research reports

no. 425 january 2002

jotun hein
jens ledet jensen
christian n.s. pedersen

recursions for statistical multiple alignment

department of theoretical statistics

university of aarhus

Recursions for Statistical Multiple Alignment

Jotun Hein

Department of Statistics, Oxford University,

The Peter Medawar Building for Pathogen Research,

South Parks Road, Oxford OX1 3SY, England

Jens Ledet Jensen

Department of Theoretical Statistics, Institute of Mathematics,

Ny Munkegade, DK-8000 Aarhus C, Denmark

Christian N.S. Pedersen

Department of Computer Science, Ny Munkegade,

DK-8000 Aarhus C, Denmark

Keywords: backward recursion, emission probability, forward recursion, hidden markov chain, states, transition probability

Abstract

Algorithms are presented that allows the calculation of the probability of a set of sequences related by a binary tree that have evolved according to the Thorne-Kishino-Felsenstein model for a fixed set of parameters. The recursions are based on a Markov chain generating sequences and their alignment at nodes in a tree. Dependent on whether the complete realization of this Markov chain is decomposed into the first transition and the rest of the realization or the last transition and the

first part of the realization, two kinds of recursions are obtained that are computationally similar, but probabilistically different. The running time of the algorithms are $O(\prod_{i=1}^{d} L_i)$, where L_i is the length of the i'th observed sequence and d is the number of sequences - leaves at the binary tree. An alternative recursion is also formulated that only uses a Markov chain involving the internal nodes of a tree.

1 Introduction

Proteins and DNA sequences evolve predominantly by substitutions, insertions and deletions of single characters or strings of these elements, where a character is either a nucleotide or an amino acid. During the last two decades, the analysis of the substitution process has improved considerably, and has increasingly been based on stochastic models. The process of insertions and deletions have not received the same attention and is presently being analysed by optimization techniques for instance maximizing a similarity score as first used by Needleman and Wunch (1970).

In 1991 Thorne, Kishino and Felsenstein proposed a well defined time reversible Markov model for insertions and deletions (denoted more briefly as the *TKF*-model) that allowed a proper statistical analysis for two sequences. Such an analysis can be used to provide maximum likelihood sequence alignments for pairs of sequences, or to estimate the evolutionary distance between two sequences. Recently, an algorithm was presented by Steel and Hein (2000) that allowed statistical alignment of sequences related by a star shaped tree - a tree with one internal node. Hein (2001) formulated an algorithm that calculate the probability of observing a set of sequences related by a given tree in $O((\prod_i L_i)^2)$ time, where L_i is the length of the i'th sequence. This is also the time required by the algorithm in Steel and Hein (2000). The present article accelerates, extends and formalizes the algorithm in Hein (2001). In particular the time requirement for the algorithm presented here is $O(\prod_i L_i)$.

In the *TKF*-model each character along the sequence develops according to the same process and independently of the other characters. During the 'lifespan' of a character (also to be denoted an individual below) the character undergoes changes according to a

reversible substitution process (identical to the site substitution process where insertions and deletions are not allowed). The character is deleted (or dies) after an exponentially distributed waiting time with mean $1/\mu$. Thus the 'death' rate is μ. While being alive a character gives rise to new characters at the rate λ. A 'newborn' character is placed immediately to the right of the character from which it is born, and the character is chosen from the stationary distribution of the substitution process. At the very left of the sequence is a socalled mortal link that never dies and gives rise to new characters at the rate λ. This prevents the process form becoming extinct.

For the *TKF*-model on a tree the defining parameters are the death rate μ and the birth rate λ as described above together with a time parameter τ for each edge of the tree. The time parameter τ defines for how long the process runs along a given edge. A scaling of the times are needed in that multiplying all the τ's by a constant c and dividing μ and λ by c will give the same process. When the process splits into two subprocesses at an internal node the two subprocesses are independent.

The main probabilistic aspects of the *TKF*-model are as follows. At equilibrium the probability that the sequence is l characters long is $(1-\frac{\lambda}{\mu})(\frac{\lambda}{\mu})^l$. Given that the sequence has length l the equilibrium probability of a specific sequence is $\prod_{i=1}^{l}\pi(S(i))$, where $S(i)$ is the i'th character in the sequence and $\pi(\cdot)$ is the equilibrium distribution of the substitution process. The probability that a character survives over a time span τ and during that time gives rise to N newborns is given by (2) below, and the same probability when the character does not survive is given in (3). For the immortal link the probability of N newborns is given by (2) without the $\exp(-\mu\tau)$ term. Thus the probability of two aligned sequences can be written in terms of the stationary probability of sequence 1 together with the product of the appropiate probabilities for the fate of each character in sequence 1 as seen in sequence 2. The latter involves either (2) or (3), the stationary probabilities for the characters of the newborns, and the substitution probability in the case of survival of a character from sequence 1 to sequence 2.

The structure of the probabilities (2) and (3) allow us to write the joint probability of observed sequences at the leaves of a tree together with the alignment and the unobserved sequences at internal nodes of the tree as a Markov chain observed until the process reaches

an absorbing state. The process of observed sequences therefore become a hidden Markov chain. Having obtained this identification we can use traditional methods for obtaining a recursion for the calculation of the probability of the observed sequences. In particular we state two recursions, one corresponding to splitting the process according to the first state of the Markov chain and one corresponding to splitting the process according to the last state of the Markov chain. In section 3.1 a state of the hidden Markov chain describes an element in the alignment for the whole tree, and this gives a recursion of complexity $O(\prod_i L_i)$. In section 3.2 we take a state of the hidden Markov chain to be an element in the alignment of the tree consisting of internal nodes only. This gives a recursion of complexity $O((\prod_i L_i)^2)$, however, this can be reduced to $O(\prod_i L_i)$ and, actually, we obtain a recursion with slightly fewer terms than the one considered in section 3.1.

We start in section 2 by defining the states of our hidden Markov chain and finding the transition probabilities of the Markov chain. This section introduces necessary notation in order to allow for a precise mathematical formulation.

2 Preliminaries

2.1 Notation

We consider a tree with d' internal nodes and d leaves. The internal nodes are numbered from 1 to d' with 1 being the root and where the ancestor $a(i)$ of i is to found in $\{1, 2, \ldots, i-1\}$. The leaves are numbered from $d'+1$ to $d'+d$ with the descendants of the

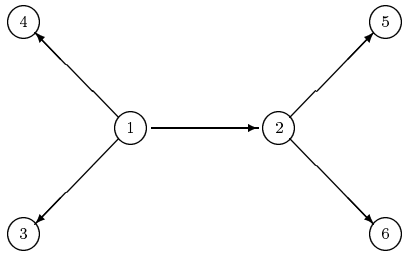

Figure 1: A rooted tree with six nodes

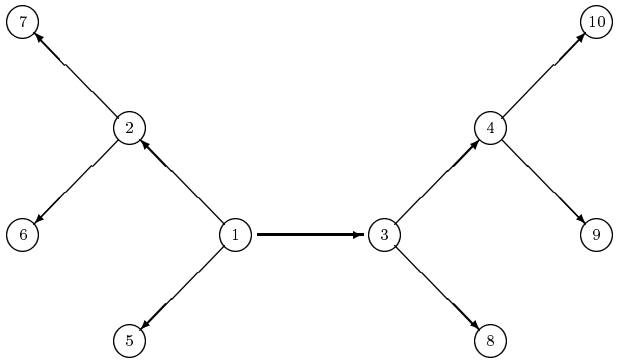

Figure 2: A rooted tree with ten nodes

internal node i being numbered before the descendants of the internal node j for $j > i$. Thus for a tree with two internal nodes and four leaves the numbering can be seen in Figure 1 and for a tree with four internal nodes and 6 leaves the numbering can be seen in Figure 2.

The evolutionary time distance from the ancestor of a node z to the node z is $\tau(z)$.

The observed sequences are S_j, for $j = d' + 1, \ldots, d' + d$, where S_j is the observed sequence at the leaf j. The length of S_j is L_j and the a'th entry of S_j is denoted $S_j(a)$. We write $S_j(a : b)$ for the entries from a to b with a and b included. We let S denote the collection of sequences, and for two d-dimensional vectors u, v indexed by $j = d' + 1, \ldots, d' + d$ and with integer entries we let $S(u : v)$ be the collection of subsequences $S_j(u_j : v_j)$. To compare two d-dimensional vectors u, v we use the notation

$$u > v \text{ if } u_j > v_j \ \forall j, \quad \text{and} \quad u \overset{w}{>} v \text{ if } u_j > v_j \text{ for some } j,$$

with similar definitions for other relations. To shorten the formulae we write for two vectors K, l with $l \geq 0$

$$S[K, l] = S((K - l + 1) : K).$$

Finally, L is the vector with entries L_j.

We write $1(E)$ for the function that is 1 when the expression E is true and 0 otherwise.

2.2 States

In this section we introduce the states of the hidden Markov chain. There is always an initial state I corresponding to the immortal link and an end state E corresponding to the end of the Markov chain. The set of other states is denoted Ξ.

To facilitate the understanding of the choice of states let us first consider a concrete realization of the *TKF*-model for the tree in Figure 1 and write this as a sequence of states. We consider a case with only one individual, or character, at the root 1. The immortal link gives rise to 1 birth at the leaf 4 and to 1 birth at the internal node 2. The latter survives in leaf 5, but not in leaf 6, and at leaf 5 there is also the birth of 1 new individual. The individual at the root 1 survives in node 2, but not in leaf 3 and 4. The individual at node 2 survives in leaf 6, but not in leaf 5 although giving rise to the birth of 2 new characters at leaf 5. At leaf 3 there is also the birth of a new character. At node 2 a new individual is born that survives in both leaf 5 and 6 and gives rise to the birth of a new character at leaf 5. We can depict this realization by columns, where each column consists of a birth and the survival of this individual along the tree.

node	column number								
	1	2	3	4	5	6	7	8	9
1				#					
2		#		#				#	
3							#		
4	#								
5		#	#		#	#		#	#
6				#				#	

(1)

We have ordered the columns in such a way that a birth at a node j comes before a birth at node $i < j$. A birth at a leaf j has # at node j only, whereas a birth at an internal node gives rise to a subtree of #'s. In our general definition of states we will use the same ordering. If we have a state with a birth at a leaf j this implies that the possible births of new individuals at leaves $l > j$ have been completed, or putting it differently, we cannot have a transition from this state to a state with a birth at the leaf $l > j$. To know

the possible transitions from a particular state we need to know the 'circumstances' that produced this state, as in (1) where we need to know that columns 2 and 3 comes from the immortal link, that columns 5-7 comes from column 4, and that column 9 comes from the combination of column 4 and 8. For this reason there will be a 'history' attached to the states.

node	column number										
	1	2	3	4	5	6	7	8	9	10	11
1	#	(#)	(#)	(#)	(#)	#	(#)	(#)	(#)	(#)	(#)
2	#	(#)	(#)	(#)	(#)	−	(−)	(−)	#	(#)	(#)
3	a_3	(a_3)	(a_3)	(a_3)	#	a_3	(a_3)	#			
4	a_4	(a_4)	(a_4)	#		a_4	#				
5	a_5	(a_5)	#						a_5	(a_5)	#
6	a_6	#							a_6	#	
	16	8	4	2	1	4	2	1	4	2	1

Table 1: States of the Markov chain for the tree in Figure 1. Any of the variables a_i can be either # of −. The last row gives the number of states of the indicated form.

For the tree in Figure 1 we have 45 states in Ξ illustrated in the columns of Table 1. In this table a_i is either # or −. The last row gives the number of states of the indicated form, that is, 2^k with k the number of a's in the column. The first 16 states (column 1) each represents the birth of the subtree $\{1, 2\}$ together with the possible survival ($a_i = \#$) or non survival ($a_i = -$) at the leaves. Column 2 represents the possibility that the subtree of column 1 gives rise to a birth of a new individual at the leaf 6. The symbols enclosed by parentheses are included to know what transitions are possible from this state. Column 3 represents the possibility that the subtree of column 1 gives rise to a birth of a new individual at the leaf 5 with the understanding that all births of new individuals at node 6 have been completed. Columns 4 and 5 have similar explanations. Column 6 is the birth of the subtree $\{1\}$ together with the possible survival ($a_i = \#$) or non survival ($a_i = -$) at the leaves attached to this subtree. There are no symbols at the leaves 5 and 6 since

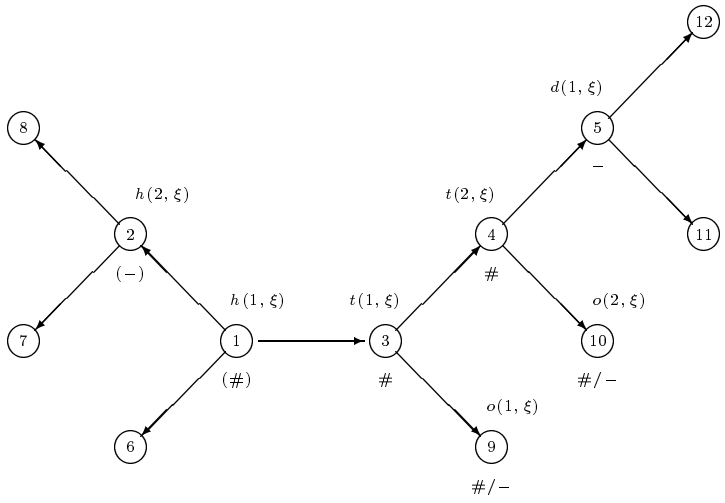

Figure 3: Notation for a subtree ξ starting at the internal node number 3 and with a history at the internal nodes 1 and 2

this subtree cannot give rise to births of new individuals at these nodes. Columns 7 and 8 have explanations as columns 2-5. Finally, column 9 is the birth of the subtree $\{2\}$ where now also is included a 'history' for those internal nodes with a number less than the root of the subtree (in this case this is node 1 and the only history possible is to have # at node 1). The 'history' attached to a state is represented by either # or $-$ enclosed by parentheses and is used to show what transitions are possible from the state, that is, at what positions can a new subtree be born. As for the leaves, where births at a node j are completed before births at the leaf $i < j$ are considered, subtrees starting at an internal node j will be completed before subtrees starting at an internal node $i < j$ are considered.

In the general case we have two kinds of states corresponding either to the birth of a subtree at a set of internal nodes (columns 1,6 and 9 in the above example) or states corresponding to the birth of a new individual at a leaf (columns 2,3,4,5,7,8,10, and 11 in the above example). We first describe a state corresponding to the birth of a subtree at a set of internal nodes. Such a state ξ has three components depicted in Figure 3:

i) A subtree consisting of # at internal nodes $t(1,\xi) < t(2,\xi) < \cdots < t(k_1(\xi),\xi)$. Furthermore, we have the symbol $-$ at the internal nodes $d(1,\xi) < d(2,\xi) < \cdots <$

$d(k_2(\xi),\xi)$ being descendants of $t(1,\xi),\ldots,t(k_1(\xi),\xi)$.

ii) The leaves being descendants of the internal nodes $t(1,\xi),\ldots,t(k_1(\xi),\xi)$ are $o(1,\xi) < o(2,\xi) < \cdots < o(k_3(\xi),\xi)$ and the value of ξ is either $\#$ or $-$ at these nodes.

iii) A 'history' consisting of $(\#)$ and $(-)$ at the internal nodes $1 = h(1,\xi) < h(2,\xi) < \cdots < h(k_4(\xi),\xi) < t(1,\xi)$ giving the part of the tree, from which ξ was born, at nodes less than $t(1,\xi)$

For the tree in Figure 2 the states ξ as described above for the birth of subtrees are illustrated in Table 2.

node													
1	#	#	#	#	#	#	(#)	(#)	(#)	(#)	(#)	(#)	(#)
2	−	#	#	#	−	−	(#)	(−)	(#)	(−)	(#)	(−)	#
3	−	−	#	#	#	#	(#)	(#)	#	#	#	#	−
4			−	#	−	#	#	#	−	−	#	#	
5	a_5	a_5	a_5	a_5	a_5	a_5							
6		a_6	a_6	a_6									a_6
7		a_7	a_7	a_7									a_7
8			a_8	a_8	a_8	a_8			a_8	a_8	a_8	a_8	
9				a_9		a_9	a_9	a_9			a_9	a_9	
10				a_{10}		a_{10}	a_{10}	a_{10}			a_{10}	a_{10}	
	2	8	16	64	4	16	4	4	2	2	8	8	4

Table 2: States of the Markov chain corresponding to the birth of a subtree for the tree in Figure 2. Any of the variables a_i can be either # of −. The last row gives the number of states of the indicated form.

To describe the states corresponding to births of new individuals at leaves we can for each state ξ described above define a state $\gamma(r,\xi)$, $1 \le r \le k_3(\xi)$, with the following components:

i) A history consisting of the history at $h(1,\xi),\ldots,h(k_4(\xi),\xi)$ from ξ together with (#) at $t(1,\xi),\ldots,t(k_1(\xi),\xi)$ and $(-)$ at $d(1,\xi),\ldots,d(k_2(\xi),\xi)$,

ii) a 'history' at leaves $o(1,\xi),\ldots,o(r-1,\xi)$ consisting of the value of ξ surrounded by parenthesis,

iii) the birth # at $o(r,\xi)$.

Two states ξ_1 and ξ_2 corresponding to the birth of a subtree of internal nodes can give rise to the same state $\gamma(r,\xi)$ for a birth at a leaf, but this has no importance when calculating transition probabilities since it is the history attached to $\gamma(r,\xi)$ that is used for this. This situation will arise when there are internal nodes with none of the descendants being leaves.

For the tree in Figure 2 the total number of states in Ξ is 271.

2.3 Transition probabilities

In this section we derive the transition probability $p(x,y)$ of going from state x to state y in our hidden Markov chain. Each transition probability can be written as the product of probabilities of independent events. As an example consider the transition from column 1 to column 3 in Table 1. This transition consists in saying that there will be no births at leaf 6 and there will be at least one birth at leaf 5. Similarly, the transition from column 3 to column 9 consists in saying that there will be no more births at leaf 5, no births at nodes 4 and 3, a birth at the internal node 2, and the possible survival or non-survival at leaves 5 and 6 as given by a_5 and a_6. That we actually have a Markov chain comes from the fact that the probabilities of the events defining a transition depend on the present state only and not on any previous states. This is exactly what we have achieved by the choice of states. To see that this is true we first rewrite the probabilities within the *TKF*-model. The probability that a mortal link survives from time zero to time τ and has N newborns is

$$\exp(-\mu\tau)(1-\lambda\beta)(\lambda\beta)^N, \tag{2}$$

and the probability that a mortal link does not survive and has N newborns is

$$(\mu\beta)^{1(N=0)}\left((1-\exp(-\mu\tau)-\mu\beta)(\lambda\beta)^{N-1}\right)^{1(N>0)}, \tag{3}$$

where

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

For the immortal link the probability of N newborns is given by (2) without the term $\exp(-\mu\tau)$. Let us write the offspring of a mortal link as (J,N), where $J=1$ if the link survives and zero otherwise and $N\geq 0$ is the number of newborns. Let $B_1,B_2,\ldots$ be an infinite sequence of zero-one variables. We then think of N as derived by

$$N=\min\{i:B_i=0\}-1,$$

i.e. N counts the number of B_i's that are 1 until the first appearance of a 0. Then

$$\begin{aligned}P(J,N=n) &= P(J,B_1=\cdots=B_n=1,B_{n+1}=0)\\ &= P(B_{n+1}=0|B_1=\cdots=B_n=1,J)\\ &\quad\times P(B_n=1|B_1=\cdots=B_{n-1}=1,J)\cdots P(B_1=1|J)P(J)\end{aligned}$$

Using (2) and (3) we find the Markov structure

$$\begin{aligned}&P(B_n=1|B_1=\cdots=B_{n-1}=1,J)=\lambda\beta\\ &\text{for } n\geq 1 \text{ if } J=1 \text{ and for } n\geq 2 \text{ if } J=0,\end{aligned} \tag{4}$$

the remaining probabilities of interest being

$$P(J=1)=\exp(-\mu\tau),\ P(B_1=0|J=0)=\frac{\mu\beta}{1-\exp(-\mu\tau)}. \tag{5}$$

For the immortal link the first part of (4) holds true.

The probabilities in (4) and (5) define all the probabilities that enter when calculating the transition probabilities for the states defined in the previous subsection. To state the transition probabilities we write for any node i

$$\begin{aligned}&b(\#,\#;i)=\lambda\beta(i),\quad b(\#,-;i)=1-b(\#,\#;i),\\ &b(-,\#;i)=1-\frac{\mu\beta(i)}{1-\exp(-\mu\tau(i))},\quad b(-,-;i)=1-b(-,\#;i),\\ &s(\#;i)=\exp(-\mu\tau(i)),\quad s(-;i)=1-s(\#;i),\end{aligned}$$

where

$$\beta(i) = \frac{1 - \exp((\lambda - \mu)\tau(i))}{\mu - \lambda \exp((\lambda - \mu)\tau(i))}.$$

Let ξ be a state corresponding to the birth of a subtree at internal nodes. Then for $1 \leq r \leq k_3(\xi)$

$$P(\xi \to \gamma(r,\xi)) = b(\xi(o(r,\xi)), \#; o(r,\xi)) \prod_{j=r+1}^{k_3(\xi)} b(\xi(o(j,\xi)), -; o(j,\xi)), \tag{6}$$

$$P(\gamma(r,\xi) \to \gamma(r,\xi)) = b(\#, \#; o(r,\xi)),$$

and for $r < s$

$$P(\gamma(s,\xi) \to \gamma(r,\xi)) b(\xi(o(r,\xi)), \#; o(r,\xi)) b(\#, -; o(s,\xi)) \prod_{j=r+1}^{s-1} b(\xi(o(j,\xi)), -; o(j,\xi)). \tag{7}$$

We next describe the transition probabilities for entering a state η corresponding to the birth of a subtree at internal nodes. When coming from ξ or $\gamma(r,\xi)$ the start position $t(1,\eta)$ of the new subtree η has to satisfy

i) $t(1,\eta) \in \{t(1,\xi), \ldots, t(k_1(\xi),\xi), d(1,\xi), \ldots, d(k_2(\xi),\xi), h(1,\xi), \ldots, h(k_4(\xi),\xi)\}$.

Furthermore the history attached to η must be the one inherited from ξ:

ii) the history $\{h(1,\eta), \ldots, h(k_4(\eta),\eta)\}$ consists of $\{i | h(i,\xi) < t(1,\eta)\} \cup \{i | t(i,\xi) < t(1,\eta)\} \cup \{i | d(i,\xi) < t(1,\eta)\}$ and the value at these nodes are the values of ξ surrounded by parentheses.

Let $A(\xi,\eta)$ be the set of internal nodes $j > t(1,\eta)$ at which a new subtree could be born when coming from the state ξ. Thus

$$A(\xi,\eta) = \{j | j > t(1,\eta) \text{ and } j = h(i,\xi) \text{ or } j = t(i,\xi) \text{ or } j = d(i,\xi) \text{ for some } i\}.$$

Then

$$\begin{aligned} P(\xi \to \eta) = & \prod_{j=1}^{k_3(\xi)} b(\xi(o(j,\xi)), -; o(j,\xi)) \prod_{j \in A(\xi,\eta)} b(\xi(j), -; j) \\ & \times b(\xi(t(1,\eta)), \#; t(1,\eta))^{1(t(1,\eta)>1)} (\lambda/\mu)^{1(t(1,\eta)=1)} \\ & \times \prod_{j=2}^{k_1(\eta)} s(\#; t(j,\eta)) \prod_{j=1}^{k_2(\eta)} s(-; d(j,\eta)) \prod_{j=1}^{k_3(\eta)} s(\eta(o(j,\eta)); o(j,\eta)), \end{aligned}$$

and for $1 \leq r \leq k_3(\xi)$

$$\begin{aligned} P(\gamma(r,\xi) \to \eta) &= \prod_{j=1}^{r} b(\xi(o(j,\xi)), -; o(j,\xi)) \prod_{j \in A(\xi,\eta)} b(\xi(j), -; j) \\ &\quad \times b(\xi(t(1,\eta)), \#; t(1,\eta))^{1(t(1,\eta)>1)} (\lambda/\mu)^{1(t(1,\eta)=1)} \\ &\quad \times \prod_{j=2}^{k_1(\eta)} s(\#; t(j,\eta)) \prod_{j=1}^{k_2(\eta)} s(-; d(j,\eta)) \prod_{j=1}^{k_3(\eta)} s(\eta(o(j,\eta)); o(j,\eta)). \end{aligned}$$

For a transition to the end state E the two terms (8) and (8) are replaced by

$$\prod_{j=1}^{r} b(\xi(o(j,\xi)), -; o(j,\xi)) \prod_{j \in A} b(\xi(j), -; j)(1 - \lambda/\mu),$$

where A is defined as above with $t(1,\eta) = 1$ and $r = k_3(\xi)$ for the replacement of (8).

Finally, the transition probabilities from the immortal state I can be calculated as if I corresponds to a birth of a subtree at internal nodes with $k_1(I) = d'$, $t(j,I) = j$, and with # at all the leaves.

3 Algorithms

In this section we present two algorithms for computing the probability of the observed sequences S_j, for $j = d'+1, \ldots, d'+d$, being related by the given evolutionary tree. Both algorithms are based on the hidden Markov chain described in the previous section but differ in their choice of states. In the first algorithm the states describe the alignment for both internal nodes and the leaves. The running time is $O(\prod_{j=d'+1}^{d'+d} L_j) = O(L_{\max}^d)$, where $L_{\max}$ is the maximum length of the observed sequences. In the second algorithm the states describe the alignment for internal nodes only. The running time is now $O(L_{\max}^{2d})$, but the algorithm can be rewritten to obtain an $O(L_{\max}^d)$ running time as in the first approach.

3.1 Approach 1: Annotation of internal nodes and leaves

3.1.1 Notation

We consider a Markov process $x_0, x_1, \ldots, x_N$ that starts in the initial state I and stops at a random time $N+1$ in the end state E. Thus $x_0 = I$, $x_i \in \Xi$, for $i = 1, \ldots, N$, and $x_{N+1} = E$. The transition probability going from x to y is $p(x,y)$ as described in

subsection 2.3. A state $\xi \in \Xi$ corresponding to the birth of a subtree at internal nodes emits a letter in those observed sequences S_z for which $z = o(j, \xi)$ for some j and $\xi(z) = \#$. For a state on the form $\gamma(r, \xi)$ the state emits a letter in the sequence $S_{o(r,\xi)}$ only. For any state $x \in \Xi$ we let $l(x)$ be a vector indexed by $j = d' + 1, \ldots, d' + d$ with

$$l_j(x) = \begin{cases} 1 & \text{if } x \text{ emits a letter in sequence } S_j \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

We write $l^i = l(x_i)$ and define

$$L^i = \sum_{r=1}^{i} l^r.$$

With this notation the state x_i emits the letters $S(L^{i-1}+1 : L^{i-1}+l^i)$, where $S_j(L_j^{i-1}+1 : L_j^{i-1}+l_j^i)$ is the empty set if $l_j^i = 0$. The probability that a state x emits the vector of letters s (with the possibility that some of the coordinates of s are equal to the empty set) is $p(s|x)$.

3.1.2 Backward recursion

Summing over the states of the Markov chain we have

$$P(S(1:L)|x_0 = I) = \sum_{n=0}^{\infty} \sum_{x_1,\ldots,x_n \in \Xi : L^n = L} p(x_n, E) \prod_{i=1}^{n} p(x_{i-1}, x_i) p(S[L^i, l^i]|x_i). \tag{9}$$

We will obtain a recursion by separating out the contribution from the first term x_1. Define for an arbitrary vector $K \geq 0$ and state $x_0 \in \Xi$

$$\begin{aligned} F(K|x_0) &= \sum_{n=0}^{\infty} \sum_{x_1,\ldots,x_n \in \Xi : K + L^n = L} p(x_n, E) \prod_{i=1}^{n} p(x_{i-1}, x_i) p(S[K + L^i, l^i]|x_i) \\ &= P(S(K+1:L)|x_0), \end{aligned}$$

that is, the probability that the sequences $S((K+1) : L)$ are produced by the states $x_1, x_2, \ldots$ given that the Markov chain starts in the state x_0. The inner sum is zero if there are no $x_1, \ldots, x_n$ with $K + L^n = L$. Clearly, $P(S(1:L)|x_0 = I) = F(0|I)$. When $K \overset{w}{<} L$ and $K \leq L$ the recursion for $F(K|x_0)$ is, with $\tilde{L}^i = \sum_{r=2}^{n} l(x_i)$,

$$\begin{aligned} F(K|x_0) &= \sum_{n=1}^{\infty} \sum_{x_1 \in \Xi} p(x_0, x_1) p(S[K + l^1, l^1]|x_1) \sum_{x_2,\ldots,x_n \in \Xi : (K + l(x_1)) + \tilde{L}^n = L} p(x_n, E) \\ &\quad \times \prod_{i=2}^{n} p(x_{i-1}, x_i) p(S[K + l^1 + \tilde{L}^i, l^i]|x_i) \\ &= \sum_{z \in \Xi} p(x_0, z) p(S[K + l(z), l(z)]|z) F(K + l(z)|z). \end{aligned} \tag{10}$$

When $K = L$ the recursion is

$$F(L|x_0) = p(x_0, E) + \sum_{z \in \Xi : l(z) = 0} p(x_0, z) F(L|z). \tag{11}$$

3.1.3 Forward recursion

In this subsection we obtain a recursion by separating out the contribution from x_n in (9). Define for an arbitrary vector $K \geq 0$ and state $x \in \Xi$

$$H(K, x) \;=\; \sum_{n=0}^{\infty} \sum_{x_1, \ldots, x_n \in \Xi : L^n = K} \left(\prod_{i=1}^{n} p(x_{i-1}, x_i) p(S[L^i, l^i] | x_i) \right) p(x_n, x), \tag{12}$$

where $x_0 = I$ and the inner sum is zero if there are no $x_1, \ldots, x_n$ with $L^n = K$. We will use below

$$H(K, x|x_0) = 0 \;\text{ if } K_j \overset{w}{<} 0.$$

With this definition (9) becomes with $L \geq 0$ and $L \overset{w}{>} 0$

$$P(S(1:L)|x_0 = I) = \sum_{x \in \Xi} p(x, E) p(S[L, l(x)]|x) H(L - l(x), x), \tag{13}$$

and splitting the sum in (12) we obtain for $K \overset{w}{>} 0$ and $K \geq 0$ the recursion

$$\begin{aligned} H(K, x) \;&=\; \sum_{n=1}^{\infty} \sum_{x_1, \ldots, x_n \in \Xi : L^n = K} \left(\prod_{i=1}^{n} p(x_{i-1}, x_i) p(S[L^i, l^i] | x_i) \right) p(x_n, x) \\ &=\; \sum_{n=1}^{\infty} \sum_{x_n \in \Xi} \sum_{x_1, \ldots, x_{n-1} \in \Xi : L^{n-1} = K - l(x_n)} \left(\prod_{i=1}^{n-1} p(x_{i-1}, x_i) p(S[L^i, l^i] | x_i) \right) \\ &\quad \times p(x_{n-1}, x_n) P(S[K, l^n] | x_n) p(x_n, x) \\ &=\; \sum_{z \in \Xi} H(K - l(z), z) P(S[K, l(z)]|z) p(z, x). \end{aligned} \tag{14}$$

When $K = 0$ the recursion becomes

$$H(0, x) = p(x_0, x) + \sum_{z \in \Xi : l(z) = 0} H(0, z) p(z, x). \tag{15}$$

Computationally there is no difference between the use of $H(K, x)$ or $F(K|x_0)$. However, the latter has an interpretation as a probability thereby making it easier to understand the recursion.

3.1.4 Emission probabilities

For a full description of the *TKF-* model we need a model for the substitution process. If a symbol a survives over a period of time τ and is being changed to b along the way we let $f(b|a;\tau)$ be the probability of the substitution of b for a given survival. The stationary probabilities for this transition matrix are denoted by π.

When a state corresponds to a birth in one of the leaves only, that is, it is of the form $\gamma(r,\xi)$, the emitted vector s has a letter at the node $o(r,\xi)$ only, and the emission probability is simply the stationary probability $\pi(s(o(r,\xi)))$. For a state ξ corresponding to a birth of a subtree at internal nodes $\{t(1,\xi),\ldots,t(k_1(\xi),\xi)\}$ the emitted vector s has letters at those nodes z for which $z = o(j,\xi)$ for some j and $\xi(z) = \#$, that is, those z for which $l_z(\xi) = 1$. If we let $a(j)$ be the ancestor of a node j we can write the emission probability as

$$\begin{aligned} p(s|\xi) &= \sum_{v_{t(1,\xi)}} \pi(v_{t(1,\xi)}) \sum_{v_z : z \in \{t(2,\xi),\ldots,t(k_1(\xi),\xi)\}} \prod_z f(v_z|v_{a(z)};\tau(z)) \\ &\quad \times \prod_{z:l_z(\xi)=1} f(s_z|v_{a(z)};\tau(z)). \end{aligned}$$

3.1.5 Implementation and analysis

Let us briefly discuss how to implement the recursion given by (10) and (11). There is a complication in that there will always be terms on the right hand side of the equations for which $K+l(z) = K$ or $l(z) = 0$. The states ξ for which $l(\xi) = 0$ are characterized by being a birth of a subtree ξ at inner nodes with $\xi(z) = -$ for all $z \in \{o(1,\xi),\ldots,o(k_3(\xi),\xi)\}$. Let us denote this class of states by C. Imagine that for some K the term $F(\tilde{K}|x)$ has been calculated for all $\tilde{K} \overset{w}{>} K$, $\tilde{K} \geq K$. For each $x \in C$ the recursion gives

$$F(K|x) = \sum_{z \in C} p(x,z)F(K|z) + \omega(x) \tag{16}$$

with $\omega(x)$ known. Let Q be the matrix with entries $p(z_1,z_2)$, $z_1, z_2 \in C$. Then since the entries are nonnegative and the sum along a row is less than 1 the matrix $I_C - Q$, where I_C is the identity matrix, is invertible, and the set of linear equations (16) has a unique solution. Having solved this system of equations we can next calculate $F(K|x)$ for $x \notin C$ directly from (10) or (11) when $K = L$.

The boundary conditions for the recursion are $F(K|x) = 0$ when $K \overset{w}{>} L$.

To run the algorithm we need to calculate $F(K|x)$ for any $K \leq L$ and for any $x \in \Xi$. The number of steps needed is therefore of the order

$$N \prod_{i=1}^{d} L_i,$$

where N is the number of elements in the set Ξ.

3.2 Approach 2: Annotation of internal nodes only

3.2.1 Notation

In section 3.1 a state described a column of the alignment for all of the internal nodes and leaves, and a state emitted at most one letter in each of the observed sequences. In this section we will instead let the states describe the internal nodes only which in turn necessitates the emission of arbitrary long subsequences among the observed sequences. This implies an extra sum in the recursions, thus seemingly making the recursions more complicated. However, we can rewrite the recursions, ending up with recursions of the same complexity as before and with less terms than in section 3.1.

More precisely, a state ξ is a birth of a subtree at internal nodes and is characterized by the subtree $t(1,\xi),\ldots,t(k_1(\xi),\xi)$ with #, the descendants of these $d(1,\xi),\ldots,d(k_2(\xi),\xi)$ among the internal nodes, where ξ has the value $-$, and the history at $h(1,\xi),\ldots,h(k_4(\xi),\xi)$. For the tree in Figure 2 the states are now given in the upper part of Table 2. As before $o(1,\xi),\ldots,o(k_3(\xi),\xi)$ are the leaves descending from the subtree $t(1,\xi),\ldots,\ t(k_1(\xi),\xi)$. Instead of (8) we now use

$$l_j(x) = \begin{cases} \geq 0 & j \in \{o(1,\xi),\ldots,o(k_3(\xi),\xi)\} \\ 0 & \text{otherwise.} \end{cases} \tag{17}$$

for the length of the emitted subsequence. We again use $l^i = l(x_i)$ and furthermore take $l^0 \geq 0$ to be the length of the subsequence emitted by the immortal link, and define

$$L^i = \sum_{r=0}^{i} l^r.$$

The emission probability $p(S(u:v)|x)$ is now both the probability of emitting subsequences of length $v_j - u_j$, $j = d'+1, \ldots, d'+d$, and the probability that the emitted symbols are $S_j(u_j : v_j)$. To state this probability we define the two sets

$$\begin{aligned} A_\xi &= \{o(1,\xi), \ldots, o(k_3(\xi),\xi)\}, \\ A_\xi(l) &= \{z \in A_\xi | l_z > 0\}, \end{aligned}$$

and use the notation

$$\begin{aligned} q(z,\#) &= \exp(-\mu\tau(z))(1-\lambda\beta(z)) \\ q(z,-) &= (1-\exp(-\mu\tau(z)) - \mu\beta(z))(1-\lambda\beta(z)). \end{aligned}$$

Furthermore, we let u be the subset of the leaves $A_\xi(l)$ for which we have survival from the ancestral internal node. Then the probability that the state ξ emits the subsequences $S(m+1:m+l)$ is

$$\begin{aligned} P(S(m+1:m+l)|\xi) &= \left(\prod_{z\in A_\xi \setminus A_\xi(l)} \mu\beta(z)\right) \sum_{u \subseteq A_\xi(l)} f(m,u,\xi) \\ &\quad \times \left(\prod_{z\in u} q(z,\#)(\lambda\beta(z))^{l_z-1}\pi(S_z(m_z+2:m_z+l_z))\right) \\ &\quad \times \left(\prod_{z\in A_\xi(l)\setminus u} q(z,-)(\lambda\beta(z))^{l_z-1}\pi(S_z(m_z+1:m_z+l_z))\right) \end{aligned}$$

where $\pi(S_j(a:b)) = \prod_{i=a}^{b}\pi(S_j(i))$. The function $f(m,u,\xi)$ is the probability of the first emitted symbol at the leaves where we have survival from the ancestral internal node. To calculate this function we let $a(z)$ be the ancestral node for the node z. Then

$$\begin{aligned} & f(m,u,\xi) \qquad (18) \\ &= \sum_{v_{t(1,\xi)}} \pi(v_{t(1,\xi)} \sum_{v_z : z\in\{t(2,\xi),\ldots,t(k_1(\xi),\xi)\}} \prod_z f(v_z|v_{a(z)};\tau(z)) \prod_{z\in u} f(S(m_z+1)|v_{a(z)};\tau(z)). \end{aligned}$$

Furthermore,

$$p(S(1:l)|I) = \prod_{d'<z\le d'+d} (1-\lambda\beta(z))(\lambda\beta(z))^{l_z-1}\pi(S_z(1:l_z)).$$

3.2.2 Backward recursion

Instead of (9) we write

$$\begin{aligned} & P(S(1:L)|x_0 = I) \qquad (19) \\ &= \sum_{n=0}^{\infty} \sum_{x_1,\ldots,x_n\in\Xi} \sum_{l^0,l^1,\ldots,l^n : L^n = L} P(S(1:l^0)|I) \left(\prod_{i=1}^{n} p(x_{i-1},x_i)p(S[L^i,l^i]|x_i)\right) p(x_n,E). \end{aligned}$$

The backward recursion is obtained by defining

$$\begin{aligned} F(K|x_0) &= \sum_{n=0}^{\infty} \sum_{x_1,\ldots,x_n \in \Xi} \sum_{l^0,l^1,\ldots,l^n : K+L^n=L} P(S(K+1:K+l^0)|x_0) \\ &\quad \times \left(\prod_{i=1}^{n} p(x_{i-1},x_i) p(S[K+L^i, l^i]|x_i) \right) p(x_n, E) \\ &= P(S(K+1:L)|x_0), \end{aligned}$$

The recursion is

$$\begin{aligned} & F(K|x_0) \qquad (20) \\ &= P(S(K+1:L)|x_0)p(x_0,E) + \sum_{\xi \in \Xi} \sum_{l(\xi)} p(x_0,\xi) p(S[K+l(\xi), l(\xi)]|\xi) F(K+l(\xi)|\xi) \end{aligned}$$

for $K \overset{w}{<} L$ and $K \leq L$ and when $K = L$ the recursion is

$$F(L|x_0) = p(x_0,E) + \sum_{\xi \in \Xi} p(x_0,\xi) P(l(\xi)=0|\xi) F(L|\xi). \qquad (21)$$

3.2.3 Forward recursion

To obtain the forward recursion we write instead of (12)

$$\begin{aligned} & H(K,x) \qquad (22) \\ &= \sum_{n=0}^{\infty} \sum_{x_1,\ldots,x_n \in \Xi} \sum_{l^0,l^1,\ldots,l^n : L^n=K} P(S(1:l^0)|I) \left(\prod_{i=1}^{n} p(x_{i-1},x_i) p(S[L^i, l^i]|x_i) \right) p(x_n, x), \end{aligned}$$

and

$$\begin{aligned} & P(S(1:L)|x_0 = I) \qquad (23) \\ &= P(S(1:L)|I)p(I,E) + \sum_{x \in \Xi} \sum_{l(x)} p(x,E) p(S[L, l(x)]|x) H(L-l(x), x). \end{aligned}$$

The recursion for $H(K,x)$ with $K \overset{w}{>} 0$ and $K \geq 0$ is

$$H(K,x) = P(S(1:K)|I)p(I,x) + \sum_{z \in \Xi} \sum_{l(z)} H(K-l(z), z) P(S[K, l(z)]|z) p(z,x). \qquad (24)$$

When $K = 0$ the recursion becomes

$$H(0,x) = p(I,x) + \sum_{z \in \Xi} H(0,z) p(l(z)=0|z) p(z,x). \qquad (25)$$

The recursion (24) corresponds to the one presented in Hein (2001), where $H(K,x)$ was erroneously referred to as a probability.

3.2.4 Reduction of complexity

For the recursions described in the previous two subsections we need to calculate either $H(K,x)$ or $F(K|x)$ for any value of $K \leq L$. This takes of the order $O(L_{\max}^d)$ steps. Each step here, however, involves the sum over l, see (20) and (24), and therefore requires of the order $O(L_{\max}^d)$ steps. The complexity of the algorithms are therefore of the order $O(L_{\max}^{2d})$ steps. The algorithms are therefore inferior to the algorithms given in Section 3.1. It turns out, though, that we can rewrite the algorithms in such a way that the resulting complexity is $O(L_{\max}^d)$ and that the constant factor is actually slightly smaller here than for the algorithms of Section 3.1.

We start by inserting (18) into the recursion (20),

$$\begin{aligned} F(K|x) &= P(S(K+1:L)|x)p(x,E) \\ &+ \sum_{\xi\in\Xi} p(x,\xi) \sum_{l_z\geq 0, z\in A_\xi; l_z=0, z\notin A_\xi} \left(\prod_{z\in A_\xi\setminus A_\xi(l)} \mu\beta(z) \right) \sum_{u\subseteq A_\xi(l)} f(K,u,\xi) \\ &\times \left(\prod_{z\in u} q(z,\#)(\lambda\beta(z))^{l_z-1}\pi(S_z(K_z+2:K_z+l_z)) \right) \\ &\times \left(\prod_{z\in A_\xi(l)\setminus u} q(z,-)(\lambda\beta(z))^{l_z-1}\pi(S_z(K_z+1:K_z+l_z)) \right) F(K+l|\xi) \end{aligned} \tag{26}$$

where, as before, u is the subset of the leaves $A_\xi(l)$ at which we have survival from the ancestral internal node. Next, we let w be the subset of the leaves $A_\xi \setminus u$ at which there is not survival, but the number of newborns is positive. Then (26) becomes

$$\begin{aligned} F(K|x) &= P(S(K+1:L)|x)p(x,E) \\ &+ \sum_{\xi\in\Xi} p(x,\xi) \sum_{u\subseteq A_\xi} f(K,u,\xi) \sum_{w\subseteq A_\xi\setminus u} \left(\prod_{z\in A_\xi\setminus(u\cup w)} \mu\beta(z) \right) \\ &\times \sum_{l_z\geq 1, z\in(u\cup w); l_z=0, z\notin(u\cup w)} \left(\prod_{z\in u} q(z,\#)(\lambda\beta(z))^{l_z-1}\pi(S_z(K_z+2:K_z+l_z)) \right) \\ &\times \left(\prod_{z\in w} q(z,-)(\lambda\beta(z))^{l_z-1}\pi(S_z(K_z+1:K_z+l_z)) \right) F(K+l|\xi) \end{aligned} \tag{27}$$

Finally, we introduce the subset v of $u\cup w$ at which $l_z \geq 2$. This gives

$$\begin{aligned} F(K|x) &= P(S(K+1:L)|x)p(x,E) \\ &+ \sum_{\xi\in\Xi} p(x,\xi) \sum_{u\subseteq A_\xi} f(K,u,\xi) \sum_{w\subseteq A_\xi\setminus u} \left(\prod_{z\in A_\xi\setminus(u\cup w)} \mu\beta(z) \right) \end{aligned} \tag{28}$$

$$
\begin{aligned}
&\times \sum_{v\subseteq(u\cup w)} \left(\prod_{z\in u} q(z,\#)\right)\left(\prod_{z\in w} q(z,-)\pi(S_z(K_z+1))\right)\\
&\times \sum_{l_z\geq 2, z\in v; l_z=1, z\in(u\cup w)\setminus v; l_z=0, z\notin(u\cup w)}\\
&\times \left(\prod_{z\in v}(\lambda\beta(z))^{l_z-1}\pi(S_z(K_z+2:K_z+l_z))\right) F(K+l|\xi)\\
= \;& P(S(K+1:L|x)p(x,E)\\
&+\sum_{\xi\in\Xi} p(x,\xi)\sum_{u\subseteq A_\xi} f(K,u,\xi)\sum_{w\subseteq A_\xi\setminus u}\left(\prod_{z\in A_\xi\setminus(u\cup w)}\mu\beta(z)\right)\\
&\times \sum_{v\subseteq(u\cup w)} \left(\prod_{z\in u} q(z,\#)\right)\left(\prod_{z\in w} q(z,-)\pi(S_z(K_z+1))\right)\\
&\times \left(\prod_{z\in v}\lambda\beta(z)\right)\sum_{\tilde{l}_z\geq 1, z\in v; \tilde{l}_z=0, z\notin v}\\
&\times \left(\prod_{z\in v}(\lambda\beta(z))^{\tilde{l}_z-1}\pi(S_z((K+1(u\cup w))_z+1:(K+1(u\cup w))_z+\tilde{l}_z))\right)\\
&\times F(K+1(u\cup w)+\tilde{l}|\xi),
\end{aligned}
$$

where

$$
1(u\cup w)_z=\begin{cases} 1 & z\in(u\cup w)\\ 0 & z\notin(u\cup w)\end{cases}
$$

Let us denote the last sum in the (28) by G, that is,

$$
G(M|\xi,v)=\sum_{m_z\geq 1, z\in v; m_z=0, z\notin v} F(M+m|\xi)\prod_{z\in v}(\lambda\beta(z))^{m_z-1}\pi(S_z(M_z+1:M_z+m_z)). \quad (29)
$$

for a nonempty subset v of A_ξ, and

$$
G(M|\xi,\emptyset)=F(M|\xi).
$$

Thus (28) becomes

$$
\begin{aligned}
F(K|x) \;=\; & P(S(K+1:L)|x)p(x,E) \qquad (30)\\
&+\sum_{\xi\in\Xi} p(x,\xi)\sum_{u\subseteq A_\xi} f(K,u,\xi)\sum_{w\subseteq A_\xi\setminus u}\left(\prod_{z\in A_\xi\setminus(u\cup w)}\mu\beta(z)\right)\\
&\times \sum_{v\subseteq(u\cup w)} \left(\prod_{z\in u} q(z,\#)\right)\left(\prod_{z\in w} q(z,-)\pi(S_z(K_z+1))\right)\\
&\times \left(\prod_{z\in v}\lambda\beta(z)\right) G(K+1(u\cup w)|\xi,v).
\end{aligned}
$$

We can obtain a recursion for G by splitting the sum in (29) into

$$\sum_{\tilde{v}\subseteq v} \quad \sum_{m_z\geq 2, z\in\tilde{v}; m_z=1, z\in(v\setminus\tilde{v}); m_z=0, z\notin v}$$

where $\tilde{v}$ can be the empty subset. This gives

$$\begin{aligned} & G(K|\xi, v) \qquad (31) \\ &= \prod_{z\in v} \pi(S_z(M_z+1)) \sum_{\tilde{v}\subseteq v} \prod_{z\in\tilde{v}} (\lambda\beta(z)) \sum_{m_z\geq 2, z\in\tilde{v}; m_z=1, z\in(v\setminus\tilde{v}); m_z=0, z\notin v} G(M+1(v)|\xi, \tilde{v}). \end{aligned}$$

Combining (30) and (31) we have established a recursion involving the functions $F(K|\xi)$ and $G(K|\xi, v)$. For the tree in Figure 2 the recursions of section 3 involves 271 terms whereas the number of terms in this section is 142. For the tree in Figure 1 the numbers are 45 and 24, respectively.

3.2.5 Implementation and analysis

For the recursion in (31) with $v \neq \emptyset$ there is no problem with self reference. For the recursion in (30) the self reference problem is handled as in subsection 3.1.5. Imagine that $F(\tilde{K}|x)$ and $G(\tilde{K}|x, v)$ have been found for all $\tilde{K} \overset{w}{>} K$, $\tilde{K} \geq K$, for all x and for all v. Then (30) takes the form

$$F(K|x) = \sum_{\xi\in\Xi} p(x, \xi) \left(\prod_{z\in A_\xi} \mu\beta(z) \right) F(K|\xi) + \omega(x) \qquad (32)$$

where $\omega(x)$ is known. Defining Q to be the matrix with entries

$$p(x_1, x_2) \left(\prod_{z\in A_{x_2}} \mu\beta(z) \right)$$

for $x_1, x_2 \in \Xi$ we have that $I_\Xi - Q$ is invertible since the entries of Q are nonnegative and the sum along a row is less than 1. Thus the set of linear equations (32) has a unique solution.

The boundary conditions for the recursions are $F(K|x) = 0$ and $G(K|\xi, v) = 0$ for $K \overset{w}{>} L$.

4 Discussion

This paper presents an algorithm that has the same complexity as the traditional nonstatistical multiple alignment algorithm (Sankoff, 1975). The statistical alignment approach to sequence analysis differs relative to the optimization approach in focusing on obtaining the probability of the sequences under the given model, not in obtaining an alignment. Among molecular biologists it is, however, popular to consider the actual alignment and the one chosen is typically the alignment that contributes the most to the probability of the observed sequences. The latter can be calculated by simple modifications of the central recursions of this paper, where a summation operator is substituted by a maximization operator. Several additional problems have to be solved to make the algorithm of this paper useful in real data analysis. Besides actually implementing the algorithm it needs to be coupled to a numerical optimization method to find maximum likelihood estimates of the unspecified parameters, such as branch lengths, substitution parameters and insertion and deletion rates. This method can then be used to analyse up to, say, four sequence of realistic lengths (hundreds of base pairs/amino acids). Elementary computational tricks can extend this to six-seven sequences, and beyond this radically different methods will have to be applied.

From the perspective of a biologist the underlying model for this paper can be criticized. Firstly, the assumption that all insertions-deletions are only one nucleotide/amino acid long does not conform to the biological reality and should be relaxed. Secondly, the assumption that all positions in a sequence evolve according to the same rates is also unrealistic. Formulating models and ways to calculate the relevant probabilities in such models is a major challenge to the field if a statistical approach to alignment is to be of widespread use.

References

Hein, J. (2001). An algorithm for statistical alignment of sequences related by a binary tree. In Altman, R.B., Dunker, A.K., Hunter, L., Lauderdale, K. and Klein, T.E. (eds). Pacific Symposium on Biocomputing. World Scientific. Singapore. pp. 179-

190.

Needleman, S.B. and Wunsch, C.D. (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol. Biol.* **48**, 443-53.

Sankoff, D. (1975). Minimal mutation trees of sequences. *SIAM J. Appl. Math.* **78**, 35-42.

Steel, M. and Hein, J. (2000). Applying the Thorne-Kishino-Felsenstein model to sequence evolution of a star tree. *Appl. Math. Lett.* **14**, 679-84.

Thorne, J.L., Kishino, H. and Felsenstein, J. (1991). An evolutionary model for maximum likelihood alignment of DNA sequences. *J. Mol. Evol.* **33**, 114-124.