BCH441 – BIOINFORMATICS

SEQUENCE ALIGNMENT

BORIS STEIPE

DEPARTMENT OF BIOCHEMISTRY – DEPARTMENT OF MOLECULAR GENETICS
UNIVERSITY OF TORONTO

section

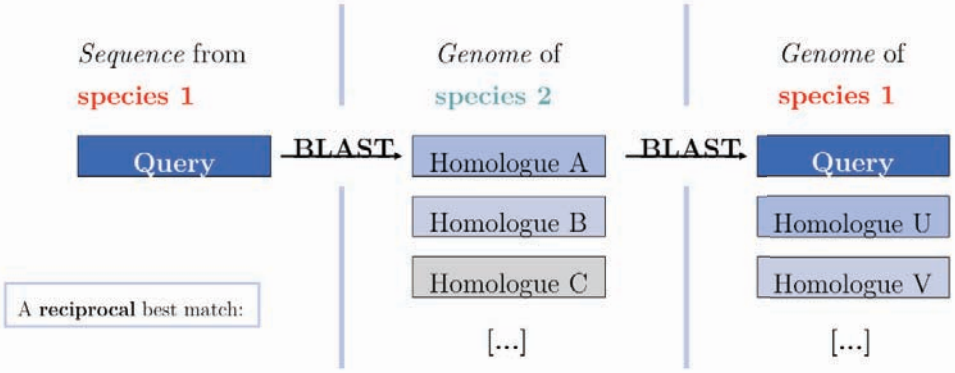
Fast Alignment: BLAST

why large search spaces?

Some procedures require **genome-wide** or **database-wide** similarity searches. Such searches are not feasible with *optimal sequence alignment* algorithms.

Example: discovering orthologues

Computationally, an orthologue has been defined when the best match in another specie's genome has the original sequence as it's own best match. This is called the "**reciprocal best-match**" criterion. *(This is a useful procedure, but not the correct definition of an orthologue.)*



An example for searches in very large search spaces is the computational definition of orthologues.

The tight integration of search capacity with database holdings is the key to the utility of the data. Public investments in sequencing **only** pay off when the sequences are easily accessible!

Some computational definitions absolutely require genome-wide searches: e.g. the computational definition of orthologues, or compiling evolutionary conservation patterns.

BLAST

Basic Local Alignment Search Tool

BLAST encompasses many different implementations and enhancements to a search algorithm that finds "**High Scoring Pairs**" of sequence alignments in databases.

It is a **Fast** way to find similar sequences.

It is **heuristic**, not exact, not optimal.

It is **not** the most **sensitive** way to search.

It is by a wide margin the **most commonly used tool** in bioinformatics.

BLAST was developed as a heuristic alternative to exact alignment, looking for a way to compute much faster, repeated searches in large search spaces than what one can do with pairwise alignments. The strategy is to pre-compute similarities and then piece a match together from quickly retrieved partial matches.

BLAST principle

1: Query Preprocessing

Break query into words

`DTLVRAIP ->  DTL, TLV, LVR, VRA ...`

Make a table of similar words

`TLV -> TLI, TIV, SLV`

2: Search query words in indexed table of database words

Find exact match between table word & db.

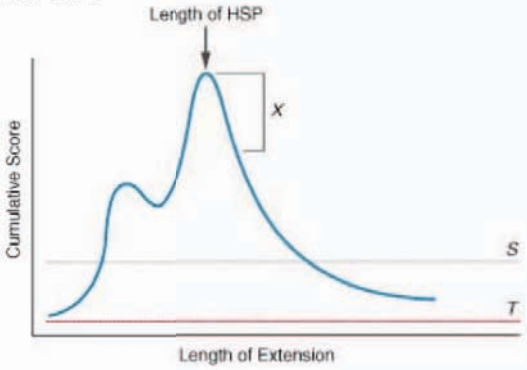
```
          TIV
SDTDGDKNADGWIETIVRALPTSD
```

3: Extend query match to HSP

– keep HSPs of significant quality.

```
          DTLVRAIP
SDTDGDKNADGWIETIVRALPTSD
```

4: Assemble HSPs into gapped local alignment

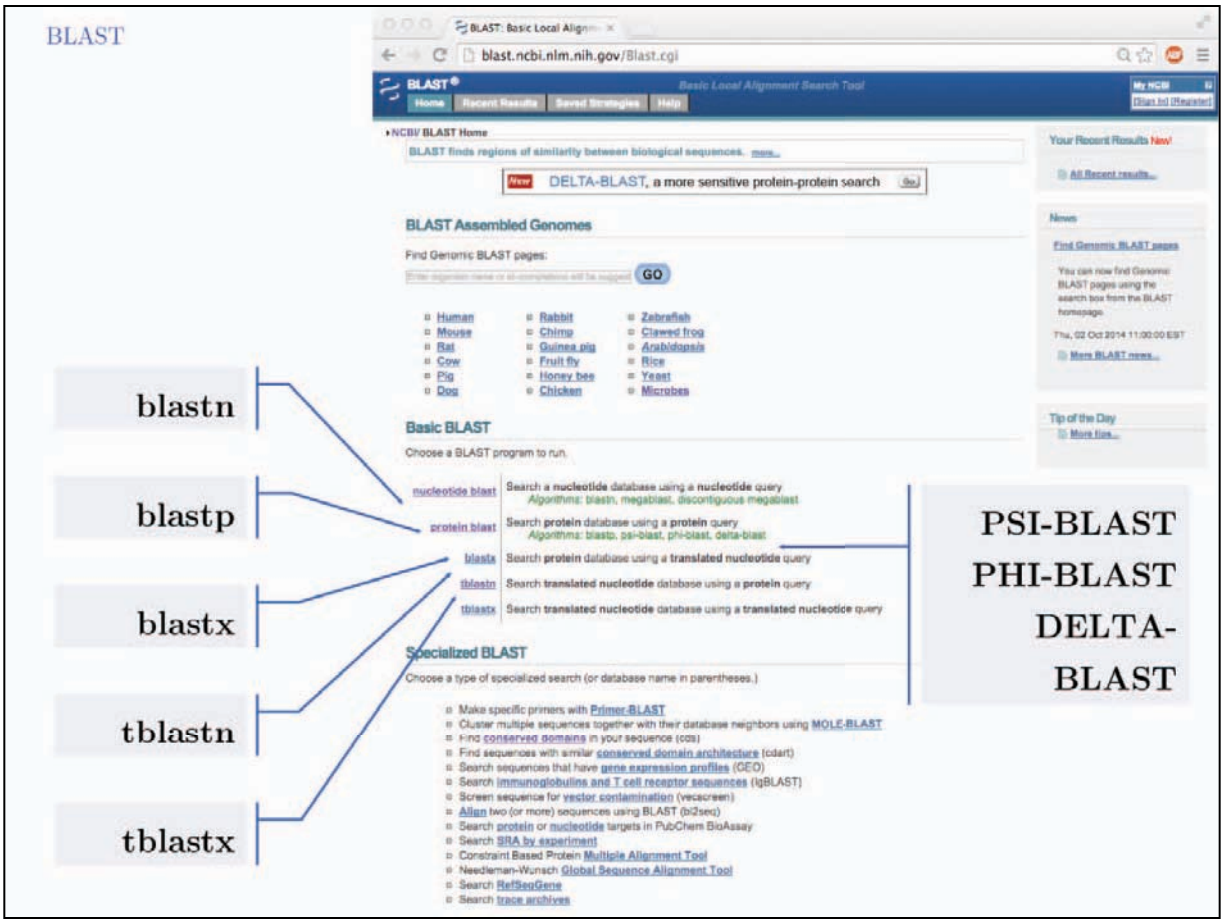


HSP (High Scoring Segment Pair): **An ungapped, high-scoring, local alignment**

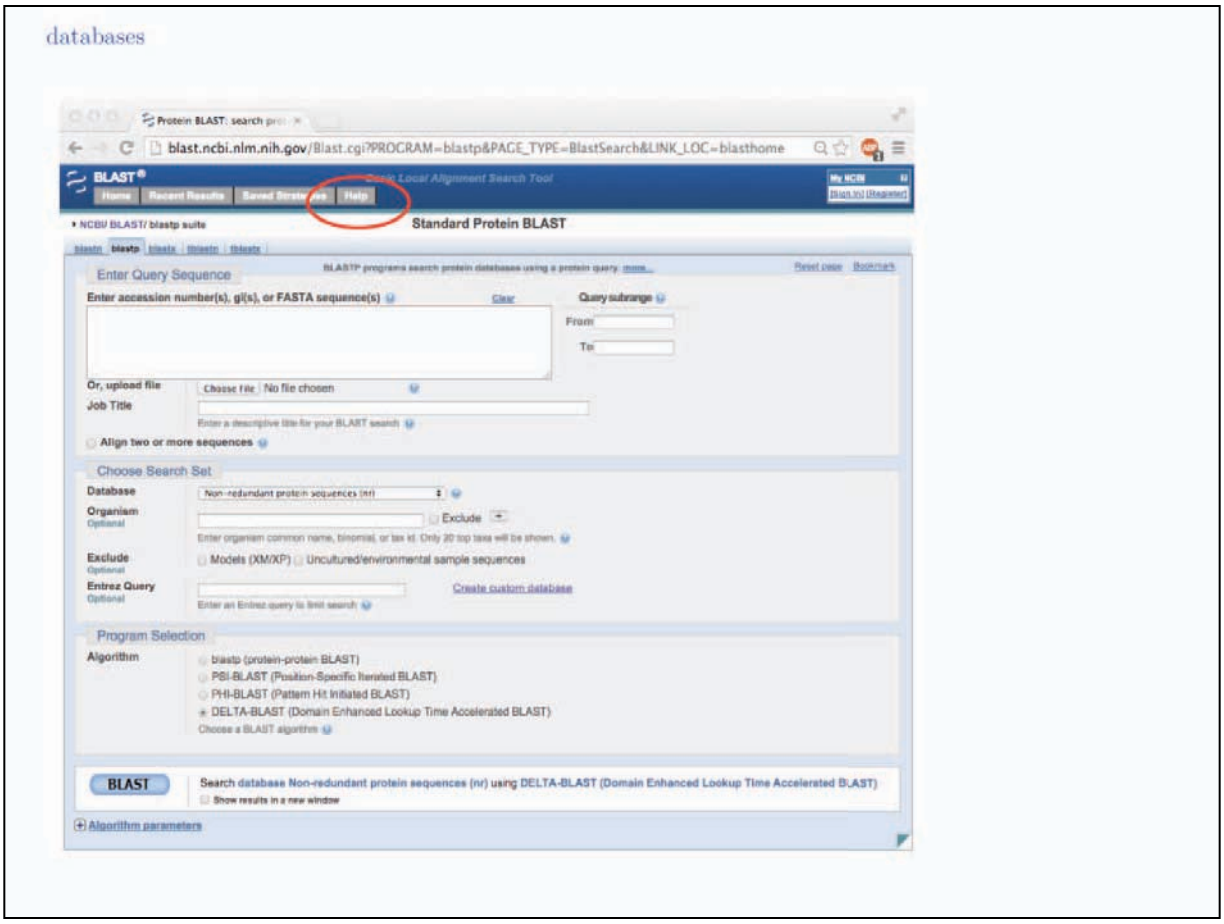
The enormous speed-up of BLAST is due to its use of an **indexed table** of database "words". The index is a list of positions at which each word occurs in the database. Using an index, it is very easy to examine every occurrence of a word in the database and try to extend the word match on both sides with additional similar sequence. The extension does not introduce gaps, because this is faster, but also because the statistics of ungapped alignments are tractable! The final step is the assembly of significant hits into longer alignments.

Note that BLAST is **heuristic**, not **optimal** and that it is a **local**, not **global** alignment algorithm.

See also: Altschul *et al.* (1990): http://www.ncbi.nlm.nih.gov/pubmed/2231712



The **BLAST** home page offers a number of different BLAST *flavours*.

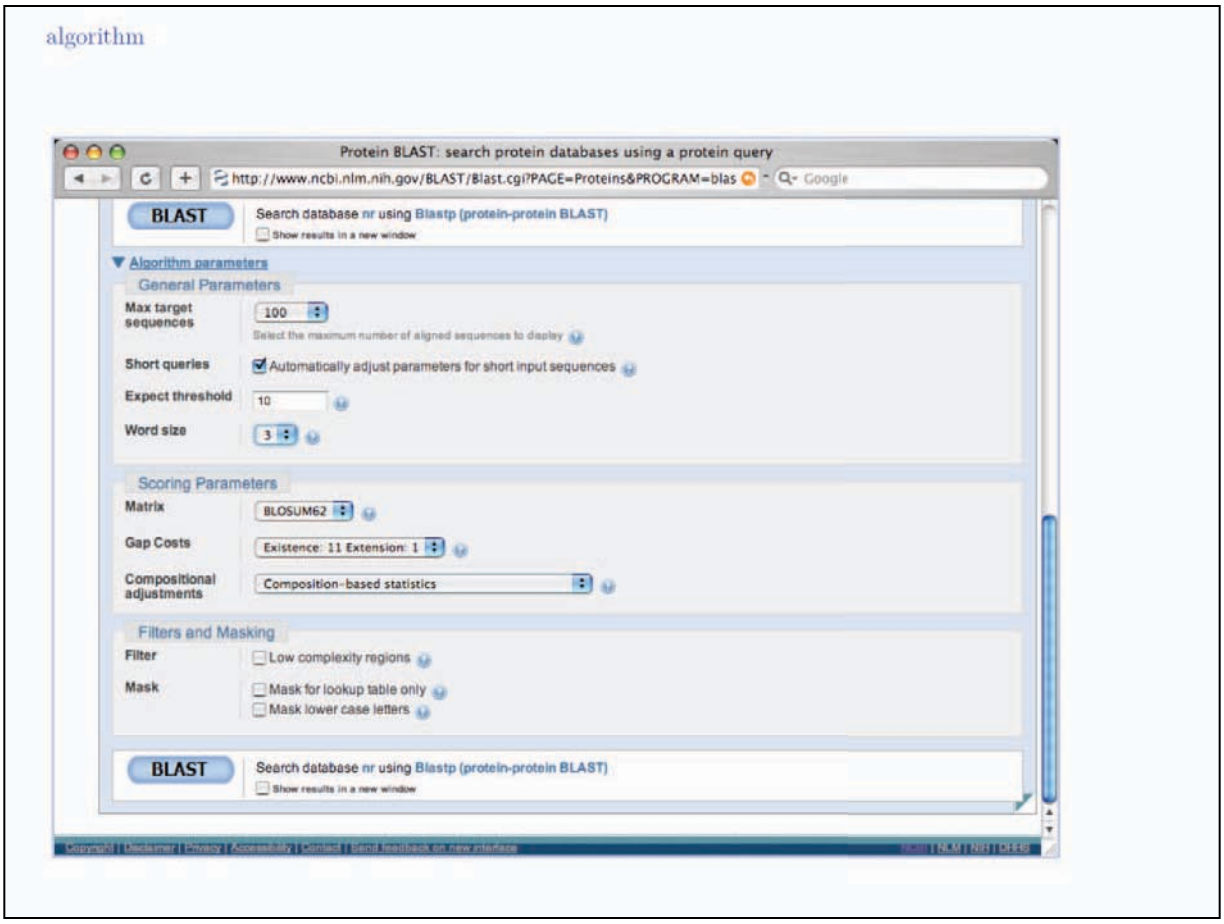


BLAST parameters: databases

Extensive help is available (and should be read!) for each of the options. Take the time to read the **Web BLAST options document**[1] and be sure to understand how to format input, what databases are available and how the choice of database influences the results. If you are not confident with the document, ask on the course list.

For example, the Help page contains guides to the **search interface** and the **report output**!

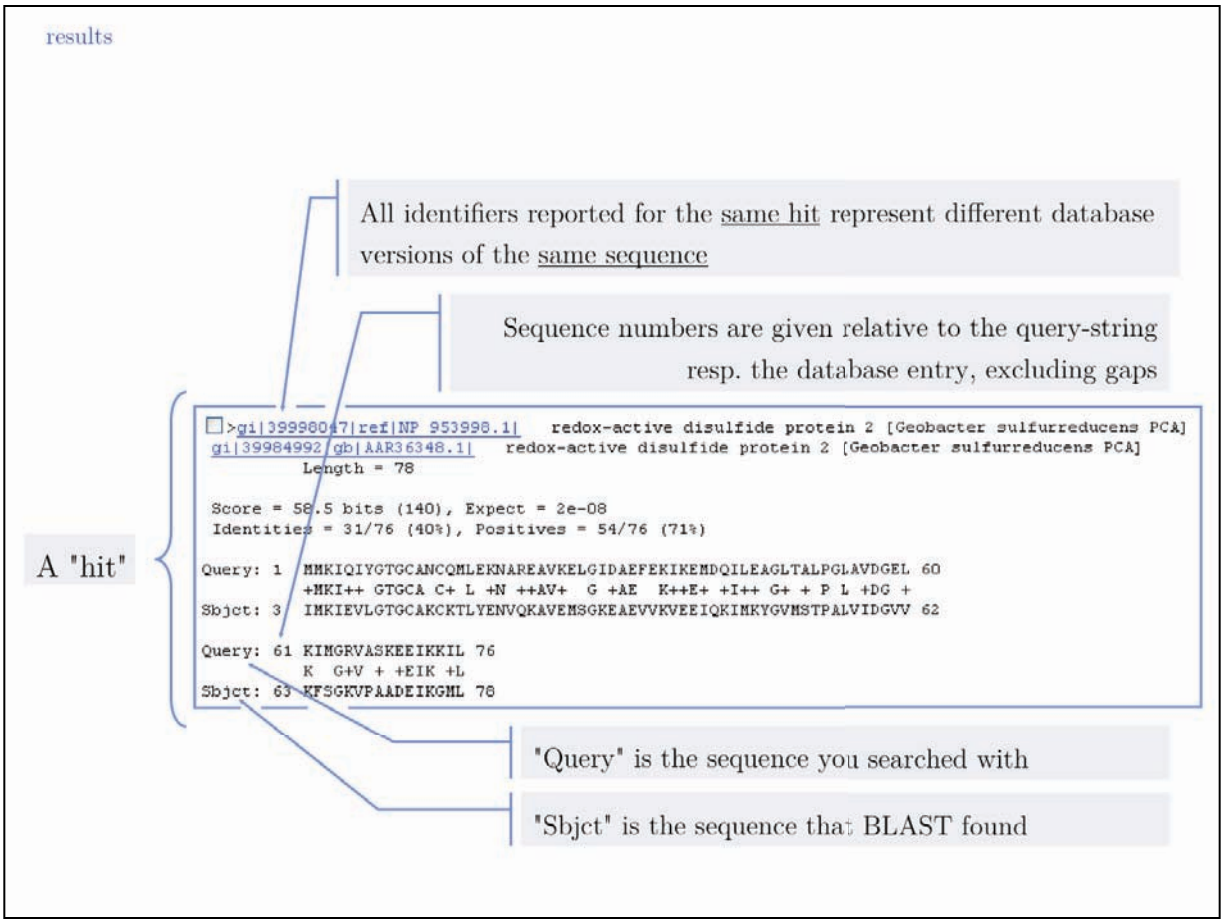
[1] http://www.ncbi.nlm.nih.gov/BLAST/blastcgihelp.shtml



BLAST parameters: algorithm

Be sure to understand the choices and their consequences for **Composition-based statistics**[1] and for **Filtering and Masking** segments of low complexity in your query. Filtering is an important option to consider especially for PSI-BLAST searches!

[1] http://www.ncbi.nlm.nih.gov/BLAST/blastcgihelp.shtml#compositional_adjustments
[2] http://www.ncbi.nlm.nih.gov/BLAST/blastcgihelp.shtml#filter



Each Blast **hit** represents an alignment that can contain one or more HSPs (High Scoring Segment Pairs). Note: If a hit is followed by a second hit and no new GI number, it identifies a second region of similarity in the **same sequence**.

E-value

The **quality** of the alignment is represented by the Score (s).

The **significance** of the alignment is computed as an E- value.

E-value (E) *Expectation value*:

The number of alignments with scores equivalent to, or better than a score ***s*** that are expected to occur in a database **of the same size** that does not contain a homologous sequence.

The smaller the E-value (the larger its negative exponent), the more significant the score.

The E-value is a statistically well founded metric that allows us to conclude the likelihood of a spurious alignment. Computing E-values is possible for HSPs since the statistics of gap-less alignments are analytically tractable, whereas gapped alignments have no theoretical description of the distribution of expected scores.

Note that E-values do not represent an assertion about the retrieved sequence, but an assertion about the score and its relation to the expected distribution of scores. Or, to rephrase this, a large E-value does not mean that your hit is not a homologue, but it means that an irrelevant sequence has a a high chance of having just as high a score due to chance similarities. To repeat: a large E-value does not mean your hit is not a homologue. However a small E-value does indeed mean that a chance alignment is unlikely.

It is important to realize that the E-value depends on the database size. Obviously, you would expect randomly high-scoring hits more often in a large database than in a small one. Thus an alignment with the **same score** will have a **smaller E-value** when searched against a particular genome than if you search it against the entire "nr" dataset of GenBank.

More detail in the NCBI tutorial "The Statistics of Sequence Similarity Scores" (http://www.ncbi.nlm.nih.gov/BLAST/tutorial/Altschul-1.html)

interpretation

E-values are very convenient because they have an obvious interpretation of significance, but they do not absolve you from using biological common sense!

Example: searching pea defensin (1JKZ) aginst nr: are these sequences homologous?

```
gi|15226880|ref|NP_178322.1|   plant defensin protein, putative (PDF2.6)
gi|11387216|sp|Q9ZUL8|THG4_ARATH   Gamma-thionin homolog At2g02140 precursor
gi|25330850|pir||D84433   proteinase inhibitor II [imported] - Arabidopsis thaliana
gi|4038038|gb|AAC97220.1|   protease inhibitor II [Arabidopsis thaliana]
gi|21592674|gb|AAM64623.1|   protease inhibitor II [Arabidopsis thaliana]
          Length = 73

 Score = 30.8 bits (68), Expect = 6.7
 Identities = 14/46 (30%), Positives = 27/46 (58%), Gaps = 1/46 (2%)

Query: 1  KTCEHLADTYRGVCFTNASCDDHCKNKAHLISGTCHNWKCFCTQNC 46
          +TCE  ++ ++GVC  + SC   C ++     G C + +C+C++ C
Sbjct: 29 RTCESPSNKFQGVCLNSQSCAKACPSEG-FSGGRCSSLRCYCSKAC 73
```

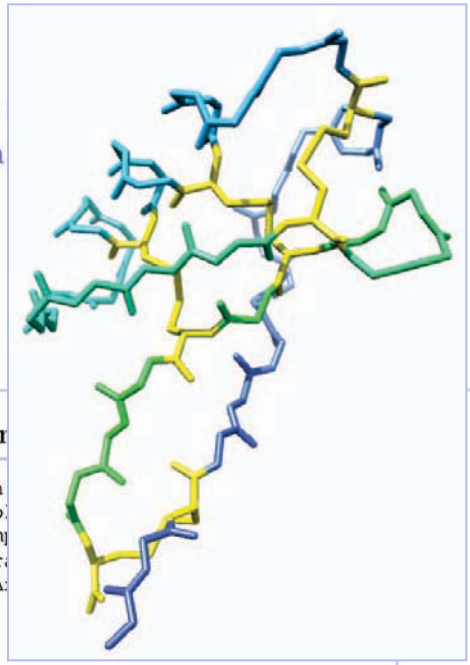
In the example above, the BLAST search of a Pea defensin - PDB structure 1JKZ - achieved an E-value of only 6.7.

interpretation

E-values are very convenient because they have an
significance, but they do not absolve you from
sense!

Always:

- **Check the annotation;**
- **Check the alignment.**



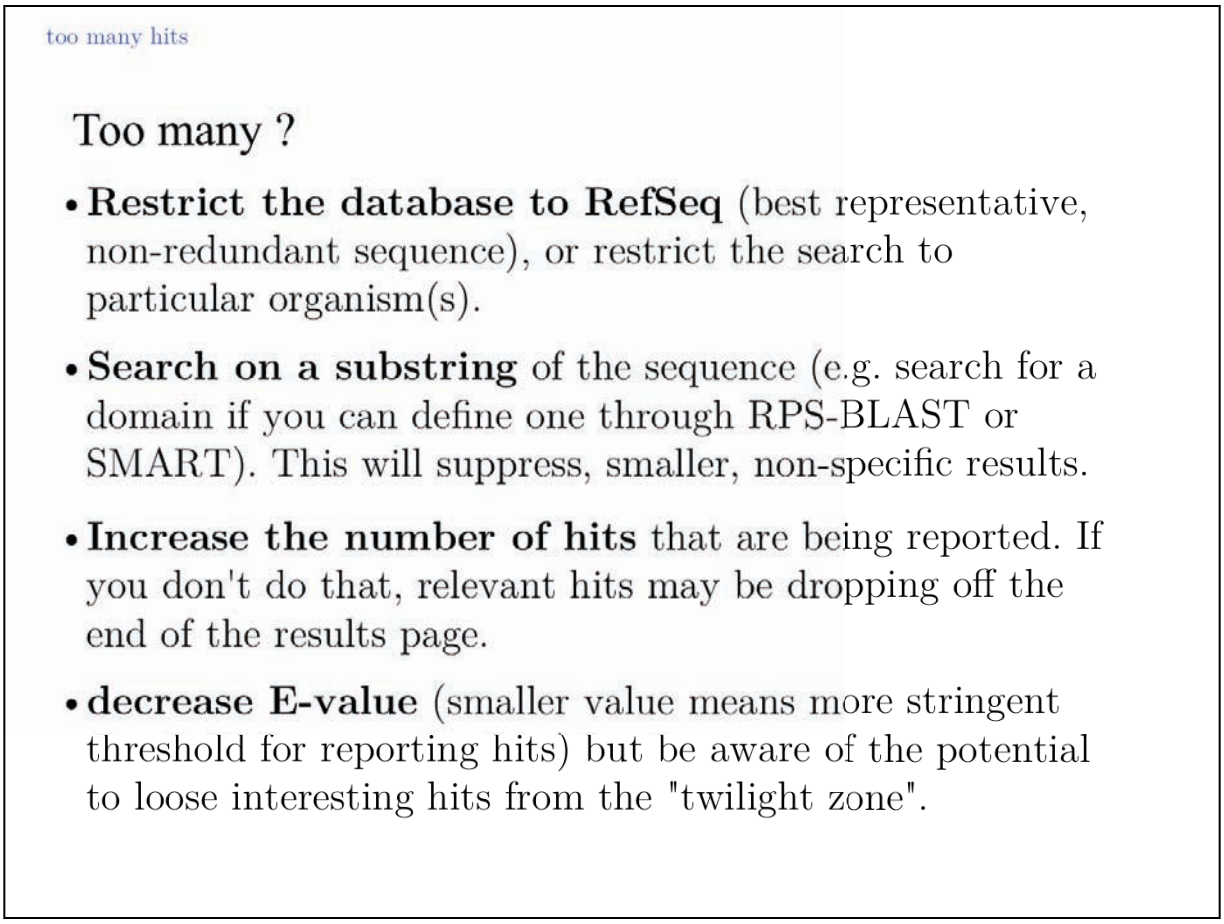
Example: searching pea defensin (1JKZ) aginst r

```
gi|15226880|ref|NP_178322.1|   plant defensin protein
gi|11387216|sp|Q9ZUL8|THG4_ARATH   Gamma-thionin homo
gi|25330850|pir||D84433   proteinase inhibitor II [im
gi|4038038|gb|AAC97220.1|   protease inhibitor II [Ar
gi|21592674|gb|AAM64623.1|   protease inhibitor II [A
          Length = 73

 Score = 30.8 bits (68), Expect = 6.7
 Identities = 14/46 (30%), Positives = 27/46 (58%), Gaps = 1/46 (2%)

Query: 1  KTCEHLADTYRGVCFTNASCDDHCKNKAHLISGTCHNWKCFCTQNC 46
          +TCE  ++ ++GVC  + SC   C ++     G C + +C+C++ C
Sbjct: 29 RTCESPSNKFQGVCLNSQSCAKACPSEG-FSGGRCSSLRCYCSKAC 73
```

However the hit that was retrieved:

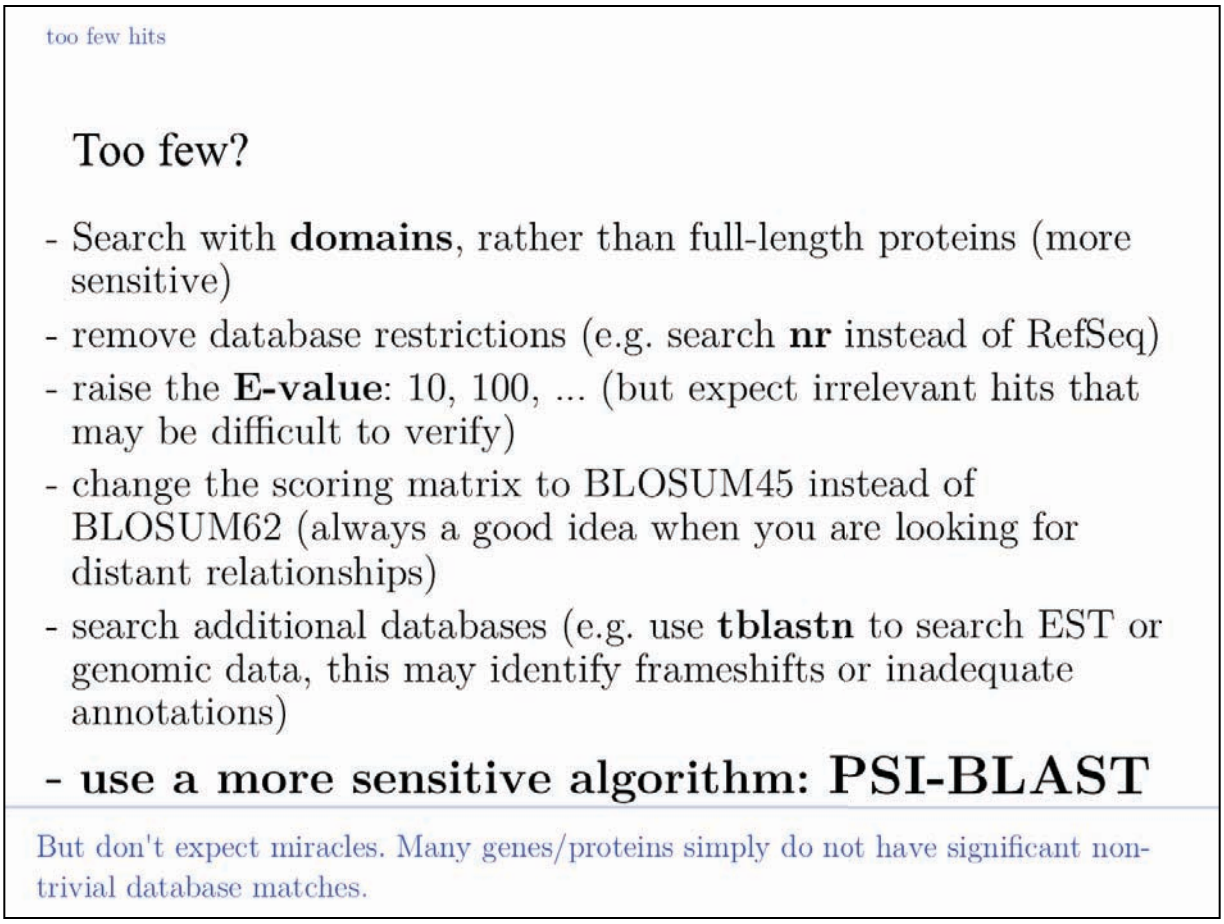
- is annotated as an *arabidopsis* **defensin**;
- has **30% identity** over the entire domain, albeit the domain is small;
- requires only **one single gap** for alignment; and
- **has each and every single cysteine conserved, when compared to the query**!

Each of these additional observations alone could have led you to conclude homology. It should be obvious for example that the aligned cysteines are extremely unlikely to be due to a random similarity of unrelated sequences! The large E-value is primarily due to the fact that the protein sequences are quite short.



How can there be too many hits, when *lots-of-hits* is what you are looking for? Either you find redundant sequences or trivially similar sequences that are obscuring the rare, interesting similarities you are looking for (GFP or other fusion proteins and ankyrin domains come to mind, for example), or you are searching in a database section that contains redundant sequences.

Note that restricting by organism does not restrict the search, but only the list of results that are being reported. The search takes just as long. Only the specialized genome search pages and some non-NCBI databases of model-organism genome projects offer BLAST searches on reduced datasets. These searches are faster.



How many genes have no homologues? That depends. Unknown genes (or "ORFans") may comprise a significant (albeit diminishing) fraction of genomes.

In general, between 10 and 30% of sequences may fall into this category and it is likely that even the most closely related species have sequences that are unique.

See Siew&Fischer (2003)[1]and a discussion of the role of viral horizontal gene transfer in ORFans by Yin and Fischer (2006)[2]

[1] http://www.ncbi.nlm.nih.gov/pubmed/12517334

[2] http://www.ncbi.nlm.nih.gov/pubmed/16914045

Searching with profiles instead of individual sequences is more sensitive!

It reduces the effect of *random drift* in individual sequences and focuses the search on the *commonalities* and the specific *tolerance to variation* that is particular to a *family* of sequences (in which all sequences have diverged from a common ancestor).

PSI-BLAST proceeds in five steps:

1. Select a query and BLAST it against a protein database
2. PSI-BLAST constructs a multiple sequence alignment from the BLAST hits, then creates a "profile" (or position-specific scoring matrix (PSSM))
3. The PSSM is used as a query against the database
4. PSI-BLAST estimates statistical significance (E values) and proposes significant hits for inclusion into the next iteration
5. Repeat steps [3] and [4] iteratively, typically 5 times. At each new search, a new profile is constructed from the previous hits and used as the new query.

profile corruption

PSI-BLAST is useful to detect weak but biologically meaningful relationships between proteins.

Not all sequences can be used for a search. For example, a query with a coiled-coil motif may detect thousands of proteins that have this motif but that are not homologous.

False positives in the search can arise from the inclusion of irrelevant sequences into the profile.

Once even a single irrelevant protein sequence is included in a PSI-BLAST profile, it will not go away.

If the number of related irrelevant proteins is large, they will *take over* the profile: **profile corruption**!

profile corruption

Apply filtering of regions with low-complexity and biased composition

Adjust E-value from 0.001 (default) to a more stringent value (e.g. 0.0001).

Visually inspect the output from each iteration:

> Suspicious hits have irrelevant locations or functions, similarities only to parts of domains, fail to conserve important motifs or disulfide bridges or have poor-quality alignments.

Remove suspicious hits from inclusion by unchecking the box.

> Be conservative regarding the sequences you include, **true positives will gradually improve their E-values** with subsequent iterations, even if they are not included in the profile. You can simply include them in later iterations (or not at all, they will still be reported, even if they don't contribute to the profile). False positives will not improve significantly.

In the end, how many false positives can we expect? Unfortunately, more than we'd think. Jones & Swindells (2002)[1] have run an analysis against decoy sequences that picked up false positives in 5% of all cases, after the fifth iteration, although the E-value threshold was set to 0.001.

Even though their methodology was a bit *ad hoc* and finding false positives about 50 times more frequently than expected is not catastrophic, we must realize that protein sequences are not random strings and that significance is often hard to evaluate, because it is hard to get the *null* hypothesis right. Use caution, use common sense and in questionable cases try to use independent confirmation of homology, such as conserved binding sites or functional motifs, if possible.

[1] http://www.ncbi.nlm.nih.gov/pubmed/11893514

filtering

Filtering is used to mask parts of sequences with **low complexity** for database searches, by replacing sequence fragments with "X" (unknown residue) symbols.

Filtering is applied to the QUERY not the database

Filtering can eliminate statistically significant but biologically uninteresting reports from BLAST output (e.g. Basic, acidic, or proline rich regions, but these **may** actually be interesting, depending on your biological question.)

Personally, I turn filtering OFF by default for standard BLAST searches, only turn it ON if low-complexity regions appear to cause problems with specificity. Informed judgement is required.

However: always use filtering for PSI-BLAST (profile corruption)!

To ...

... identify regions of the polypeptide chain that fold independently, that are stable on their own

(folding units; initiation sites for folding)

... identify gene fusion or gene insertion events from analysis of the 3D structure

(understand evolutionary history)

... understand protein mechanism as an additive/cooperative result of domain function

(CDART, SMART - domain architecture)

... allow for meaningful structural classification of proteins

(SCOP, CATH classifications)

section

Domains

The discovery and catalogization of the universe of domains in protein sequences is the greatest achievement of profile– and model based sequence alignment algorithms.

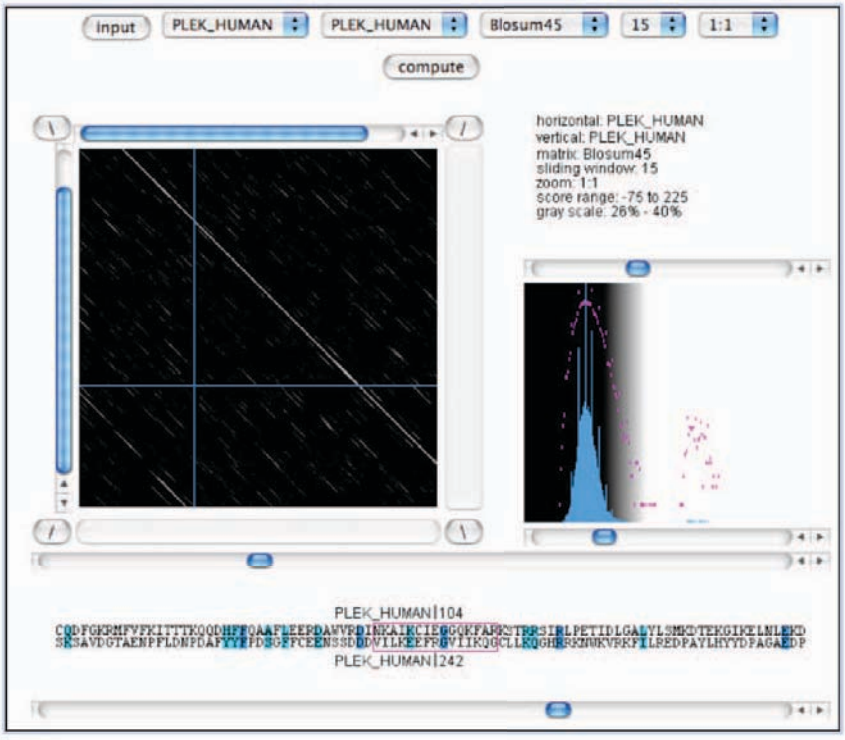
Domains are modules for ...

- inheritance
- distinct function
- separate folding

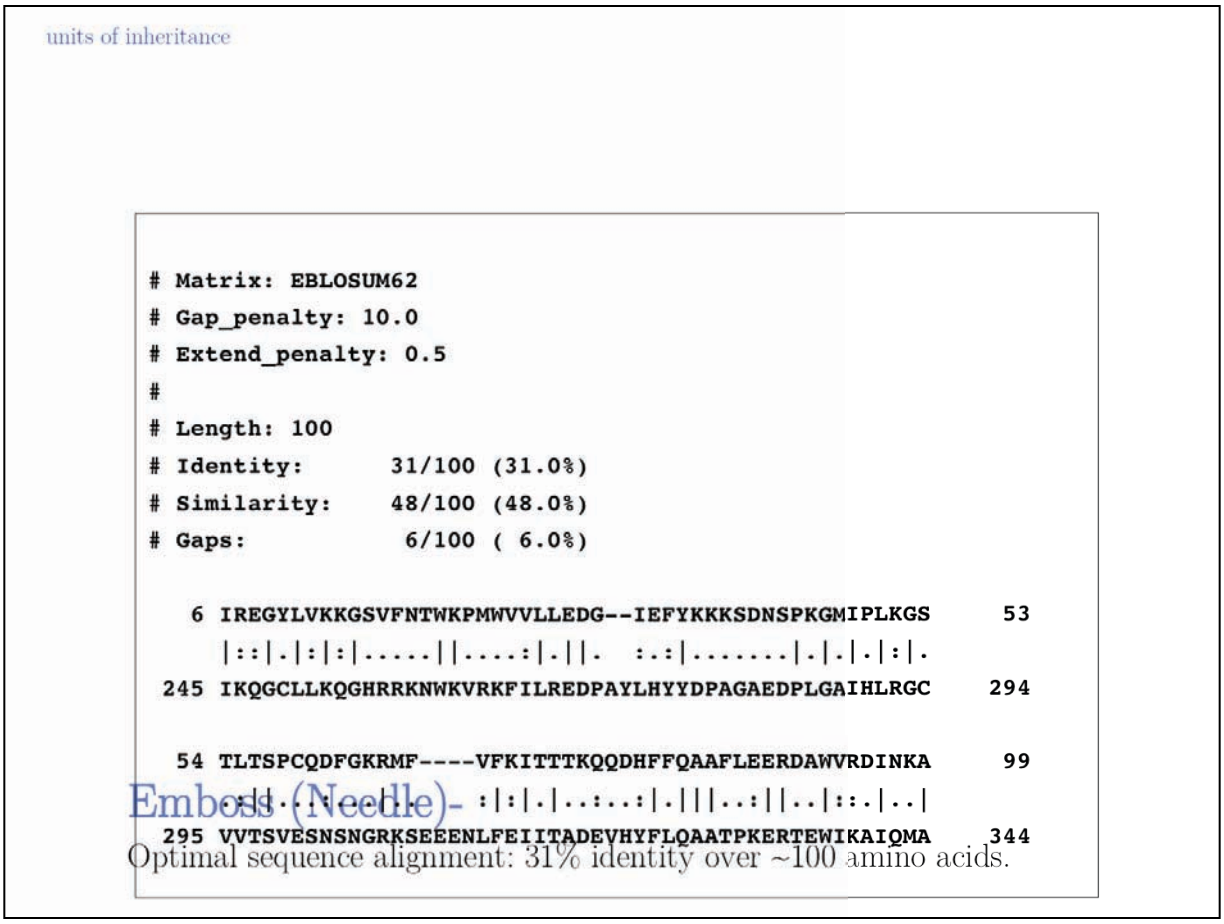
The *domain* is the natural unit of analysis of protein structure.

Dotlet -

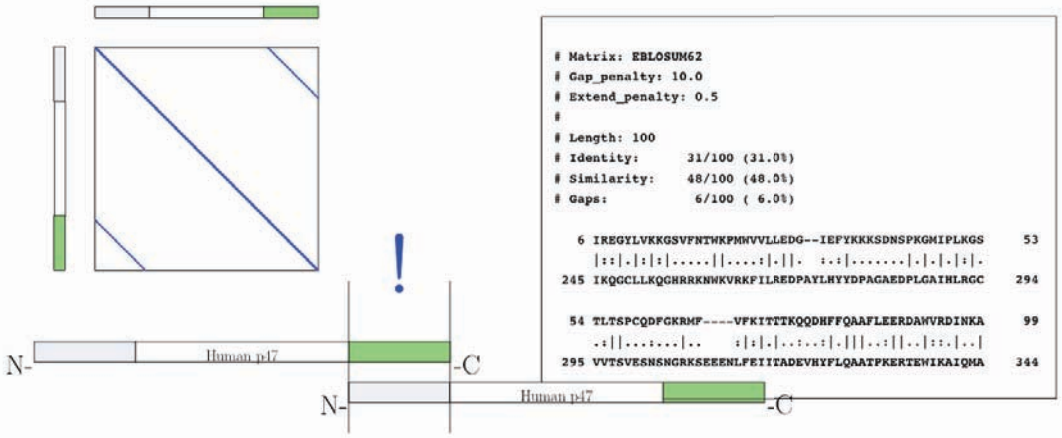
A dotplot of Pleckstrin (p47) reveals similarity between N-and C terminus !



Here is an example of how a domain was discovered from sequence alignments.



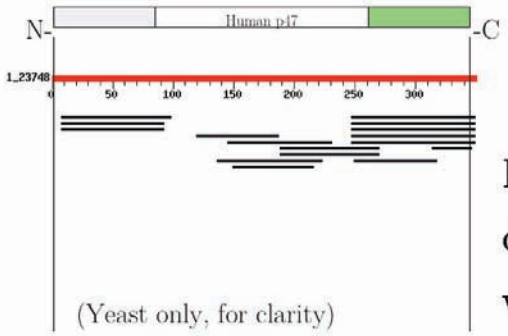
The alignment shows high similarity between N- and C-terminus.




```
# Matrix: EBLOSUM62
# Gap_penalty: 10.0
# Extend_penalty: 0.5
#
# Length: 100
# Identity:      31/100 (31.0%)
# Similarity:    48/100 (48.0%)
# Gaps:           6/100 ( 6.0%)

   6 IREGYLVKKGSVFNTWKPMHVVLLEDG--IEFYKKKSDNSPKGMIPLKGS     53
     |::|.|:|:|.....||....:|.||.  :.:|.......|.|.|:|.
 245 IKQGCLLKQGHRRKNWKVRKFILREDPAYLHYYDPAGAEDPLGAIHLRGC    294

  54 TLTSPCQDFGKRMF----VFKITTTKQQDHFFQAAFLEERDAWVRDINKA     99
     .:||...:...|..    :|:|.|..:..:|.|||..:||..|::.|..|
 295 VVTSVESNSNGRKSEEENLFEIITADEVHYFLQAATPKERTEWIKAIQMA    344
```

Overlapping alignments may define domain boundaries ! We can search a database with this knowledge ...

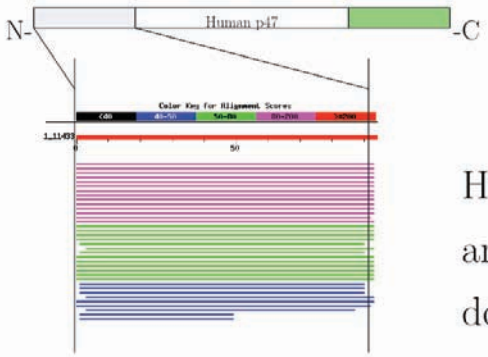
BLAST search with *full-length* Pleckstrin sequence



(Yeast only, for clarity)

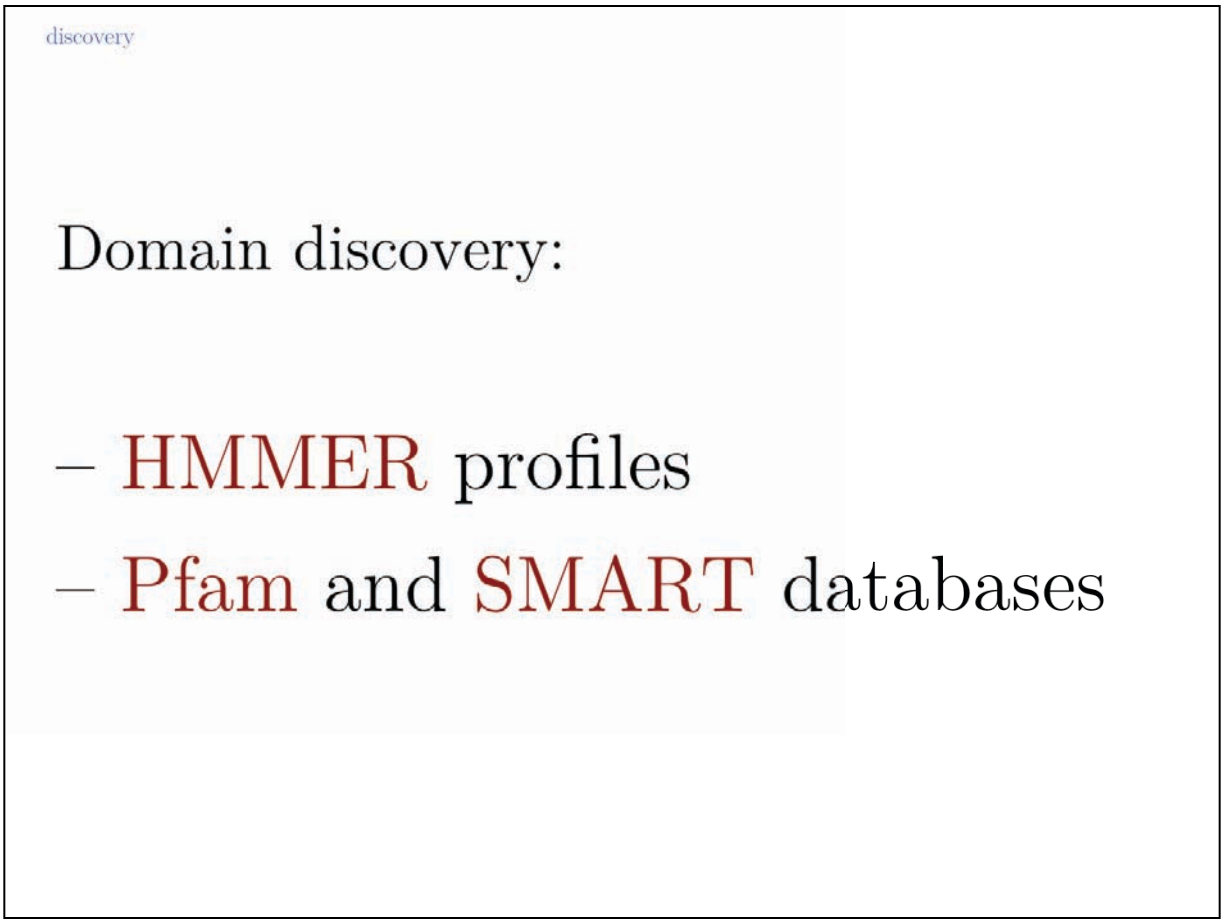
Hits extend over the entire domain. PSI -BLAST would be difficult ...

BLAST search with Pleckstrin Domain *only* ...



Hits are smoothly bounded and extend over the entire domain. PSI-BLAST is straightforward.

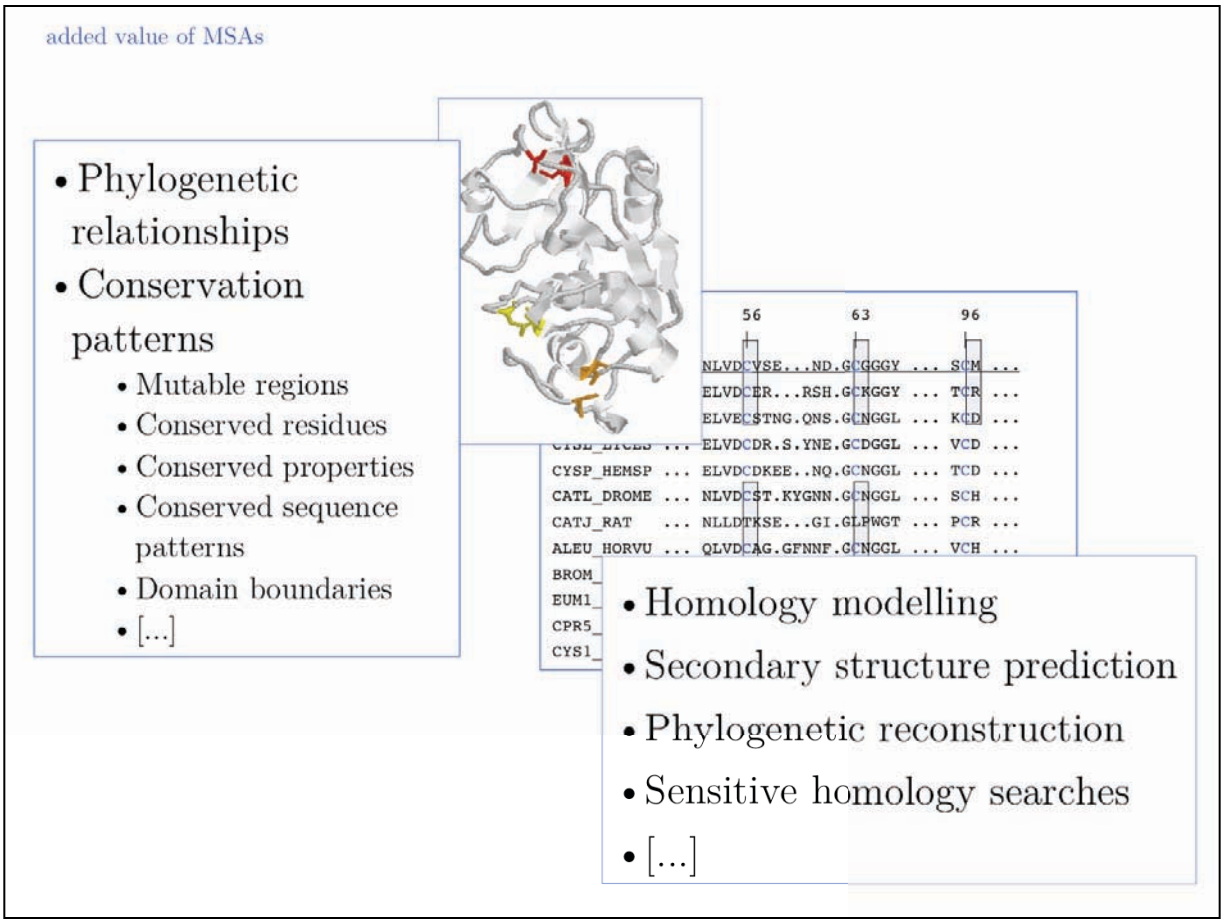
486 hits ... etc.



Domain discovery on a large scale has been made possible through Hidden Markov Model alignments, implemented in Sean Eddy's HMMER program. This has been used to compile large databases like Pfam that curate domain profiles. These profiles can be scanned against an unknown sequence, thus allowing the annotation of the sequence with the domains it contains. In many cases this allows to assign at least a coarse description of function and mechanism.

section

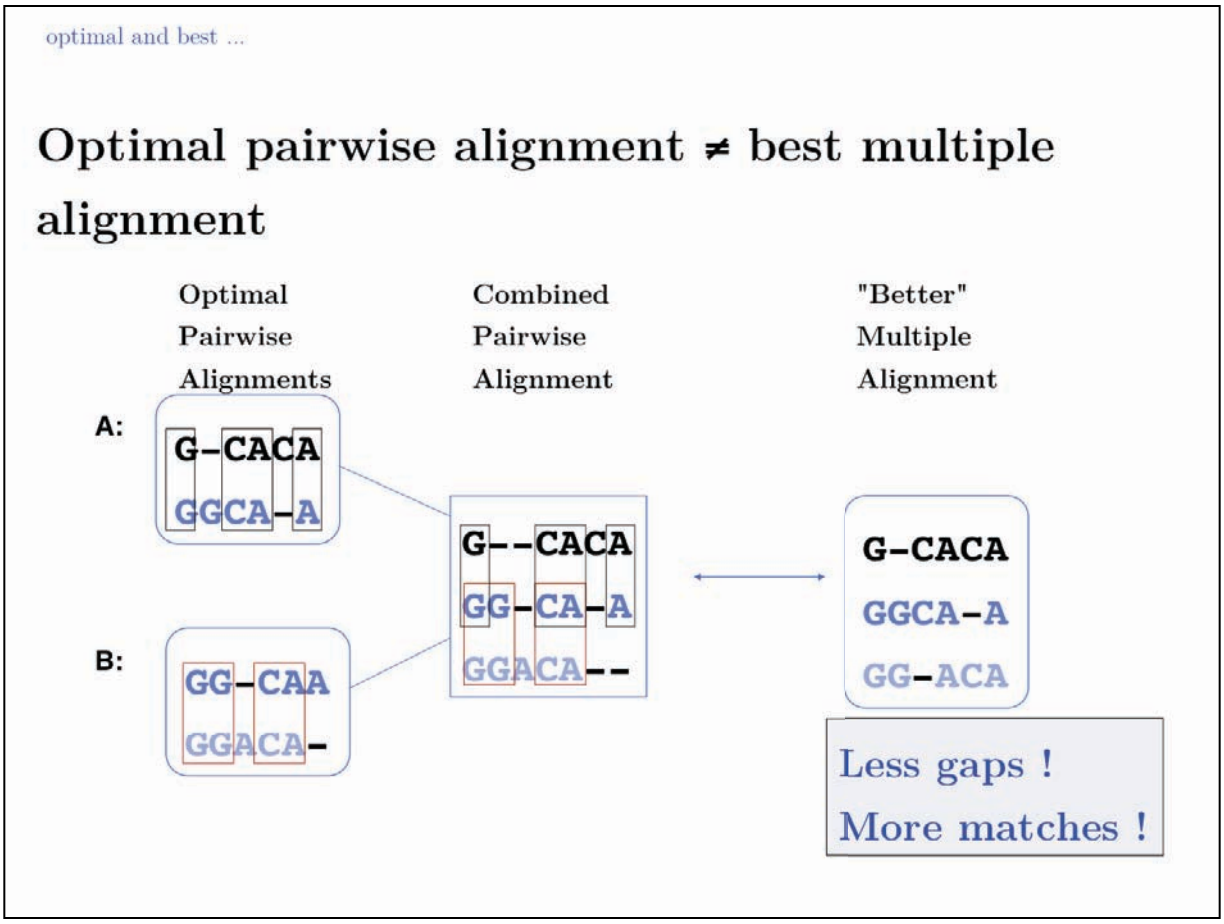
Multiple Sequence Alignment



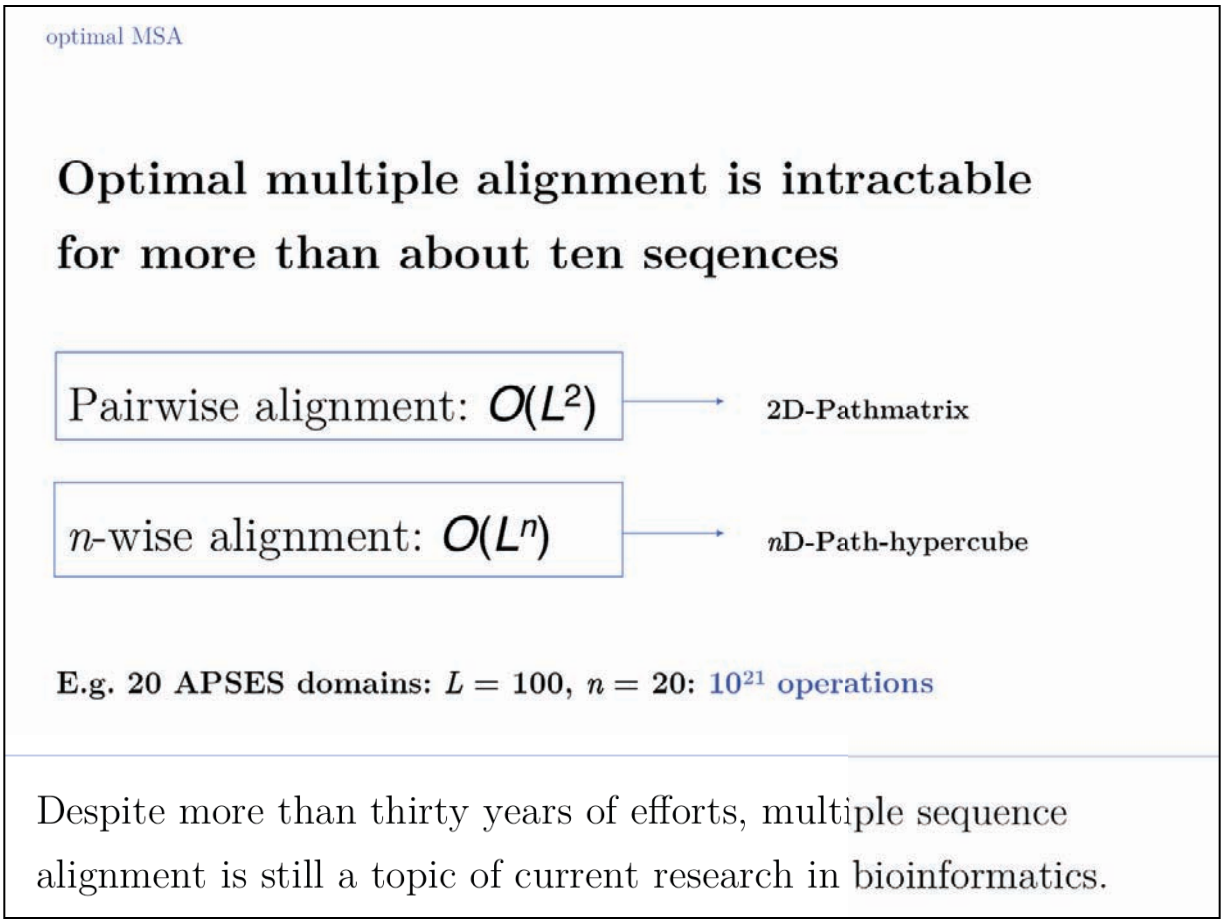
MSAs show conservation patterns.

Multiple sequence alignments don't just match residues. They also give information on how strongly a residue is conserved, what it can be replaced with, which species share particular sequence patterns, and where in the sequence indels can be tolerated. An analysis of conservation even allows to distinguish between structurally and functionally conserved residues! This makes multiple sequence alignments the method of choice for many applications.

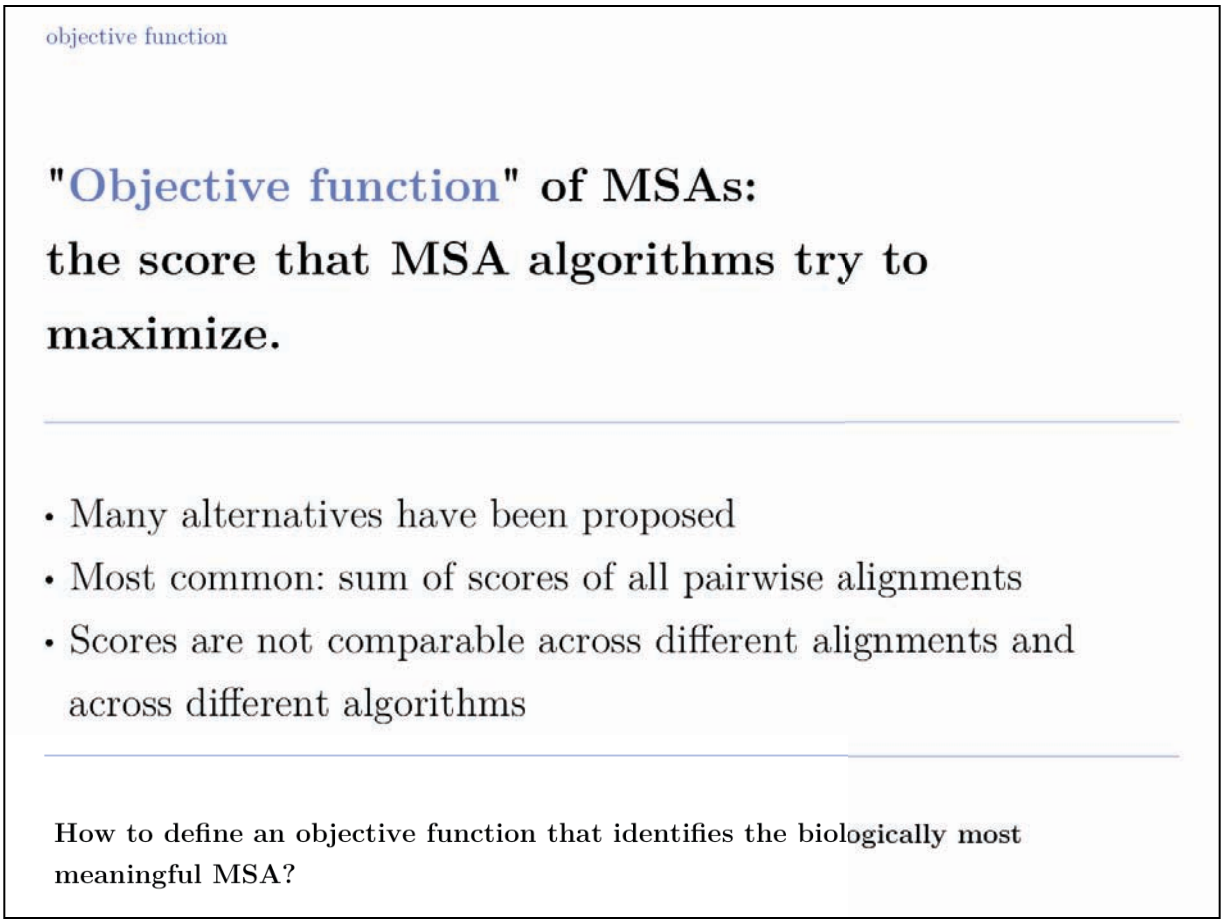
- Multiple sequence alignments are more accurate than pairwise alignments, thus they are the method of choice for starting **homology modeling** projects;
- Combined information from numerous sequences is invaluable for **secondary structure prediction** and **sensitive sequence database searches**;
- They contain the information needed for inferences about **evolutionary relationships**, i.e. the order in which particular sequence changes occurred.



Multiple alignments cannot necessarily be **constructed** by merging pairwise alignments. Moreover, it may be actually be impossible to merge three mutually pairwise alignments into a non-contradicting multiple alignment. However the inverse is always possible: a multiple alignment can be **decomposed** into pairwise alignments.



Besides being intractable, it is questionable how meaningful the objective function of *optimal sequence alignments* is for *multiple alignments*. This pair score maximizes the score derived from a mutation data matrix, for pairs of aligned residues. But – for example – the pair score does not optimize the pattern of indel placements, or whether a particular motif is well-conserved.



If we want an algorithm to optimize anything at all, we first must define how we can measure the quality of the result. This metric defines the **target function** or **objective function**.

(Note that "objective" here is not used in the sense of "unbiased" but in the sense of being a "target", or "goal".)

biologically meaningful?

Biologically motivated objectives for multiple alignments:

- Minimize number of indels, not length
- Minimize number of sites at which indels are tolerated
- Maximize sequence similarity
- Retain conserved motifs and patterns
- Recapitulate phylogeny
- Concentrate on alignable regions not on gapped regions
- [...]

Alignment objectives are based on the biological models we apply to multiple alignments - they attempt to capture constraints on sequence similarity that go beyond optimizing a pairwise alignment score (which is based on a **context-independent** mutation data matrix and on an **empirical model** for the probability of indels).

Reasonable alignment metrics are based on models of how evolution has shaped a family of related sequences.

Each of the reasonable biological objectives suggests a different alignment strategy! The most modern algorithms currently available attempt to satisfy these heuristics simultaneously. Note that these are '''heuristics''', they are not the result of some rigorously applied theory, but reflect the complex relationship between protein sequence, structure, evolution and selection.

strategies

Computational strategies for multiple alignment algorithms derive from the nature of the objective function. The objective function can reflect biological heuristics.

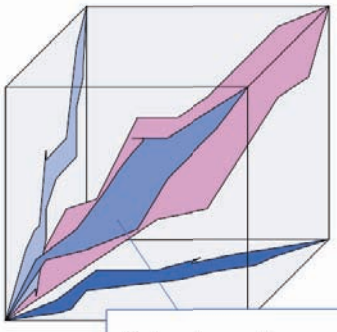
Objective	Algorithm	Type of alignment algorithm
Maximize similarity, Minimize gaps	Bounded optimal solution search	**Exact**
Align according to phylogeny	Align most similar first, then merge together	**Progressive**
Retain conserved regions	Conserved regions guide alignment	**Consistency based**
Maximize similarity to model	Create a model, align each sequence to that	**Probabilistic**
"Learn" about important regions and extend the alignment from secure seeds	Improve alignment from draft alignments	**Iterated**

Alignment algorithms that operate according to one or more of these principles are easily accessible online *via* the EBI:

http://www.ebi.ac.uk/Tools/msa/

Exact MSA (... maximize alignment scores)

MSA:
Multidimensional dynamic programming; search space in n-dimensional path matrix can be bounded by optimal pairwise alignments. Optimizes sum-of-pairs score which may not be the most biologically meaningful score anyway. More accurate than e.g. progressive methods but compute intensive. Practical limit ~ 10 seq. of L = 200.



Pairwise alignments bound the search volume in n-space!

Novel developments:
DCA (divide and conquer: split into subsequences, then recombine),
OMA (iterated improvement of splits).

Exact methods certainly have their place where it comes to analyzing and improving algorithms; they are especially of interest to computer science because high-dimensional optimal alignment is a difficult problem. However they cannot compete in terms of result-quality with modern heuristic methods. This is not only because they really don't scale to current genome-scale questions or even modest sized protein families, but also because optimizing the score derived from a pair-score mutation data matrix plus an empirical affine gap model is not a really a very good objective for MSAs that inform about biology in the first place.

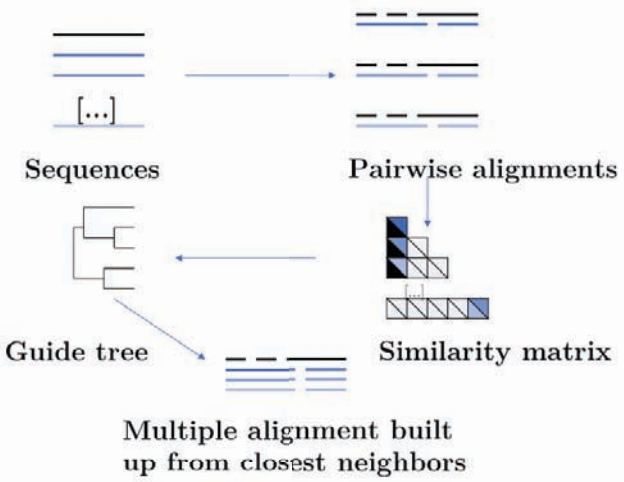
progressive MSA

(... align the most similar sequences first)

Clustal W, X:
Considered by many to be *The Standard* method.

1. Build pairwise similarity matrix
2. **Build guide tree.**
3. join neighbors into profiles according to guide tree.
4. Align according to tree.

Key limitation: persistence of early errors. Best performance is for globally alignable, gap-poor sequence sets. Performance progressively worse for multidomain proteins and distant similarities.



Bottom line: Don't use CLUSTAL W. There are much better and equally convenient algorithms available.

Progressive alignment is one of the fundamental algorithmic approaches to MSA. Pure progressive alignment algorithms are only of historical interest today, since they suffer from unacceptable degradation of accuracy for sequences below ~30% ID due to the fact that early alignment errors cannot be corrected.

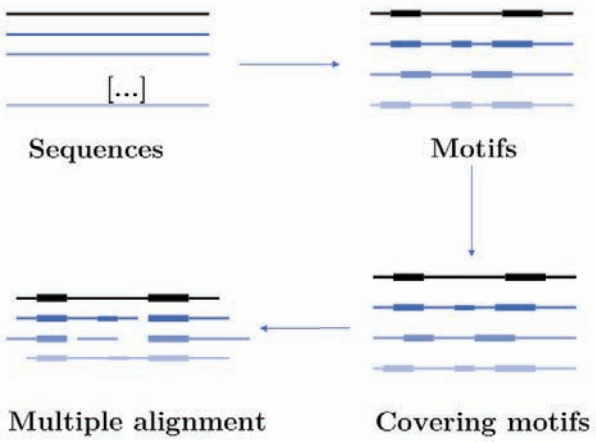
consistency based MSA

Consistency based multiple alignment: alignment based on motifs and patterns

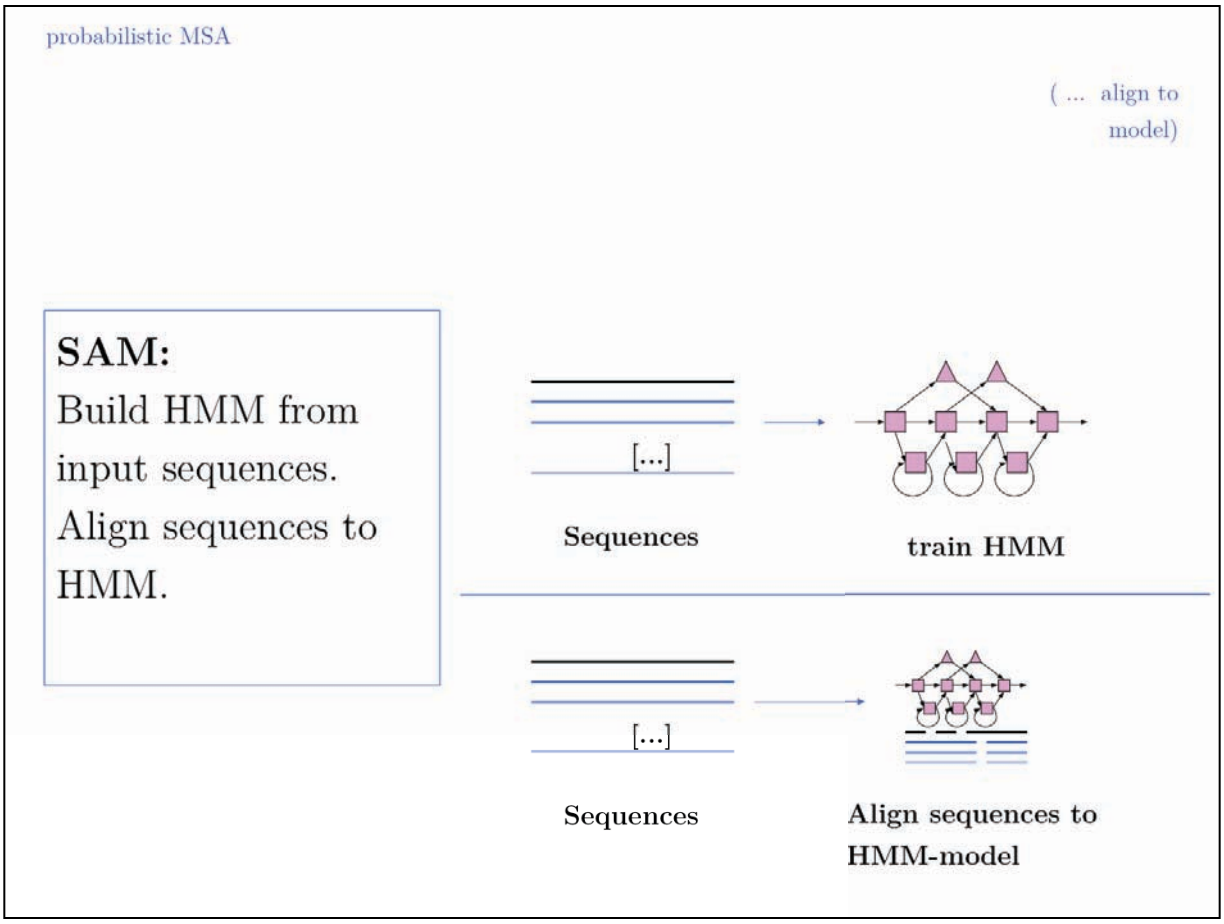
(... emphasize conserved –i.e. important– regions)

MUSCA:
(I) TEIRESIAS motif discovery.
(II) Use set covering algorithm to select motifs that are common to sequence set.

MEME:
Postulate motif, compare residue composition of motif with background, choose motifs to maximize composition difference, output.



Consistency based multiple alignment is one of the fundamental algorithmic approaches to MSA. Many modern algorithms have a consistency based step included, however none of them relies solely on consistency, since problems from spurious local similarity can corrupt the alignment.



Probabilistic multiple alignment is one of three fundamental algorithmic approaches to MSA.

A statistical model of the sequences is built, then the alignment can be generated by aligning the sequences to the model. Of course, aligning sequences to a profile is a special case of this procedure: PSI BLAST can thus be used as an alignment algorithm. The most widely used algortihm is Sean Eddy's **HMMER**[1], a profile hidden Markov model tool, which is also used in the generation of the **Pfam** domain database[2].

[1] http://hmmer.janelia.org/

[2] http://pfam.sanger.ac.uk/Pfam

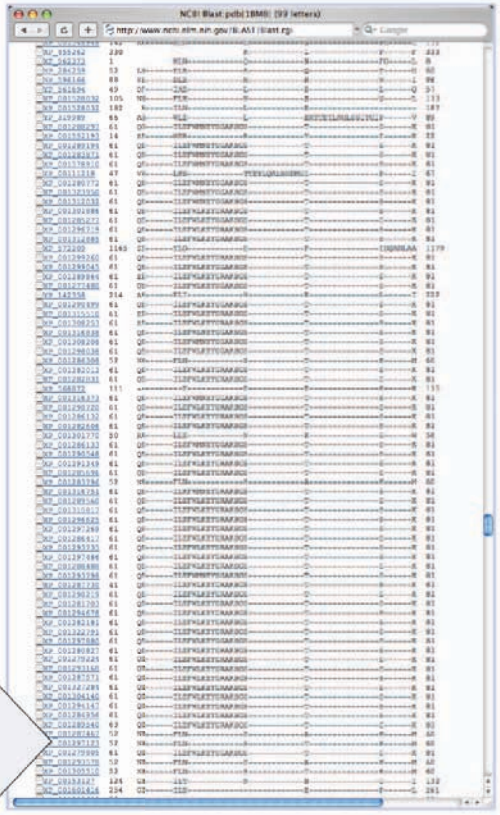
profile based

The MSA derived from aligning sequences to a profile in a **PSI-BLAST** search is also a model based alignment.

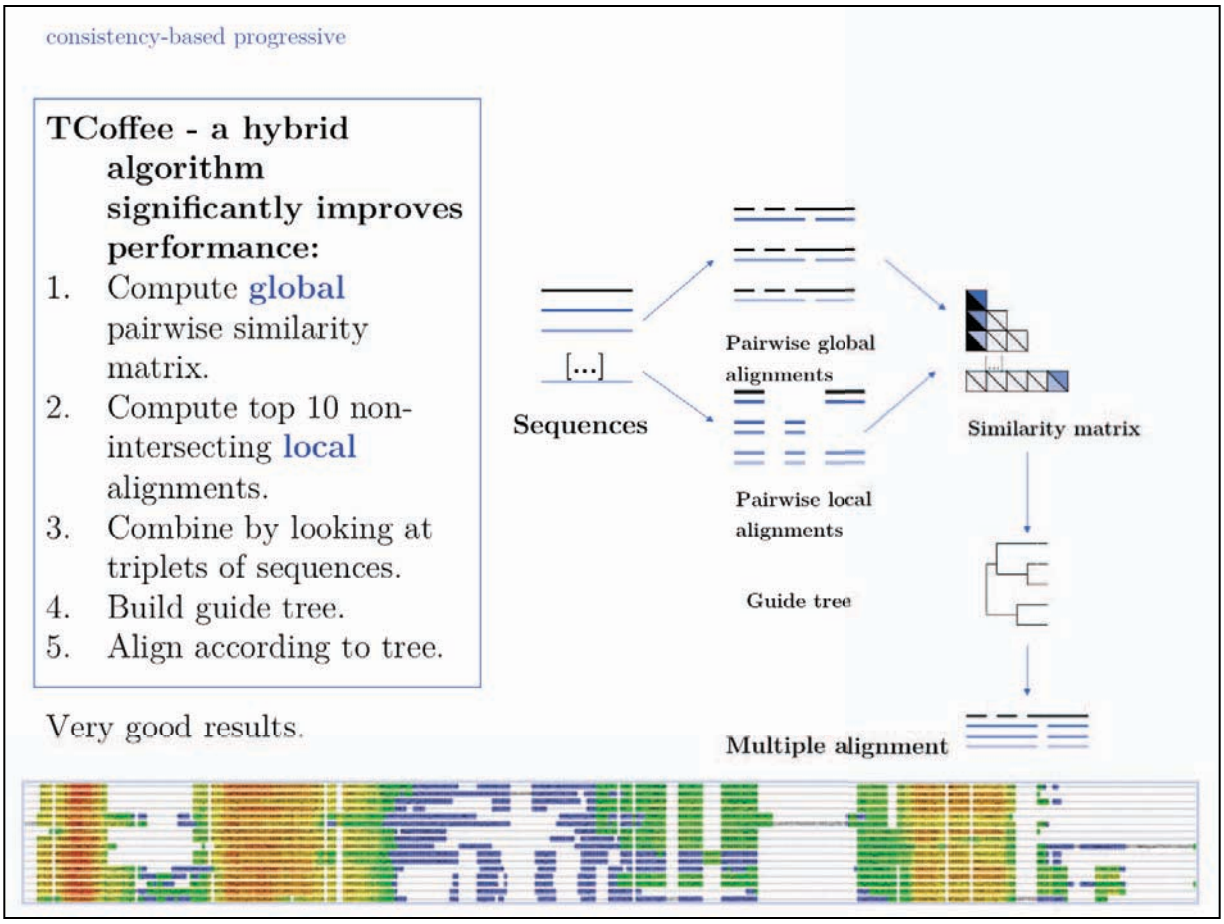
PSI-BLAST

1. Begin with BLAST search
2. Identify significant hits
3. Align to query
4. Compile into position specific scoring matrix (PSSM or "sequence profile")
5. **Repeat search with profile**
6. Add new aligned hits to PSSM
7. Iterate until no new sequences can be added

Results can be displayed as an MSA.
Choose formatting option:
"Flat query-anchored with letters for identities"



Altschul *et al.* (1998) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Research* **25**:3389-3402 (http://www.ncbi.nlm.nih.gov/pubmed/9254694)



I personally rate TCoffee as one of the most useful and useable tools that are currently available. It is robust, fast, and gives reasonable results for many cases. Usually it is **very** noticeably better then CLUSTAL and I would reject any result based on CLUSTAL for that reason.

Run TCoffee via the EBI **TCoffee** server which is very easy to use (although alignment size is limited;. Source code can be obtained and a local installation on UNIX machines is straightforward. The TCoffee Web page[1] links to another Web server and also offers 3DCoffee, a variant that automatically fetches related structures and incorporates structural alignments for increased accuracy.

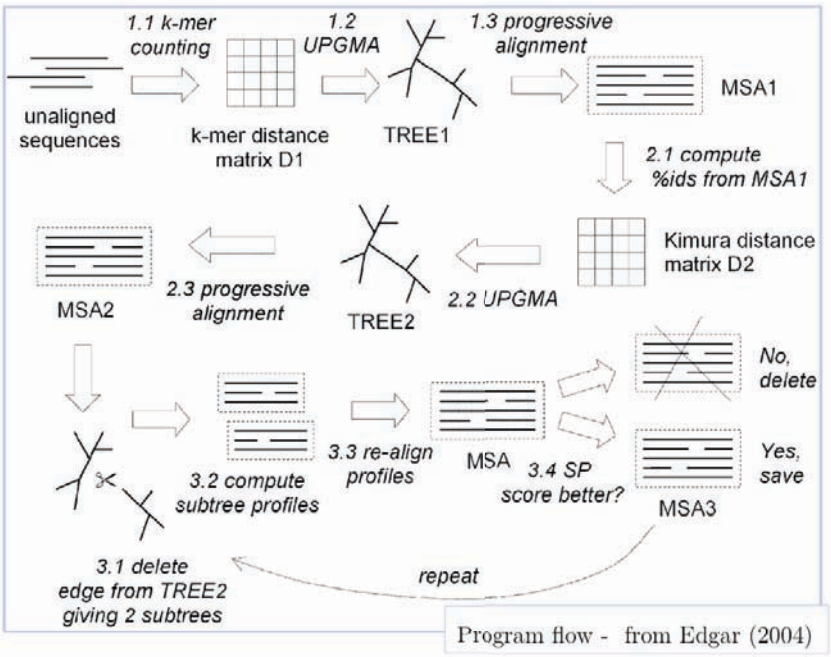
The inset image shows one of the useful features of TCoffee: an alignment output in which sequence is coloured according to the local quality of the alignment. This makes reliable and unreliable regions easy to spot, and immediately highlights outliers that could for example be due to sequence errors, such as frameshifts in exons. (MSA taken from the Mbp1 full-length protein alignment).

MUSCLE

Progressive draft alignment

Iterated improvement

Refinement, by aligning to groups of profiles



Program flow - from Edgar (2004)

Better than CLUSTAL, faster than ProbCons - one of the general purpose algorithms of first choice with the capacity to align thousands of sequences in one run.

Run MUSCLE MSAs via the EBI MSA server, which is very easy to use, or via the Berkeley **MUSCLE server**[1], courtesy of Kimmen Sjolander's lab. Source code and compiled code can be obtained from the MUSCLE homepage[2] and a local installation on UNIX and Windows machines is straightforward. That site also hosts the PREFAB multiple alignment benchmark.

MUSCLE is one of the algorithms provided in the R package **msa**.

[1] http://phylogenomics.berkeley.edu/cgi-bin/muscle/input_muscle.py

[2] http://www.drive5.com/muscle/ Muscle

Practical considerations for MSAs:

Principle: try to use sequences that are well distributed on the evolutionary tree. If a group of sequences biases the alignment, then not all information contributes equally to the result.

Include sequences with known structure wherever possible.

You may include more sequences than the ones that you are actually interested in comparing.

Ensure that your selection of sequences is appropriate to your question, e.g. clearly distinguish between orthologues and paralogues when trying to study function.

Spend some time and thought **before** you run the MSA to review the sequences that you are planning to align. Including un-alignable sequence **will** lead the algorithms astray and has the potential to degrade the entire alignment.

The requirement not to align **non-hmologous** sequence should really be extended not to align (or at least: not to evaluate) sequence segments that have evolved in different context, such as in different local structural environments after insertions or deletions have occurred. The reason is: if the structural environment is not conserved, the mutation data matrix scores are irrelevant for the residues that are paired up. They may be "aligned" by the algorithm, but they are really not equivalent in structure or function, thus whether they have a good or poor similarity score is meaningless.

Rule #1 of pairwise alignment applies:

Don't align non-homologous sequence!

Here are some heuristics for preparing the sequences for MSAs:

- Remove unalignable (non-homologous) domains and N- and C-terminal extensions.
- Consider what the alignment is supposed to show! Include no more sequence than that which is needed.
- Be cautious when including widely divergent outlier sequences - better to align those to a profile derived from the result, than to create errors in the sensitive early stages of progressive alignments.
- If individual sequences do not align well, reconsider the evidence for their homology: the alignment may try to tell you that they are not homologous after all and should be removed from the set.
- If sequence segments behave as outliers, they may indicate sequencing frameshifts, skipped exons or erroneously translated introns: review the evidence for the gene-model that was used to define the sequence.
- If domain annotations exist: align domains separately.

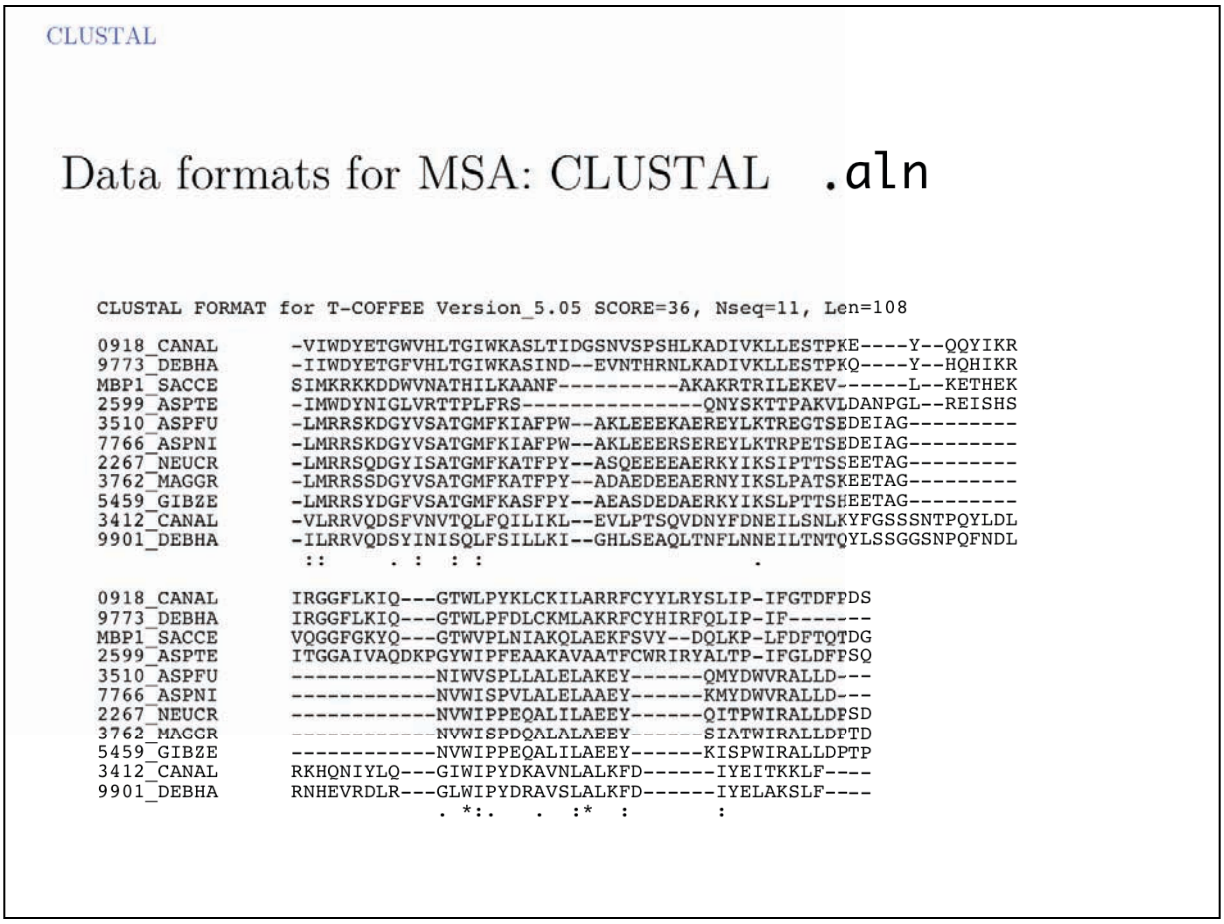
multi FASTA

Data formats for MSA: multi FASTA .mfa

```
>0918_CANAL XP_710918 256..352
-VIWDYETGWVHLTGIWKASLTIDGSNVSPSHLKADIVKLLESTPKE----Y--QQYIKR
IRGGFLKIQ---GTWLPYKLCKILARRFCYYLRYSLIP-IFGTDFPDS
>9773_DEBHA XP_459773 187..274
-IIWDYETGFVHLTGIWKASIND--EVNTHRNLKADIVKLLESTPKQ----Y--HQHIKR
IRGGFLKIQ---GTWLPFDLCKMLAKRFCYHIRFQLIP-IF-------
>MBP1_SACCE NP_010227 024..107
SIMKRKKDDWVNATHILKAANF----------AKAKRTRILEKEV------L--KETHEK
VQGGFGKYQ---GTWVPLNIAKQLAEKFSVY--DQLKP-LFDFTQTDG
>2599_ASPTE XP_001212599 130..218
-IMWDYNIGLVRTTPLFRS---------------QNYSKTTPAKVLDANPGL--REISHS
ITGGAIVAQDKPGYWIPFEAAKAVAATFCWRIRYALTP-IFGLDFPSQ
>3510_ASPFU XP_753510 089..163
-LMRRSKDGYVSATGMFKIAFPW--AKLEEEKAEREYLKTREGTSEDEIAG---------
------------NIWVSPLLALELAKEY------QMYDWVRALLD---
>7766_ASPNI XP_657766 089..163
-LMRRSKDGYVSATGMFKIAFPW--AKLEEERSEREYLKTRPETSEDEIAG---------
------------NVWISPVLALELAAEY------KMYDWVRALLD---
>2267_NEUCR XP_962267 085..162
-LMRRSQDGYISATGMFKATFPY--ASQEEEEAERKYIKSIPTTSSEETAG---------
------------NVWIPPEQALILAEEY------QITPWIRALLDPSD
>3762_MAGGR XP_363762 084..161
-LMRRSSDGYVSATGMFKATFPY--ADAEDEEAERNYIKSLPATSKEETAG---------
------------NVWISPDQALALAEEY------SIATWIRALLDPTD
>5459_GIBZE XP_385459 077..154
-LMRRSYDGFVSATGMFKASFPY--AEASDEDAERKYIKSLPTTSHEETAG---------
------------NVWIPPEQALILAEEY------KISPWIRALLDPTP
>3412_CANAL XP_723412 087..178
-VLRRVQDSFVNVTQLFQILIKL--EVLPTSQVDNYFDNEILSNLKYFGSSSNTPQYLDL
RKHQNIYLQ---GIWIPYDKAVNLALKFD------IYEITKKLF----
>9901_DEBHA XP_459901 067..158
-ILRRVQDSYINISQLFSILLKI--GHLSEAQLTNFLNNEILTNTQYLSSGGSNPQFNDL
RNHEVRDLR---GLWIPYDRAVSLALKFD------IYELAKSLF----
```

Three common formats exist for MSA results. An **aligned** multi FASTA file contains FASTA formatted sequences into which gap characters have been inserted.

Of course, multi FASTA files can also be unaligned and they are the most common way of formatting **input files** for MSAs.



Three common formats exist for MSA results.

The CLUSTAL format is not the same as the CLUSTAL algorithm. A CLUSTAL formatted alignment is the format in most common use.

Take care when formatting input FASTA files to ensure the **first 10 characters in your input file are unique** and contain **no special characters**! These are the characters that are usually used for the sequence names of the .aln files. I have seen programs break if they contain blanks, hyphens and **|** (the pipe character). The latter is especially annoying, since the **|** character is used in NCBI FASTA files to separate the database identifier from the accession number.

MSF

Data formats for MSA: MSF .msa

```
   MSF:  108  Type: P    Check:  3302   ..

 Name: 0918_CANAL oo  Len:  108  Check:  8295  Weight:  1.000
 Name: 9773_DEBHA oo  Len:  108  Check:  3488  Weight:  1.000
 Name: MBP1_SACCE oo  Len:  108  Check:   808  Weight:  1.000
 Name: 2599_ASPTE oo  Len:  108  Check:   241  Weight:  1.000
 Name: 3510_ASPFU oo  Len:  108  Check:  9082  Weight:  1.000
 Name: 7766_ASPNI oo  Len:  108  Check:  9952  Weight:  1.000
 Name: 2267_NEUCR oo  Len:  108  Check:  5383  Weight:  1.000
 Name: 3762_MAGGR oo  Len:  108  Check:  3063  Weight:  1.000
 Name: 5459_GIBZE oo  Len:  108  Check:  4901  Weight:  1.000
 Name: 3412_CANAL oo  Len:  108  Check:  5134  Weight:  1.000
 Name: 9901_DEBHA oo  Len:  108  Check:  2955  Weight:  1.000
//


0918_CANAL       .VIWDYETGW VHLTGIWKAS LTIDGSNVSP SHLKADIVKL LESTPKE... .Y..QQYIKR
9773_DEBHA       .IIWDYETGF VHLTGIWKAS IND..EVNTH RNLKADIVKL LESTPKQ... .Y..HQHIKR
MBP1_SACCE       SIMKRKKDDW VNATHILKAA NF........ ..AKAKRTRI LEKEV..... .L..KETHEK
2599_ASPTE       .IMWDYNIGL VRTTPLFRS. .......... ....QNYSKT TPAKVLDANP GL..REISHS
3510_ASPFU       .LMRRSKDGY VSATGMFKIA FPW..AKLEE EKAEREYLKT REGTSEDEIA G.........
7766_ASPNI       .LMRRSKDGY VSATGMFKIA FPW..AKLEE ERSEREYLKT RPETSEDEIA G.........
2267_NEUCR       .LMRRSQDGY ISATGMFKAT FPY..ASQEE EEAERKYIKS IPTTSSEETA G.........
3762_MAGGR       .LMRRSSDGY VSATGMFKAT FPY..ADAED EEAERNYIKS LPATSKEETA G.........
5459_GIBZE       .LMRRSYDGF VSATGMFKAS FPY..AEASD EDAERKYIKS LPTTSHEETA G.........
3412_CANAL       .VLRRVQDSF VNVTQLFQIL IKL..EVLPT SQVDNYFDNE ILSNLKYFGS SSNTPQYLDL
9901_DEBHA       .ILRRVQDSY INISQLFSIL LKI..GHLSE AQLTNFLNNE ILTNTQYLSS GGSNPQFNDL


0918_CANAL       IRGGFLKIQ. ..GTWLPYKL CKILARRFCY YLRYSLIP.I FGTDFPDS
9773_DEBHA       IRGGFLKIQ. ..GTWLPFDL CKMLAKRFCY HIRFQLIP.I F.......
MBP1_SACCE       VQGGFGKYQ. ..GTWVPLNI AKQLAEKFSV Y..DQLKP.L FDFTQTDG
2599_ASPTE       ITGGAIVAQD KPGYWIPFEA AKAVAATFCW RIRYALTP.I FGLDFPSQ
3510_ASPFU       .......... ..NIWVSPLL ALELAKEY.. ....QMYDWV RALLD...
7766_ASPNI       .......... ..NVWISPVL ALELAAEY.. ....KMYDWV RALLD...
2267_NEUCR       .......... ..NVWIPPEQ ALILAEEY.. ....QITPWI RALLDPSD
3762_MAGGR       .......... ..NVWISPDQ ALALAEEY.. ....STATWI RALLDPTD
5459_GIBZE       .......... ..NVWIPPEQ ALILAEEY.. ....KISPWI RALLDPTP
3412_CANAL       RKHQNIYLQ. ..GIWIPYDK AVNLALKFD. .....IYEIT KKLF....
9901_DEBHA       RNHEVRDLR. ..GLWIPYDR AVSLALKFD. .....IYELA KSLF....
```

Three common formats exist for MSA results.

MSF is a legacy format from the GCG package of sequence alignments, also produced by the EMBOSS tool EMMA, and supported as a valid input format for many programs. Gaps are denoted by periods and checksums are calculated for the sequences and for the alignment.

manual editing

Manual editing can often improve MSAs because you can take context into account whereas algorithms typically calculate scores based on objective functions that are computed over columns only.

(Context: e.g. where are indels placed, which residues are part of a functional site ...)

- Move sequences of different length (thus having obviously different structure) for dissimilar families into separate columns. Try not to include non-alignable residues in the same column and create the impression they are alignable.
- Move indels into regions adjacent to secondary structure elements, even if this creates a lower sequence alignment score. This is where they are commonly **accommodated** in structure, even if they have been generated elsewhere.
- Move two/four/six-residue insertions equally to both sides of a conserved beta-turn, rather than postulating only a single insertion on one side.
- Adjust gap positions to minimize the number of indel **sites.**
- Consider the evolutionary tree to minimize number of indel **events**.
- Conserve hydrophobic patterns in regions of secondary structure, e.g. alternating hydrophobic residues in beta-strands and $i \longrightarrow i+4$ patterns in alpha-helices.
- Preserve conservation especially in binding sites and functional motifs, even at the cost of larger indels.

It is common and perfectly permissible to manually edit a MSA with some biologically motivated heuristic in mind **as long as you document what you have done**! In the early days of MSAs, editing was simply required since the results were often obviously inadequate. In all cases in which the algorithm uses only the input sequences for the alignment, this still holds true. However, regarding the more modern template-based procedures (e.g. SPEM, PROMALS or PRALINE) I would be more reluctant to edit, since we may be actively ignoring/discarding the additional information the algorithm has used.

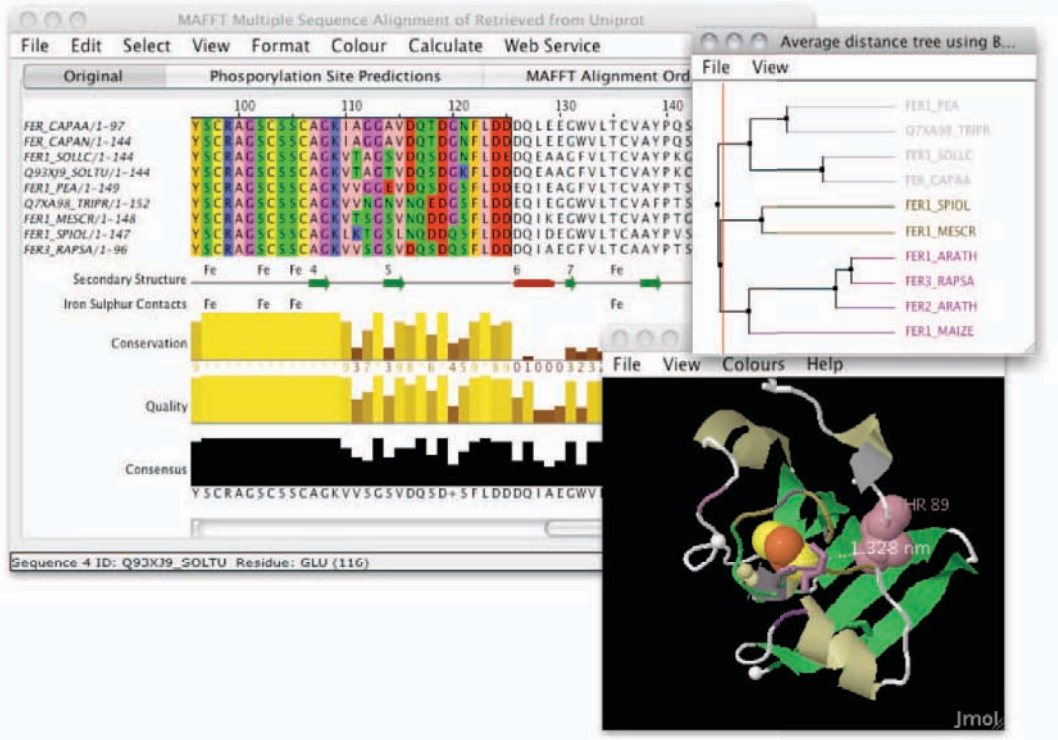
MS Word

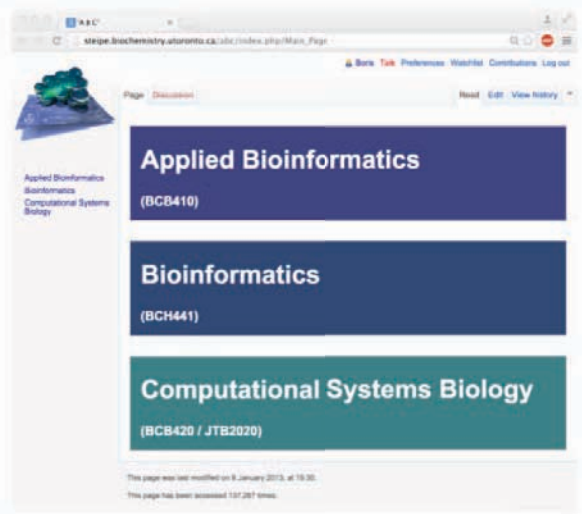
You **can** use a text-editor to edit MSA's, (e.g. MS Word) ...

1. Format your sequences in a **fixed-width** font (e.g. `Courier New`).
2. Display the entire sequence in one line: small font, landscape -page setup, legal- or custom wide paper size, small margins, magnified view.
3. Press the **<ALT>** key
4. You can now select **columns** and **blocks** of text.
5. You can **format** your selection; you can **delete**, **copy**, **cut**...
6. You can also **paste** entire columns or column ranges into the alignment.

... but you should avoid it, because **much better tools are freely available**.

JALVIEW


MAFFT Multiple Sequence Alignment of Retrieved from Uniprot
File Edit Select View Format Colour Calculate Web Service
Original
Phosporylation Site Predictions
MAFFT Alignment Ord
FER_CAPAA/1-97
FER_CAPAN/1-144
FER1_SOLLC/1-144
Q93XJ9_SOLTU/1-144
FER1_PEA/1-149
Q7XA98_TRIPR/1-152
FER1_MESCR/1-148
FER1_SPIOL/1-147
FER3_RAPSA/1-96
Secondary Structure
Iron Sulphur Contacts
Conservation
Quality
Consensus
Sequence 4 ID: Q93XJ9_SOLTU Residue: GLU (116)
Average distance tree using B...
File View
FER1_PEA
Q7XA98_TRIPR
FER1_SOLLC
FER_CAPAA
FER1_SPIOL
FER1_MESCR
FER1_ARATH
FER3_RAPSA
FER2_ARATH
FER1_MAIZE
File View Colours Help
HR 89
1.328 nm
Jmol



http://steipe.biochemistry.utoronto.ca/abc

B O R I S . S T E I P E @ U T O R O N T O . C A

DEPARTMENT OF BIOCHEMISTRY & DEPARTMENT OF MOLECULAR GENETICS
UNIVERSITY OF TORONTO, CANADA