

**Seminar**  
**Aktuelle Themen der Bioinformatik SS 07**

Fachbereich Informatik und Mathematik  
Johann Wolfgang Goethe-Universität, Frankfurt am Main

Thema

**Gibbs sampler for statistical multiple alignment - Jensen, Hein (2005)**

von Marten Rosselli

# 1. Einleitung

## Alinierung/Alignment:

- Vergleich von Sequenzen (z.B. Nucleinsäuren (DNA/RNA) mit u.a. Basen (Adenin (A), Guanin (G), Cytosin (C), Thymin (T) oder Uracil (U) ) als Bausteine oder Proteine mit Aminosäuren (23 bisher bekannt) als Bausteine)
- funktionelle oder evolutionäre Verwandtschaft (Homologie)
- paarweises Alignment (global, lokal, semiglobal) mit z.B. Needleman-Wunsch (1970) (global) oder Smith und Waterman (1981) (lokal) (dyn. Prog. mit Scorefunktion). Laufzeit:  $O(nm)$
- multiples Alignment steht im Gegensatz für das Analysieren mehrerer Sequenzen mit z.B. CLUSTALW (progressive Strategie) oder Gibbs-Sampler (statist. Ansatz) f. mult. Alig..

Vorteil von stat. Ansatz: Miteinbeziehung der stochastischen Natur des Evolutionsprozesses!

## Beispiel für multiples Alignment:

### ClustalW-Output-Files (Alignment von 5 Aminosäuresequenzen)

Output: CLUSTAL W (1.83) multiple sequence alignment

```
FOS_RAT      MMFSGFNADYEASSSRCSSASPAGDSLSYYHSPADSSMGVNTQDFCADLSVSSANF
FOS_MOUSE   MMFSGFNADYEASSSRCSSASPAGDSLSYYHSPADSSMGVNTQDFCADLSVSSANF
FOS_CHICK   MMYQGFAGEYEAPSSRCSSASPAGDSLTYYPSPADSSMGVNSQDFCTDLAVSSANF
FOSB_MOUSE  -MFQAFPGDYDS-GSRCSS-SPSAESQ--YLSSVDSSPPAASQE-CAGLGEMPGSF
FOSB_HUMAN  -MFQAFPGDYDS-GSRCSS-SPSAESQ--YLSSVDSSPPAASQE-CAGLGEMPGSF
          *:.:. *  .:.*:  .***** **:.:*  *  *..***  . :*:  *:.*.  ...*
```

- \* = Identische in allen 5 Zeilen
- . = Sehr ähnliche(semi-conserved substitutions)
- : = Sehr ähnliche (conserved substitutions)

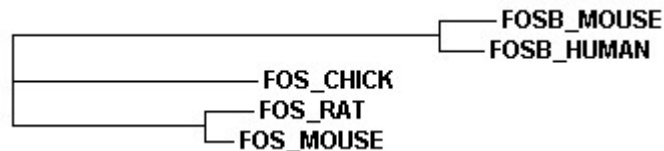


Abbildung 1: Phylogram Tree (ClustalW)

„Gibbs sampler for statistical multiple alignment“ von Jens Ledet Jensen und Jotun Hein (2005):

- gibt für  $S_1, \dots, S_n$  die Seq. der Vorfahren und Alignments an.
- Betrachtet in jedem Schritt *3-star-tree*
- Sämtliche Sequenzen sind in einem binären Baum gespeichert
- Grundlage ist Sequenz-Evolutionsmodell von Thorne, Kishino und Felsenstein (1991) mit Deletionen und Insertionen → HMMs und damit eff. Algos

Info: Wird später noch mit dem zeitgleich entwickelten Algo. von Holmes und Bruno (2001) verglichen.

## 2. Gibbs-Sampling-Algorithmus (allgemein)

- Nach Physiker Josiah Willard Gibbs benannt.

Wikipedia:

[...] Gibbs-Sampling ist ein Algorithmus, um eine Folge von Stichproben der gemeinsamen Wahrscheinlichkeitsverteilung zweier oder mehrerer Zufallsvariablen zu erzeugen. Das Ziel ist es dabei, die unbekannte gemeinsame Verteilung zu approximieren. [...]

[...] Gibbs-Sampling eignet sich besonders dann, wenn die gemeinsame Verteilung eines Zufallsvektors unbekannt, jedoch die bedingte Verteilung einer jeden Zufallsvariable bekannt ist. Das Grundprinzip besteht darin, in wiederholender Weise eine Variable auszuwählen und gemäß ihrer bedingten Verteilung einen Wert in Abhängigkeit der Werte der anderen Variablen zu erzeugen. Die Werte der anderen Variablen bleiben in diesem Iterationsschritt unverändert. Aus der entstehenden Folge von Stichprobenvektoren lässt sich eine Markow-Kette herleiten. [...]

### 3. Grundlage: Das TKF91-Modell und Markovketten

- Sequenz-Evolutionsmodell mit Insertionen und Deletionen (und Berücksichtigung von z.B. Transversionen und Transitionen)

- Jede Position entwickelt sich unabhängig

- Neuer Buchstabe wird durch eine Verteilung  $\pi$  bestimmt

- Existierender Buchstabe unterliegt Markov'schen Substitutionsprozess mit stationärer Ws  $\pi$  und Transitions/Substitution-Ws  $f(w_2|w_1, \tau)$ .

- Die stationäre Verteilung einer Sequenz S der Länge L:

$$P(S) = (1 - \gamma) \gamma^L \prod_{i=1}^L \pi(S[i]) \quad \text{mit} \quad \gamma = \frac{\lambda}{\mu} \quad (1)$$

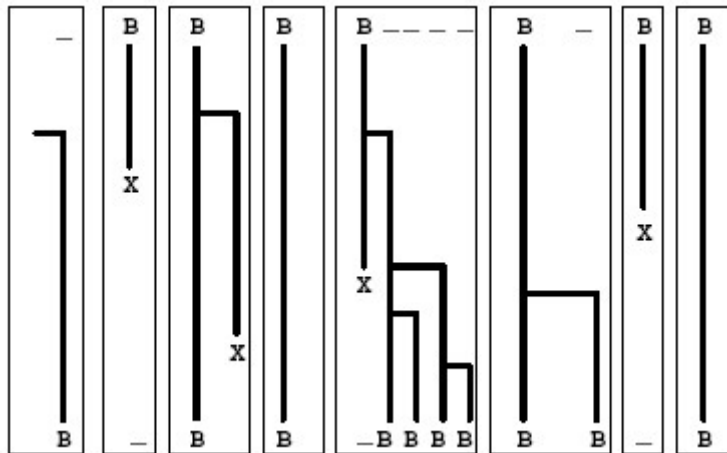
- Geburtsrate  $\lambda$  (typischer Wert z.B.  $\lambda = 0,099$ )

- Sterberate  $\mu > \lambda$  (typischer Wert  $\mu = 0,1$ ) angegeben.

- Immortal-Rate ganz links

- Buchstabe kann überleben, substituiert werden, austerben oder geboren werden.

- Neuer Buchstabe (Geburt) wird rechts angefügt



## Zusammenhang von TKF91-Modell, HMMs und Alignments

- Evolution durch Deletionen, Insertionen, Substitution beschreibbar (Markovkette mit 3 Zuständen)
- Zustände:  $M$ ,  $D$  und  $I$  (plus Start- und End-Zustand).
- Aus Markovkette (genauer: durch den Weg durch die Markovkette) Alignment direkt ablesbar

### Info:

- Symbol # für Präsenz, Symbol - für Abstinenz eines Buchstabens.

- Def.:  $\gamma = \frac{\lambda}{\mu}$  und  $\beta = \frac{(1 - \exp(-(\mu - \lambda)\tau))}{(1 - \gamma \exp(-(\mu - \lambda)\tau))}$

Um Transitionsmatrix angeben zu können:

$b(\#, \#) = \gamma\beta$  (Match und Geburt) und  $b(\#, -) = 1 - b(\#, \#)$  (Match und Geburt)

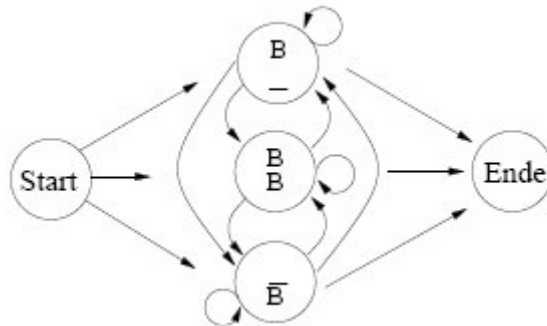
$b(-, \#) = 1 - \left(\frac{\beta}{(1 - \exp(-\mu\tau))}\right)$  (Gap aber Geburt) und  $b(-, -) = 1 - b(-, \#)$  (Gap u. keine Geburt)

$s(\#) = \exp(-\mu\tau)$  (Überleben) und  $s(-) = 1 - s(\#)$  (Aussterben)

### Transitionsmatrix (3)

	<b>M</b>	<b>D</b>	<b>I</b>	<b><math>\epsilon</math></b>
<b>M</b>	$b(\#, -)\gamma_S(\#)$	$b(\#, -)\gamma_S(-)$	$b(\#, \#)$	$b(\#, -)(1-\gamma)$
<b>D</b>	$b(-, -)\gamma_S(\#)$	$b(-, -)\gamma_S(-)$	$b(-, \#)$	$b(-, -)(1-\gamma)$
<b>I</b>	$b(\#, -)\gamma_S(\#)$	$b(\#, -)\gamma_S(-)$	$b(\#, \#)$	$b(\#, -)(1-\gamma)$

Markovkette:



Aber: Aussehen von Matrix und MK für mult. Alignment?

## 4. Grundlage: HMMs

- Übernimmt Markov-Eigenschaften von *normaler* MK (nächste Zustand darf nur vom jetzigen Zustand abhängen, Ausnahme: MKs höherer Ordnungen).

Besonderheit:

Neben Zustandsraum noch Raum mit den für uns sichtbaren Emissionen

Formal:

Eine unbeobachtbare Markovkette  $X_1, X_2, \dots$  auf einem Zustandsraum  $Z$  mit Übergangswahrscheinlichkeiten  $P_{xy} = \mathcal{W}_S(X_i = y | X_{i-1} = x)$  und Startverteilung  $P_x = \mathcal{W}_S(X_i = x)$  emittiert beobachtbare zufällige Signale  $S_1, S_2, \dots$  aus einem Alphabet  $A$ . Die Emissionswahrscheinlichkeiten für  $S_i$  hängen jeweils nur von  $X_i$  ab, d.h. für  $y \in Z$  und  $b \in A$  gilt:

$$e_y(b) := \mathcal{W}_S(S_i = b | X_i = y) = \mathcal{W}_S(S_i = b | X_i = y, X_1, X_2, \dots, S_1, \dots, S_{i-1}, S_{i+1}, \dots)$$

## 5. Gibbs sampler für mult. Alignment

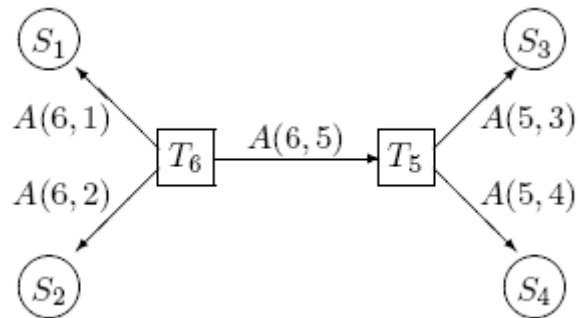
### 5. 1. Notation

Sequenzen in Baum einordnen:

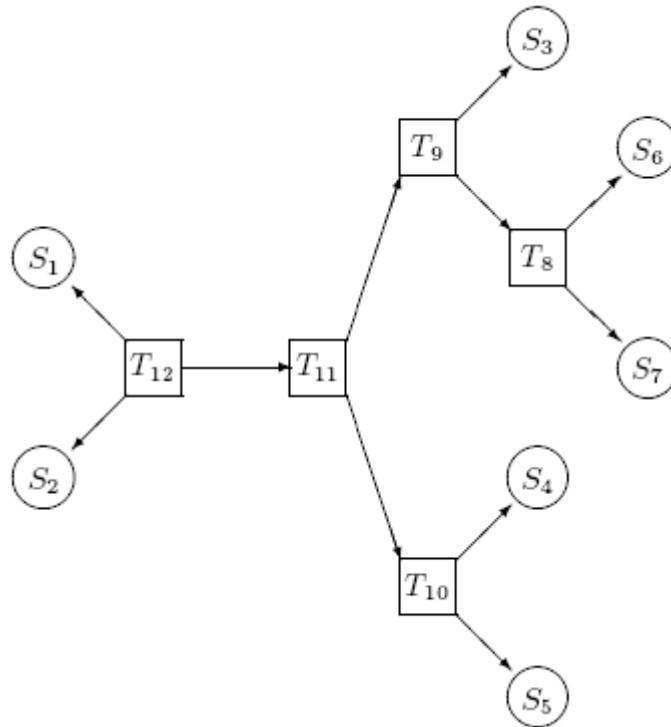
- $n$  beobachtbare Sequenzen  $S_1, \dots, S_n$  in Blättern des binären Baumes
- Dann  $v = n - 2$  innere Knoten  $T_{n+1}, \dots, T_{n+v}$
- Wurzel:  $T_{n+v}$
- Jeder innere Knoten  $n+1 \leq u < n+v$  hat Vorgänger  $a(i)$  von den inneren Knoten  $i+1, \dots, n+v$  und zwei Nachfolger  $d_1(i)$  und  $d_2(i)$  von den inneren Knoten  $n+1, \dots, n+i-1$  und den Blättern des Baumes.
- Da die Wurzel keinen Vorgänger  $a(n+v)$  hat, wird dort der Vorgänger durch einen Nachkommen ersetzt.
- Für ein Blatt  $j$  ist der Vorgänger  $a(j)$  immer aus der Menge der inneren Knoten.

Beispiele:

Binärer Baum mit 4 Sequenzen  $S_1, \dots, S_4$ , mit inneren Knoten  $T_5, T_6$  und der Wurzel  $T_6$ :



Baum mit 7 Sequenzen  $S_1, \dots, S_7$ , mit inneren Knoten  $T_8, \dots, T_{12}$  und mit Wurzel  $T_{12}$  :



## Alignments in Baum einordnen:

- Eine Kante vom Knoten  $a(j)$  zum Knoten  $j$  wird mit  $j$  bezeichnet, so dass die Kantenmenge  $j=1, \dots, n+v-1$  ist.
- Die Kante  $j$  hat die Länge  $\tau_j$
- Ein Alignment  $A(a(j), j)$  besteht aus einer Sequenz mit den Zuständen  $M$ ,  $D$  und  $I$ .

## Bedeutung für Algorithmus

- Betrachtet in jedem Schritt *3-star-tree*
- Wir betrachten für alle  $r=n+1, \dots, n+v$  einen *3-star-tree* mit inneren Knoten  $r$  und Blättern  $a(r)$ ,  $d_1(r)$  und  $d_2(r)$  und simulieren neue Werte für Sequenz  $T_r$  und Alignments  $A(r, a(r))$ ,  $A(r, d_1(r))$  und  $A(r, d_2(r))$ .

(möglich durch Ws bedingt auf die Seq in den drei Blättern (11))

## 5. 2. Zustände und Transitions-Ws

Betrachtung von mult. Alignments (jeweils ein *3-star-tree*) und deren HMMs im Rahmen des TKF91-Modells. *3-star-tree* mit  $T$  als innerer Knoten und Blättern  $S_1, S_2, S_3$ , evolutionäre Zeiten  $\tau_1, \tau_2, \tau_3$  entlang Kanten.

→ #Zustände steigt von 3 auf 15 und Transitionsmatrix wird komplizierter

### Zwei Mengen von Zuständen:

1. Menge der Blätter  $J$  in denen ein Zeichen aus  $T$  in einem bestimmten Blatt überlebt.  
 $j \in J$  bedeutet, dass der Buchstabe im Blatt  $j$  überlebt hat und  $j \notin J$  dass er nicht überlebt hat.

Zustandsbezeichnung:  $M(J)$ , wobei  $J=0$  (da es möglich sein muss, dass die Position in allen drei Blättern ausstirbt, d.h. u.a. auch dass Deletionszustände überflüssig)

2. Teilmenge (keinen Gaps am Anfang oder Gaps beliebig verlängern) von  $J$ , mit den  $j$  wo eine Geburt (Insertion) eines neuen Zeichens stattgefunden hat.

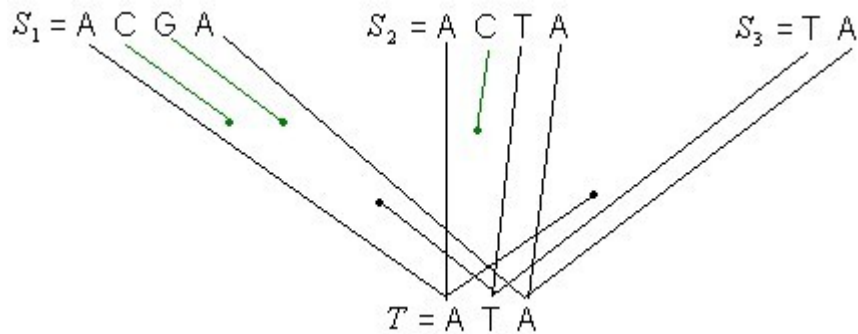
Zustandsbezeichnung:  $I(J)$  mit  $J \neq 0$ .

Von einem Zustand  $M(J)$  können wir in jeden anderen Zustand wechseln. Von einem Zustand  $I(J_1)$  können wir in jeden Zustand  $M(J_2)$ , aber nur in Zustände  $I(J_2)$  mit  $J_2 \subseteq J_1$

Durch die Teilmengenrestriktion wird sichergestellt, dass der Algorithmus eine bestehende Sequenz mit einer Insertion mit weiteren Insertionen verlängern kann, aber die Anzahl der Gaps der inneren Sequenz so nicht größer als die längste Insertionssequenz wird.

Die Menge der 15 Zustände wird mit im Folgenden mit  $E$  bezeichnet.

Beispiel:



Markovkette:

$$M(\{1,2\}) \rightarrow I(\{1,2\}) \rightarrow I(\{1\}) \rightarrow M(\{2,3\}) \rightarrow M(\{1,2,3\})$$

## Transitionsmatrix

-  $b(\#, -; j)$  ,  $b(-, \#; j)$  ,  $b(-, -; j)$  ,  $s(\#; j)$  und  $s(-; j)$  wie oben.

- Für Zustand  $x$  der Form  $M(J)$  oder  $I(J)$  für  $j=1,2,3$  ist  $x_j=\#$  wenn  $j \in J$  und  $x_j=-$  wenn  $j \notin J$  .

Die Transitionsmatrix der Übergangswahrscheinlichkeiten  $p(x, y)$  :

	$y=M(J_2)$	$y=I(J_2)$	$y=\varepsilon$
$y=M(J_1)$	$B(\#, \#)\gamma\left(\prod_{j=1}^3 s(y_j; j)\right)$	$B(\#, -)$	$B(\#, \#)(1-\gamma)$
$y=I(J_1)$	$B(-, \#)\gamma\left(\prod_{j=1}^3 s(y_j; j)\right)$	$B(-, -)$	$B(-, \#)(1-\gamma)$

(4)

Mit:

$$B(\#, \#) = \left( \prod_{j=1}^3 b(x_j, -; j) \right) , \quad B(\#, -) = \left( \prod_{j=1}^3 b(x_j, y_j; j) \right) , \quad B(-, \#) = \left( \prod_{j=1}^3 b(\#, -; j) \right) \quad \text{und}$$

$$B(-, -) = \left( \prod_{j=1}^3 b(\#, y_j; j) \right) .$$

- Startzustand entspricht  $M = (\{1, 2, 3\})$  und emittiert keine Zeichen.
- Zustand  $M(J)$  emittiert ein Zeichen  $w_0$  am inneren Knoten und emittiert ein Zeichen  $w_j$  an den Blättern  $j \in J$ .
- Zustand  $I(J)$  emittiert nur ein Zeichen  $w_j$  an den Blättern  $j \in J$ .

Emissionswahrscheinlichkeiten:

$$p_e^0(w | M(J)) = \pi(w_0) \prod_{\{j \in J\}} f(w_j | w_0; \tau_j) , \quad p_e^0(w | I(J)) = \prod_{\{j \in J\}} \pi(w_j) \quad (5)$$

Summieren wir über alle möglichen  $w_0$  erhalten wir

$$p_e(w | M(J)) = \sum_{w_0} \pi(w_0) \prod_{\{j \in J\}} f(w_j | w_0; \tau_j) , \quad p_e^0(w | I(J)) = \prod_{\{j \in J\}} \pi(w_j) \quad (6)$$

### 5. 3. Simulation eines „3-star-trees“

- Länge der Sequenz  $S_j$  sei  $L_j$ ,  $j=1,2,3$ .

- Teilsequenz  $S_j[a:b]$  bezeichnet.  $a > b$  sei dTeilsequenz  $S_j[a:b]$  leer.

- Für Spaltenvektoren  $u$  und  $v$  mit Integereinträgen bezeichne  $S[u:v]$  die drei Teilsequenzen  $S_j[u_j:v_j]$ ,  $j=1,2,3$ .

- Für Zustand  $x = M(J)$  definieren wir  $t(x)=1$  und einen 3-dimensionalen Vektor  $l(x)$  mit Einseinträgen an den Koordinaten  $j \in J$  und Nulleinträgen an den restlichen Stellen.

- Für Zustand  $x = I(J)$  definieren wir  $t(x)=0$  und  $l(x)$  wie oben.

- Längen von den entsprechenden Sequenzen bis zu einer Position  $i$ :  $L^i = l(x^1) + \dots + l(x^i)$  und  $t^i = t(x^1) + \dots + t(x^i)$

- Mult. Alignment eines 3-star-trees:  $x^0, \dots, x^N$  (eine Folge aus  $\Xi$ ).  $x^0$  sei der Startzustand,  $x^i \in \Xi$  mit  $i=1, \dots, N$  die „inneren“ Zustände und  $x^{N+1}$  Endzustand  $\varepsilon$ .

Gemeinsame Verteilung der Sequenzen und der Alignments unter Verwendung der Formel für die Emissionswahrscheinlichkeiten:

$$P(N=n, x^1, \dots, x^n, T, S) = p(x^n, \varepsilon) \prod_{i=1}^n p(x^{(i-1)}, x^i) p_e^0(T[t^{(i-1)}+1:t^i], S[L^{(i-1)}+1:L^i] | x^i) \quad (7)$$

Wobei  $n$  und  $x^0, \dots, x^N$  so gewählt werden, dass  $L^n = l(x^1) + \dots + l(x^n) = L$  und  $L$  der Vektor der Längen der (äußeren) Sequenzen ist.

Summieren wir diesen Ausdruck über die möglichen Buchstaben der inneren Sequenz  $T$  wird  $p_e^0$  durch  $p_e(S[L^{(i-1)}+1:L^i] | x^i)$ .

Info: Der Ausdruck  $T[t^{(i-1)}+1:t^i]$  ist Null für den Fall  $t^{(i-1)} = t^i$  und ansonsten gibt  $T[t^{(i-1)}+1:t^i]$  eine Position innerhalb der Sequenz  $T$  an.

Wahrscheinlichkeit für einen Teil des Alingments:

Funktion  $F(K | x^0)$ .  $K$  Spaltenvektor mit Integer-Einträgen und  $x^0$  irgendein (Initial-)Zustand aus  $\Xi$ .

$$F(K | x^0) = \sum_{n=0}^{\infty} \sum_{x^1, \dots, x^n \in \Xi: K + L^n = L} p(x^n, \varepsilon) \prod_{i=1}^n p(x^{(i-1)}, x^i) p_e(S[K + L^{(i-1)} + 1: K + L^i] | x^i) \quad (8)$$

Wobei die innere Summe Null ist wenn es keine  $x^0, \dots, x^N$  mit  $K + L^n = L$  gibt.

D.h.  $F(K | x^0) = 0$  wenn es  $j$ 's mit  $K_j > L_j$  gibt.

Diese Funktion gibt uns die Wahrscheinlichkeit der Sequenzen  $S[K+1:L]$  an, bei Start in  $x^0$ .

→ Ws für 3 Sequenzen in Blättern:  $P(S)=F(0|I)$  (0 weil man alle Positionen der Sequenzen berücksichtigt, und  $I$  weil man im Startzustand der Markovkette anfangen möchte)

Aus (7) (komplette Ws) und (8) (Ws von Teilstrings):

$$P(N \geq k, x^1, \dots, x^k, T[1:t^k], S) \\ = \left( \prod_{i=1}^k p(x^{(i-1)}, x^i) p_e^0(T[t^{(i-1)}+1:t^i], S[L^{(i-1)}+1:L^i] | x^i) \right) F(L^k | x^k)$$

Diesen Ausdruck geteilt durch die Ws für die 3 Seq. in den Blättern  $P(S)=F(0|I)$  :

$$P(N \geq k, x^1, \dots, x^k, T[1:t^k] | S) \\ = \left( \prod_{i=1}^k p(x^{(i-1)}, x^i) p_e^0(T[t^{(i-1)}+1:t^i], S[L^{(i-1)}+1:L^i] | x^i) \right) F \frac{(L^k | x^k)}{(F(0|I))} \quad (10)$$

Nun teilen wir diesen Ausdruck durch den gleichen Ausdruck jedoch mit  $k$  ausgetauscht durch  $k-1$  und erhalten:

$$P(N \geq k, x^k, T[t^{(k-1)}+1:t^k] | S, N \geq (k-1), x^1, \dots, x^{(k-1)}, T[1:t^{(k-1)}]) \\ = p(x^{(k-1)}, x^k) p_e^0(T[t^{(k-1)}+1:t^k], S[L^{(k-1)}+1:L^k] | x^k) F \frac{(L^k | x^k)}{(F(L^{(k-1)} | x^{(k-1)}))} \quad (11)$$

Mit (11) können wir nacheinander  $(x^1, T[1:t^1])$ ,  $(x^2, T[1:t^2])$ , ... simulieren wenn wir  $F(K|x)$  für alle  $x$  kennen.

$F(K|x)$  berechnen wir mit Rekursion

$$F(K|x) = \sum_{x \in \mathcal{E}} p(x, z) p_e(S[K+1:K+l(z)]|x) F(K+l(z)|z)$$

Wir haben also (8) aufgebrochen in einmal die Summe nur über  $x^1$  und dann den Rest ausgedrückt durch  $F(K+l(z)|z)$ .

Haben wir das Alignment  $x^1, \dots, x^N$  simuliert für einen *3-star-tree* mit innerem Knoten  $r$  und Blättern  $a(r)$ ,  $d_1(r)$  und  $d_2(r)$  dann können wir direkt die Alignments  $A(r, a(r))$ ,  $A(r, d_1(r))$  und  $A(r, d_2(r))$  ablesen.

#### 5. 4. Unterschied zu Holmes und Bruno

- Bei Holmes/Bruno im ersten Schritt jeweils drei „normale“ Alignments von der inneren Sequenz  $T$  zu den äußeren Sequenzen  $S_1, \dots, S_3$
- Nur Zustände  $x^1, \dots, x^N$  der Form  $M$ ,  $D$  oder  $I$
- Schritt 2: Drei unabhängigen Alignments zu einem einzigen Alignment mit Zuständen der Form  $M(J)$  und  $I(J)$  zusammenführen.

## 6. Mixing

- Mixing? Bei schnellem *mixing* werden Zustände der Kette schnell gewechselt. Man hängt nicht lokal in der Kette fest.

- Bei Jensen/Hein-Algo simuliert die innere Sequenz und die Alignments direkt aus der Ws konditioniert über die Sequenzen in den Blättern. (in einem Schritt und in Abhängigkeit von allen 3 Seq. gleichzeitig)

→ per Konstruktion bei einem *3-star-tree* keine Korrelation

- Bei mehr als 3 Seq. geringe Korrelation (s. Abb. 6)

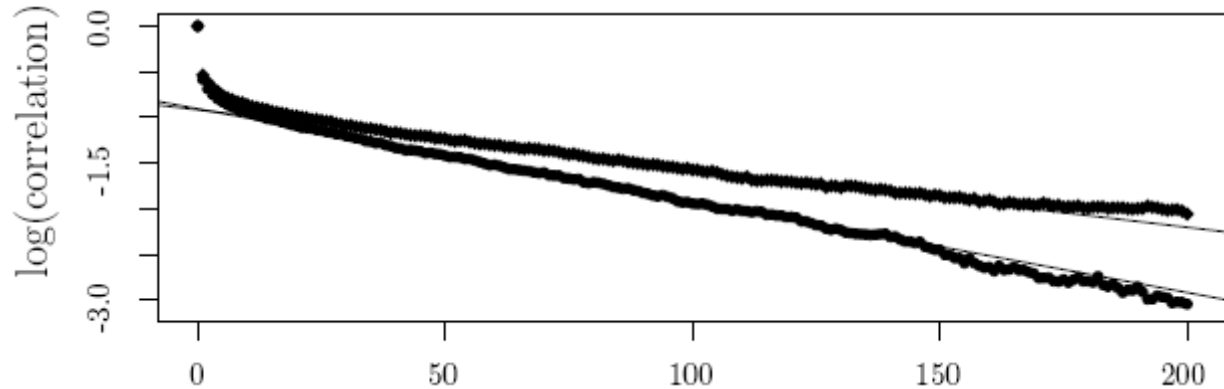
- Holmes/Bruno: Zustände wechseln langsamer

Wie kann man das erkennen? Beobachtung der Werte der Autokorrelationen über einen Zeitraum hinweg.

Wikipedia über Autokorrelation: [...] Vergleicht man eine Folge mit sich selbst, so spricht man von Autokorrelation. Da jede unverschobene Folge mit sich selbst am Ähnlichsten ist, hat die Autokorrelation für die unverschobenen Folgen den höchsten Wert. Wenn zwischen den Gliedern der Folge eine Beziehung besteht, die mehr als zufällig ist, hat auch die Korrelation der ursprünglichen Folge mit der verschobenen Folge in der Regel einen Wert, der signifikant von Null abweicht. Man sagt dann, die Glieder der Folge sind autokorreliert. [...]

## Holmes/Bruno

- die ersten 200 Autokorrelationen aufgetragen
- oberen Kurve entspricht 3 Seq. mit jeweils Länge von 150
- unteren Kurve 3 Seq. mit jeweils Länge von 75.

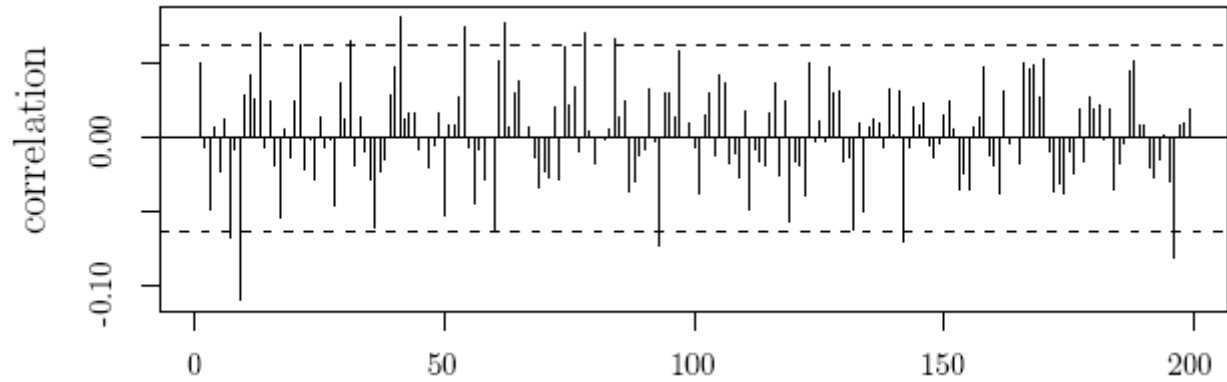


(Abbildung 5, Autokorrelation bei Holmes-Bruno)

Beachte: Log von Werten zwischen 0 bis 1 ist negativ  
Deshalb: Je "negativer" je unkorrelierter.

## Jensen-Hein

- Korrelation schwankt um 0 (hier bei 4 Seq)



(Abbildung 6, Autokorrelation bei Jensen-Hein)

Um nun zu sehen, wie oft man nach Holmes-Bruno simulieren muss um die gleiche Präzision verglichen mit der Situation von  $n$  (wenn  $n$  groß ist) unabhängigen Werten zu haben nutzen wir eine Formel aus der Statistik.

$n \text{VAR}(x) \approx 1 + 2 \sum_k r_k$ , wobei:

$x$  Durchschnitt,  $\sum_k r_k$  Summe der Autokorrelationen,  $n \rightarrow \infty$ .

Ist z.B.  $1 + 2 \sum_k r_k = 100$ , so ist die Interpretation dass wir  $100n$  mal simulieren müssen um die Genauigkeit verglichen mit der Situation von  $n$  zufälligen Werten zu haben.

- Man muss Holmes/Bruno-Algo länger laufen lassen um die Genauigkeit von Jensen/Hein zu erreichen.

- Aber Laufzeit von Holmes/Bruno ist dafür besser: Jensen und Hein  $O(n^3)$ , Holmes/Bruno  $O(n^2)$ ,  $n$  typische Länge einer Sequenz

Für Holmes/Bruno gilt:

<b>Länge</b>	3-star, 75	3-star, 150	4-Seq, 75	4-Seq, 150	7-Seq, 75
<b>Steigung</b>	-0,0101	-0,0064	-0,0061	-0,0061	-0,0055
$1 + 2 \sum_k r_k$	82	127	130	140	176

Fairer Vergleich:

	3-star, 75	3-star, 150	4-seq, 75	4-seq, 150	7-seq, 75
H&B [s]	4,99	15,04	9,94	27,77	18,77
J&H [s]	158,5	775,4	330,3	1524,0	830,2
Ratio	32	52	33	55	44
$\frac{(1 + 2 \sum_k r_k)}{Ratio}$	2,6	2,5	3,9	2,6	4,0

## 7. Parameterschätzung

### 7.1. ML-Ansatz

Allgemein: Um die Parameter des Modells zu schätzen können wir den Maximum-Likelihood (ML)-Ansatz verwenden. Für eine einzige Sequenz bedeutet das erstmal: Gegeben sei eine Sequenz  $s_1, s_2, \dots, s_n \in A$  von Beobachtungen aus einem HMM auf einem Zustandsraum  $Z$ . Gesucht seien nun die Parameterwerte

$$\theta = (P_{xy}, e_x(s))_{(x, y \in Z, s \in A)} .$$

$\hat{\theta}$  ist der *Maximum-Likelihood-Schätzer* und das ist der Wert für  $\theta$  der die Daten möglichst wahrscheinlich werden lässt.

$$\hat{\theta} := \arg \max_{\theta} WS_{\theta}(S = (s_1, \dots, s_n))$$

Gesucht ist also die Maximalstelle der Likelihood-Funktion  $L_{(s_1, \dots, s_n)}(\theta) := WS_{\theta}(S = (s_1, \dots, s_n))$  .

## Konkret bei Jensen/Hein:

- zwei Sequenzen  $S_1$  und  $S_2$
- Alignment  $A = \{z^1, \dots, z^n\}$ ,  $z^i$  der Form  $M$ ,  $D$  oder  $I$  :

$$P(S_2, A | S_1; \tau) = \tilde{p}(z^n, \varepsilon; \tau) \prod_{i=1}^n \tilde{p}(z^{(i-1)}, z^i; \tau) \tilde{p}_e^c(S_2[L_2^{(i-1)}+1:L_2^i] | S_1[L_1^{(i-1)}+1:L_1^i], z^i; \tau)$$

$L_1$  und  $L_2$  sind die Längen der Sequenzen,  
 $\tilde{p}(\cdot, \cdot; \tau)$  ist die Transitions-Ws aus (3),  
und  $\tilde{p}_e^c$  ist die bedingte Emissions-Ws und  $z^1$  ist ein Zustand der Form  $M$ ,  $D$  oder  $I$ .

In unserem Fall mit mehreren Sequenzen und mult. Alignments können wir diese Formel nutzen (für alle  $j$ ) und damit eine *Likelihood-Funktion* angeben.

$$L_f(\theta) = P(T_{(n+v)}) \prod_{j=1}^{n+v-1} P(S_j, A(a(j), j) | T_{(a(j))}; \tau_j)$$

Gesucht ist hier der Parameter  $\theta$  der den oberen Ausdruck maximiert.

## 7. 2. EM-Algorithmus

- *Expectation-Maximization-Algorithmus*

- um einen unbekanntem Parameter zu schätzen.

- EM-Algorithmus rechnet mit Erwartungswerte (effizienter)

- Parameter  $\theta$  dieses Modells:

stationären Wahrscheinlichkeiten  $\pi(\cdot)$ ,

die Geburtsrate  $\lambda$ ,

die Sterberate  $\mu$ ,

die Evolutionsdistanzen entlang der Äste  $\tau$

zusätzlicher Parameter  $\psi$  für Transversionen (häufiger) und Transitionen (seltener) beeinflusst.

- Schätzung von  $\pi$  aus der empirischen Häufigkeit in beobachteten Sequenzen  $S_1, \dots, S_n$ .

-  $\gamma = \frac{\lambda}{\mu}$  schätzen wir aus der durchschnittlichen Länge der beobachteten Sequenzen.

$$\hat{\pi}(a) = \sum_{j=1}^n \sum_{i=1}^{L_j} 1(S_j[i]=a) / \sum_{j=1}^n L_j,$$

$$\hat{\gamma} = \frac{(\bar{L})}{(1 + \bar{L})} \quad \text{mit} \quad \bar{L} = \frac{1}{n} \sum_{i=1}^n L_i.$$

Wenn nun  $\pi$  und  $\gamma$  konstant sind, können wir die *Maximum-Likelihood-Funktion* schreiben als:

$$L_f(\theta) = \prod_{j=1}^{n+v-1} \{ b(\#, \#; j)^{N(\#, \#; j)} b(\#, -; j)^{N(\#, -; j)} b(-, \#; j)^{N(-, \#; j)} b(-, -; j)^{N(-, -; j)} \\ \times s(\#; j)^{N(\#; j)} s(-; j)^{N(-; j)} \prod_{w_1, w_2} f(w_2 | w_1; j)^{K(w_1, w_2; j)} \} \quad (20)$$

mit  $\theta = (\mu, \psi, \tau_j : j=1, \dots, n+v-1)$  .

$N(\#, \#; j)$  zählt wie oft der  $b(\#, \#; j)$  in dem Alignment  $A(a(j), j)$  vorkommt.

$K(w_1, w_2; j)$  ist die Anzahl der Substitutionen von  $w_1$  zu  $w_2$  entlang des Astes  $j$  .

- Simulation der Erwartungswerte (E-Schritt) der Zählerexponenten gegeben die beobachtbaren Sequenzen unter z.B. dem Parameter  $\theta_1$  .

- Dann finden wir einen neuen (besseren) Parameterwert  $\theta_2$  wenn wir (20) maximieren (M-Schritt).

- Wir verwenden diese iterative Prozedur um (20) zu weiter zu maximieren.

Zuerst finden wir mit  $\phi = (\mu, \psi) : \hat{\tau}_j(\phi)$  ,  $j=1, \dots, n+v-1$  .

- Dann suchen wir einen neuen Wert für  $\phi = (\mu, \psi)$  indem wir  $L_f(\hat{\phi}, \hat{\tau}(\phi) : j=1, \dots, n+v-1)$  maximieren.

Unsere Hoffnung ist das der Parameter  $\hat{\phi}$  gegen  $\tilde{\phi}$  konvergiert und wir iterieren solange bis wir einen guten Wert haben.

Anwendung von EM-Algo auf *3-star-tree* und Baum mit 4 Seq..

(je 30 Iterationen und für jede Iteration 1000 Update-Schritte für Alignments und den Vorgänger).

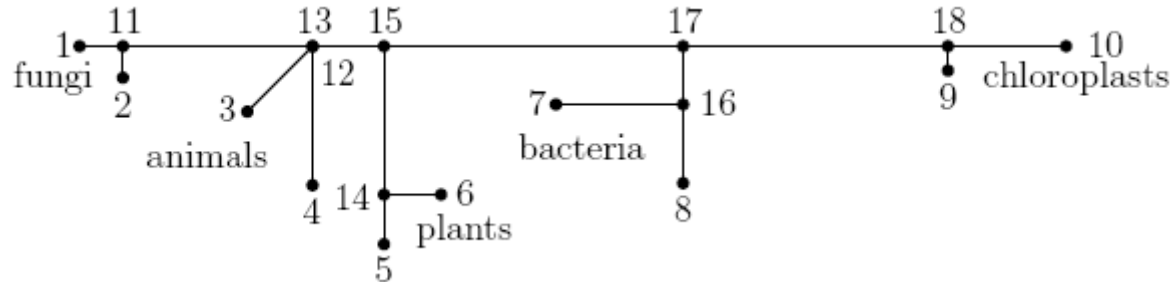
$\psi$  ist hier aber konstant (=0.2), also  $\theta=(\mu, \tau_1, \tau_2, \tau_3)$  bzw.  $\theta=(\mu, \tau_1, \tau_2, \tau_3, \tau_4, \tau_5)$ . Die andere Werte sind:  
 $\pi=(0.2,0.3,0.2,0.3)$ ,  $\mu=0,1$ ,  $\lambda=0,099$

<b>3-seq</b>	$\mu$	$\tau_1$	$\tau_2$	$\tau_3$	$\bar{l}_f$ , $\log(L_f)=l_f$
Start	0.100	0.80	0.80	0.80	19.41
Iteration 5	0.085	1.15	0.87	1.12	-2.65
Iteration 10	0.079	1.27	0.83	1.39	-6.58
Iteration 15	0.078	1.32	0.76	1.48	-4.44
Iteration 20	0.077	1.35	0.68	1.53	-1.82
Iteration 25	0.077	1.38	0.63	1.57	-0.87
Iteration 30	0.078	1.39	0.61	1.59	0.00

<b>4-seq</b>	$\mu$	$\tau_1$	$\tau_2$	$\tau_3$	$\tau_4$	$\tau_5$	$\bar{l}_f$
Start	0.100	0.80	0.80	0.80	0.80	0.80	-1.68
Iteration 5	0.101	0.94	0.79	0.71	0.72	0.75	4.83
Iteration 10	0.102	1.01	0.76	0.68	0.69	0.71	7.40
Iteration 15	0.107	1.09	0.72	0.70	0.67	0.67	3.58
Iteration 20	0.110	1.11	0.66	0.69	0.66	0.67	3.14
Iteration 25	0.113	1.14	0.64	0.70	0.66	0.67	1.34
Iteration 30	0.115	1.15	0.62	0.69	0.65	0.66	0.00

## 8. Beispiel

- 10 *5S-RNA* (Kinder im Baum)
- Vorfahren sind die inneren Knoten
- gemeinsamer Vorfahre hat vermutlich vor 3 Milliarden Jahren existiert
- gesamte Kantenlänge im phylogenetischen Baum ca. 10 Mill. Jahre



Die 10 Sequenzen sind:

1: *Auricularia auricula-judae* ((Speise-)Pilz), 2: *Auricularia edulis* (Pilz), 3: *Caenorhabditis elegans* (ein Fadenwurm), 4: *Gallus gallus* (Haushuhn), 5: *Equisetum arvense* (Ackerschachtelhalm), 6: *Cycad revoluta*, 7: *Bacillus brevis*, 8: *Bacillus firmus*, 9: *Jungermannia subulata* und 10: *Dryopteris acuminata*.



- Man lässt den hier vorgestellten Algorithmus laufen um mehrere Beispielsequenzen von der a posteriori Verteilung der Alignments gegeben die beobachtbaren Sequenzen zu erhalten.
- Hein/Jensen verwendeten die Parameter des EM-Algorithmus und sie simulierten 1000 Updateschritte und berechneten die Häufigkeit mit der eine bestimmte Spalte identisch in allen Stellen ist.
- Die meisten Stellen die auch von CLUSTALW markiert wurden hatten eine a posteriori Wahrscheinlichkeit identisch zu sein von teilweise über 95%. In wenigen Fällen war die Wahrscheinlichkeit unter 80% und in einem Fall war sie 16%.

Solche prozentualen Aussagen kann man mit einem Programm wie CLUSTALW alleine nicht treffen.

## 9. Fazit

- Simulation eines mult. Alignments im Rahmen des TKF-Modells mit Gibbs-Sampler möglich
- Korrelation sehr gering und Zustände werden schnell(er) gewechselt (als bei Holmes/Bruno)
- Gesamtlaufzeit (um Faktor 2 bis 4) besser als bei Holmes/Bruno

Der hier vorgestellte Algorithmus ist auch nicht allein beschränkt auf das TKF-Modell sondern ein allgemeineres Markovkettenmodell wäre denkbar wenn man weitere Parameter einführt.