[1 Chapter 2 - Algorithms 5](#_Toc105495581)

[1.1 Name 8 algorithm strategies 5](#_Toc105495582)

[1.2 Describe the branch and bound algorithm 5](#_Toc105495583)

[1.3 Describe the steps of the divide-and-conquer algorithm 5](#_Toc105495584)

[1.4 Describe the principle of dynamic programming 5](#_Toc105495585)

[1.5 Describe the pre-order tree-traversal. 5](#_Toc105495586)

[1.6 Describe the post-order tree-traversal 5](#_Toc105495587)

[1.7 Describe in-order tree-traversal 5](#_Toc105495588)

[1.8 Describe breadth-first traversal of a tree 6](#_Toc105495589)

[2 Chapter 3 – Pairwise Alignment 7](#_Toc105495590)

[2.1 Define string and alphabet 7](#_Toc105495591)

[2.2 Define concatenation 7](#_Toc105495592)

[2.3 Define substring 7](#_Toc105495593)

[2.4 Reverse complement 7](#_Toc105495594)

[2.5 What are the building blocks of a fasta-file? 7](#_Toc105495595)

[2.6 How is a dot matrix composed? 7](#_Toc105495596)

[2.7 How are common substrings (reversed, displaced, repeated) visible in the dot matrix? 7](#_Toc105495597)

[2.8 What parameters are used to reduce noise in a dot matrix? 7](#_Toc105495598)

[2.9 What runtime does the dot matrix have? 7](#_Toc105495599)

[2.10 Define pairwise global alignment. 7](#_Toc105495600)

[2.11 What two characters CAN’T be aligned in a pairwise alignment? 8](#_Toc105495601)

[2.12 What assumption is made with scoring such that the scoring scheme can be additive? 8](#_Toc105495602)

[2.13 What is the formula of the Minkowski metric? What are its first two instances called? 8](#_Toc105495603)

[2.14 What is the formula of the Hamming distance? What is an intuitive explanation for it? 8](#_Toc105495604)

[2.15 What is the definition of the Levenshtein (or edit) distance? 8](#_Toc105495605)

[2.16 What is the definition of an edict transcript? 8](#_Toc105495606)

[2.17 What are the three essential components of dynamic programming? 8](#_Toc105495607)

[2.18 What is a prefix? 8](#_Toc105495608)

[2.19 What is the distance matrix defined as? In which direction goes an insertion, in which a deletion? 8](#_Toc105495609)

[2.20 What is the formula for the score S of an alignment A? 9](#_Toc105495610)

[2.21 What is the definition of a substitution matrix? 9](#_Toc105495611)

[2.22 What is the formula for the conditional probability of A given B? 9](#_Toc105495612)

[2.23 What is the formula for the joint probability? 9](#_Toc105495613)

[2.24 What are the entries of a substitution matrix S representing? How are they computed? 9](#_Toc105495614)

[2.25 Kochrezept: How can you compute your own substitution matrix? 9](#_Toc105495615)

[2.26 Name two commonly used substitution matrices. 9](#_Toc105495616)

[2.27 What does BLOSUM stand for? What is the requirement if there is an integer behind the word? 9](#_Toc105495617)

[2.28 How are the entries in the BLOSUM matrix computed? 9](#_Toc105495618)

[2.29 What are some possible ways to classify amino acids? 10](#_Toc105495619)

[2.30 What is the formula for a function that assigns a constant gap penalty of gap length g? 10](#_Toc105495620)

[2.31 What is the formula for a function that assigns a gap penalty that increases with gap extension? Is e usually smaller or bigger than d? 10](#_Toc105495621)

[2.32 Define subsequence of X and Y. 10](#_Toc105495622)

[2.33 What is the formula for the number of possible alignments between X and Y? 10](#_Toc105495623)

[2.34 What is another way to write the formula for the number of possible alignments between X and Y? 10](#_Toc105495624)

[2.35 What does Stirling’s formula look like? For which n and m does it hold? 10](#_Toc105495625)

[2.36 Define optimal global alignment. 10](#_Toc105495626)

[2.37 What are the three cases for the Fitch algorithm (in words and in formula)? What other conditions are defined for its initialization? 10](#_Toc105495627)

[2.38 Reproduce the pseudocode for the Needleman-Wunsch algorithm. 11](#_Toc105495628)

[2.39 What is the complexity of the Needleman-Wunsch algorithm, and why? 11](#_Toc105495629)

[2.40 Where do we apply pairwise local alignment? 11](#_Toc105495630)

[2.41 Define local alignment. 11](#_Toc105495631)

[2.42 What is the difference between the Smith-Waterman algorithm to the Needleman-Wunsch-algorithm? What is the formula for F(i,j) with the Smith-Waterman algorithm? 11](#_Toc105495632)

[2.43 Reproduce the pseudocode for the Smith-Waterman algorithm. 12](#_Toc105495633)

[2.44 What is the complexity of the Smith-Waterman algorithm? 12](#_Toc105495634)

[2.45 Under what condition does the Smith-Waterman algorithm work? Include the formula. 12](#_Toc105495635)

[3 Chapter 4 – BLAST 13](#_Toc105495636)

[3.1 What is the formula for sensitivity? 13](#_Toc105495637)

[3.2 What is the formula for specificity? 13](#_Toc105495638)

[3.3 What does BLAST stand for? 13](#_Toc105495639)

[3.4 What are the two components of BLAST? 13](#_Toc105495640)

[3.5 What is the algorithm strategy of BLAST? 13](#_Toc105495641)

[3.6 Describe the seed lemma by Owolabi and McGregor. Does this describe the worst or the best case? 13](#_Toc105495642)

[3.7 Describe the goal of BLAST. 13](#_Toc105495643)

[3.8 Describe the alignment score using the BLAST terms “query”, “database”, “segment” and “scoring matrix”. 13](#_Toc105495644)

[3.9 What three parameters are used in BLAST? 13](#_Toc105495645)

[3.10 Describe the preprocessing for BLAST. 14](#_Toc105495646)

[3.11 What are the steps of the BLAST search algorithm? 14](#_Toc105495647)

[3.12 What are typical default parameters for word size W for proteins and DNA sequences? 14](#_Toc105495648)

[3.13 Describe the keyword tree used in BLAST. 14](#_Toc105495649)

[3.14 Name and describe the different variants of the BLAST program. 14](#_Toc105495650)

[3.15 What is the underlying distribution under BLASTs statistical considerations? What is the formula? 14](#_Toc105495651)

[3.16 What is the formula of the E-value? What is its description? 14](#_Toc105495652)

[3.17 What is the formula of the p-value in the BLAST context? 14](#_Toc105495653)

[3.18 Describe the BLAST bit score. What is its formula? 15](#_Toc105495654)

[3.19 How are E-value and bit score S’ related (formula)? 15](#_Toc105495655)

[4 Chapter 5 – Multiple Alignments 16](#_Toc105495656)

[4.1 What constitutes a multiple sequence alignment? What are its goals? 16](#_Toc105495657)

[4.2 What is the advantage of MSA in comparison to pairwise alignments? 16](#_Toc105495658)

[4.3 How is an MSA obtained? What are the requirement concerning the resulting and original sequences in it? 16](#_Toc105495659)

[4.4 How is an MSA scored? 16](#_Toc105495660)

[4.5 Given a column of length r, how many different columns can be obtained by inserting a gap symbol at certain places instead of the original character? Give the formula. 16](#_Toc105495661)

[4.6 Why do we need the sum of pairs score? What are the ‘reasonable asumptions’ so that we can compute the score in this way? 16](#_Toc105495662)

[4.7 For the sum of pairs score, hat are the output values of the scoring function for a given combination of residues and gaps? 17](#_Toc105495663)

[4.8 Reproduce the formal definition of the sum of pairs score. 17](#_Toc105495664)

[4.9 How would you compute the Dynamic Programming/Fitch algorithm for a three-sequence MSA? How many cases of the scoring function for a certain column composition would there be? 17](#_Toc105495665)

[4.10 What is the space complexity of the generalized Dynamic Programming algorithm? What is the time complexity? Is it NP-complete? 17](#_Toc105495666)

[4.11 What are the steps of progressive alignment? 17](#_Toc105495667)

[4.12 Where do different progressive alignment implementations differ? 17](#_Toc105495668)

[4.13 Reproduce the pseudocode for progressive alignments 18](#_Toc105495669)

[4.14 Describe the process of pair-guided alignment. 18](#_Toc105495670)

[4.15 What is the main difference between profile alignment and pair-guided alignment? 18](#_Toc105495671)

[4.16 Provide the formula for the sum of pairs score for an MSA A\* computed from two profiles A1\* and A2\*. Explain each resulting term. 18](#_Toc105495672)

[4.17 Define guide trees. 18](#_Toc105495673)

[4.18 Describe the steps of the Feng-Doolittle algorithm. What is its purpose? 18](#_Toc105495674)

[4.19 What is the formula for the distance score as defined by Feng-Doolittle for two sequences X and Y? 19](#_Toc105495675)

[4.20 In the Feng-Doolittle formula, how is Sobs defined? 19](#_Toc105495676)

[4.21 In the Feng-Doolittle formula, how is Sid defined? What is the formula? 19](#_Toc105495677)

[4.22 In the Feng-Doolittle formula, how is Srand defined? What is the formula? 19](#_Toc105495678)

[4.23 What are the steps of the ClustalW algorithm? 19](#_Toc105495679)

[4.24 What distance functions does ClustalW use? Provide formulas if applicable. 19](#_Toc105495680)

[4.25 What is one distinctive feature of ClustalW? 20](#_Toc105495681)

[4.26 Which of the steps in ClustalW is the most time-consuming? 20](#_Toc105495682)

[4.27 Name other programs for MSAs. 20](#_Toc105495683)

# Chapter 2 - Algorithms

## Name 8 algorithm strategies

### Recursive

### Dynamic programming

### Backtracking

### Heuristic

### Branch and bound

### Divide and conquer

### Greedy

### Brute force

## Describe the branch and bound algorithm

* + 1. Find a solution of a combinatorial problem with each solution associated with a cost, such that the whole solution cannot be smaller than any partial solution. Remember the lowest-cost solution and keep that as a lower bound for other solutions. Problems: Search can still become exponential

## Describe the steps of the divide-and-conquer algorithm

* + 1. 1. Divide a larger problem into smaller subproblems of the same type
    2. 2. Solve each subproblem recursively
    3. 3. Combine smaller solutions into a solution for the whole problem
    4. Divide-and-conquer is a top-down approach.

## Describe the principle of dynamic programming

* + 1. Divide each problem into a succession of stages, in which the first stage only consists of trivial solutions and each partial solution in a later stage can be calculated by recurring on only a fixed number of partial solutions in an earlier stage. DP is bottom-up.
    2. It consists of four steps:
    3. 1. Characterisation of the structure of the optimal solution
    4. 2. Recursive definition of the value of an optimal solution
    5. 3. Computation of the optimum using recursion
    6. 4. Construction of an optimal solution through the computed optimal value

## Describe the pre-order tree-traversal.

* + 1. Pre-order tree-traversal is top-down. Examine the root of a tree T. Then traverse its left sub-tree in pre-order, then traverse its right sub-tree in pre-order.

## Describe the post-order tree-traversal

* + 1. Post-order tree-traversal is bottom-up. First traverse the left subtree of the root in post-order then traverse the right subtree of the root in post-order. Finally examine the root.

## Describe in-order tree-traversal

* + 1. First examine thee left subtree of a tree in in-order, then examine the root, then examine the right subtree of a tree in in-order.

## Describe breadth-first traversal of a tree

### Put the root in a queue. Repeat the following until the queue is empty: Pop a node off the beginning of the queue and examine it. Then, if the node has any children, push them onto the end of the queue. First, always examine the root and then the left and right child-node of the root. Then examine the left-and right child node of the left node, then of the right. Stop if a node has no child-nodes.

# Chapter 3 – Pairwise Alignment

## Define string and alphabet

* + 1. An alphabet is a finite, non-empty set of symbols and letters. A string is obtained by writing a sequence of elements s1s2s3… of symbols of the alphabet. The length of a string is the number of symbols in it denoted as |S|.

## Define concatenation

* + 1. Let S and T be two strings on an alphabet. We will use ST to denote the concatenation of S and T.

## Define substring

* + 1. Let S and T be two strings on an alphabet. We call S a substring of T if there exists two strings U, V from the alphabet such that T = USV. We denote a substring of S as S[i,j].

## Reverse complement

Let S be a sequence of length n on the DNA alphabet {A,G,C,T}. The sequence S bar is computed with s\_bar\_i, which is:

A if s\_n-i+1 = T

G if s\_n-i+1 = C

T if s\_n-i+1 = A

C if s\_n-i+1 = G

## What are the building blocks of a fasta-file?

Header line starts with “>” and contains identifier for sequence. Successive lines contain the sequence.

## How is a dot matrix composed?

### Write Sequence S in row zero. Write sequence T in column zero. Place a dot at each cell for which the corresponding symbols match. M[i,j] is 1 where s\_i = t\_i, and 0 everywhere else

## How are common substrings (reversed, displaced, repeated) visible in the dot matrix?

### Common: Diagonal line from top left to bottom right

### Reversed common: Diagonal line from bottom let to top right

### Displaced common: Diagonal line NOT on the main diagonal

### Repeated common: diagonal lines at the same horizontal/vertical position as other diagonal lines

## What parameters are used to reduce noise in a dot matrix?

Window size w (how many characters are looked at?) and stringency s (how many characters need to be equal in the next w position?).

## What runtime does the dot matrix have?

Quadratic.

## Define pairwise global alignment.

A pairwise alignment is obtained by inserting dashes into two strings X and Y such that: (1) The resulting strings X’ and Y’ are of the same length (2) They can be written on top of each other so each character in one string is opposite a unique character in the other string and (3) removing all gaps results in the original strings (X’\{-}=X).

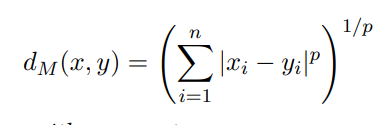
## What two characters CAN’T be aligned in a pairwise alignment?

Two dashes.

## What assumption is made with scoring such that the scoring scheme can be additive?

Evolution of all sites within each sequence is independent of each other.

## What is the formula of the Minkowski metric? What are its first two instances called?



Manhattan distance, Euclidean distance

## What is the formula of the Hamming distance? What is an intuitive explanation for it?

### 

The Hamming distance is the number of characters between two strings that are not equal at the same positions.

## What is the definition of the Levenshtein (or edit) distance?

The minimum number of replacements, deletions or insertions that are needed to transform sequence X into sequence Y.



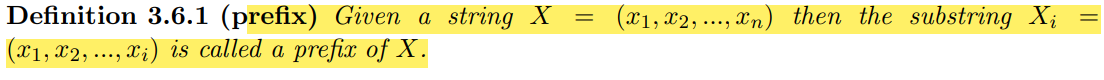
## What is the definition of an edict transcript?

Using M for match, R for replacement, D for deletion and I for insertion, the edit transcript is a string that describes the transformation of a string X into a string Y.

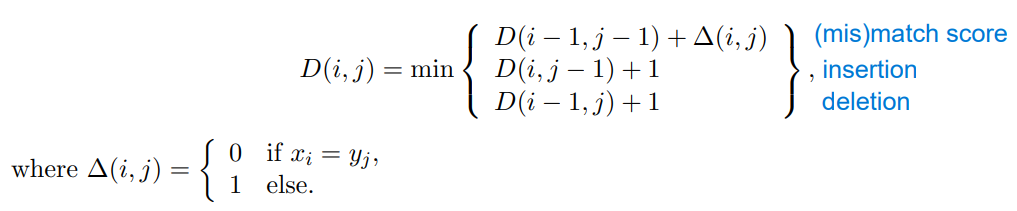
## What are the three essential components of dynamic programming?

Recurrence relation, tabular computation and traceback

## What is a prefix?

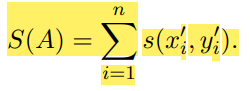


## What is the distance matrix defined as? In which direction goes an insertion, in which a deletion?



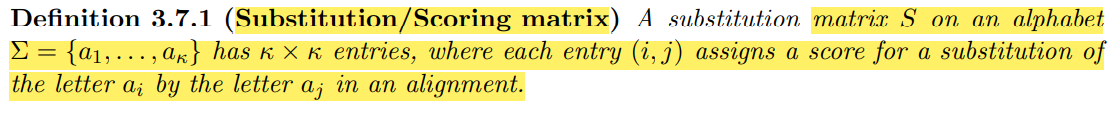
In the first column of D, we assume all deletions. In the first row of D, we assume all insertions.

## What is the formula for the score S of an alignment A?



Where X’ and Y’ are strings in which there have possibly been inserted gaps.

## What is the definition of a substitution matrix?



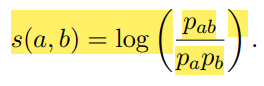
## What is the formula for the conditional probability of A given B?



## What is the formula for the joint probability?



## What are the entries of a substitution matrix S representing? How are they computed?



They represent the log-odds-ratio of the probability that the two elements are derived from a common ancestor divided by the probability that they occur randomly. Because it is the log, we can add up the elements of the scoring matrix for a score.

## Kochrezept: How can you compute your own substitution matrix?

### Compute match model: Derive the frequency of each substitution in the given alignments.

### Compute random model: Derive the frequency of each symbol

* + 1. Compute each entry s(x,y) with the formula given above (joint probability divided by single probabilities).

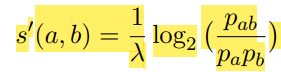
## Name two commonly used substitution matrices.

BLOSUM(x), PAM

## What does BLOSUM stand for? What is the requirement if there is an integer behind the word?

BLOSUM stands for BLOcks Substitution Matrix. The integer means that if two sequences have at least this amount of sequence identity they are placed in the same cluster. All counts are based on comparisons between pairs of sequences in different clusters. The more similar the sequences should be, the higher the integer.

## How are the entries in the BLOSUM matrix computed?



Rounded to the nearest integer value.

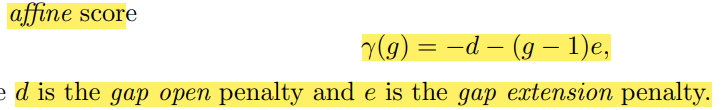
## What are some possible ways to classify amino acids?

Side chain polarity (polar vs. nonpolar), side chain charge (positive, neutral or negative), hydropathy index, size..

## What is the formula for a function that assigns a constant gap penalty of gap length g?

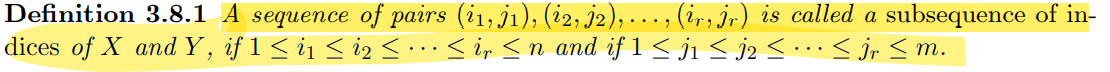


## What is the formula for a function that assigns a gap penalty that increases with gap extension? Is e usually smaller or bigger than d?



Usually, e < d, which means that less isolated gaps are produced.

## Define subsequence of X and Y.

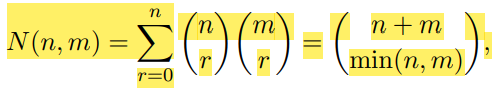


## What is the formula for the number of possible alignments between X and Y?

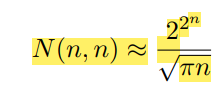


This equals the number of possible subsequences, ergo the combinatorial possibilities to choose r mismatch/match positions in each sequence.

## What is another way to write the formula for the number of possible alignments between X and Y?

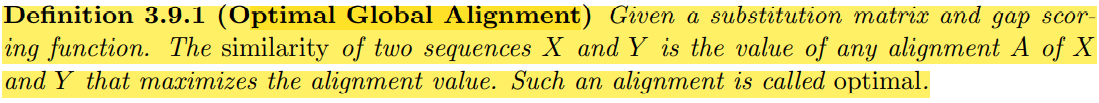


## What does Stirling’s formula look like? For which n and m does it hold?

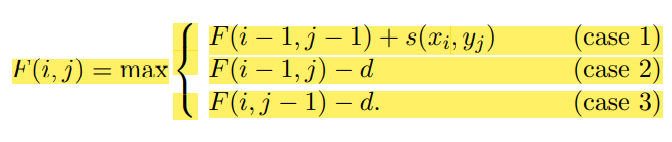


If n = m

## Define optimal global alignment.



## What are the three cases for the Fitch algorithm (in words and in formula)? What other conditions are defined for its initialization?



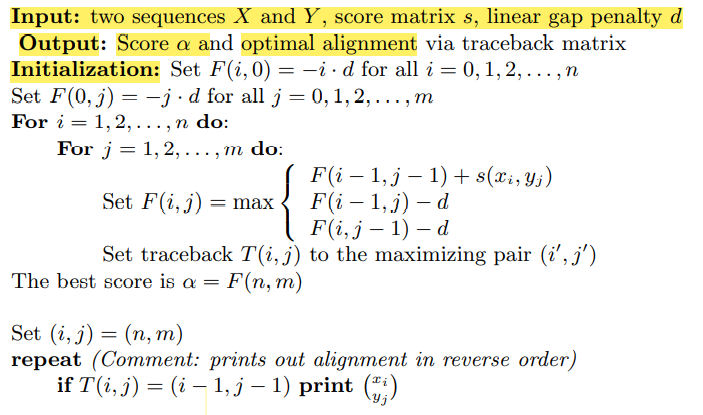
Case 1: Add score for the two aligned characters.

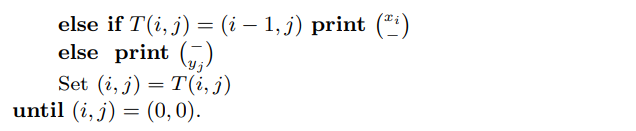
Case 2: Add score for gap score aligned with x

Case 3: Add score for gap score aligned with y

Additional initialization conditions: 1. F(0,0) = 0 2. F(i,0) > 0 for all i = 1,…,n and F(j, 0) > 0 for all j=1,…,m.

## Reproduce the pseudocode for the Needleman-Wunsch algorithm.





## What is the complexity of the Needleman-Wunsch algorithm, and why?

The algorithm takes O(nm) time and memory. This is because we need to store (n+1)x(m+1) numbers and each number takes three additions and a max to compute.

## Where do we apply pairwise local alignment?

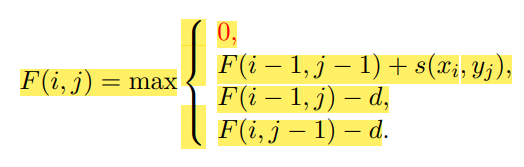
When we find out the alignment of two substrings that have a higher score than the score between the full strings.

## Define local alignment.

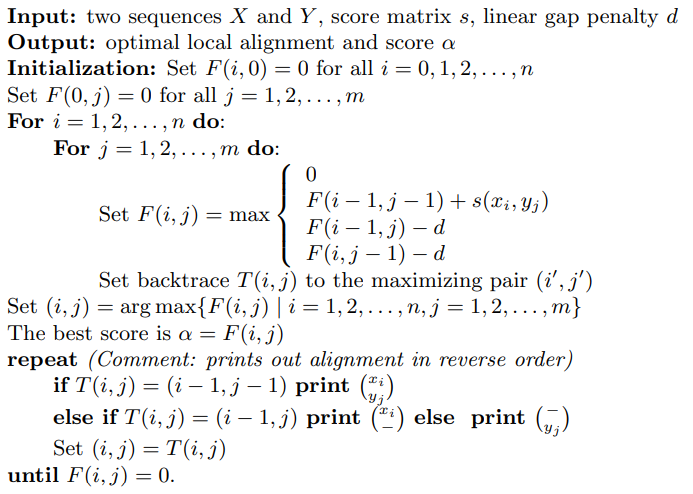
### 

## What is the difference between the Smith-Waterman algorithm to the Needleman-Wunsch-algorithm? What is the formula for F(i,j) with the Smith-Waterman algorithm?

The optimal score is never negative, thus we set the score to zero if all attainable values of NW at position (i, j) are negative. For the traceback, we start at the cell with the highest score and end at any cell with score 0.



## Reproduce the pseudocode for the Smith-Waterman algorithm.

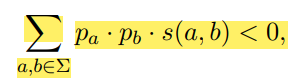


## What is the complexity of the Smith-Waterman algorithm?

O(nm) (like Needleman-Wunsch)

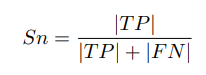
## Under what condition does the Smith-Waterman algorithm work? Include the formula.

The expected score of a random match to be negative.

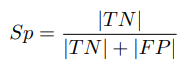


# Chapter 4 – BLAST

## What is the formula for sensitivity?



## What is the formula for specificity?



## What does BLAST stand for?

Basic Local Alignment Search Tool

## What are the two components of BLAST?

1. Search algorithm 2. Computation of statistical significance of solutions

## What is the algorithm strategy of BLAST?

Seed and extend. First find small exact matches (hits), then possibly extend to find long inexact ones (High-scoring Segment Pairs). Extending to the left or right of an HSP would lead to a lower score.

## Describe the seed lemma by Owolabi and McGregor. Does this describe the worst or the best case?

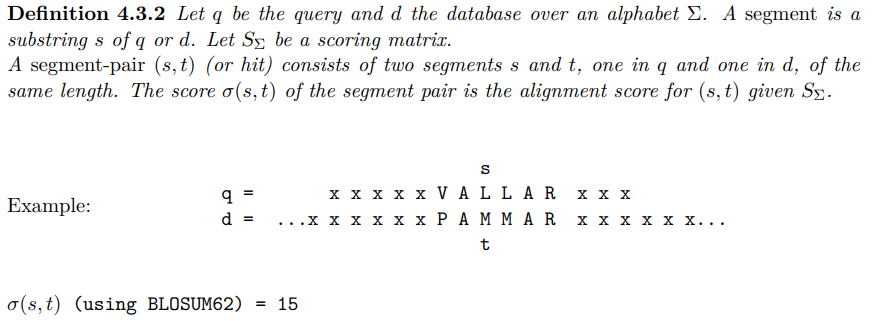


It describes the worst case.

## Describe the goal of BLAST.

Goal: Compute all High-scoring segment Pairs between two sequences with score >= S

## Describe the alignment score using the BLAST terms “query”, “database”, “segment” and “scoring matrix”.



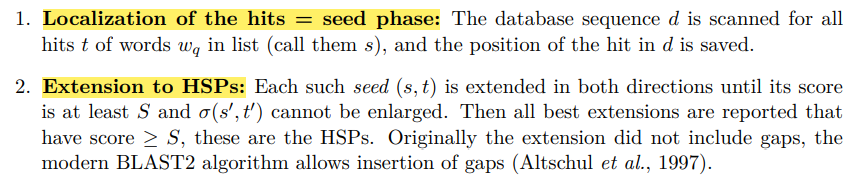
## What three parameters are used in BLAST?

Word size W, word similarity threshold T and HSP score S.

## Describe the preprocessing for BLAST.

Generate all words of length W from a query sequence q. Generate a list of all W-mers out of the alphabet that have similarity >= T to some word in the query sequence q.

## What are the steps of the BLAST search algorithm?



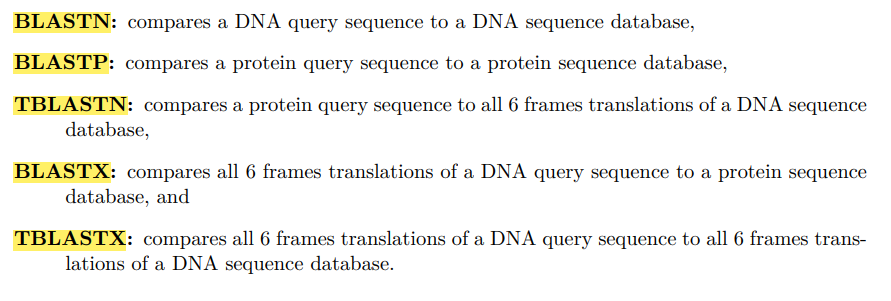
## What are typical default parameters for word size W for proteins and DNA sequences?

DNA: W= 11, Proteins: W = 3

## Describe the keyword tree used in BLAST.

A keyword tree is a finite state automaton used in pattern matching. It’s a rooted tree built from all words, edges are labelled with exactly one letter, only one edge per letter is leaving each node. Each word therefore maps to one node in the tree. We move forward through the tree if we can with a specific word, and jump back to the root if we can’t.

## Name and describe the different variants of the BLAST program.

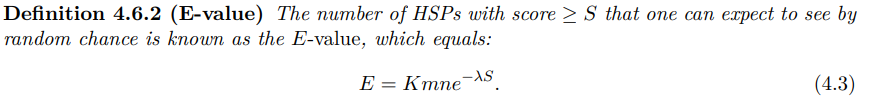


## What is the underlying distribution under BLASTs statistical considerations? What is the formula?

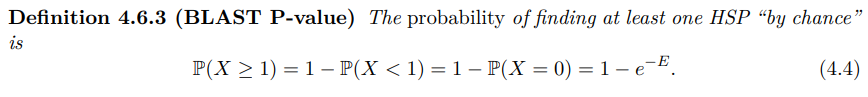
The Poisson distribution.



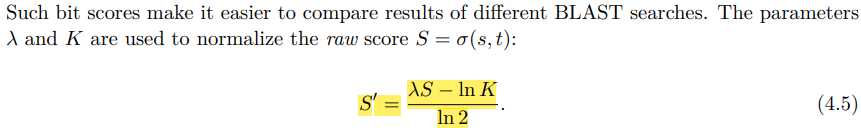
## What is the formula of the E-value? What is its description?



## What is the formula of the p-value in the BLAST context?



## Describe the BLAST bit score. What is its formula?



## How are E-value and bit score S’ related (formula)?



# Chapter 5 – Multiple Alignments

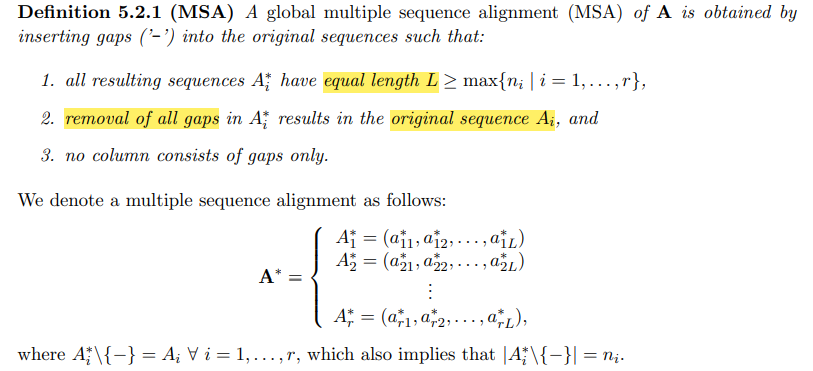
## What constitutes a multiple sequence alignment? What are its goals?

A multiple sequence alignment is an alignment of more than two sequences. The MSA arranges protein sequences into rectangular array with the goal that the residues in a given column are homologous (derived from a single position in an ancestral sequence), superposable (in a rigid local structural alignment) or play a common functional role. Detect homologous residues and place them in the same column of the MSA.

## What is the advantage of MSA in comparison to pairwise alignments?

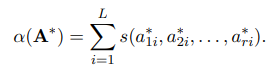
The chance off random similarities decreases with the increasing number of aligned sequences.

## How is an MSA obtained? What are the requirement concerning the resulting and original sequences in it?

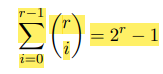


## How is an MSA scored?

Because we can assume independence of the different columns of an MSA, the score alpha can be defined as a sum off column scores. The function returns a score for every combination of r symbols (including the gap symbol).



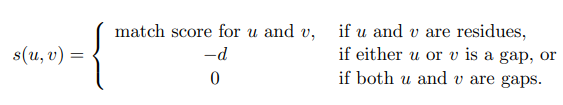
## Given a column of length r, how many different columns can be obtained by inserting a gap symbol at certain places instead of the original character? Give the formula.



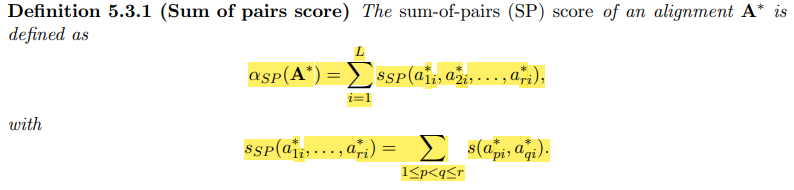
## Why do we need the sum of pairs score? What are the ‘reasonable asumptions’ so that we can compute the score in this way?

We need the sum of pairs score because constructing a substitution matrix for a whole specific column would be to laboursome. Instead, we can add up the single scores for each combination. Assumption 1: The score must be independent of order of gaps in the column. Assumption 2: Similar or identical aligned residues should get a high score, while unrelated or gaps a negative one.

## For the sum of pairs score, hat are the output values of the scoring function for a given combination of residues and gaps?



## Reproduce the formal definition of the sum of pairs score.



## How would you compute the Dynamic Programming/Fitch algorithm for a three-sequence MSA? How many cases of the scoring function for a certain column composition would there be?

You would construct a three-dimension cube, progressing in the three-dimensional direction of the optimal score. There would be seven cases for three dimensions.

## What is the space complexity of the generalized Dynamic Programming algorithm? What is the time complexity? Is it NP-complete?

Space complexity: O(nr). Run time complexity: O(r2n22r). Computing an MSA with optimal SP-score is NP-complete.

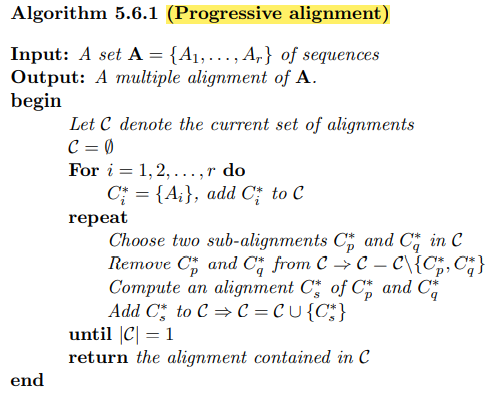
## What are the steps of progressive alignment?

1. Calculate a distance matrix, representing he distance between each pair of sequences
2. From this matrix, build a phylogenetic tree
3. Use this tree as guide to progressively align the sequences

## Where do different progressive alignment implementations differ?

1. Order of the aligned sequences
2. Whether a single multiple alignment is generated or several ones, following a tree structure
3. Which scoring function is used

## Reproduce the pseudocode for progressive alignments



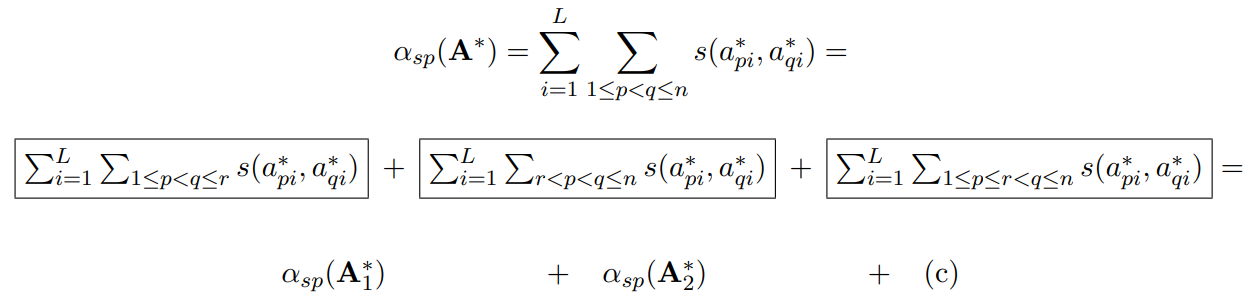
## Describe the process of pair-guided alignment.

Choose two sequences, one from each profile. Align these. By adjusting the gaps in the other sequences, the full alignment is produced by following this alignment.

## What is the main difference between profile alignment and pair-guided alignment?

In profile alignment, whole columns of gaps are inserted into either profile, without further changing the alignment of either of the two profiles.

## Provide the formula for the sum of pairs score for an MSA A\* computed from two profiles A1\* and A2\*. Explain each resulting term.



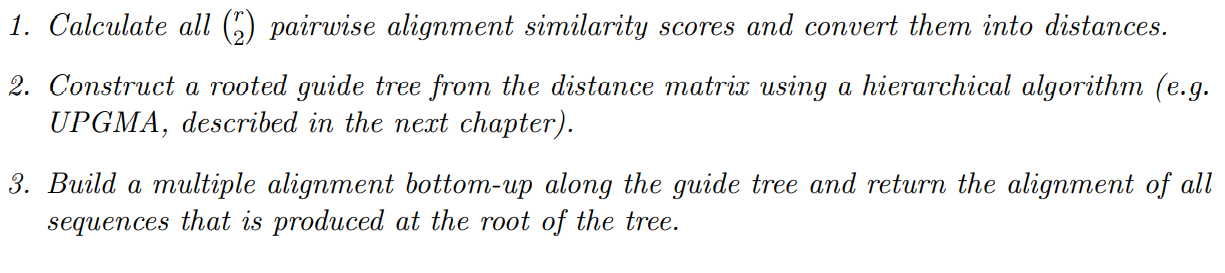
The first two scores are the alignment scores of each of the profiles. The term c) is the optimized alignment score between one sequence in the first and second profile respectively.

## Define guide trees.

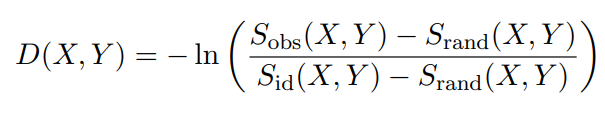
A guide tree is a phylogenetic tree whose leaves are labelled by the sequences that are to be aligned.

## Describe the steps of the Feng-Doolittle algorithm. What is its purpose?

Its purpose is building a multiple sequence alignment.



## What is the formula for the distance score as defined by Feng-Doolittle for two sequences X and Y?

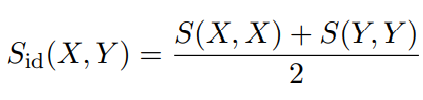


## In the Feng-Doolittle formula, how is Sobs defined?

Here, Sobs is the observed similarity score computed by pairwise alignment of X and Y.

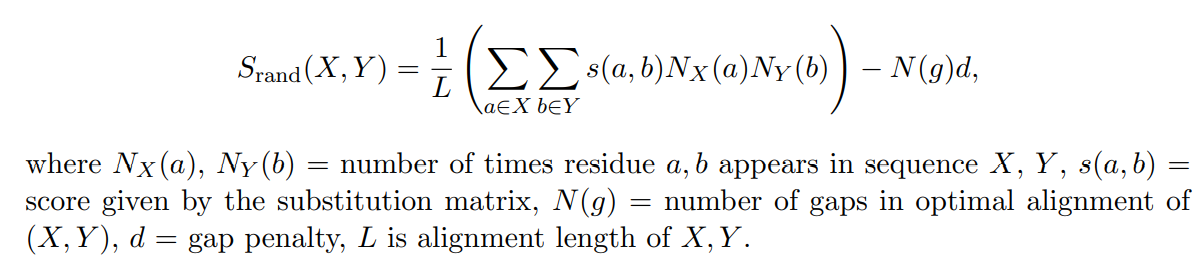
## In the Feng-Doolittle formula, how is Sid defined? What is the formula?

Sid is the average score of both aligned with each other, given by

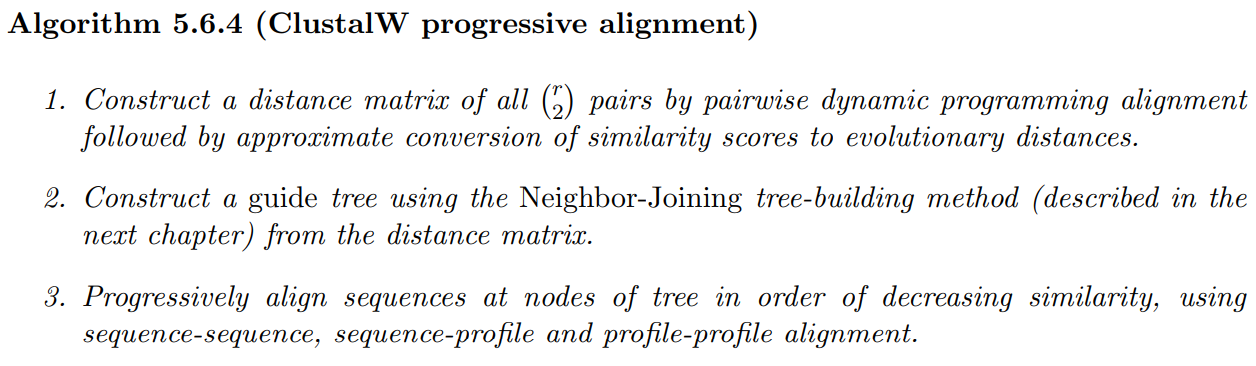


## In the Feng-Doolittle formula, how is Srand defined? What is the formula?

And Srand is the expected score of an alignment of two random sequences of the same length and composition as X and Y.

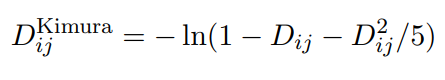


## What are the steps of the ClustalW algorithm?



## What distance functions does ClustalW use? Provide formulas if applicable.

1. Kimura distance.



Here, Dij is the number of mismatches in an optimal pairwise alignment divided by the length of the sequence.

1. K-tuple distance measure of Wilbur and Lipman for unaligned sequences. Computes the number of exactly matching k-tuples (for proteins k=2) using a hash-table approach.

## What is one distinctive feature of ClustalW?

Once a gap, always a gap.

## Which of the steps in ClustalW is the most time-consuming?

Computation of the initial pairwise alignments (> 90% of time)

## Name other programs for MSAs.

Muscle, T-Coffee, Clustal Omega, MAFFT

# Chapter 6 – Phylogeny

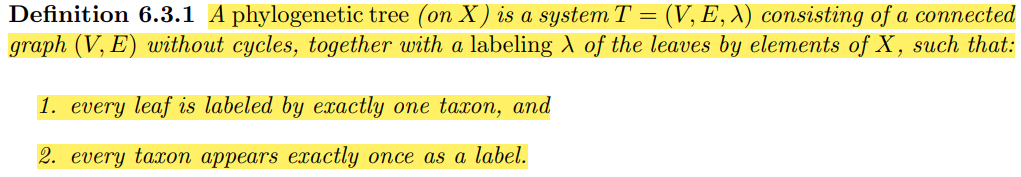
## What is the goal of phylogenetic analysis?

Describe the evolutionary relationship between a collection of extant species.

## Describe the basic structure of phylogenetic trees.

A tree consists of nodes connected by edges. Terminal nodes (leaves) represent sequences or species for which we have data. Internal nodes represent hypothetical ancestors.

## Provide the formal definition off phylogenetic trees.



Where V is the set of nodes and E is the set off edges.

## When are two species related by definition?

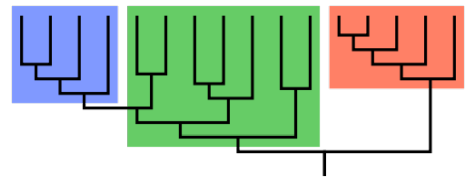
Two species are related if they share a recent common ancestor.

## Define a Cluster

A set of species.

## Define a Clade.

A cluster C is called a clade or monophyletic group if all species in C are more closely related to each other than to any species outside of C; in other words: all species in C are descended from a common ancestor, which is not an ancestor of any other species under consideration. In the example below, the green cluster is not a clade.



## Name different ways to draw a dendrogram.

Rooted, unrooted, slanted or traditional, circular.

## How many nodes and edges does a phylogenetic tree with n taxa have?

### 

## How many unrooted and rooted trees can be constructed from n taxa?

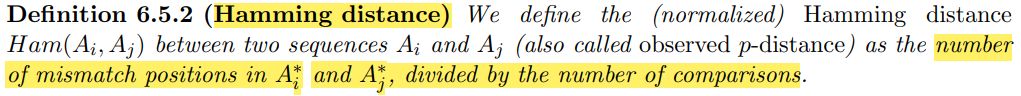
Unrooted: (2n – 5)!! Rooted: (2n-3)!!

## Name some strategies to construct phylogenetic trees

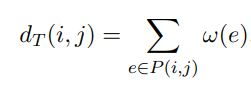
Distance method, maximum parsimony, Maximum Likelihood, Bayesian approaches.

## What is the normalized Hamming distance defined as?

Columns where both sequences contain a gap are ignored. If only one sequencye has a gap we can ignore it, or treat it as a match or a mismatch, depending on the data.



## Define the tree distance of a phylogenetic tree (with formula).

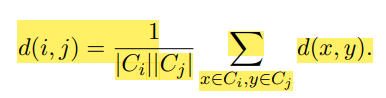


This is with weighted edges, the edge weight is omega.

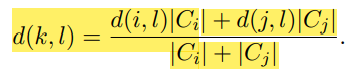
## Describe the steps that are taken in UPGMA.

* Input it the distance matrix D on all taxa, they form n clusters
* Starting from the leaves, identify the off-diagonal entry that is minimal
* These two leaves are merged into a new cluster with one internal new node
* Update the distance matrix by recomputing the distances of the not yet merged elements with the new cluster.
* At the next step the two elements with minimal distance are clustered and merged to a single cluster, distances are updated, etc., until only one cluster is left

## What is the formula for the average distance between pairs of taxa from each cluster?



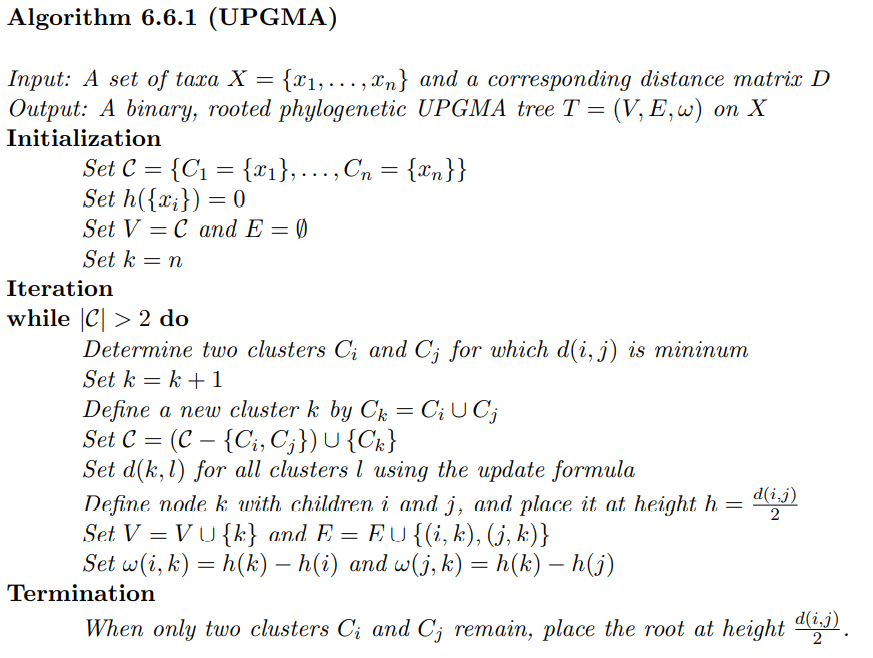
## What is the update formula for a Cluster k (union of clusters i and j) and any other cluster l?



## What are the three cases that are distinguished for determining edge lengths in a tree? What are the corresponding formulas?

1. Both child-nodes are leaves: ω(i, k) = ω(j, k) = h(k) = d(i,j)/2
2. One or two of the two children is an internal node: For the leave: see above, for the other one: ω(j, k) = h(k) – h(j) = d(i, j)/2 − d(a, b)/2
3. Both children are internal nodes: See number 2

## Provide the pseudocode for the UPGMA algorithm.



## What is the molecular clock hypothesis?

The rate of evolutionary events of sequences was constant over time and equal for all lineages in the tree.

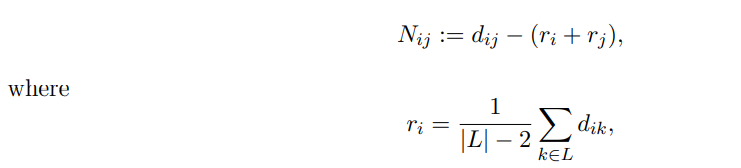
## Is UPGMA suitable or unsuitable for data that doesn’t fulfil the molecular clock hypothesis? What about NJ?

UPGMA: Unsuited. NJ: Suitable for all.

## What is the main difference in the algorithm process between UPGMA and NJ?

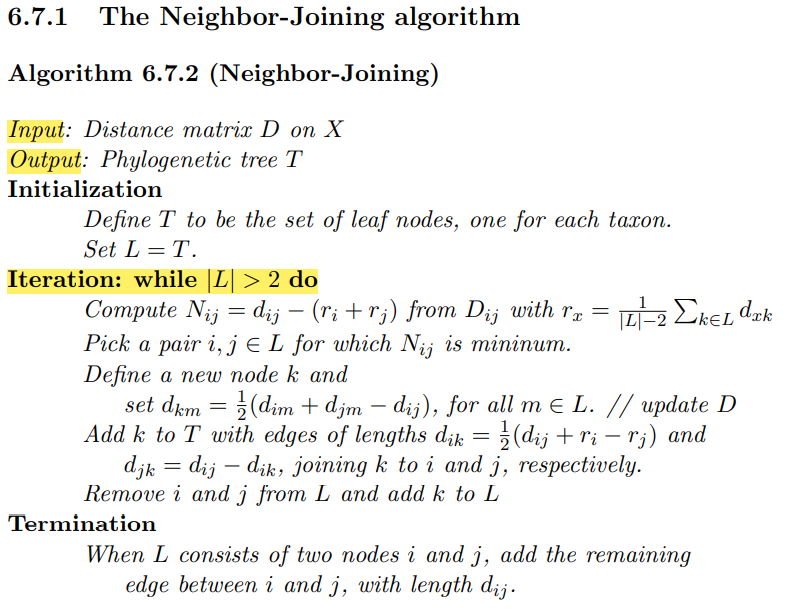
NJ joins branches which minimize the total branch length while UPGMA just chooses the minimal branch length between any two clusters.

## How is the neighboring matrix made up (with formulas)?



L is the set of clusters still present at the respective iteration step.

## Provide the pseudocode for the Neighbor-Joining algorithm



## What is the complexity of UGMA and Neighbor-joining?

UPGMA: runtime complexity O(n3). There are O(n) iteration steps (n-1) internal nodes in a rooted tree, each iteration involves joining a cluster-pair and reducing the input distance matrix (needs O(n) time). Choosing a cluster pair (global minimum) takes O(n2) time.

NJ: Also O(n3). Compute the N matrix to find the pair of clusters to join takes O(n2) time. There are O(n) iteration steps (there are n-2 internal nodes in an unrooted tree).

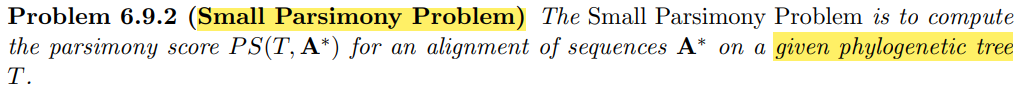
## What are the two assumptions made for maximum parsimony?

Changes in different sites are independent (each column is treated independently). Changes in different lineages across the phylogeny are independent (changes on each edge can be summed up for the overall score).

## How is the parsimony score of a tree computed?

This is done by taking the minimum of all possible labelings of the sum of all non-normalized Hamming distances

## Define the small parsimony problem.



## What algorithm is used to solve the small parsimony problem?

The Fitch algorithm.

## What are the two stages of the Fitch algorithm and what is their purpose? Are they bottom-up or top-down?

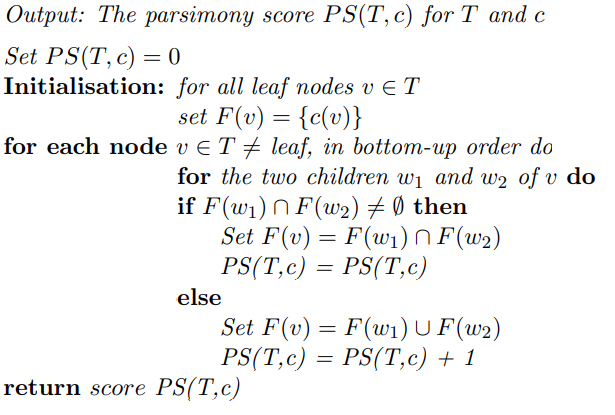
Forward pass is bottom up. It computes the parsimony score and a set F of possible states for each node. Backward pass is top down. It chooses a specific state for each node.

## What are the steps of the forward pass?

* Set parsimony score to zero.
* Start at the leaves and go along the tree up the root
* At each internal node, check if the intersection of states is empty. If not, take intersection and make it the label of the new parent node. If yes, add +1 to parsimony score, label parent node with union set of states.

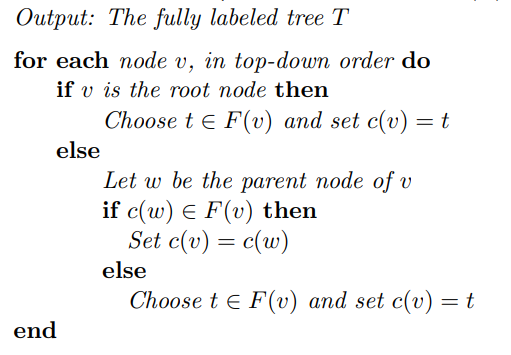
## Provide the pseudocode for the forward-pass

Input: A binary phylogenetic tree T, a state c(w) for each leaf (v)) w of T



## Provide the pseudocode of the backward-pass.

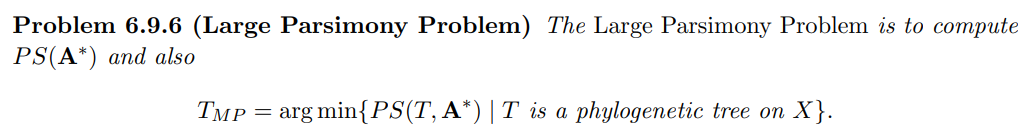
Input: A phylogenetic tree T and the set F(w) for every node w in T



## What is the run time complexity of the Fitch algorithm?

O(nL), because it requires O(n) steps for every column, for the forward-pass as well as for the backward-pass.

## What is the Large Parsimony Problem?



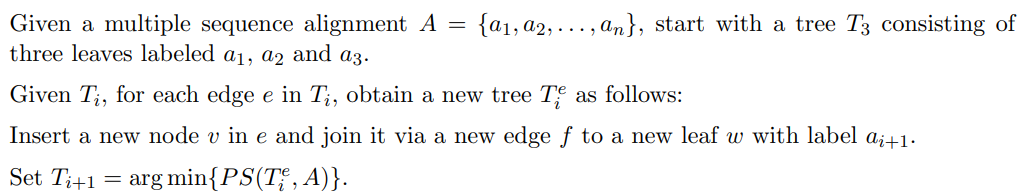
## What is the problem with the Large Parsimony Problem?

It is NP-hard. You have to compute the parsimony score for every unrooted tree. Therefore, we need a heuristic.

## What is the process of the heuristic for choosing a parsimony tree?

We start with three nodes, choose the one with the best parsimony score and then add one node. We join it to the tree and compute the best parsimony score for all options again. Repeat until we have the desired number of nodes.

Formally:



## What is the difference between orthologous genes and paralogous genes?

Orthologous genes are genes with a common ancestor, diverged after a speciation event. Paralogous genes are genes that arose from a duplication event (multiple copies in the genome)