**Course: Advance Bio Informatics**

**Module Title: Cloud Computing for MSA**

**Module No: 143**

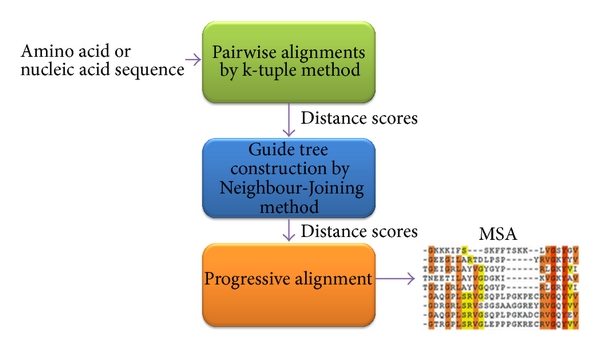
**Multiple Sequence Alignment**

Multiple sequence alignment (MSA) of DNA, RNA, and protein sequences is one of the most essential techniques in the fields of molecular biology, computational biology, and bioinformatics. Next-generation sequencing technologies are changing the biology landscape, flooding the databases with massive amounts of raw sequence data. MSA of ever-increasing sequence data sets is becoming a significant bottleneck. In order to realize the promise of MSA for large-scale sequence data sets, it is necessary for existing MSA algorithms to be run in a parallelized fashion with the sequence data distributed over a computing cluster or server farm. Combining MSA algorithms with cloud computing technologies is therefore likely to improve the speed, quality, and capability for MSA to handle large numbers of sequences.

In this module we will discuss ClustalW and Clustal Omega algorithms. Cloud computing technologies and concepts are outlined, and the next generation of cloud base MSA algorithms is introduced.

**ClustalW**

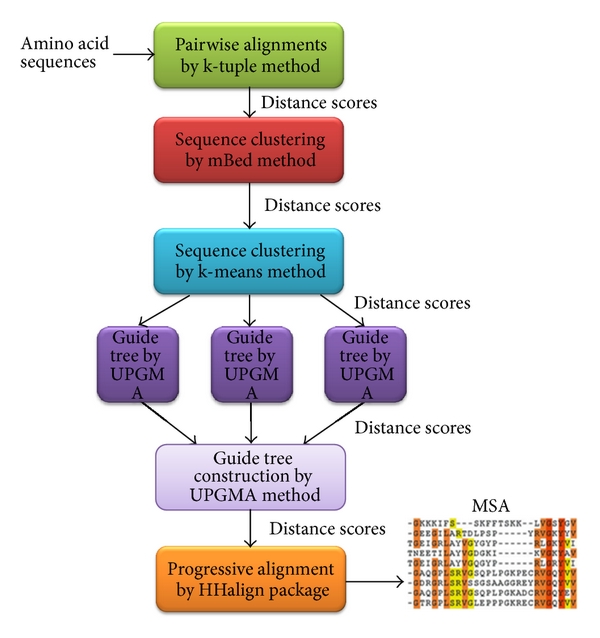
ClustalW was introduced by Thompson et al. in 1994 and quickly became the method of choice for producing multiple sequence alignments as it presented a dramatic increase in alignment quality, sensitivity, and speed in comparison with other algorithms. ClustalW incorporates a novel position-specific scoring scheme and a weighting scheme for down weighting overrepresented sequence groups, with the” representing “weights.” Firstly, the algorithm performs a pairwise alignment of all the sequences (nucleotide or amino acid) using the k-tuple method by Wilbur and Lipman which is a fast, albeit approximate, method or the Needleman-Wunsch method which is known as the full dynamic programming method. These methods calculate a matrix which shows the similarity of each pair of sequences. The similarity scores are converted to distance scores, and then the algorithm uses the distance scores to produce a guide tree, using the Neighbor-Joining (NJ) method for guide tree construction. The last step of the algorithm is the construction of the multiple sequence alignment of all the sequences. The MSA is constructed by progressively aligning the most closely related sequences according to the guide tree previously produced by the NJ method



ClustalW algorithm, which works by taking an input of amino acid or nucleic acid sequence, completing a pairwise alignment using the k-tuple method that guide tree construction using neighbor-joining method, following by a progressive alignment to output a multiple sequence alignment.

**Clustal Omega**

Clustal Omega is the latest MSA algorithm from the Clustal family. This algorithm is used to align protein sequences only (though nucleotide sequences are likely to be introduced in time). The accuracy of Clustal Omega on small numbers of sequences is similar to other high-quality aligners; however, on large sequence sets, Clustal Omega outperforms other MSA algorithms in terms of completion time and overall alignment quality. Clustal Omega is capable of aligning 190,000 sequences on a single processor in a few hours. The Clustal Omega algorithm produces a multiple sequence alignment by firstly producing pairwise alignments using the k-tuple method. Then, the sequences are clustered using the mBed method. This is followed by the k-means clustering method. The guide tree is next constructed using the UPGMA method. Finally, the multiple sequence alignment is produced using the HHalign package, which aligns two profile hidden Markov models.

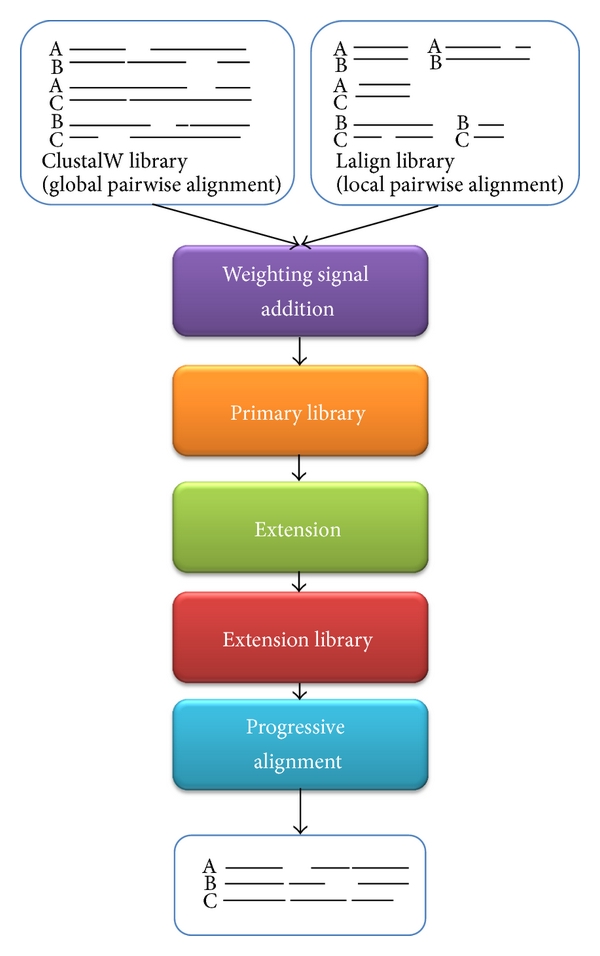


Clustal Omega algorithm, which works by taking an input of amino acid sequences, completing a pairwise alignment using the k-tuple method, sequence clustering using mBed method, and k-means method, guide tree construction using the UPGMA method, followed by a progressive alignment using HHalign package to output a multiple sequence alignment.

**T-Coffee**

T-Coffee, which stands for tree-based consistency objective function for alignment evolution, is an iterative MSA algorithm. T-Coffee provides a simple and flexible means of producing multiple sequence alignments by using heterogeneous data sources which are provided to T-Coffee via library of global and local pairwise alignments. In the progressive alignment, pairwise alignments are completed first in order to produce a distance matrix. This matrix is then used to produce a guide tree using the Neighbor-Joining method. The tree is then used to group the sequences together during the multiple sequence alignment process. The closest two sequences on the tree are aligned first using normal dynamic programming method. The alignment uses weighting in the extended library as shown in Figure 3.

This is done in order to align the residues in two sequences. The next two closest sequences suggested by the guide tree or prealigned group of sequences are always joined. This continues until all the sequences have been aligned. To align two groups of prealigned sequences, the scores from the extended library are used; however, the average library scores in each column of existing alignment are taken. T-Coffee increases the accuracy of the alignments 5–10% in comparison to ClustalW; however, the algorithm presents disadvantages such as weak scalability. T-Coffee can only align maximum 100 sequences without loss of accuracy.



T-COFFEE diagram. Steps involved in producing multiple sequence alignment by T-Coffee method.

**MAFFT**

Another good quality, highly accurate multiple sequence alignment is an algorithm called MAFFT. MAFFT uses two novel techniques; firstly, homologous regions are identified by the fast Fourier transform (FFT). In this method, the amino acid sequences are converted to a sequence composed of volume and polarity values of each amino acid residue. Secondly, a simplified scoring system is introduced which reduces CPU time and increases the accuracy of alignments.

MAFFT uses two-cycle heuristics, the progressive method (FFT-NS-2) and iterative refinement method (FFT-NS-i). In the (FFT-NS-2) method, low-quality all-pairwise distances are rapidly calculated, a provisional MSA is constructed, refined distances are calculated from the MSA, and then the second method (FFT-NS-i) is performed. (FFT-NS-i) is a one cycle progressive method; it is faster and less accurate than the FFT-NS-2. Part tree option is available to alignments of ~50,000 sequences, and this method allows scalability.

**Kalign**

Kalign is yet another good quality multiple sequence alignment algorithm. The algorithm follows a strategy that is very similar to the standard progressive methods for sequence alignments, such as pairwise distances which are calculated firstly by using k-tuple method adopted from ClustalW. The guide tree is constructed using either UPGMA or Neighbor-Joining method, and progressive alignment is completed by following the guide trees. In contrast to the existing methods, what makes this algorithm different is the use of Wu-Manber approximate string-matching algorithm. This method is used in the distance calculation and in the dynamic programming used to align the profiles. This method allows string matching with mismatches. Also, the distances between two strings are measured using Levenshtein edit distance.

**MUSCLE**

MUSCLE stands for multiple sequence comparison by log expectation. MUSCLE uses two distance measures, kmer distance for unaligned pairs of sequences and the Kimura distance method for aligned pairs of sequences. Guide trees are produced using UPGMA method. A progressive alignment is then constructed following the order of the guide tree. This process produces an initial multiple sequence alignment. The program carries out stage two which is completed in order to improve the progressive alignment. The initial guide tree is re-estimated using Kimura distance method, and this method is known to be much more accurate than kmer, and however it requires an alignment. Once the distances are computed, the UPGMA method reclusters the sequences producing second guide tree. A progressive alignment is calculated following second tree, producing second multiple sequence alignment. A new multiple sequence alignment is produced using both the first multiple sequence alignment and the second one. New multiple sequence alignment is produced by realigning the two profiles. If the SP score is improved on the second MSA, then the new alignment is kept and the old is discarded; otherwise, it is deleted and the first alignment is used.

**Reason of down fall**

* Heuristics based
* Computational complexity
* No systematic Benchmarks
* Competitor for Clustal Omega

**FASTA for cloud**

Hadoop / Map Reduce framework and MPP DB

* Time   and number of sequences
* Number of nodes & no of alignments
* Theory of parallelization

**Hadoop & Map Reduce**

* + Small sub-problems
  + Distributed file system that stores data on nodes
  + Distributed processing of large datasets across multiple computer nodes
  + Fault tolerant parallelized analysis