**Sequence Alignment**

* Method to compare 2 or more sequences (DNA / protein) to identify characters that are identical or similar in the sequences.
* **Why use Sequence comparison ?**

**Biological basis –** many genes + proteins 🡪 members of families which have a similar biological function / share common evolutionary origin.

1. To define evolutionary relationships
2. To identify conserved patterns
3. To find similar domains that imply similar functions – when dealing with a sequence of unknown function

**Multiple Sequence Alignment (MSA)**

* Method to align multiple related sequences (identified through – database similarity searching) to achieve optimal matching of the sequences – according to a particular scoring function – to achieve maximum SP (Sum of Pairs) score.
* **Arrangement of sequences such that –** evolutionary equivalent positions across all sequences are matched.
* **Advantages –**

1. Reveals more biological information than many PSA.
2. Allows identification of conserved sequence patterns and motifs in the whole sequence family.
3. Identification of many conserved and functionally critical amino acid residues in a protein multiple alignment.
4. Phylogenetic analysis of sequence families
5. Prediction of protein secondary and tertiary structures.
6. Designing degenerate PCR (polymerase chain reaction) primers based on multiple related sequences.

* **Disadvantages –**

1. As number of sequences increases, amount of computing time + memory it requires increases exponentially 🡪 Full dynamic programming cannot be applied for datasets of more than 10 sequences.

* **SCORING FUNCTION**
* Based on the concept of – **Sum of Pairs (SP) =** sum of the scores of all possible pairs of sequences in a multiple alignment based on a particular scoring matrix.
* Each column scored by – summing the scores of all possible pairwise matches + mismatches + gap costs
* Score of entire alignment = Sum of all of the column scores
* **2 APPROACHES USED IN MSA –**

1. Exhaustive algorithms
2. Heuristic algorithms

**1. Exhaustive Algorithms**

* Involves examining all possible aligned positions simultaneously
* Requires **– Multidimensional search matrix –** to use dynamic programming for MSA. Eg – for 3 sequences – (3 – dimensional) matrix required
* **Back – tracking –** applied through the multidimensional matrix – to find the highest scored path that represents optimal alignment
* **Limitation –** full dynamic programming limited to small datasets of <10 short sequences

**Reason – ↑** No. of sequences **🡪** ↑ Amount of computational time + memory space required \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_exponentially

↓

Hence the need for “brute force approach” called DCA

↓

* **DCA (Divide – and – Conquer Alignment) –**
* Web – based program
* **Semi – exhaustive –** certain steps of computation reduced to heuristics
* **Mechanism of DCA –**

**Breaks each of the sequences into 2 smaller sections ;**

Further divisions carried out 🡪 if sections are not short enough

Breaking points determined based on – **regional similarity** of the sequences

↓

Lengths of sequences 🡪 reach a predefined threshold 🡪 dynamic programming applied for aligning each set of subsequences

↓

Resulting short alignments 🡪 joined together head to tail 🡪 to yield multiple alignment of the entire length of all sequences

* fastDCA (even more heuristic compared to DCA – performs global alignment – requires input sequences to be of similar lengths and domain structures

**2. Heuristic Algorithms**

3 categories –

1. Progressive Alignment type
2. Iterative Alignment type
3. Block – based Alignment type
4. **Progressive Alignment Type**

* **Depends on –** stepwise assembly of multiple alignment
* **Nature –** Heuristic
* **Steps –**

1. Conducts pairwise alignments for each possible pair of sequences – using the **Needleman –Wunsch global alignment method**
2. Records these similarity scores from the pairwise comparisons.

Scores can be – **%identity or %similarity** – based on particular substitution matrix.

Both scores correlate with – evolutionary distances between sequences

1. Scores – converted – to 🡪 evolutionary distances – to generate a distance matrix for all the sequences involved
2. Phylogenetic analysis performed – based on the distance matrix 🡪 to group sequences based on pairwise distance scores
3. Phylogenetic tree generated – using Neighbourhood Joining (NJ) Method 🡪 tree reflects – evolutionary proximity among all sequences.

**NOTE –** generates approximate tree ; lacks rigor of formally constructed phylogenetic tree

Used as **guide tree –** for directing realignment of the sequences.

1. According to guide tree –
2. **2 most closely related sequences –** first re-aligned using the Needleman–Wunsch algorithm.
3. **Aligning additional sequences** 🡪 2 already aligned sequences converted – to 🡪 consensus sequence (with gap positions fixed) 🡪 consensus then treated as a – single sequence in the subsequent step.
4. Using dynamic programming – next closest sequence based on the guide tree is aligned with the consensus sequence
5. More distant sequences / sequence profiles – subsequently added – according to their relative positions on the guide tree.
6. After realignment 🡪 process repeated from step 6(b) until all sequences 🡪 aligned

* **CLUSTAL –**
* Progressive multiple alignment program
* **Available either as a –** stand – alone / online program
* Stand – alone program – runs on UNIX and Macintosh – has 2 variants –

1. **ClustalW –** provides a simple text-based interface
2. **ClustalX –** provides a more user – friendly graphical interface

* **Features of Clustal –**

1. **Flexibility of using Substitution Matrices –**

* Does not rely on a single substitution matrix
* Applies different scoring matrices when aligning sequences – depending on degrees of similarity
* **Choice of a matrix** – depends on the evolutionary distances measured from the guide tree

1. **BLOSUM62 / PAM120** – for aligning closely related sequences (in the initial steps)
2. **BLOSUM45 / PAM250 –** for aligning more divergent sequences (later steps)
3. **Adjustable gap penalties –**

* Allow more insertions and deletions in regions that are outside the conserved domains – but fewer in conserved regions
* Penalties of gaps near Hydrophobic residues >>> Penalties of gaps near Hydrophilic residues
* Penalties of gaps too close to each other >>> Penalties of gaps occurring in isolated loci

1. **Weighing scheme –**

* Downweighing redundant and closely related groups of sequences in the alignment by a certain factor
* **Advantage –** prevents similar sequences from dominating the alignment
* **Weight factor** for each sequence – determined by its branch length on the guide tree
* Branch lengths – normalized by how many times sequences share a basal branch from the root of the tree
* **Obtained value** for each sequence is subsequently – used to multiply the raw alignment scores of residues from that sequence
* **Goal –**

1. Toincrease the reliability of aligning divergent sequences (sequences with less than 25% identity)
2. to decrease the matching scores of frequent characters in a multiple alignment

+ increasing the ones of infrequent characters

* **DRAWBACKS OF PROGRESSIVE ALIGNMENT METHOD**

1. Global alignment – based method 🡪 **Not suitable for comparing sequences of different lengths**
2. **Uses affine gap penalties** 🡪 long gaps not allowed 🡪 **limits the accuracy** of the method.
3. **“Greedy” nature –** depends on initial pairwise alignment + gaps fixed after early steps of alignment 🡪 errors made in these steps cannot be corrected 🡪 final alignment = far from optimal

* **T – COFFEE**
* Tree-based Consistency Objective Function for alignment Evaluation
* provides a graphical output of the alignment results, with coloured boxes to display degree of agreement in the alignment library for various sequence regions
* Performs both global and local pairwise alignment for all possible pairs involved –

1. Global pairwise alignment – using the Clustal program.
2. Local pairwise alignment – using Lalign program

* **LIBRARY EXTENSION PROCESS of T – Coffee –**
  1. Collection of local and global sequence alignments – pooled to form a library
  2. Evaluation of consistency of the alignments 🡪 **Consistency score** calculated for every pair of residues in a pair of sequences – for both global and local alignments.
  3. Each pairwise alignment – further aligned with a possible third sequence
  4. The result is used to refine the original pairwise alignment

↓

* 1. Based on the refined pairwise alignments – Distance matrix built to derive a guide tree – used to direct multiple alignment using the progressive approach.

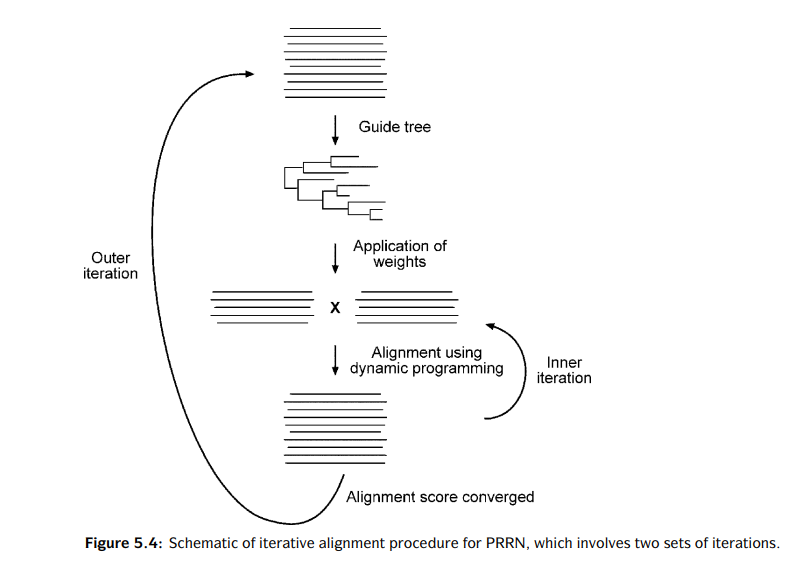
**Advantages of T - COFFEE –**

1. Avoids the problem of getting stuck in the suboptimal alignment regions
2. Minimizes errors in the early stages of alignment assembly
3. Outperforms Clustal when aligning moderately divergent sequences.

**Disadvantage of T - COFFEE –**

1. Slower than Clustal
2. Extra computing time necessary for the calculation of consistency scores
3. **Iterative Alignment Type**

* **Principle –** Optimal solution – found by repeatedly modifying existing suboptimal solutions
* **Process –** Producing a low – quality alignment and gradually improves it by iterative realignment through well – defined procedures until no more improvements in the alignment scores can be achieved
* **Advantage –** removes the “greedy” problem of Progressive strategy 🡪 since the order of the sequences used for alignment is different in each iteration
* **Disadvantage –** Heuristic nature – does not have guarantees for finding the optimal alignment

****

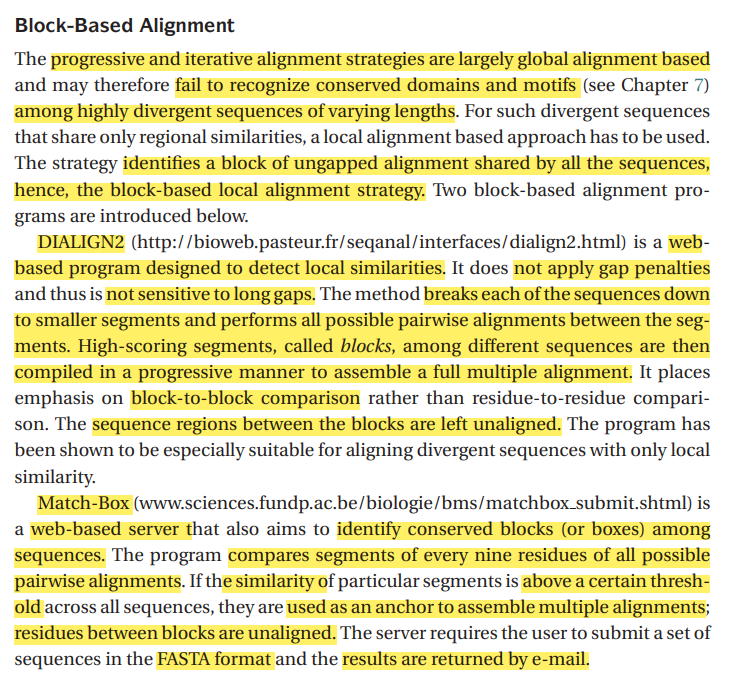
* **PRRN –**
* A web – based program
* Uses a double nested iterative strategy for multiple alignment
* Performs multiple alignment through two sets of iterations –

1. **Outer iteration –** Initial random alignment generated 🡪 used to derive a UPGMA tree 🡪 Weights applied to optimize the alignment
2. **Inner iteration –** sequences randomly divided into two groups 🡪 Randomized alignment used for each group in the initial cycle 🡪 alignment positions in each group are fixed 🡪 2 groups – each treated as a single sequence – aligned to each other using global dynamic programming

* Process repeated through many cycles until no increase in total SP score i.e., no further improvement in the overall alignment scores

1. **Block – based Alignment Type**

REFER PAVESNER



REMAINING – PRACTICAL ISSUES

* **Editing**
* Automated alignment often contains misaligned regions
* Hence the need for the user to check the alignment carefully for biological relevance + edit the alignment if necessary
* Involves introducing / removing gaps – to maximize biologically meaningful matches
* Portions that are ambiguously aligned + deemed to be incorrect 🡪 have to deleted
* Empirical evidence or mere experience 🡪 required for manual editing
* Word processor 🡪 to edit the text – based alignment
* Other dedicated softwares –

1. **BioEdit –** multifunctional sequence alignment editor for Windows 🡪 provides coloring scheme for nucleotide / amino acid residues 🡪 BLAST searches + plasmid drawing + restriction mapping
2. **Rascal –** web – based program – automatically refines a MSA

**PHYLOGENETICS**

* **Evolution**
* Development of a biological form from other preexisting forms or its origin to the current existing form through natural selections and modifications.
* **Driving force –** natural selection in which “unfit” forms are eliminated through changes of environmental conditions or sexual selection so that only the fittest are selected.
* Underlying mechanism of evolution – genetic mutations that occur spontaneously 🡪 provide the biological diversity
* **Phylogenetics**
* Phylogenetics = study of the evolutionary history of living organisms using tree – like diagrams to represent pedigrees of these organisms.
* Phylogeny = The tree branching patterns representing the evolutionary divergence
* **2 ways to study Phylogeny –**

**1. Fossil Records –**

* Contain morphological information about ancestors of current species and the timeline of divergence
* **Limitations –**

1. Available only for certain species
2. Existing fossil data can be fragmentary and their collection is limited by factors such as – abundance, habitat, geographic range, etc.
3. Descriptions of morphological traits – often ambiguous due to multiple genetic factors. Hence phylogenetic relationships become biased.
4. For microorganisms – fossils are non – existent
5. **Molecular data –**

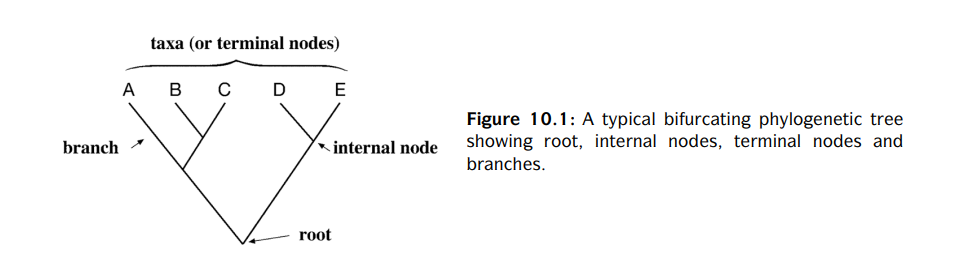
* Found in the form of DNA or protein sequences
* As organisms evolve 🡪 the genetic materials accumulate mutations over time 🡪 causing phenotypic changes – serve as **molecular fossils.**
* Comparative analysis of the molecular fossils from a number of related organisms 🡪 evolutionary history of the genes + organisms revealed
* **Advantages –**

1. Molecular data – more numerous than fossil records
2. Easier to obtain
3. No sampling bias involved – which helps to mend the gaps in real fossil records
4. Construction of more clear – cut and robust phylogenetic trees

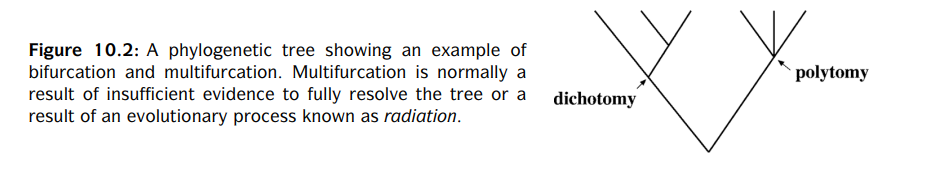
* **Molecular Phylogenetics**
* Study of evolutionary relationships of genes and other biological macromolecules by analyzing mutations at various positions in their sequences and developing hypotheses about the evolutionary relatedness of the biomolecules.
* **Major Assumptions –**

1. Molecular sequences used in phylogenetic construction – homologous 🡪 share a common origin and subsequently diverged through time.
2. Phylogenetic divergence 🡪 bifurcating – a parent branch splits into two daughter branches at any given point.
3. Each position in a sequence – evolved independently. Variability among sequences – sufficiently informative for constructing unambiguous phylogenetic trees

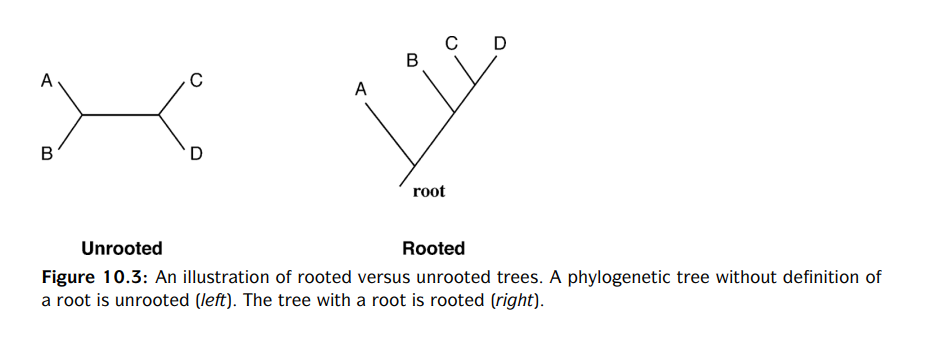
* **Terminology**



|  |  |
| --- | --- |
| **TERMS** | **DEFINITIONS** |
| Branches | Lines in the tree |
| Taxa (singular – taxon) / Operational Taxonomic Units | Present – day species or sequences at the tips of the branches |
| Node | Connecting point where two adjacent branches join 🡪 represents an inferred ancestor of extant taxa. |
| Root node | Bifurcating point at the very bottom of the tree 🡪 represents the common ancestor of all members of the tree. |
| Clade / Monophylectic group | A group of taxa descended from a single common ancestor. |
| Sister taxa | 2 taxa sharing a unique common ancestor not shared by any other taxa in a monophyletic group. |
| Lineage | Branch path depicting an ancestor – descendant relationship on a tree |
| Paraphylectic group | A group of taxa that share more than one closest common ancestors |
| Tree Topology | Branching pattern in a tree |



|  |  |
| --- | --- |
| **TERMS** | **DEFINITIONS** |
| Dichotomy | When all branches bifurcate on a phylogenetic tree 🡪 each ancestor divides and gives rise to two descendants. |
| Multifurcating node | A branch point on a phylogenetic tree having more than two descendants. |
| Polytomy + Radiation | **Polytomy** – Phylogeny with multifurcating branches  ↓  A result of –   1. **Radiation** – an ancestral taxon giving rise to more than two immediate descendants simultaneously during evolution 2. an unresolved phylogeny in which the exact order of bifurcations cannot be determined precisely |



|  |  |
| --- | --- |
| **TERMS** | **DEFINITIONS** |
| Unrooted phylogenetic tree | - Does not assume knowledge of a common ancestor  - Only positions the taxa to show their relative relationships  - No direction of an evolutionary path |
| Rooted tree | All the sequences under study have a common ancestor or root node from which a unique evolutionary path leads to all other nodes. |

* **Conversion of Unrooted Tree – to 🡪 Rooted tree**

Defining the root – 2 ways –

1. **Outgroups –**

* Sequence that is homologous to the sequences under consideration – but separated from those sequences at an early evolutionary time
* Determined from – Independent sources of information
* Required to be distinct from the ingroup sequences – but not too distant from the ingroup

↓

Using too divergent sequences as an outgroup leads to errors in tree construction.

* **Eg –** a bird sequence – used as a root for the phylogenetic analysis of mammals

1. **Midpoint Rooting Approach –**

Midpoint of the two most divergent groups 🡪 assigned as the root 🡪 determined by overall branch lengths

**Assumptions –**

1. Divergence from root to tips for both branches is equal
2. Follows the **“Molecular Clock” hypothesis.**

* Molecular sequences evolve at constant rates.
* Amount of accumulated mutations α evolutionary time