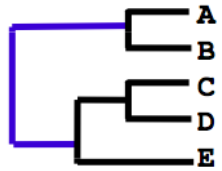


	A	B	E	CD
A		17	77	59
B			53	49
E				31

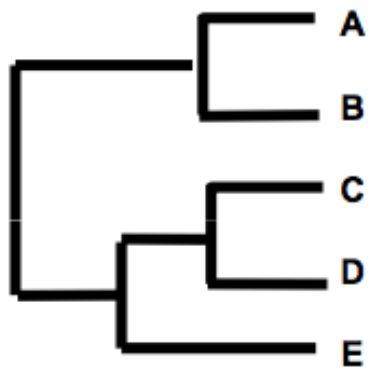


	E	CD	AB
E		31	65
CD			54





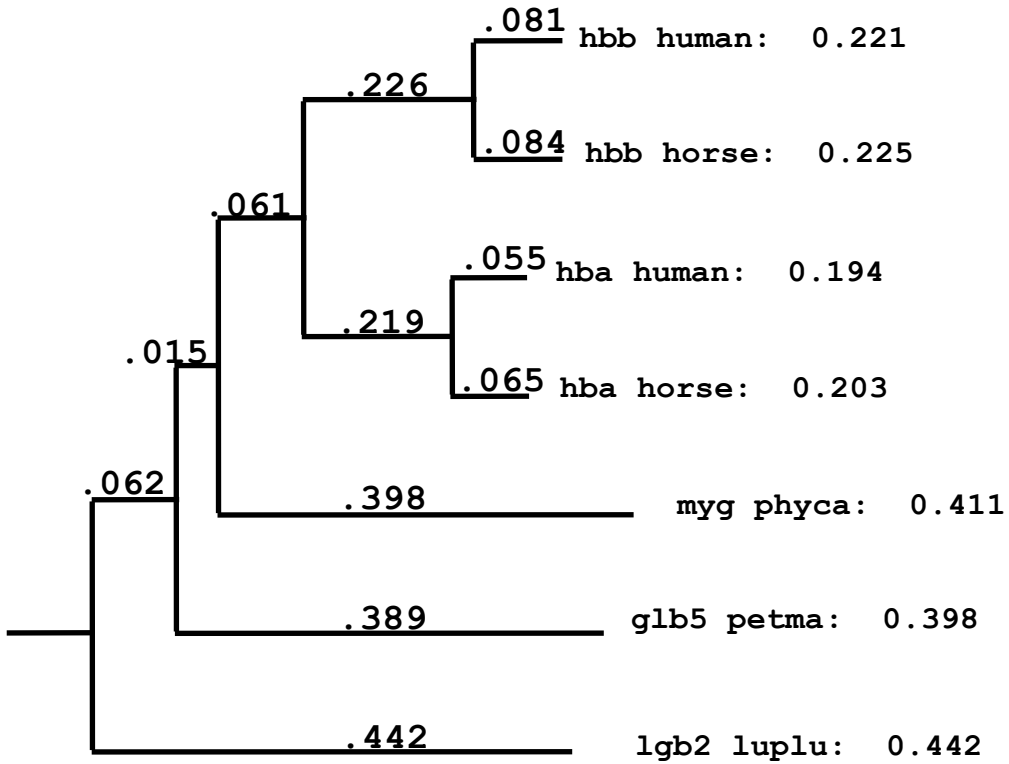
Example




```
C   PADKTNVKAAWGKVGAHAGEYGA
D   AADKTNVKAAWSKVGGHAGEYGA

A   PEEKSAVTALWGKVNVDEYGG
B   GEEKAAVLALWDKVNEEEYGG

C   PADKTNVKAAWG_KVGAHAGEYGA
D   AADKTNVKAAWS_KVGGHAGEYGA
E   AA__TNVKTAWSSKVGGHAPA__A

A   PEEKSAV_TALWG_KVN__VDEYGG
B   GEEKAAV_LALWD_KVN__EEEYGG
C   PADKTNVKAA_WG_KVGAHAGEYGA
D   AADKTNVKAA_WS_KVGGHAGEYGA
E   AA__TNVKTA_WSSKVGGHAPA__A
```

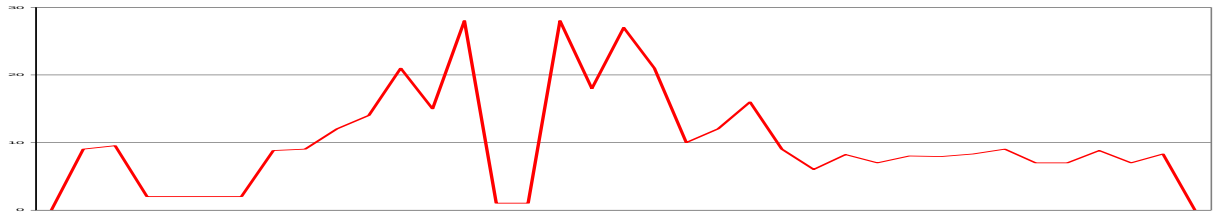
Clustalw

- Widely used progressive alignment approach
- Basic flow
  - Find pairwise scores
  - Build guide tree from pairwise distances (neighbor joining algorithm, discussed later)
  - Progressively align according to guide tree
- Need a way to score aligning partial alignments
  - Weighted average of pairwise scores
  - Weights correct for unequal sampling across evolutionary distance



Gaps in Clustalw

- Opening and extension penalties depend on
  - score matrix
  - sequence similarity,
  - sequence length,
  - position of gaps
  - residues at gaps –gaps cost less in a hydrophilic region
- Gaps should cost more if they break up a structural element and less if they are in a loop
- For further details, see Thompson *et al*., **NAR** 1994, **22:**4673 or **Methods in Enz**. 1996, **266:**article 22.



```
HLTPEEKSAVTALWGKVN--VDEVGGEALGRLLVVYPWTQRFFESFGDL
QLSGEEKAAVLALWDKVN--EEEVGGEALGRLLVVYPWTQRFFDSFGDL
VLSPADKTNVKAAWGKVGAHAGEYGAEALERMFLSFPTTKTYFPHFDLS
VLSAADKTNVKAAWSKVGGHAGEYGAEALERMFLGFPTTKTYFPHFDLS
```

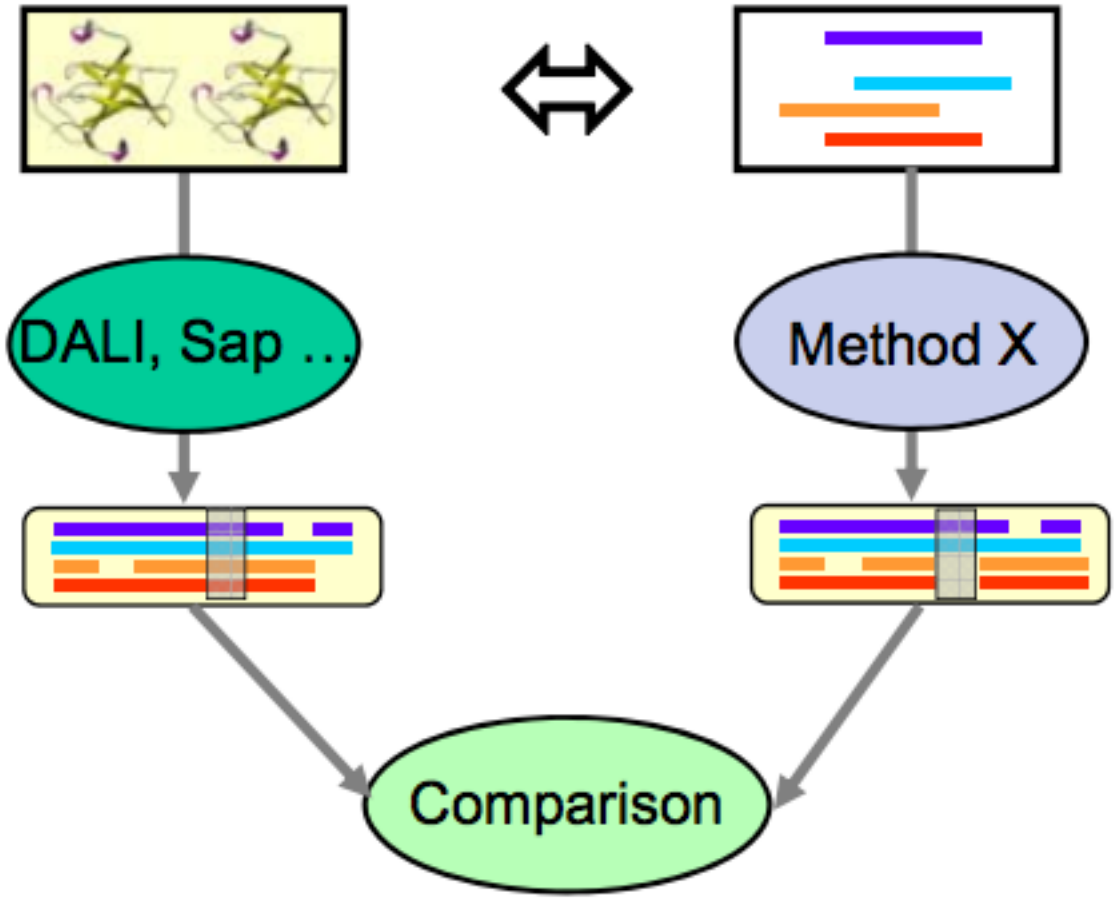
Problem with progressive alignment


```
CLUSTALW (Score=20, Gop=-1, Gep=0, M=1)

  SeqA GARFIELD THE LAST FA-T CAT
  SeqB GARFIELD THE FAST CA-T ---
  SeqC GARFIELD THE VERY FAST CAT
  SeqD -------- THE ---- FA-T CAT
```

```
CORRECT (Score=24)

  SeqA GARFIELD THE LAST FA-T CAT
  SeqB GARFIELD THE FAST ---- CAT
  SeqC GARFIELD THE VERY FAST CAT
  SeqD -------- THE ---- FA-T CAT
```

Solution: iterative refinement

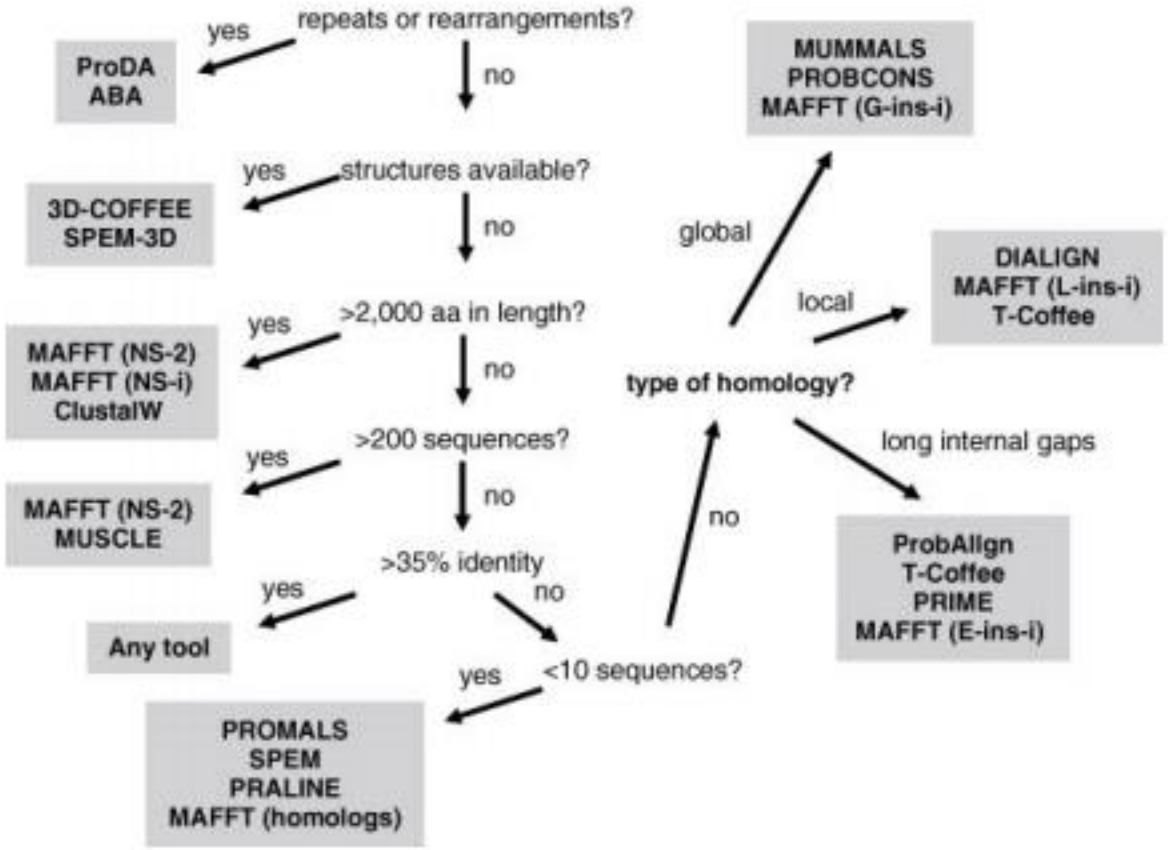
BaliBase: Reference MSA based on structure



Aligner	Performance*	Time
DIALIGN	57.2	12 h, 25 min
CLUSTALW	58.9	2 h, 57 min
T-Coffee	63.6	***144 h, 51 min***
MUSCLE	64.8	3 h, 11 min
MAFFT	64.8	***2h,36min***
ProbCons	66.9	19 h, 41 min
ProbCons-ext	***68.0***	37 h, 46 min

Do et al, Genome Research, 2005

Which program to choose

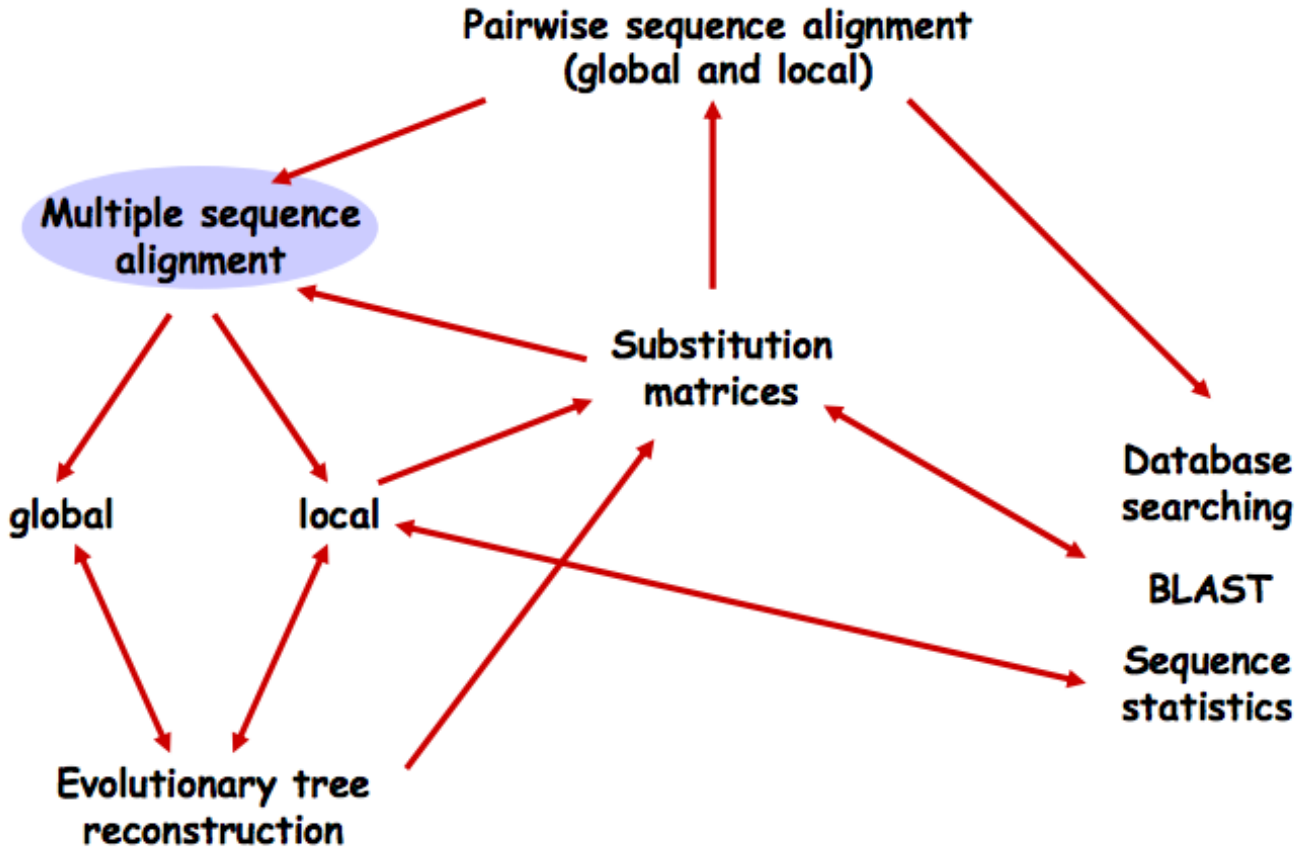


Do and Katoh, 2008

Multiple Alignments Summary

- Even below the 10-20% identity twilight zone, the best programs correctly align 47% of residues on average
- Iterative algorithms are superior, but with a large trade-off in use of computational resources
- Global generally performs better than local
- **No single ‘best’ program exists**
- For reviews, see:
  - P. Briffeuil *et al*., **Bioinformatics** 1998, **14**:357
  - J. D. Thompson *et al*., **NAR** 1999, **27**:2682

Complex dependence structure



Slide credit: Dannie Durand