

## Bioinformatics: Life Sciences on Your Computer

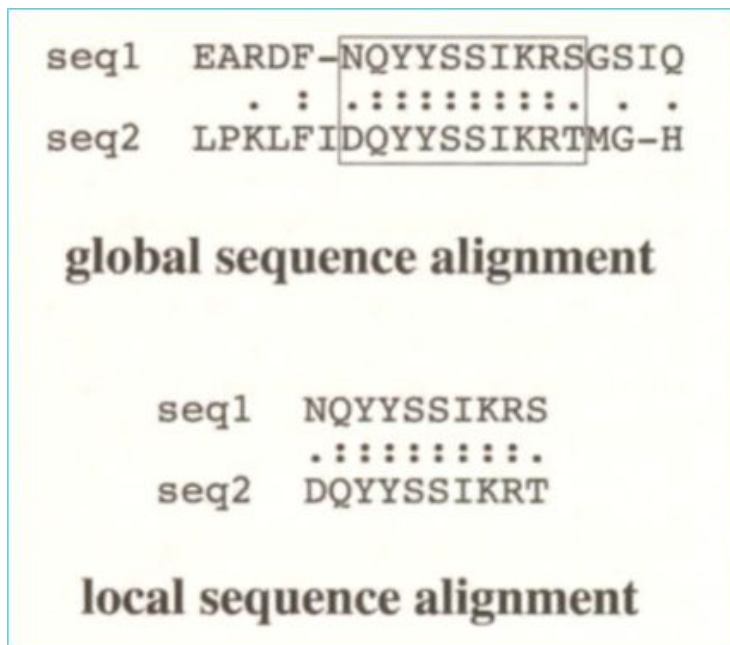
### Global & Local Alignment

*This is supplemental reading to the video on Dynamic Programming.*

#### Global and local alignments

Global alignment came before local alignment and the goal was to align the entire sequence from beginning to end and make as many matches as possible. Global alignment is better for very similar sequences and especially for sequences of equal length.

Local alignment was a modification of global alignment that looks for stretches with high density of matches to align. It generates islands of matched areas and is better for most sequences.



#### Methods of sequence alignment

An alignment can be visualized with dot matrix analysis (dotplots). Sequences are placed on two axes and dots are placed on graphs for matches. Most pairwise alignment algorithms are based on dynamic programming, which finds the optimal path for best fit sequence. There are slightly different algorithms for global and local alignments.

Finally, word (or k-tuple) methods search for identical short stretches of sequence before using dynamic programming. That's a shortcut designed to improve the speed of an algorithm.

#### Needleman-Wunsch (1970)

The Needleman-Wunsch algorithm was a global alignment strategy using dynamic programming. It aligns the sequence from start to finish and finds the best path to maximize scores. That usually means putting in gaps in just the right place to find a better path. The algorithm is similar to what phone companies used to use to find the best path for long distance calls.

**Discussion topic: From the video lecture: Why are vertical lines and horizontal lines gaps? Are there any other alignments that score equally in our exercises? How does global, local alignment differ in**

## this exercise?

So in finding the optimal path from corner to corner, because the ends of the sequences must align, you will get a good alignment with closely related proteins of similar size.

With the two proteins below, try EMBOSS, and use needle program for proteins.

<http://www.ebi.ac.uk/Tools/psa/>

```
>human beta globin (NP_000509)
MVHLTPEEKSAVTALWGKVNVEVGGEALGRLLVVYPWTQRFFESFGDLSTPDVAVMGNPKVKAHGKKVLG
AFSDGLAHLNLDNLKGTATLSELHCDKLHVDPENFRLLGNVLCVLAHHFGKEFTPPVQAAYQKVVAGVAN
ALAHKYH
```

```
>human alpha 1 globin (NP_000549)
MVLSPADKTNVKAAGKVGAGAHAGEYGAEALERMFSLFPTTKTYFPHFDLSHGSAQVKGHGKKVADALTNA
VAHVDDMPNALSALSDLHAHKLKRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTISK
YR
```

**Discussion topic: Any trouble with the alignment program? Most are very similar. Discuss results. Try changing the matrix or gap penalties.**

### Varying parameters

A Low Gap Length Penalty de-emphasizes gap length and results in higher identity, however it may take many gaps to artificially achieve that identity. Usually the gap length penalty is much higher than the length/existence penalty. A high mismatch penalty has a similar effect--it will create gaps to force mismatches.

A high gap penalty minimizes gaps. The best scores go to straight matches and that contributes to global alignment. It's good for near-identical sequences with few deletions.

The end gap penalty is what, perhaps more than anything else, distinguishes local from global alignment.

### End gaps & Smith-Waterman (1981)

The Smith-Waterman (1981) was a modification of the Needleman-Wunsch global alignment that results in a local alignment. The path may begin and end internally, not necessarily at beginning, end (or corners of the dot plot).

There is no end gap penalty!! That's the hallmark difference, but other adjustments to the algorithm were also made.

```

1 MKVWVALLLLAAWAAAERDCRVSSFRVKENFDKARFSGTWYAMAKKDPEG 50 RBP
. ||| | . |. . . | : .|||.:.| :
1 ...MKCLLLALALTCGAQALIVT..QTMKGLDIQKVAGTWYSLAMAASD. 44 lactoglobulin

51 LFLQDNIVAEFSVDETGQMSATAKGRVR.LLNNWD..VCADMVGTFTDTE 97 RBP
: | | | | : : | . | . || | : || | |
45 ISLLDAQSAPLRV.YVEELKPTPEGDLEILLQKWENGECQAQKKIIAEKTK 93 lactoglobulin

98 DPAKFKMKYWGVASFLQKGNDDHWIVDTDYDTYAV.....QYSC 136 RBP
|| ||. | :.|||| | . .|
94 IPAVFKIDALNENKVL.....VLDTDYKKYLLFCMENSAEPEQSLAC 135 lactoglobulin

137 RLLNLDGTCADSYSFVFSF...NGLPPEAQKIVRQRQ.EELCLARQYRLIV 185 RBP
. | | | | : || . | || |
136 QCLVRTPEVDDEALEKFDK...KALPMHIRLSFNPTQLEEQCHI..... 178 lactoglobulin

```

**Internal  
gap**

**Terminal  
gap**

There is a [detailed comparison](#) of Needleman-Wunsch (global) to Smith-Waterman (local) on the web in powerpoint format. The [SSEARCH program](#) is an example of a detailed Smith-Waterman alignment that is slow but exhaustive.

### End gap effect

End gaps: global vs. local

|       |   |     |   |   |    |   |   |    |   |      |
|-------|---|-----|---|---|----|---|---|----|---|------|
|       | M | -   | N | A | L  | S | D | R  | T |      |
|       | M | G   | S | D | R  | T | T | E  | T |      |
| score | 6 | -12 | 1 | 0 | -3 | 1 | 0 | -1 | 3 | = -5 |

### With End Gap Penalty

Using the Needleman-Wunsch (global) algorithm the end gaps are penalized, which forces ends to align to avoid those penalties.

```

-----
      M  N  A  L  S  D  R  T
      -  -  M  G  S  D  R  T  T  E  T
score  0  0 -1 -4  2  4  6  3  0  0  0 = 10

```

### Without End Gap Penalty

This Smith-Waterman (local) alignment has more appeal and much higher score. The gaps are on the ends, where they do not count. It finds the conserved SDRT without scoring end gaps. That was missed with end gap penalties.

### Global-local example

```

CURRENT--T-OPIC-S
CURRENTLYTROPICAL

```

Above is a global alignment (fictitious!) with lots of gaps. There is almost 100% sequence identity, but it seems forced. There is just one S-L mismatch but with a forced match of T in second word.

## CURRENT---TOPICS- CURRENTLYTROPICAL

The local alignment has only one internal gap. There are two mismatches (T-R and S-A) but without worry about where sequence ends, the alignment seems more realistic.

### Gap opening vs. gap extension

Which alignment is "better?"

GCTACTAG-T-T--CGC-T-TAGC  
GCTACTAGCTCTAGCGCGTATAGC  
0 mismatches, 5 gaps

GCTACTAGTT-----CGCTTAGC  
GCTACTAGCTCTAGCGCGTATAGC  
3 mismatches, 1 gap

Most algorithms use a high penalty for starting gap and a low penalty for the length of the gap. In above example, use:

G = -5 (gap starting penalty)  
L = -1 (gap length penalty)

**Discussion topic: What is the difference in gap penalties in that example. What if you used -10, -1 instead of -5, -1? Which penalties would overcome that kind of difference in mismatches?**

### Dynamic programming summary

Ignore low complexity regions. Those are stretches of the same character repeated (like a string of Qs in MTYHQQQQQQQQQWE). These are usually unrelated to function and noninformative. Then connect diagonal lines.

For global alignment, connect along entire length of sequence. That often results in lots of short gaps and lots of exact matches.

For local alignment, focus on best matching regions. That results in few gaps, but usually longer gaps than are found in global alignment. For the most part, local was a significant improvement over global, but there are still areas of bioinformatics where global is still useful.

**Global alignment:** Higher % identity, lots of gaps, often small

**Local alignment:** Lower % identity, few gaps, sometimes long

### Word (k-tuple) methods

A word method is used by both FASTA and BLAST to align sequences quickly. This is referred to as a **heuristic** method or approach. In bioinformatics, "heuristic" is usually a euphemism for shortcut. The idea is to search for strings of sequence identity and look for hits (identical matches). Words are later joined in the alignment by dynamic programming. Basically, finding word matches saves time, at the expense of some sensitivity.

### Pairwise alignment web servers

FASTA program suite (LALIGN): [http://fasta.bioch.virginia.edu/fasta/fasta\\_list.html](http://fasta.bioch.virginia.edu/fasta/fasta_list.html)

BLAST 2 sequence alignment: <http://blast.ncbi.nlm.nih.gov/Blast.cgi>

EBI pairwise suite: <http://www.ebi.ac.uk/Tools/psa/>

### FASTA algorithm

The FASTA program called LALIGN (local alignment) is slower but more accurate than BLAST, on average. The k-tup parameter is used to increase speed while sacrificing sensitivity. The default is ktup=2. If you switch to ktup=1, you will get a slower alignment that may be more accurate.

FASTA does not investigate every hit, but looks for hits with nearby hits that may be connected.

### BLAST algorithm

BLAST is the Basic Local Alignment Search Tool, found on the NCBI site. The default settings use the BLOSUM62 matrix for proteins. The word length is set to 3 for proteins and 11 for nucleic acids. The default has only one window, but click the check box that reads "align two or more sequences" (see below) and you'll get the second window for pairwise alignment.

The image shows the NCBI BLAST web interface. At the top, there's a navigation bar with 'Home', 'Recent Results', 'Saved Strategies', and 'Help'. Below this, the 'blastp' suite is selected. The main section is titled 'Enter Query Sequence' and includes a large text box for 'Enter accession number(s), gi(s), or FASTA sequence(s)', a 'Clear' button, and a 'Query subrange' section with 'From' and 'To' fields. Below the text box is an 'Or, upload file' section with a 'Browse...' button. A 'Job Title' field is also present. A checkbox labeled 'Align two or more sequences' is highlighted with a red circle. Below this is the 'Choose Search Set' section, which includes a 'Database' dropdown menu (currently set to 'Reference proteins (refseq\_protein)'), an 'Organism' field with a search suggestion box, an 'Exclude' button, and checkboxes for 'Models (XM/XP)' and 'Uncultured/environmental sample sequences'. There is also an 'Entrez Query' field. At the bottom, the 'Program Selection' section shows 'blastp (protein-protein BLAST)' as the selected algorithm.