## Chapter 3

# Multiple Sequence Alignments 

Q5E940_BOVIN ---------MPREDRATWKSNYFLKIIQLLDDYPKCFIVGADNVGSKQMQQIRMS LRGK-AVVLMGKNTMMRKAIRGHLENN--PAL RLAO_HUMAN ----------MPREDRATWKSNYFLKIIQLLDDYPKCFIVGADNVGSKQMQQIRMSLRGK-AVVLMGKNTMMRKAIRGHLENN--PALE RLA0_MOUSE ----------MPREDRATWKSNYFLKIIQLLDDYPKCFIVGADNVGSKQMQQIRMS LRGK-AVVLMGKNTMMRKAIRGHLENN--PALE RLAO RAT -----------MPREDRATWKSNYFLKIIQLLDDYPKCFIVGADNVGSKQMQQIRMSLRGK-AVVLMGKNTMMRKAIRGHLENN--PALE RLA0_CHICK ----------MPREDRATWKSNYFMKIIQLLDDYPKCFVVGADNVGSKQMQQIRMSLRGK-AVVLMGKNTMMRKAIRGHLENN--PALE RLAO_RANSY ----------MPREDRATWKSNYFLKIIQLLDDYPKCFIVGADNVGSKQMQQIRMSLRGK-AVVLMGKNTMMRKAIRGHLENN--SALE
Q7ZUG3_BRARE ----------MPREDRATWKSNYFLKIIQLLDDYPKCFIVGADNVGSKQMQTIRLSLRGK-AVVLMGKNTMMRKAIRGHLENN--PALE RLA0 ICTPU -----------MPREDRATWKSNYFLKIIQLLNDYPKCFIVGADNVGSKQMQTIRLSLRGK-AIVLMGKNTMMRKAIRGHLENN--PALE RLA0 ${ }^{-}$DROME RLAO ${ }^{-}$DICDI Q54LP0 ${ }^{-}$DICDI RLAO_PLAF8 RLA0 SULAC RLA0 SULTO -AO- SULSO RLAO_AERPE RLA0 ${ }^{-}$PYRAE RLA0_METAC RLA0 METMA RLA0 ARCFU RLA0 - METKA RLAO-METTH RLAO METTL RLA0 METVA RLA0 ${ }^{-}$METJA RLAO - PYRAB ---------- MVRENKAAWKAOYFTKVVELFDEFPKCFIVGADNVGSKOMONIRTS LRGL-AVVLMGKNTMMRKATRGHLENN--POLE ----------MSGAG-SKRKKLFIEKATKLFTTYDKMIVAEADFVGSSQLQKIRKS IRGI-GAVLMGKKTMIRKVIRDLADSK--PELD ----------MSGAG-SKRKNVFIEKATKLFTTYDKMIV AE ADFVGSSQLQKIRKSIRGI-GAVLMGKKTMIRKVIRDLADSK--PELD ---------MAKLSKQQKKQMYIEKLSSLIQQYSKILIVHVDNVGSNQMASVRKSLRGK-ATILMGKNTRIRTALKKNLQAV--PQIE -----MIGLAVTTTKKIAKWKVDEVAELTEKLKTHKTIIIANIEGFPADKLHEIRKKLRGK-ADIKVTKNNLFNIALKNAG-----YDTK - - - MRIMAVITQERKIAKWKIE EVKELEQKLREYHTIIIANIEGFPADKLHDIRKKMRGM-AEIKVTKNTLFGIAAKNAG-----LDVS - - - MKRLALALKQRKVASWKLEEVKELTELIKNSNT ILIGNLEGFPADKLHEIRKKLRGK-ATIKVTKNTLFKIAAKNAG-----IDIE MSVVSLVGQMYKREKPIPEWKTLMLRELEELFSKHRVVLFADLTGTPTFVVQRVRKKLWKK-YPMMVAKKRIILRAMKAAGLE ---LDDN -MMLAIGKRRYVRT RQYP ARKVKIVSEATELLQKYPYVFLFDLHGLSSRILHEYRYRLRRY-GVIKIIKPTLFKIAFTKVYGG---IPAE -----MAEERHHTEHIPQWKKDEIENIKELIQSHKVFGMVGIEGILATKMQKIRRDLKDV-AVLKVSRNTLTERALNQLG-----ETIP ------MAEERHHT EHIPQWKKDEIENIKELIQSHKVFGMVRIEGILATKIQKIRRDLKDV-AVLKVSRNTLTERALNQLG-----ESIP ------MAAVRGS ---PPEYKVRAVEEIKRMISSKPVVAIVSFRNVPAGQMQKIRREFRGK-AEIKVVKNTLLERALDALG-----GDYL MAVKAKGQPPSGYEPKVAEWKRREVKELKELMDEYENVGLVDLEGIPAPQLQEIRAKLRERDTIIRMSRNTLMRIALEEKLDER--PELE -------MITAESEHKIAPWKIE EVNKLKELLKNGQIVALVDMMEVPARQLQEIRDKIR-GTMTLKMSRNTLIERAIKEVAEETGNPEFA RLAO PYRHO ----MAHVAEWKKKEVEELANLIKSYPVIALVDVSSMPAYPLSQMRRL IRENGGLLRVSRNTLIE LAIKKAAQE LGKPELE RLAO_PYRFU -------------MAHVAEWKKKEVEELANLIKSYPVVALVDVSSMPAYPLSQMRRLIRENNGLLRVSRNTLIELAIKKVAQELGKPELE RLA0_PYRKO -------------MAHVAEWKKKEVEELANIIKSYPVIALVDVAGVPAYPLSKMRDKLR-GKALLRVSRNTLIELAIKRAAQELGQPELE RLAO HALMA -----MSAESERKTETIPEWKQEEVDAIVEMIESYESVGVVNIAGIPSRQLQDMRRDLHGT-AELRVSRNTLLERALDDVD-----DGLE RLAO-HALVO -----MSESEVRQTEVIPQWKREEVDELVDFIESYESVGVVGVAGIPSRQLQSMRRELHGS-AAVRMSRNTLVNRALDEVN-----DGFE RLAO_HALSA ----MSAEEQRTTEEVPEWKRQEVAELVDLLETYDSVGVVNVTGIPSKQLQDMRRGLHGQ-AALRMSRNTLLVRALEEAG-----DGLD RLAO_THEAC -------------MKEVSQQKKELVNEITQRIKASRSVAIVDTAGIRTRQIQDIRGKNRGK-INLKVIKKTLLFKALENLGD----EKLS RLA0 THEVO -------------MRKINPKKKE IVSELAQDITKSKAVAIVDIKGVRTRQMQDIRAKNRDK-VKIKVVKKTLLFKALDSIND----EKLT RLAO PICTO ------------MTEPAQWKIDFVKNLENEINSRKVAAIVSIKGLRNNEFQKIRNSIRDK-ARIKVSRARLLRLAIENTGK----NNIV


By Miguel Andrade at English Wikipedia, https://commons.wikimedia.org/w/index.php?curid=3930704

## Multiple alignment algorithms

Definition. A multiple alignment of sequences $X^{1}, \ldots, X^{n}$ is a series of gapped sequences $\tilde{X}^{1}, \ldots, \tilde{X}^{n}$ such that
(i) $\tilde{X}^{i}$ is an extension of $X^{i}$ obtained by insertions of spaces;
(ii) $\left|\tilde{X}^{1}\right|=\left|\tilde{X}^{2}\right|=\cdots=\left|\tilde{X}^{n}\right|$.

Why are we interested in multiple alignments?

- A multiple alignment carries more information than a pairwise one, as a protein can be matched against a family of proteins instead of only against another one.
- Multiple similarity of (protein) sequences suggests
- a common structure,
- a common function,
- a common evolutionary source.


## The alignment hyper-cube

Best multiple alignment of $r$ sequences:
Best path through $r$-dimensional hyper-cube $D$.


$$
\begin{array}{lllllll}
V & S & N & - & S & & \\
- & S & N & A & - & & \\
- & - & - & A & S & & \\
1 & 1 & 1 & 0 & 1 & & \\
0 & 1 & \varepsilon_{1} \\
0 & 0 & 0 & 1 & 1 & & \lambda \\
\hline
\end{array}
$$

Alignment path for three example sequences.

## Dynamic Programming Solution

- Best multiple alignment of $r$ sequences: Best path through $r$-dimensional hyper-cube.
- Define $S\left(j_{1}, j_{2}, \ldots, j_{r}\right)$ as as the best score for aligning the prefixes of lengths $j_{1}, j_{2}, \ldots, j_{r}$ of the sequences $X^{1}, X^{2}, \ldots, X^{r}$.
- We define $S(0,0, \ldots, 0)=0$, and we calculate

$$
\left.\begin{array}{rl}
S\left(j_{1}, j_{2}, \ldots, j_{r}\right)=\max _{\left(\epsilon_{1}, \ldots, \epsilon_{r}\right): \epsilon_{i} \in\{0,1\}, \epsilon \neq 0} & {[ }
\end{array} S\left(j_{1}-\epsilon_{1}, j_{2}-\epsilon_{2}, \ldots, j_{r}-\epsilon_{r}\right)\right] \text {. }
$$

where $s$ is the scoring function (example $s(a, b, 0)$ : joint score for aligning characters $a, b$ and a gap) and
$\boldsymbol{\epsilon}=\left(\epsilon_{1}, \ldots, \epsilon_{r}\right)$ is a binary vector that indicates the directions of the alignment progress in the hyper-cube.

## Dynamic Programming Solution: Complexity

- The size of the hyper-cube is $O\left(\prod_{j=1}^{r} n_{j}\right)\left(n_{j}=\right.$ length of $\left.x_{j}\right)$.
- Computation of each entry considers $2^{r}-1$ other entries. Example: 000, 001, 010, 011, 100, 101, 110, 111
- If $n_{1}=n_{2}=\cdots=n_{r}=n$, the space complexity is of $O\left(n^{r}\right)$ and the time complexity is of $O\left(2^{r} n^{r}\right)$.



## Scoring Metrics

- A scoring scheme should take into account that...

1. some positions are more conserved than others $\rightsquigarrow$ position specific scoring;
2. the sequences are not independent, but are related by a phylogenetic tree.

- Ideal scoring: Complete probabilistic model of evolution $\leadsto$ Probability of a multiple alignment is composed of the probabilities of all evolutionary events necessary to produce the alignment.
- In practice, we do not have such a model
$\leadsto$ simplifying assumptions: Two main concepts:

1. Position specific, but ignoring the phylogenetic tree;
2. explicit tree model, but position independent.

## Multiple alignments by Profile HMM training

- Suppose we have successfully trained a profile HMM from a set of labeled sequences.
How can we use this HMM to derive the multiple alignment of $n$ sequences?
- Answer: align all $n$ sequences to the profile using the Viterbi algorithm $\rightsquigarrow$ most probable state paths for all sequences.
- Characters aligned to the same match state are aligned in columns.
- Multiple alignments from HMMs are approximations of type one:
- Score is position specific,
- but sequences are treated as independent objects.


## Computing the multiple alignment: example



Multiple Alignment

$$
\begin{aligned}
& \begin{array}{llll}
x_{1}^{1} & x_{2}^{1} & x_{3}^{1} & x_{4}^{1} \\
\hline x_{1}^{2} & x_{2}^{2} & x_{3}^{2} & x_{4}^{2} \\
x_{5}^{2} & x_{6}^{2} \\
\hline
\end{array} \\
& \hline
\end{aligned}
$$

## Computing the multiple alignment: Real example



$$
\begin{aligned}
& \text { FPHF-DLS-----HGSAQ } \\
& \text { FESFGDLSTPDAVMGNPK } \\
& \text { FDRFKHLKTEAEMKASED } \\
& \text { FTQFAG-KDLESIKGTAP } \\
& \text { FPKFKGLTTADQLKKSAD } \\
& \text { FS-FLK-GTSEVPQNNPE } \\
& \text { FG-FSG----AS---DPG }
\end{aligned}
$$

Figure 6.4 A model (top) estimated from an alignment (bottom). The characters in the shaded area of the alignment were treated as inserts.

Durbin et al., Cambridge University Press. https://doi.org/10.1017/CBO9780511790492.004

## Computing the multiple alignment: Real example



FPHF-Dls.....HGSAQ FESFGDlstpdavMGNPK FDRFKHlkteaemKASED FTQFAGkdlesi.KGTAP FPKFKGlttadqlKKSAD FS-FLKgtsevp. QNNPE FG-FSGas.....--DPG

Figure 6.6 Left: the alignment of the seven sequences is shown with lowercase letters meaning inserts. The dots are just space-filling characters to make the matches line up correctly. Right: the alignment is shown after a new sequence was added to the set. The new sequence is shown at the top, and because it has more inserts more space-filling dots were added.

Durbin et al., Cambridge University Press. https://doi.org/10.1017/CBO9780511790492.004

## Multiple Alignments by Profile HMM training

- For parameter estimation in Profile HMMs, aligned training sequences are often unavailable
$\rightsquigarrow$ usually we only have a sample of unaligned sequences, the state paths are unknown.
- Idea: Use EM algorithm for iterative parameter optimization (Baum-Welch algorithm).
- Recall: for the EM algorithm, we need the forward and backward probabilities in the E-step for calculating
- $E_{b l}$ (the expected emission counts) and
- $A_{l^{\prime} l}$ (expected transition counts).


## Simpler Multiple Alignment Algorithms

- Alternative to the probabilistic HMM formulation: Sum of Pairs score:
Sum of scores between all pairs of sequences.
- The SP score for a column $m_{j}$ of the multiple alignment is

$$
S\left(m_{j}\right)=\sum_{k<l} \underbrace{s\left(m_{j}^{k}, m_{j}^{l}\right)}_{\text {from scoring matrix }}
$$

- SP scores lack a probabilistic justification:

Correct log-odds score for 3-way alignment would be

$$
s(a, b, c)=\log \frac{p_{a b c}}{q_{a} q_{b} q_{c}} \neq \underbrace{\log \frac{p_{a b}}{q_{a} q_{b}}+\log \frac{p_{b c}}{q_{b} q_{c}}+\log \frac{p_{a c}}{q_{a} q_{c}}}_{\mathrm{SP} \text { score }}
$$

## Approximation Algorithms for MSA

- Even for SP scores, MSA has exponential time complexity.
- Denote by $D(S, T)$ the minimum cost of aligning $S$ with $T$.
- Let $\sigma(x, y)$ be our cost function, i.e. the cost of aligning the character $x$ with the character $y$, for $x, y \in \Sigma \cup\{-\}$.
- Here we minimize costs $\sigma$ instead of maximizing scores $s$. Example transformation: $\sigma(x, y)=\exp (-\lambda s(x, y))$.
- We assume that $\sigma(-,-)=0, \sigma(x, y)=\sigma(y, x)$, and that the triangle inequality holds: $\sigma(x, y) \leq \sigma(x, z)+\sigma(z, y)$

Problem: The SP alignment problem.
INPUT: A set of sequences $\mathcal{S}=\left\{S_{1}, \ldots, S_{k}\right\}$.
QUESTION: Compute a global multiple alignment $\mathcal{M}$ with minimum SP-costs, given the above assumptions on $\sigma(\cdot, \cdot)$.

## The Center Star Method for Alignment

Approximation algorithm for calculating the optimal multiple alignment under the SP metric with approximation ratio of two.

- Center string: String that minimizes $\sum_{S_{j} \in \mathcal{S}} D\left(S_{c}, S_{j}\right)$.
- Center star: A star tree of $k$ nodes, center node labeled $S_{c}$, each of the $k-1$ remaining nodes labeled by $\mathcal{S} \backslash\left\{S_{c}\right\}$.


Type-2 approximation: explicit (star-)tree model, but position independent scoring.

## The Center Star Algorithm

1. Find $S_{t} \in \mathcal{S}$ minimizing $\sum_{i \neq t} D\left(S_{i}, S_{t}\right)$ and let $\mathcal{M}=\left\{S_{t}\right\}$

2. Add sequences in $\mathcal{S} \backslash\left\{S_{t}\right\}$ to $\mathcal{M}$ one by one so that the pairwise alignment of every newly added sequence with $S_{t}$ is optimal. Add spaces, when needed, to all pre-aligned sequences.

## The Center Star Algorithm

## Pair：

| Given： |
| :--- |
| ATTGCCATT |
| ATGGCCATT |
| ATCCAATTTT |
| ATCTTCTT |
| ATTGCCGATT |

ATGGCCATT ATTGCCATT

ATC－CAATTTT ATTGCCATT－－

ATCTTC－TT ATTGCCATT

ATTGCCGATT ATTGCC－ATT

## Alignment：

ATTGCCATT
ATGGCCATT

ATTGCCATT－－
ATGGCCATT－－ ATC－CAATTTT

ATTGCCATT－－ ATGGCCATT－－ ATC－CAATTTT ATCTTC－TTー－

ATTGCC－ATT－－
ATGGCC－ATT－－
ATC－CA－ATTTT
ATCTTC－－TTー－
ATTGCCGATTー－

## The Center Star Algorithm: Analysis

- $\mathcal{M}$ : Multiple alignment produced by the center-star algorithm.
- $d(i, j)$ : Cost of the resulting pairwise alignment of $S_{i}$ and $S_{j}$, induced by $\mathcal{M}$.
Note that

- SP-costs of center-star alignment: $\sigma(\mathcal{M})=\sum_{i=1}^{k} \sum_{j=1, j \neq i}^{k} d(i, j)$
- $\mathcal{M}^{*}$ : Optimal SP-alignment of all strings in $\mathcal{S}$ with $\operatorname{costs} \sigma\left(\mathcal{M}^{*}\right)$.


## Theorem 1.

$$
\frac{\sigma(\mathcal{M})}{\sigma\left(\mathcal{M}^{*}\right)}=\leq \frac{2(k-1)}{k} \leq 2
$$

Theorem 2. The running time of the center star algorithm for $k$ strings with length $\leq n$ is $O\left(k^{2} \cdot n^{2}\right)$.

Proofs: see exercises.

## Progressive alignment heuristics

Idea: Use a binary "guide tree" instead of a star tree
(Guide tree defines a model of evolution)
Leaves: sequences, inner nodes: alignments (sequence-sequence, sequence-profile, or profile-profile).

myoglobin
$\longleftarrow$ haemoglobins

Durbin et al., Cambridge University Press. https://doi.org/10.1017/CBO9780511790492.004

## Progressive alignment: ClustalW

ClustalW is a software package for multiple alignment (implementing an algorithm of Thompson, Higgins, Gibson 1994).

1. Calculate all pairwise alignment scores, convert to pairwise distances.
2. Use Neighbor-Joining algorithm to build a tree from the distances.
3. Align sequence - sequence, sequence - profile, profile - profile.

This algorithm makes use of many ad-hoc rules such as weighting, different matrix scores
 and special gap scores.

