SEQUENCE ANALYSIS

The term "**sequence analysis**" in biology implies subjecting a DNA or peptide sequence to sequence alignment, sequence databases, repeated sequence searches, or other bioinformatics methods on a computer.

In bioinformatics, a **sequence alignment** is a way of arranging the sequences of DNA, RNA, or protein to identify regions of similarity that may be a consequence of functional, structural, or evolutionary relationships between the sequences. Aligned sequences of nucleotide or amino acid residues are typically represented as rows within a matrix. Gaps are inserted between the residues so that identical or similar characters are aligned in successive columns.

Since the development of methods of high-throughput production of gene and protein sequences during the 90s, the rate of addition of new sequences to the databases increases continuously. Such a collection of sequences does not, by itself, increase the scientist's understanding of the biology of organisms. However, comparing sequences with known functions with these new sequences is one way of understanding the biology of that organism from which the new sequence comes. Thus, sequence analysis can be used to assign function to genes and proteins by the study of the similarities between the compared sequences. Nowadays there are many tools and techniques that provide the sequence comparisons (sequence alignment) and analyze the alignment product to understand the biology.

Sequence analysis in molecular biology and bioinformatics is an automated, computer-based examination of characteristic fragments, e.g. of a DNA strand. It basically includes relevant topics:

1. The comparison of sequences in order to find similarity and dissimilarity in compared sequences (sequence alignment)
2. Identification of gene-structures, reading frames, distributions of introns and exons and regulatory elements
3. Finding and comparing point mutations or the single nucleotide polymorphism (SNP) in organism in order to get the genetic marker.
4. Revealing the evolution and genetic diversity of organisms.
5. Function annotation of genes.

If two sequences in an alignment share a common ancestor, mismatches can be interpreted as point mutations and gaps as indels (that is, insertion or deletion mutations) introduced in one or both lineages in the time since they diverged from one another. In sequence alignments of proteins, the degree of similarity between amino acids occupying a particular position in the sequence can be interpreted as a rough measure of how conserved a particular region or sequence motif is among lineages. The absence of substitutions, or the presence of only very conservative substitutions (that is, the substitution of amino acids whose side chains have similar biochemical properties) in a particular region of the sequence, suggest that this region has structural or functional importance. Although DNA and RNA nucleotide bases are more similar to each other than are amino acids, the conservation of base pairs can indicate a similar functional or structural role.

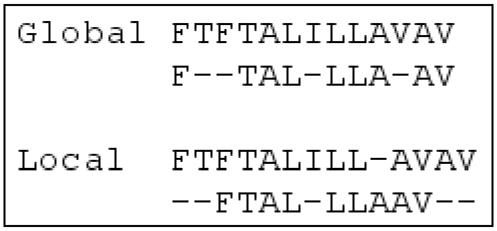
Very short or very similar sequences can be aligned by hand. However, most interesting problems require the alignment of lengthy, highly variable or extremely numerous sequences that cannot be aligned solely by human effort. Instead, human knowledge is applied in constructing algorithms to produce high-quality sequence alignments, and occasionally in adjusting the final results to reflect patterns that are difficult to represent algorithmically (especially in the case of nucleotide sequences). Computational approaches to sequence alignment generally fall into two categories: *global alignments* and *local alignments*. Calculating a global alignment is a form of global optimization that "forces" the alignment to span the entire length of all query sequences. By contrast, local alignments identify regions of similarity within long sequences that are often widely divergent overall. Local alignments are often preferable, but can be more difficult to calculate because of the additional challenge of identifying the regions of similarity. A variety of computational algorithms have been applied to the sequence alignment problem, including slow but formally optimizing methods like dynamic programming, and efficient, but not as thorough heuristic algorithms or probabilistic methods designed for large-scale database search.

Alignments are commonly represented both graphically and in text format. In almost all sequence alignment representations, sequences are written in rows arranged so that aligned residues appear in successive columns. In text formats, aligned columns containing identical or similar characters are indicated with a system of conservation symbols. As in the image above, an asterisk or pipe symbol is used to show identity between two columns; other less common symbols include a colon for conservative substitutions and a period for semi conservative substitutions. Many sequence visualization programs also use color to display information about the properties of the individual sequence elements; in DNA and RNA sequences, this equates to assigning each nucleotide its own color. In protein alignments, such as the one in the image above, color is often used to indicate amino acid properties to aid in judging the conservation of a given amino acid substitution. For multiple sequences the last row in each column is often the consensus sequence determined by the alignment; the consensus sequence is also often represented in graphical format with a sequence logo in which the size of each nucleotide or amino acid letter corresponds to its degree of conservation.

Sequence alignments can be stored in a wide variety of text-based file formats, many of which were originally developed in conjunction with a specific alignment program or implementation. Most web-based tools allow a limited number of input and output formats, such as FASTA format and GenBank format and the output is not easily editable. Several conversion programs are available, READSEQ or EMBOSS having a graphical interfaces or command line interfaces, while several programming packages like BioPerl, BioRuby provide functions to do this.

GLOBAL and LOCAL Alignment

Global alignments, which attempt to align every residue in every sequence, are most useful when the sequences in the query set are similar and of roughly equal size. (This does not mean global alignments cannot end in gaps.) A general global alignment technique is the Needleman-Wunsch algorithm, which is based on dynamic programming. Local alignments are more useful for dissimilar sequences that are suspected to contain regions of similarity or similar sequence motifs within their larger sequence context. The Smith-Waterman algorithm is a general local alignment method also based on dynamic programming. With sufficiently similar sequences, there is no difference between local and global alignments.



```
Global  FTFTALILLAVAV
        F--TAL-LLA-AV

Local   FTFTALILL-AVAV
        --FTAL-LLAAV--
```

Figure 1: Illustration of global and local alignments demonstrating the 'gappy' quality of global alignments that can occur if sequences are insufficiently similar

The following is an example of global sequence alignment using Needleman/Wunsch techniques. For this example, the two sequences to be globally aligned are

G A A T T C A G T T A (sequence #1)
G G A T C G A (sequence #2)

So M = 11 and N = 7 (the length of sequence #1 and sequence #2, respectively)

A simple scoring scheme is assumed where

- $S_{i,j}$ = 1 if the residue at position i of sequence #1 is the same as the residue at position j of sequence #2 (match score); otherwise
- $S_{i,j}$ = 0 (mismatch score)
- w = 0 (gap penalty)

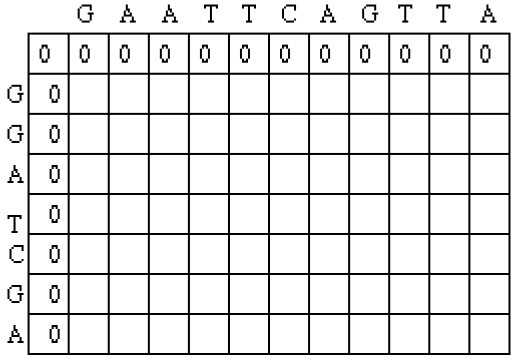
Three steps in global alignment

1. Initialization
2. Matrix fill (scoring)
3. Traceback (alignment)
4.

1. Initialization Step

The first step in the global alignment dynamic programming approach is to create a matrix with M + 1 columns and N + 1 rows where M and N correspond to the size of the sequences to be aligned.

Since this example assumes there is no gap opening or gap extension penalty, the first row and first column of the matrix can be initially filled with 0.



		G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0	0
G	0											
G	0											
A	0											
T	0											
C	0											
G	0											
A	0											

2. Matrix Fill Step

One possible (inefficient) solution of the matrix fill step finds the maximum global alignment score by starting in the upper left hand corner in the matrix and finding the maximal score $M_{i,j}$ for each position in the matrix. In order to find $M_{i,j}$ for any i,j it is minimal to know the score for the matrix positions to the left, above and diagonal to i, j. In terms of matrix positions, it is necessary to know $M_{i-1,j}$, $M_{i,j-1}$ and $M_{i-1,\, j-1}$.

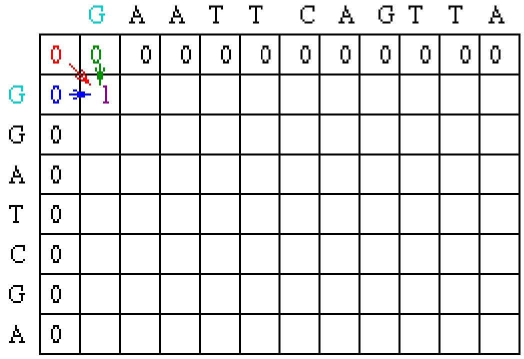
For each position, $M_{i,j}$ is defined to be the maximum score at position i,j; i.e.

$$\mathbf{M_{i,j} = MAXIMUM[}$$
$$\quad \mathbf{M_{i-1,\, j-1} + S_{i,j}} \text{ (match/mismatch in the diagonal),}$$
$$\quad \mathbf{M_{i,j-1} + w} \text{ (gap in sequence \#1),}$$
$$\quad \mathbf{M_{i-1,j} + w} \text{ (gap in sequence \#2)}\mathbf{]}$$

Note that in the example, $M_{i-1,j-1}$ will be red, $M_{i,j-1}$ will be green and $M_{i-1,j}$ will be blue.

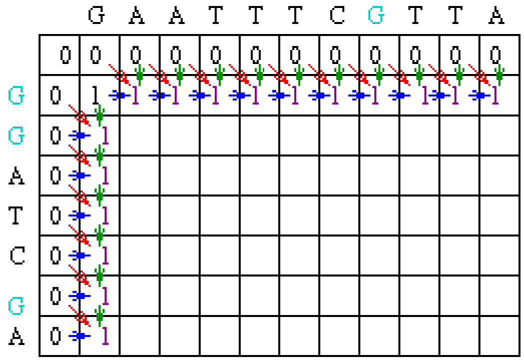
Using this information, the score at position 1,1 in the matrix can be calculated. Since the first residue in both sequences is a G, $S_{1,1} = 1$, and by the assumptions stated at the beginning, w = 0. Thus, $M_{1,1} = MAX[M_{0,0} + 1, M_{1,\,0} + 0, M_{0,1} + 0] = MAX\ [1, 0, 0] = 1$.

A value of 1 is then placed in position 1,1 of the scoring matrix.



		G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0	0
G	0	1										
G	0											
A	0											
T	0											
C	0											
G	0											
A	0											

Since the gap penalty (w) is 0, the rest of row 1 and column 1 can be filled in with the value 1. Take the example of row 1. At column 2, the value is the max of 0 (for a mismatch), 0 (for a vertical gap) or 1 (horizontal gap). The rest of row 1 can be filled out similarly until we get to column 8. At this point, there is a G in both sequences (light blue). Thus, the value for the cell at row 1 column 8 is the maximum of 1 (for a match), 0 (for a vertical gap) or 1 (horizontal gap). The value will again be 1. The rest of row 1 and column 1 can be filled with 1 using the above reasoning.

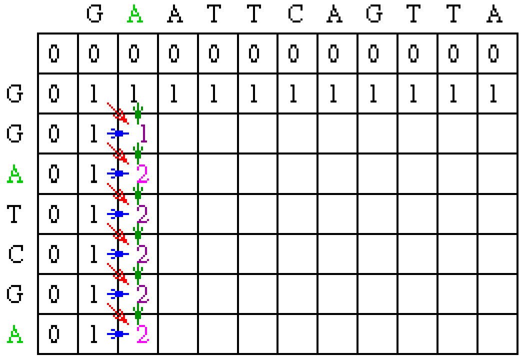


		G	A	A	T	T	T	C	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0	0
G	0	1	1	1	1	1	1	1	1	1	1	1
G	0	1										
A	0	1										
T	0	1										
C	0	1										
G	0	1										
A	0	1										

Now let's look at column 2. The location at row 2 will be assigned the value of the maximum of 1(mismatch), 1(horizontal gap) or 1 (vertical gap). So its value is 1.

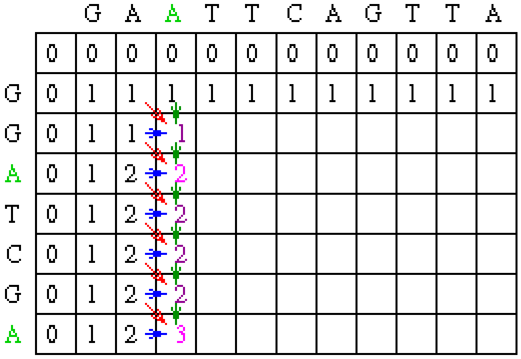
At the position column 2 row 3, there is an A in both sequences. Thus, its value will be the maximum of 2(match), 1 (horizontal gap), 1 (vertical gap) so its value is 2.

Moving along to position colum 2 row 4, its value will be the maximum of 1 (mismatch), 1 (horizontal gap), 2 (vertical gap) so its value is 2. Note that for all of the remaining positions except the last one in column 2, the choices for the value will be the exact same as in row 4 since there are no matches. The final row will contain the value 2 since it is the maximum of 2 (match), 1 (horizontal gap) and 2(vertical gap).



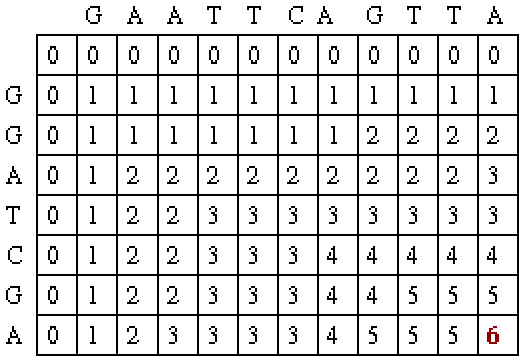
		G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0	0
G	0	1	1	1	1	1	1	1	1	1	1	1
G	0	1	1									
A	0	1	2									
T	0	1	2									
C	0	1	2									
G	0	1	2									
A	0	1	2									

Using the same techniques as described for column 2, we can fill in column 3.



		G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0	0
G	0	1	1	1	1	1	1	1	1	1	1	1
G	0	1	1	1								
A	0	1	2	2								
T	0	1	2	2								
C	0	1	2	2								
G	0	1	2	2								
A	0	1	2	3								

After filling in all of the values the score matrix is as follows:



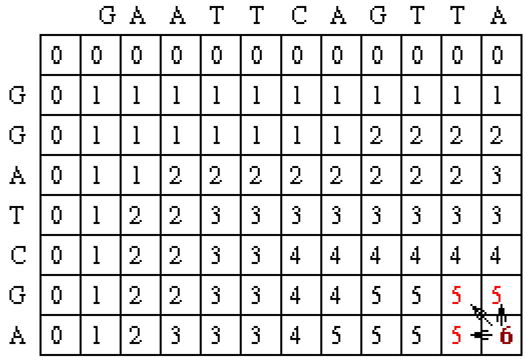
		G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0	0
G	0	1	1	1	1	1	1	1	1	1	1	1
G	0	1	1	1	1	1	1	1	2	2	2	2
A	0	1	2	2	2	2	2	2	2	2	2	3
T	0	1	2	2	3	3	3	3	3	3	3	3
C	0	1	2	2	3	3	3	4	4	4	4	4
G	0	1	2	2	3	3	3	4	4	5	5	5
A	0	1	2	3	3	3	3	4	5	5	5	6

3. Traceback Step

After the matrix fill step, the maximum alignment score for the two test sequences is 6. The traceback step determines the actual alignment(s) that result in the maximum score. Note that with a simple scoring algorithm such as one that is used here, there are likely to be multiple maximal alignments.

The traceback step begins in the M,J position in the matrix, i.e. the position that leads to the maximal score. In this case, there is a 6 in that location.

Traceback takes the current cell and looks to the neighbor cells that could be direct predacessors. This means it looks to the neighbor to the left (gap in sequence #2), the diagonal neighbor (match/mismatch), and the neighbor above it (gap in sequence #1). The algorithm for traceback chooses as the next cell in the sequence one of the possible predacessors. In this case, the neighbors are marked in red. They are all also equal to 5.

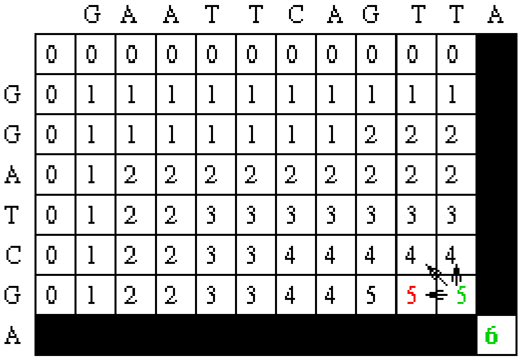


		G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0	0
G	0	1	1	1	1	1	1	1	1	1	1	1
G	0	1	1	1	1	1	1	1	2	2	2	2
A	0	1	1	2	2	2	2	2	2	2	2	3
T	0	1	2	2	3	3	3	3	3	3	3	3
C	0	1	2	2	3	3	4	4	4	4	4	4
G	0	1	2	2	3	3	4	4	5	5	5	5
A	0	1	2	3	3	3	4	5	5	5	5	6

Since the current cell has a value of 6 and the scores are 1 for a match and 0 for anything else, the only possible predacessor is the diagonal match/mismatch neighbor. If more than one possible predacessor exists, any can be chosen. This gives us a current alignment of

```
(Seq #1)    A
            |
(Seq #2)    A
```

So now we look at the current cell and determine which cell is its direct predacessor. In this case, it is the cell with the red 5.

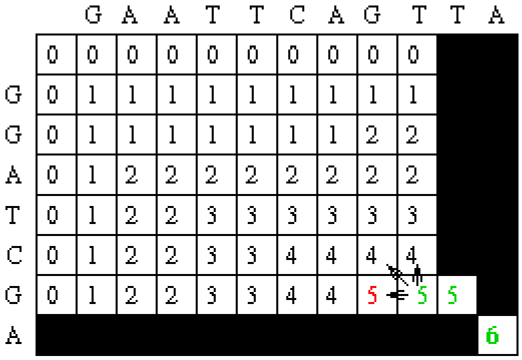


		G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0	0	
G	0	1	1	1	1	1	1	1	1	1	1	
G	0	1	1	1	1	1	1	1	2	2	2	
A	0	1	2	2	2	2	2	2	2	2	2	
T	0	1	2	2	3	3	3	3	3	3	3	
C	0	1	2	2	3	3	4	4	4	4	4	
G	0	1	2	2	3	3	4	4	5	5	5	
A												6

The alignment as described in the above step adds a gap to sequence #2, so the current alignment is

(Seq #1)   T A
             |
(Seq #2)   _ A

Once again, the direct predacessor produces a gap in sequence #2.

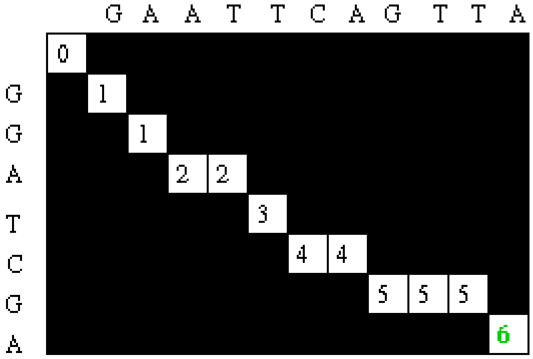


		G	A	A	T	T	C	A	G	T	T	A
	0	0	0	0	0	0	0	0	0	0		
G	0	1	1	1	1	1	1	1	1	1		
G	0	1	1	1	1	1	1	1	2	2		
A	0	1	2	2	2	2	2	2	2	2		
T	0	1	2	2	3	3	3	3	3	3		
C	0	1	2	2	3	3	4	4	4	4		
G	0	1	2	2	3	3	4	4	5	5	5	
A												6

After this step, the current alignment is

(Seq #1)   T T A
               |
           _ _ A

Continuing on with the traceback step, we eventually get to a position in column 0 row 0 which tells us that traceback is completed. One possible maximum alignment is :



		G	A	A	T	T	C	A	G	T	T	A
	0											
G		1										
G			1									
A				2	2							
T						3						
C							4	4				
G									5	5	5	
A												6

Giving an alignment of :

G A A T T C A G T T A
| | | | | |
G G A _ T C _ G _ _ A

Hybrid methods, known as semiglobal or "glocal" methods, attempt to find the best possible alignment that includes the start and end of one or the other sequence. This can be especially useful when the downstream part of one sequence overlaps with the upstream part of the other sequence. In this case, neither global nor local alignment is entirely appropriate: a global alignment would attempt to force the alignment to extend beyond the region of overlap, while a local alignment might not fully cover the region of overlap.

Sequence alignment is broadly divided into Pair-wise alignment and Multiple Sequence Alignment.

1. PAIRWISE ALIGNMENT

Pairwise sequence alignment methods are used to find the best-matching piecewise (local) or global alignments of two query sequences. Pairwise alignments can only be used between two sequences at a time, but they are efficient to calculate and are often used for methods that do not require extreme precision (such as searching a database for sequences with high similarity to a query). The three primary methods of producing pairwise alignments are **dot-matrix methods**, **dynamic programming**, and **word methods**. Although each method has its individual strengths and weaknesses, all three pairwise methods have difficulty with highly repetitive sequences of low information content - especially where the number of repetitions differ in the two sequences to be aligned. One way of quantifying the utility of a given pairwise alignment is the 'maximum unique match', or the longest subsequence that occurs in both query sequence. Longer MUM sequences typically reflect closer relatedness.

a) Dot-Matrix Method

The dot-matrix approach, which implicitly produces a family of alignments for individual sequence regions, is qualitative and conceptually simple, though time-consuming to analyze on a large scale. In the absence of noise, it can be easy to visually identify certain sequence features—such as insertions, deletions, repeats, or inverted repeats—from a dot-matrix plot. To construct a dot-matrix plot, the two sequences are written along the top row and leftmost column of a two-dimensional matrix and a dot is placed at any point where the characters in the appropriate columns match—this is a typical recurrence plot. Some implementations vary the size or intensity of the dot depending on the degree of similarity of the two characters, to accommodate conservative substitutions. The dot plots of very closely related sequences will appear as a single line along the matrix's main diagonal.

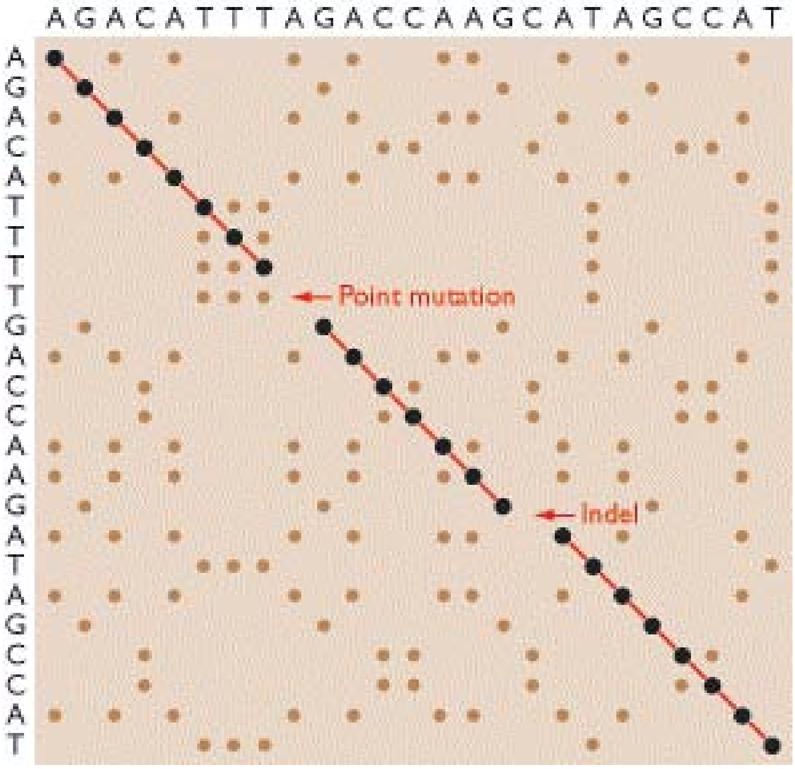

A G A C A T T T A G A C C A A G C A T A G C C A T
A G A C A T T T T G A C C A A G A T A G C C A T
Point mutation
Indel

Figure 2: The dot matrix technique for sequence alignment

Problems with dot plots as an information display technique include: noise, lack of clarity, non-intuitiveness, difficulty extracting match summary statistics and match positions on the two sequences. There is also much wasted space where the match data is inherently duplicated across the diagonal and most of the actual area of the plot is taken up by either empty space or noise, and, finally, dot-plots are limited to two sequences.

b) Dynamic Programming

The technique of dynamic programming can be applied to produce global alignments via the Needleman-Wunsch algorithm, and local alignments via the Smith-Waterman algorithm. In typical usage, protein alignments use a substitution matrix to assign scores to amino-acid matches or mismatches, and a gap penalty for matching an amino acid in one sequence to a gap in the other. DNA and RNA alignments may use a scoring matrix, but in practice often simply assign a positive match score, a negative mismatch score, and a negative gap penalty.

Dynamic programming can be useful in aligning nucleotide to protein sequences, a task complicated by the need to take into account frameshift mutations (usually insertions or deletions). The framesearch method produces a series of global or local pairwise alignments between a query nucleotide sequence and a search set of protein sequences, or vice versa.

Although the method is very slow, its ability to evaluate frameshifts offset by an arbitrary number of nucleotides makes the method useful for sequences containing large numbers of indels, which can be very difficult to align with more efficient heuristic methods. In practice, the method requires large amounts of computing power or a system whose architecture is specialized for dynamic programming. The BLAST and EMBOSS suites provide basic tools for creating translated alignments (though some of these approaches take advantage of side-effects of sequence searching capabilities of the tools.

c) Word Method

Word methods, also known as *k*-tuple methods, are heuristic methods that are not guaranteed to find an optimal alignment solution, but are significantly more efficient than dynamic programming. These methods are especially useful in large-scale database searches where it is understood that a large proportion of the candidate sequences will have essentially no significant match with the query sequence. Word methods are best known for their implementation in the database search tools FASTA and the BLAST family. Word methods identify a series of short, nonoverlapping subsequences ("words") in the query sequence that are then matched to candidate database sequences. The relative positions of the word in the two sequences being compared are subtracted to obtain an offset; this will indicate a region of alignment if multiple distinct words produce the same offset. Only if this region is detected do these methods apply more sensitive alignment criteria; thus, many unnecessary comparisons with sequences of no appreciable similarity are eliminated.

In the FASTA method, the user defines a value *k* to use as the word length with which to search the database. The method is slower but more sensitive at lower values of *k*, which are also preferred for searches involving a very short query sequence. The BLAST family of search methods provides a number of algorithms optimized for particular types of queries, such as searching for distantly related sequence matches. BLAST was developed to provide a faster alternative to FASTA without sacrificing much accuracy; like FASTA, BLAST uses a word search of length *k*, but evaluates only the most significant word matches, rather than every word match as does FASTA. Most BLAST implementations use a fixed default word length that is optimized for the query and database type, and that is changed only under special circumstances, such as when searching with repetitive or very short query sequences. Implementations can be found via a number of web portals, such as EMBL FASTA and NCBI BLAST.

2. MULTIPLE SEQUENCE ALIGNMENT

Multiple sequence alignment is an extension of pairwise alignment to incorporate more than two sequences at a time. Multiple alignment methods try to align all of the sequences in a given query set. Multiple alignments are often used in identifying conserved sequence regions across a group of sequences hypothesized to be evolutionarily related. Such conserved sequence motifs can be used in conjunction with structural and mechanistic information to locate the catalytic active sites of enzymes. Alignments are also used to aid in establishing evolutionary relationships by constructing phylogenetic trees.

a) Dynamic Programming

The technique of dynamic programming is theoretically applicable to any number of sequences; however, because it is computationally expensive in both time and memory, it is rarely used for more than three or four sequences in its most basic form. This method requires constructing the *n*-dimensional equivalent of the sequence matrix formed from two sequences, where *n* is the number of sequences in the query. Standard dynamic programming is first used on all pairs of query sequences and then the "alignment space" is filled in by considering possible matches or gaps at intermediate positions, eventually constructing an alignment essentially between each two-sequence alignment. Although this technique is computationally expensive, its guarantee of a global optimum solution is useful in cases where only a few sequences need to be aligned accurately.

b) Progressive Methods

Progressive, hierarchical, or tree methods generate a multiple sequence alignment by first aligning the most similar sequences and then adding successively less related sequences or groups to the alignment until the entire query set has been incorporated into the solution. The initial tree describing the sequence relatedness is based on pairwise comparisons that may include heuristic pairwise alignment methods similar to FASTA. Progressive alignment results are dependent on the choice of "most related" sequences and thus can be sensitive to inaccuracies in the initial pairwise alignments. Most progressive multiple sequence alignment methods additionally weight the sequences in the query set according to their relatedness, which reduces the likelihood of making a poor choice of initial sequences and thus improves alignment accuracy.

Many variations of the Clustal progressive implementation are used for multiple sequence alignment, phylogenetic tree construction, and as input for protein structure prediction. A slower but more accurate variant of the progressive method is known as T-Coffee

c) Iterative Methods

Iterative methods attempt to improve on the weak point of the progressive methods, the heavy dependence on the accuracy of the initial pairwise alignments. Iterative methods optimize an objective function based on a selected alignment scoring method by assigning an initial global alignment and then realigning sequence subsets. The realigned subsets are then themselves aligned to produce the next iteration's multiple sequence alignment. Various ways of selecting the sequence subgroups and objective function are reviewed in.

d) Motif Finding

Motif finding, also known as profile analysis, constructs global multiple sequence alignments that attempt to align short conserved sequence motifs among the sequences in the query set. This is usually done by first constructing a general global multiple sequence alignment, after which the highly conserved regions are isolated and used to construct a set of profile matrices. The profile matrix for each conserved region is arranged like a scoring matrix but its frequency counts for each amino acid or nucleotide at each position are derived from the conserved region's character distribution rather than from a more general empirical distribution. The profile matrices are then used to search other sequences for occurrences of the motif they characterize. In cases where the original data set contained a small number of sequences, or only highly related sequences, pseudocounts are added to normalize the character distributions represented in the motif.

Techniques inspired by computer science

A variety of general optimization algorithms commonly used in computer science have also been applied to the multiple sequence alignment problem. Hidden Markov models have been used to produce probability scores for a family of possible multiple sequence alignments for a given query set; although early HMM-based methods produced underwhelming performance, later applications have found them especially effective in detecting remotely related sequences because they are less susceptible to noise created by conservative or semiconservative substitutions. Genetic algorithms and simulated annealing have also been used in optimizing multiple sequence alignment scores as judged by a scoring function like the sum-of-pairs method. More complete details and software packages can be found in the main article multiple sequence alignment.

Phylogenetics and sequence alignment are closely related fields due to the shared necessity of evaluating sequence relatedness. The field of phylogenetics makes extensive use of sequence alignments in the construction and interpretation of phylogenetic trees, which are used to classify the evolutionary relationships between homologous genes represented in the genomes of divergent species. The degree to which sequences in a query set differ is qualitatively related to the sequences' evolutionary distance from one another. Roughly speaking, high sequence identity suggests that the sequences in question have a comparatively young most recent common ancestor, while low identity suggests that the divergence is more ancient. This approximation, which reflects the "molecular clock" hypothesis that a roughly constant rate of evolutionary change can be used to extrapolate the elapsed time since two genes first diverged (that is, the coalescence time), assumes that the effects of mutation and selection are constant across sequence lineages. Therefore it does not account for possible difference among organisms or species in the rates of DNA repair or the possible functional conservation of specific regions in a sequence. (In the case of nucleotide sequences, the molecular clock hypothesis in its most basic form also discounts the difference in acceptance rates between silent mutations that do not alter the meaning of a given codon and other mutations that result in a different amino acid being incorporated into the protein.) More statistically accurate methods allow the evolutionary rate on each branch of the phylogenetic tree to vary, thus producing better estimates of coalescence times for genes. Progressive multiple alignment techniques produce a phylogenetic tree by necessity because they incorporate sequences into the growing alignment in order of relatedness

Significance of Sequence Alignment

Sequence alignments are useful in bioinformatics for identifying sequence similarity, producing phylogenetic trees, and developing homology models of protein structures. However, the biological relevance of sequence alignments is not always clear. Alignments are often assumed to reflect a degree of evolutionary change between sequences descended from a common ancestor; however, it is formally possible that convergent evolution can occur to produce apparent similarity between proteins that are evolutionarily unrelated but perform similar functions and have similar structures.

In database searches such as BLAST, statistical methods can determine the likelihood of a particular alignment between sequences or sequence regions arising by chance given the size and composition of the database being searched. These values can vary significantly depending on the search space. In particular, the likelihood of finding a given alignment by chance increases if the database consists only of sequences from the same organism as the query sequence. Repetitive sequences in the database or query can also distort both the search results and the assessment of statistical significance; BLAST automatically filters such repetitive sequences in the query to avoid apparent hits that are statistical artifacts.

Statistical significance indicates the probability that an alignment of a given quality could arise by chance, but does not indicate how much superior a given alignment is to alternative alignments of the same sequences. Measures of alignment credibility indicate the extent to which the best scoring alignments for a given pair of sequences are substantially similar.

The choice of a **scoring function** that reflects biological or statistical observations about known sequences is important to producing good alignments. Protein sequences are frequently aligned using substitution matrices that reflect the probabilities of given character-to-character substitutions. A series of matrices called PAM matrices (Point Accepted Mutation matrices, originally defined by Margaret Dayhoff and sometimes referred to as "Dayhoff matrices") explicitly encode evolutionary approximations regarding the rates and probabilities of particular amino acid mutations. Another common series of scoring matrices, known as BLOSUM (Blocks Substitution Matrix), encodes empirically derived substitution probabilities. Variants of both types of matrices are used to detect sequences with differing levels of divergence, thus allowing users of BLAST or FASTA to restrict searches to more closely related matches or expand to detect more divergent sequences. Gap penalties account for the introduction of a gap - on the evolutionary model, an insertion or deletion mutation - in both nucleotide and protein sequences, and therefore the penalty values should be proportional to the expected rate of such mutations. The quality of the alignments produced therefore depends on the quality of the scoring function.

It can be very useful and instructive to try the same alignment several times with different choices for scoring matrix and/or gap penalty values and compare the results. Regions where the solution is weak or non-unique can often be identified by observing which regions of the alignment are robust to variations in alignment parameters.

Sequenced RNA, such as expressed sequence tags and full-length mRNAs, can be aligned to a sequenced genome to find where there are genes and get information about alternative splicing and RNA editing. Sequence alignment is also a part of genome assembly, where sequences are aligned to find overlap so that *contigs* (long stretches of sequence) can be formed. Another use is SNP analysis, where sequences from different individuals are aligned to find single basepairs that are often different in a population.

The methods used for biological sequence alignment have also found applications in other fields, most notably in natural language processing and in social sciences. Techniques that generate the set of elements from which words will be selected in natural-language generation algorithms have borrowed multiple sequence alignment techniques from bioinformatics to produce linguistic versions of computer-generated mathematical proofs. In the field of historical and comparative linguistics, sequence alignment has been used to partially automate the comparative method by which linguists traditionally reconstruct languages. Business and marketing research has also applied multiple sequence alignment techniques in analyzing series of purchases over time.

Sequence Databases

The repositories for the genomic sequences are

1. National Center for Biotechnology Information (**NCBI**) is part of the United States National Library of Medicine (NLM), a branch of the National Institutes of Health. The NCBI is located in Bethesda, Maryland and was founded in 1988 through legislation sponsored by Senator Claude Pepper. The NCBI houses genome sequencing data in GenBank and an index of biomedical research articles in PubMed Central and PubMed, as well as other information relevant to biotechnology. All these databases are available online through the Entrez search engine. The NCBI is directed by David Lipman, one of the original authors of the BLAST sequence alignment program and a widely respected figure in Bioinformatics. The NCBI has had responsibility for making available the GenBank DNA sequence database since 1992. GenBank coordinates with individual laboratories and other sequence databases such as those of the European Molecular Biology Laboratory (EMBL) and the DNA Data Bank of Japan (DDBJ). Since 1992, NCBI has grown to provide other databases in addition to GenBank. NCBI provides Online Mendelian Inheritance in Man, the Molecular Modeling Database (3D protein structures), dbSNP a database of single-nucleotide polymorphisms, the Unique Human Gene Sequence Collection, a Gene Map of the human genome, a Taxonomy Browser, and coordinates with the National Cancer Institute to provide the Cancer Genome Anatomy Project. The NCBI assigns a unique identifier (Taxonomy ID number) to each species of organism. The NCBI has software tools that are available by WWW browsing or by FTP. For example, BLAST is a sequence similarity searching program. BLAST can do sequence comparisons against the GenBank DNA database in less than 15 seconds. The **NCBI Bookshelf** is a collection of freely available, downloadable, on-line versions of selected biomedical books. The Bookshelf has various titles covering aspects of molecular biology, biochemistry, cell biology, genetics, microbiology, a couple of disease states from a molecular and cellular point of view, research methods, and virology. Some of the books are online versions of previously published books, while others, such as Coffee Break (book), are written and edited by NCBI staff. The Bookshelf is a complement to the Entrez PubMed repository of peer-reviewed publication abstracts in that Bookshelf contents provide established perspectives on evolving areas of study and a context in which many disparate individual pieces of reported research can be organized.

2. European Molecular Biology Laboratory (**EMBL**) is a molecular biology research institution supported by 20 European countries and Australia as associate member state. The EMBL was created in 1974 and is a non-profit organisation funded by public research money from its member states. Research at EMBL is conducted by approximately 85 independent groups covering the spectrum of molecular biology. The Laboratory operates from five sites: the main Laboratory in Heidelberg, and Outstations in Hinxton (the European Bioinformatics Institute (**EBI**)), Grenoble, Hamburg, and Monterotondo near Rome. Each of the sites has a research specific field. At EBI, the research is oriented towards computational biology and bioinformatics. At Grenoble and Hamburg the research is in the field of structural biology. At Monterotondo the research is focused mainly on mouse models for clinical research. At the headquarters in Heidelberg, there are big departments in Cell Biology and Gene Expression as well as smaller complementing the aforementioned research fields. The cornerstones of EMBL's mission are: to perform basic research in molecular biology and molecular medicine, to train scientists, students and visitors at all levels, to offer vital services to scientists in the member states, to develop new instruments and methods in the life sciences, and to actively engage in technology transfer. EMBL's international PhD Programme has a student body of about 170. The Laboratory also sponsors an active Science and Society programme. Many scientific breakthroughs have been made at EMBL, most notably the first systematic genetic analysis of embryonic development in the fruit fly by Christiane Nüsslein-Volhard and Eric Wieschaus, for which they were awarded the Nobel Prize for Medicine in 1995.

3. DNA Data Bank of Japan (DDBJ) is a DNA data bank. It is located at the National Institute of Genetics (NIG) in the Shizuoka prefecture of Japan. It is also a member of the International Nucleotide Sequence Database Collaboration or INSDC. It exchanges its data with European Molecular Biology Laboratory at the European Bioinformatics Institute and with GenBank at the National Center for Biotechnology Information on a daily basis. Thus these three databanks contents the same data at any given time. DDBJ began data bank activities since 1986 at NIG and it boasts to be the only nucleotide sequence data bank in Asia. Although DDBJ mainly receives its data from Japanese researchers, however it can accept data from a contributor belonging to any other country. DDBJ is primarily funded by the Japanese Ministry of Education, Culture, Sports, Science and Technology (MEXT). DDBJ has an international advisory committee which consists of nine members, 3 members each from Europe, US, and Japan. This committee advice DDBJ about its maintenance, management and future plans once a year. Apart from this DDBJ also has an international collaborative committee which advises on various technical issues related to international collaboration and consists of working-level participants.

Softwares used in Sequence Alignment

	Name	Function	Website Link
1	ALIGN	Sequence Analysis	http://www.ebi.ac.uk/Tools/emboss/align
2	CENSOR	Sequence Analysis	http://www.ebi.ac.uk/Tools/censor/
3	CLUSTALW2	Sequence Analysis	http://www.ebi.ac.uk/Tools/clustalw2/
4	CpG Plot / CpGreport	Sequence Analysis	http://www.ebi.ac.uk/Tools/emboss/cpgplot/
5	Genewise	Sequence Analysis	http://www.ebi.ac.uk/Tools/Wise2/
6	Kalign	Sequence Analysis	http://www.ebi.ac.uk/Tools/kalign
7	MAFFT	Sequence Analysis	http://www.ebi.ac.uk/Tools/mafft/
8	MUSCLE	Sequence Analysis	http://www.ebi.ac.uk/Tools/muscle/
9	Pepstats/Pepwindow/Pepinfo	Sequence Analysis	http://www.ebi.ac.uk/Tools/emboss/pepinfo/
10	PromoterWise	Sequence Analysis	http://www.ebi.ac.uk/Tools/Wise2/promoterwise.html
11	SAPS	Sequence Analysis	http://www.ebi.ac.uk/Tools/saps/
12	T-coffee	Sequence Analysis	http://www.ebi.ac.uk/Tools/t-coffee/
13	Transeq	Sequence Analysis	http://www.ebi.ac.uk/Tools/emboss/transeq/
14	COBALT	Sequence Analysis	http://www.ncbi.nlm.nih.gov/tools/cobalt/
15	Genome Workbench	Sequence Analysis	http://www.ncbi.nlm.nih.gov/projects/gbench/
16	ORF Finder	Sequence Analysis	http://www.ncbi.nlm.nih.gov/gorf/gorf/html
17	Primer - BLAST	Sequence Analysis	http://www.ncbi.nlm.nih.gov/tools/primer-blast

18	ProSplign	Sequence Analysis	http://www.ncbi.nlm.nih.gov/sutils/static/prosplin/prosplign.html
19	Splign	Sequence Analysis	http://www.ncbi.nlm.nih.gov/sutils/splign/
20	VecScreen	Sequence Analysis	http://www.ncbi.nlm.nih.gov/VecScreen/VecScreen.html
21	Sequence Analysis	Sequence analysis	http://www.informagen.com/SA/
22	SeWeR	Sequence analysis	http://www.bioinformatics.org/sewer/
23	Motif Search	Sequence analysis	http://nbc11.biologie.uni-kl.de/framed/left/menu/auto/right/motifsearch2/index.pl
24	DNA Translator	Sequence analysis	http://nbc11.biologie.uni-kl.de/framed/left/menu/auto/right/JDT/
25	Non coding RNA Gene Finder	Sequence analysis	http://nbc11.biologie.uni-kl.de/framed/left/menu/auto/right/ncRnaGeneFinder/index.pl
26	TransTerm	Sequence analysis	http://nbc11.biologie.uni-kl.de/framed/left/menu/auto/right/transterm/
27	QRNA	Sequence analysis	http://nbc11.biologie.uni-kl.de/framed/left/menu/auto/right/qrna/
28	Clustalformatter 5	Sequence analysis	http://nbc11.biologie.uni-kl.de/framed/left/menu/auto/right/ClustalFormatter/
29	BioEdit	Sequence Alignment Editor	http://www.mbio.ncsu.edu/BioEdit/bioedit.html
30	FASTA	Sequence Similarity Search	http://www.ebi.ac.uk/Tools/fasta/
31	HMMER	Homology of protein	http://hmmer.janelia.org/
32	JAligner	Pairwise seq. alignment	http://jaligner.sourceforge.net/
33	JSTRING	Java Search for Tandem Repeats IN Genomes	http://bioinf.dms.med.uniroma1.it/JSTRING/
34	NCBI BLAST	Aligning Sequences	http://blast.ncbi.nlm.nih.gov/Blast.cgi
35	Gene Runner/ Motif Runner	Motif based sequence analysis	http://www.generunner.net/

36	GoCore	Protein Seq. Alignment & Analysis	http://www.helsinki.fi/project/ritvos/GoCore/
37	MAFFT	Multiple alignment	http://mafft.cbrc.jp/alignment/server/index.html
38	MAUVE	Multiple alignment	http://gel.ahabs.wisc.edu/mauve/
39	MEME Suite	Motif based sequence analysis	http://meme.nbcr.net/
40	CORAL (CDTree)	Aligning Core Conserved Regions	http://www.ncbi.nlm.nih.gov/Structure/cdtree/cdtree.shtml
41	BlastAlign	Align N Seq. with large INDELs	http://www.bioafrica.net/blast/BlastAlign.html
42	ARB software	Sequence DB Handling and Data Analysis	http://www.arb-home.de/
43	Automated Codon Usage Analysis Software - ACUA	Nucleotide Analysis	http://www.bioinsilico.com/acua
44	AnnHyb	Nucleotide Analysis	http://www.bioinformatics.org/annhyb/
45	SOAP2	Short read Alignment	http://soap.genomics.org.cn/
46	ACT (Artemis Comparison Tool)	DNA Sequence Comparison	http://www.sanger.ac.uk/resources/software/act/
47	WU-BLAST	Multiple Sequence Alignment	www.ebi.ac.uk/Tools/blast2/
48	CLUSTALW2	multiple sequence alignment	http://www.ebi.ac.uk/Tools/clustalw2/

After sequence alignment of proteins, there is a need for its structural prediction. The first step involved is **structural alignments**, which are usually specific to protein and sometimes RNA sequences, use information about the secondary and tertiary structure of the protein or RNA molecule to aid in aligning the sequences. These methods can be used for two or more sequences and typically produce local alignments; however, because they depend on the availability of structural information, they can only be used for sequences whose corresponding structures are known (usually through X-ray crystallography or NMR spectroscopy). Because both protein and RNA structure is more evolutionarily conserved than sequence, structural alignments can be more reliable between sequences that are very distantly related and that have diverged so extensively that sequence comparison cannot reliably detect their similarity.

Structural alignments are used as the "gold standard" in evaluating alignments for homology-based protein structure prediction because they explicitly align regions of the protein sequence that are structurally similar rather than relying exclusively on sequence information. However, clearly structural alignments cannot be used in structure prediction because at least one sequence in the query set is the target to be modeled, for which the structure is not known. It has been shown that, given the structural alignment between a target and a template sequence, highly accurate models of the target protein sequence can be produced; a major stumbling block in homology-based structure prediction is the production of structurally accurate alignments given only sequence information.