PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY B

Phil. Trans. R. Soc. B (2008) **363**, 3913–3919
doi:10.1098/rstb.2008.0170
Published online 7 October 2008

A model of evolution and structure for multiple sequence alignment

Ari Löytynoja* **and Nick Goldman**

EMBL-European Bioinformatics Institute, Hinxton, UK

We have developed a phylogeny-aware progressive alignment method that recognizes insertions and deletions as distinct evolutionary events and thus avoids systematic errors created by traditional alignment methods. We now extend this method to simultaneously model regional heterogeneity and evolution. This novel method can be flexibly adapted to alignment of nucleotide or amino acid sequences evolving under processes that vary over genomic regions and, being fully probabilistic, provides an estimate of regional heterogeneity of the evolutionary process along the alignment and a measure of local reliability of the solution. Furthermore, the evolutionary modelling of substitution process permits adjusting the sensitivity and specificity of the alignment and, if high specificity is aimed at, leaving sequences unaligned when their divergence is beyond a meaningful detection of homology.

Keywords: sequence alignment; insertion–deletion processes; character homology; evolutionary process heterogeneity

1. INTRODUCTION

Sequence alignment aims to match homologous characters, nucleotides or amino acids that are descended from a common ancestor. This is complicated by base substitutions that decrease similarity between sequences over evolutionary time and insertions and deletions that add and remove sequence in different evolutionary lineages. From the end user's point of view, the sequence alignment problem is about placing homologous residues in the same alignment columns and positioning gaps to indicate inserted and deleted sequence.

Depending on the aim of the analysis, sequences in an alignment can be seen as descendants of an ancestral sequence or a set of sequences sharing a common or a related biological function. Hence, multiple sequence alignment methods have traditionally modelled either the hierarchical relationships among the sequences (Hogeweg & Hesper 1984; Thompson *et al.* 1994) or the varying structural and functional constraints along the sequence sites (Eddy 1998; Karplus *et al.* 1998). There have been few attempts to combine the two alternative approaches (e.g. Edgar & Sjölander 2003; Arribas-Gil *et al.* 2007), but so far these methods have been either not suitable for alignment of several sequences and genome-scale analyses or computationally too hard to be biologically realistic (e.g. Satija *et al.* 2008).

We present a method that combines the strengths of tree- and profile-based alignment algorithms and simultaneously describes the evolution and regional heterogeneity, from here on called *sequence structure*, of multiple sequences. Our approach is based on a pairwise alignment model that consists of a moderate number of evolutionary processes, each describing a set of differently evolving sequence sites or a sequence region. Distinct processes are depicted with structure classes, the moves among the structure classes described as a Markov process, and the whole alignment process is described with a two-level Hidden Markov Model (HMM) outputting pairs of aligned characters. The model of a sequence pair is extended to progressive multiple alignment using a modification of the phylogeny-aware algorithm that distinguishes insertions from deletions (Löytynoja & Goldman 2005), a method that can be seen as a greedy 'short cut' towards full evolutionary alignment (e.g. Thorne *et al.* 1991; Hein *et al.* 2003; Holmes 2003).

We have implemented the method described in the alignment program PRANK. Analyses of real data show that the algorithm successfully uses different model states and the posterior probabilities for alternative structure classes in different parts of the alignment well match the known genomic structures of the sequences.

2. MATERIAL AND METHODS

We have implemented our pairwise alignment algorithm for sequences with a structure as an extension of the homogeneous model that distinguishes insertions and deletions and handles insertions in an evolutionarily meaningful way (Löytynoja & Goldman 2005). Similar to the basic homogeneous model, the structure model can be extended to multiple alignments by iteration of pairwise alignment according to a guide tree, though, for clarity, we ignore the correction for pre-existing insertions here and present the algorithm for a standard pairwise alignment with affine gap penalties. General descriptions of HMMs and pair HMMs for sequence alignment are given by Rabiner (1989) and Durbin *et al.* (1998), respectively.

(a) *Model states and state transitions*

The model can be seen as a two-level HMM (figure 1): on the higher level, the HMM consists of start and end states (S and E, respectively) and of two or more structure classes; on the lower level, each structure class h consists of three character-emitting states X_h, Y_h and M_h emitting a character against a

* Author for correspondence (ari@ebi.ac.uk).

One contribution of 17 to a Discussion Meeting Issue 'Statistical and computational challenges in molecular phylogenetics and evolution'.

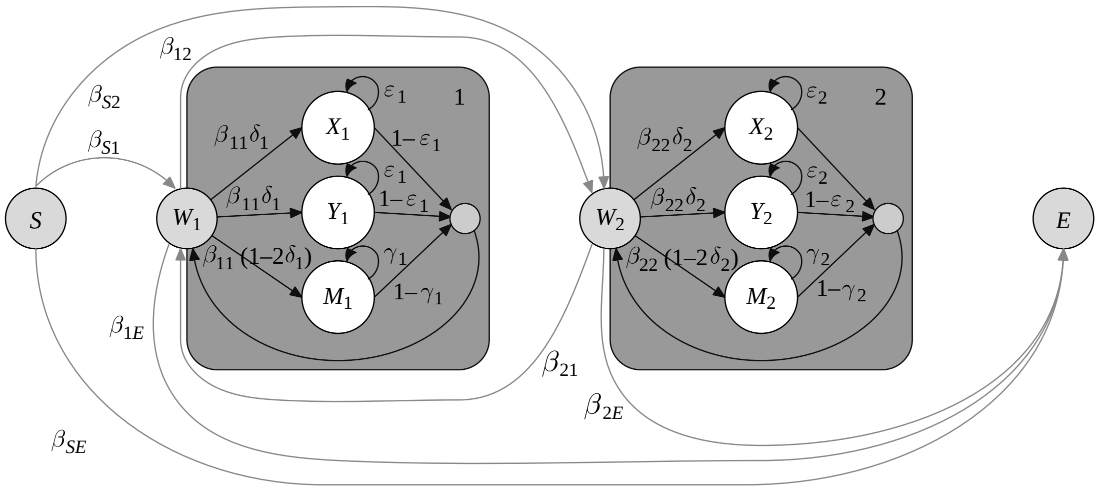

Figure 1. The simplest non-homogeneous alignment model consists of non-emitting start and end states (light grey circles; S and E) and two structure classes (grey boxes; 1 and 2), each describing an evolutionary process of its own. Moves between structure classes and moves within a structure class are denoted with grey and black arrows, respectively. For clarity, the moves from character emitting states (white circles; X_i, Y_i and M_i) back to a non-emitting linker state (light grey; W_i) are drawn via a dummy state (light grey, empty circles).

gap, a gap against a character and two characters matching, respectively, and a silent linker state (W_h) connecting the two levels. Structure classes describe distinct evolutionary processes (such as a fast or slowly evolving site or region, or a codon site), and the moves between the classes define a sequence structure (i.e. regions/sites evolving differently); the moves within a structure class describe the character matching process within a given evolutionary process.

Probabilities β_{gh} for transitions between structure classes g and h (figure 1) are predefined and fixed. Transition within a structure is structure class specific: for structure class h, δ_h is the probability of moving to one of the gap states, and $1-2\delta_h$ to a match state; ε_h and γ_h of staying in a gap state or in a match state, respectively; and $1-\varepsilon_h$ and $1-\gamma_h$ of moving back to the wait state W_h. ε_h and γ_h are fixed ($\gamma_h=0$ makes sites independent), whereas δ_h is jointly defined by a structure-specific insertion–deletion rate r_h and the evolutionary time

$$\delta_h = 1-\mathrm{e}^{-r_h(|\boldsymbol{x}|+|\boldsymbol{y}|)}, \qquad (2.1)$$

where $|\boldsymbol{x}|$ and $|\boldsymbol{y}|$ are the evolutionary distances from the ancestral node to the two child nodes (either extant or reconstructed ancestral sequences) to be aligned (figure 2*a*).

(b) *Character emission and evolutionary match scores*

We consider a pairwise alignment of sequences x and y consisting of characters $x_1 \ldots x_n$ and $y_1 \ldots y_m$. Sequence sites are described with vectors of probabilities, $p_a^h(x_i)$, that the site i in sequence x has character a given that the process is in structure class h. If no sequence structure is imposed, the observed character at a terminal node is given a probability of 1 and others are set to 0; if the observed character is ambiguous, the probability is shared among different characters. For sequences with a known structure (e.g. gene annotation), character probabilities for some structure classes can be set positive and for other classes zero. At internal nodes, $p_a^h(x_i)$ is defined from the pairwise alignment of the two child nodes as a conditional probability of all possible parent characters, given the child sites and all their descendants related by a phylogenetic tree and the process defined for structure class h.

Character emission is defined by an evolutionary substitution model (such as that of Jukes & Cantor (1969) or Hasegawa *et al.* (1985)) and the evolutionary distance between the parent and the child sequences. In state M_h, a conditional probability for a parent character a at ancestral position z_k, given the child character distributions, is defined by

$$L_{z_k}^{(M_h)}(a) = \sum_b s_{ab}^h(x)p_b^h(x_i)\sum_c s_{ac}^h(y)p_c^h(y_j), \qquad (2.2)$$

where $s_{ab}^h(x)$ is the substitution probability between characters a and b given $|\boldsymbol{x}|$, the evolutionary distance between sequence x and its immediate ancestor, and an evolutionary substitution model in structure class h (and similarly for $s_{ac}^h(y)$). As z_k cannot be known, the probabilities are summed over all possible character assignments a at the parent site, and an evolutionary score, $d^h_{x_i,y_j}$, for a match in structure class h is obtained by dividing the probability of observed character emissions by the probability of the process emitting the same output randomly

$$d^h_{x_i,y_j} = \frac{\sum_a q_a^h L_{z_k}^{(M_h)}(a)}{\sum_b q_b^N p_b^h(x_i)\sum_c q_c^N p_c^h(y_j)}, \qquad (2.3)$$

where q_a^h denotes the equilibrium frequency of character a in structure class h, and the superscript N denotes the homogeneous null model (i.e. no structure).

In states X_h and Y_h, the probability depends only on the existing child and is defined by the equilibrium frequencies of the possible characters at the child site and their conditional probabilities, given the subtree below that child. The score for a gap in sequence y, $d^h_{x_i,-}$, is given by

$$d^h_{x_i,-} = \frac{\sum_a q_a^h p_a^h(x_i)}{\sum_a q_a^N p_a^h(x_i)}, \qquad (2.4)$$

and is similarly defined for a gap in sequence x.

The expected number of insertions and deletions observed between two sequences depends on their evolutionary distance (equation (2.1)), whereas their length distribution is not expected to be time dependent. Typically, $d^h_{x_i,-}$ and d^h_{-,y_j} are close to 1, and the alignment is dominated by $d^h_{x_i,y_j}$ and the expected similarity between the sequences given their evolutionary divergence.

The evolutionary modelling of substitution and insertion–deletion processes ensures that the structure classes are correctly scaled for the alignment of sequences that are differently diverged. The character substitution processes are described by instantaneous rate matrices (figure 2*b*), and given the evolutionary distances between the two nodes to align, substitution probability matrices that correctly reflect the expected divergence between the sequences are computed

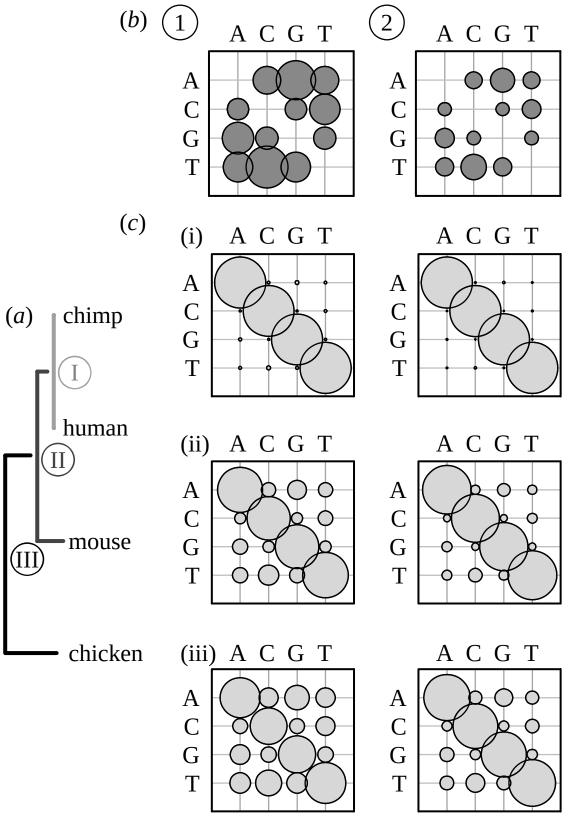

Figure 2. (*a*) A multiple alignment is built from pairwise alignments performed in order of decreasing relatedness ((I), (II) and (III)), each alignment describing the ancestral node for the two nodes (extant or ancestral sequences) to be aligned. (*b*) The substitution process in each structure class is described by an instantaneous rate matrix $\boldsymbol{Q}_i$, here indicated by plots ① and ② showing the rates between different nucleotides as relative sizes of bubbles. In this example, structure classes 1 and 2 model regions of DNA sequence that evolve at the rate that is 150 and 50 per cent of the average rate, respectively. (*c*) For each pairwise alignment, indicated by different shades in the tree (*a*), substitution probability matrices for every structure class are computed from the corresponding matrix $\boldsymbol{Q}_i$. The evolutionary divergence between the sequence/ancestral node pairs to be aligned varies, as shown by the relative length of horizontal bars in the tree, and the alignments contain unequal amounts of information to distinguish the two evolutionary processes. (i) Between human and chimpanzee, both fast and slowly evolving regions (left and right matrix, respectively) are mostly conserved and the diagonal bubbles indicating no change are dominant. In the alignment of (ii) primate ancestor to mouse and (iii) mammalian ancestor to chicken, the fast evolving regions (left matrix) contain greater numbers of substitutions and the off-diagonal bubbles are relatively bigger.

(figure 2*c*). Closely related sequences are expected to be similar just by their recent ancestry and, if the base frequencies in the structure classes are not drastically different, the sequences contain little information to distinguish the regions evolving under different processes.

However, correct homology between sequences, especially on nucleotide level that has low information content, may not be detectable even between moderately diverged sequence pairs (Pollard *et al.* 2004). By modelling the substitution process between the sequences aligned, our approach allows for setting an upper limit for the accepted pairwise distance and thus adjusting the sensitivity and specificity of the alignment. By setting the distance low, the sequence matching becomes stringent and, while it aligns the conserved parts normally, the method leaves the more uncertain diverged sequence regions unmatched.

(c) *Pairwise and progressive multiple alignment*

Given the probabilities for state transitions and character emissions, two sequences are aligned by searching the most probable state path through the HMM. The algorithm finding the path is generally called Viterbi algorithm, and for a pair HMM similar to ours, it resembles the affine gap-cost algorithm of Gotoh (1982) as described by Durbin *et al.* (1998). Our approach differs from the standard algorithm in two respects: (i) the recursions are not only computed over the sequence sites $x_0 \dots x_n$ and $y_0 \dots y_m$, but also over the structure classes $1 \dots r$ such that the state transition probability β_{gh} contributes to the probability of each move, and (ii) we correctly allow for the moves between states X_h and Y_h indicating independent insertion–deletion events at the same or neighbouring sites.

The recursion for pairwise alignment with an affine gap penalty is described in algorithm A.1 (appendix), and its extension to progressive multiple alignment is straightforward (Löytynoja & Goldman 2005). For the latter, we define the probability vector $p_a^h(z_k)$ for parent site z_k as the conditional probabilities of characters a, given the child sites in the pairwise alignment. Given equation (2.2) and defining $L_{z_k}^{(X_h)}(a)$ for the single child x_i as

$$L_{z_k}^{(X_h)}(a) = \sum_b s_{ab}^h(x) p_b^h(x_i), \tag{2.5}$$

(and similarly $L_{z_k}^{(Y_h)}$ for the single child y_j), in an internal node $p_a^h(z_k) = L_{z_k}^{(\cdot)}(a)$, where $\cdot$ denotes M_h, X_h or Y_h depending on which is the most probable character-matching event.

Given all the sites on the alignment path, the ancestral sequence is fully defined and can be aligned with another sequence. Ancestral sequences are technically not treated differently from extant ones.

(d) *Structure class posterior probabilities*

The posterior probability of the process being in a certain state at a given moment is traditionally computed using forward/backward algorithms (Rabiner 1989). We use a similar approach to compute the posterior probabilities for alternative structure classes across the sites of a pairwise alignment.

For sites $1 \dots l$, the probability of observing site z_k, an ancestor for column k in the alignment, that either matches two sites or creates a gap using structure class h is given by

$$L^{(\cdot)}(z_k) = \sum_a q_a^h L_{z_k}^{(\cdot)}(a), \tag{2.6}$$

where $\cdot$ denotes M_h, X_h or Y_h depending on which is the most probable character-matching event and $L_{z_k}^{(\cdot)}(a)$ is given by equations (2.2) and (2.5). Then, forward moves from the site z_{k-1} to a matching site z_k, and to a site z_k that aligns x_i against a gap, in structure class h are defined as

$$f^h(z_k) = \sum_g f^g(z_{k-1}) h_{\cdot M}^{gh} L_h^{(M_h)}(z_k) \quad \text{and}$$

$$f^h(z_k) = \sum_g f^g(z_{k-1}) h_{\cdot X}^{gh} L_h^{(X_h)}(z_k), \tag{2.7}$$

respectively, where $\cdot$ denotes either X, Y or M depending if the previous site was one of the two gaps or a match, respectively (algorithm A.1). This is similarly defined for moves aligning the site y_j against a gap.

For the forward computation, the initialization and termination conditions are defined as in algorithm A.1 except that we denote them $f^h(z_0)$ and $f^h(z_{l+1})$, respectively. For the backward computation, the initialization and termination terms, $b^h(z_{l+1})$ and $b^h(z_0)$, simply change their places and then a backward move from a matching site z_{k+1} to the site z_k in structure class h is defined as

$$b^h(z_k) = \sum_g h^{hg}_{\cdot M} L_g^{(M_g)}(z_{k+1}) b^g(z_{k+1}), \tag{2.8}$$

and is similarly computed for moves from sites aligning a site against a gap.

Given the forward and backward algorithms, the relative probability of being in structure classes h at the site z_k is

$$P^h(z_k) = \frac{f^h(z_k) b^h(z_k)}{f^E}, \tag{2.9}$$

where f^E denotes the full probability of the forward recursion, i.e. the sum of all possible paths through the structure classes.

(e) *Alignment reliability*

Our approach requires normalization of the match and gap scores (equations (2.3) and (2.4)) and does not allow for the computation of an unnormalized probability for a specific solution. However, we can still use forward and backward computation similar to Durbin *et al.* (1998), sum the probabilities of all possible moves in the alignment recursion (*cf. max*( ) in algorithm A.1) and calculate the proportion of the total score supporting the transition in the alignment path.

As the support score is defined for a given alignment solution, we sum the probabilities of transitions that give the same alignment of characters (i.e. moves to either X_h, Y_h or M_h) across all structure classes h. The score is computed for each transition on the Viterbi path in each pairwise alignment and, if the insertion-aware algorithm (Löytynoja & Goldman 2005) is used, the computation in ancestral nodes skips over the pre-existing insertions. The support score can be seen as a measure of the local reliability of a specific alignment solution.

3. APPLICATION

We have implemented the recursions described above in the alignment program PRANK that is downloadable under http://www.ebi.ac.uk/goldman/prank. Our implementation allows for defining different alignment HMMs in text files, such that the method can easily be adapted to the alignment of sequences from any alphabet using models of any complexity. Here, we describe results from alignment of genomic sequences.

(a) *Test data and alignment model*

We aligned the CAPZA2 gene from 15 mammalian species using a simple nucleotide model and a more complex codon model. Genomic sequences for the protein coding region and 500 bases of upstream and downstream flanking sequence were extracted from the multiple alignment of ENCODE target region 1 (The ENCODE Project Consortium 2007), and alignment gaps were removed. The alignment guide trees were based on the ENCODE neutral tree.

A model is described by state transition probabilities β_{gh} and parameters q^h_a, $s^h_{ab}(x)$, δ_h and ε_h for the evolutionary processes in different structure classes. The alignment model FAST/SLOW consists of two classes, F and S, describing fast and slowly evolving sequence sites. The average lengths of fast and slow regions are 200 and 50 bases, respectively, and the gap opening rate and the expected gap length are higher in the former (1/20 subst. versus 1/40 subst.; and 10 bases versus 2 bases, respectively). The transition–transversion ratio is set to 2 and character frequencies q^h_a are defined by the empirical estimate π in both classes, whereas the instantaneous rate matrices (to define the substitution rate matrices $s^h_{ab}(x)$) are based on $\boldsymbol{Q}$: in class S, $\boldsymbol{Q}$ is scaled down giving a substitution rate that is 0.75 of the estimated rate, and in class F, it is scaled up such that, given the equilibrium distribution of the structure classes, the average rate of the model equals the estimated rate.

The alignment model CODON is an extension of the fast–slow model and consists of five structure classes, the two single-character classes S and F and three consecutive nucleotide classes modelling a codon. The character-matching states in the three codon classes are connected and, when two characters are matched in the first class, characters have also to be matched in the second and third class; similarly, the lengths of alignment gaps are always multiples of three and gaps are only possible in phase 0. The average lengths of non-coding and coding sequences are 500 and 100 bases, respectively, and moves to and from a codon are only possible through the S state. The gap-opening rate and the expected gap length in the codon are 1/40 substitutions and 3 bases, respectively. The evolutionary process in states S and F is as described above; for the three codon sites, q^h_a and the instantaneous rate matrices are defined by first computing the parameters for a codon with the selection parameter ω value 0.25 (Nielsen & Yang 1998), and then collapsing the parameters π and $\boldsymbol{Q}$ for the three distinct sites.

In both cases, the q^N_a equals the empirical π.

(b) *Results*

The true alignment for the given genomic region is obviously not known. Instead, we assume that the use of a structure state that more accurately describes the underlying evolutionary process produces improved alignments, and compare the posterior probabilities of being in different structure classes across the sequence sites to the known biological features, namely the 10 protein-coding exons.

The simplistic model FAST/SLOW is able to identify the protein-coding exons along the alignment of human and mouse CAPZA2 sequences (figure 3*a*). However, the posterior probability curve is rather smooth and the accuracy of exon prediction based on any cut-off value would be poor. Also, the model describes single unlinked sites and many conserved non-protein-coding regions obtain high probabilities of being aligned by the slowly evolving class (such as 5′ and 3′ UTRs and sequence immediately flanking the exons). With the model CODON, which adds three classes describing the periodicity of codon, the separation between the protein-coding exons and the conserved splicing signal becomes clearer, though parts of the 5′ and 3′ UTRs still obtain high probabilities for the codon classes (figure 3*b*). The model detects protein-coding regions purely based on the periodicity of substitution rates and gap lengths of multiples of three, and it may be misled by few random substitutions or gaps that happen to be in the right frame.

The performance of our method in the pairwise alignment of human and mouse seems satisfactory but the benefits of structure modelling should be more significant in multiple alignments. First, the alignments of closely related more similar sequences should

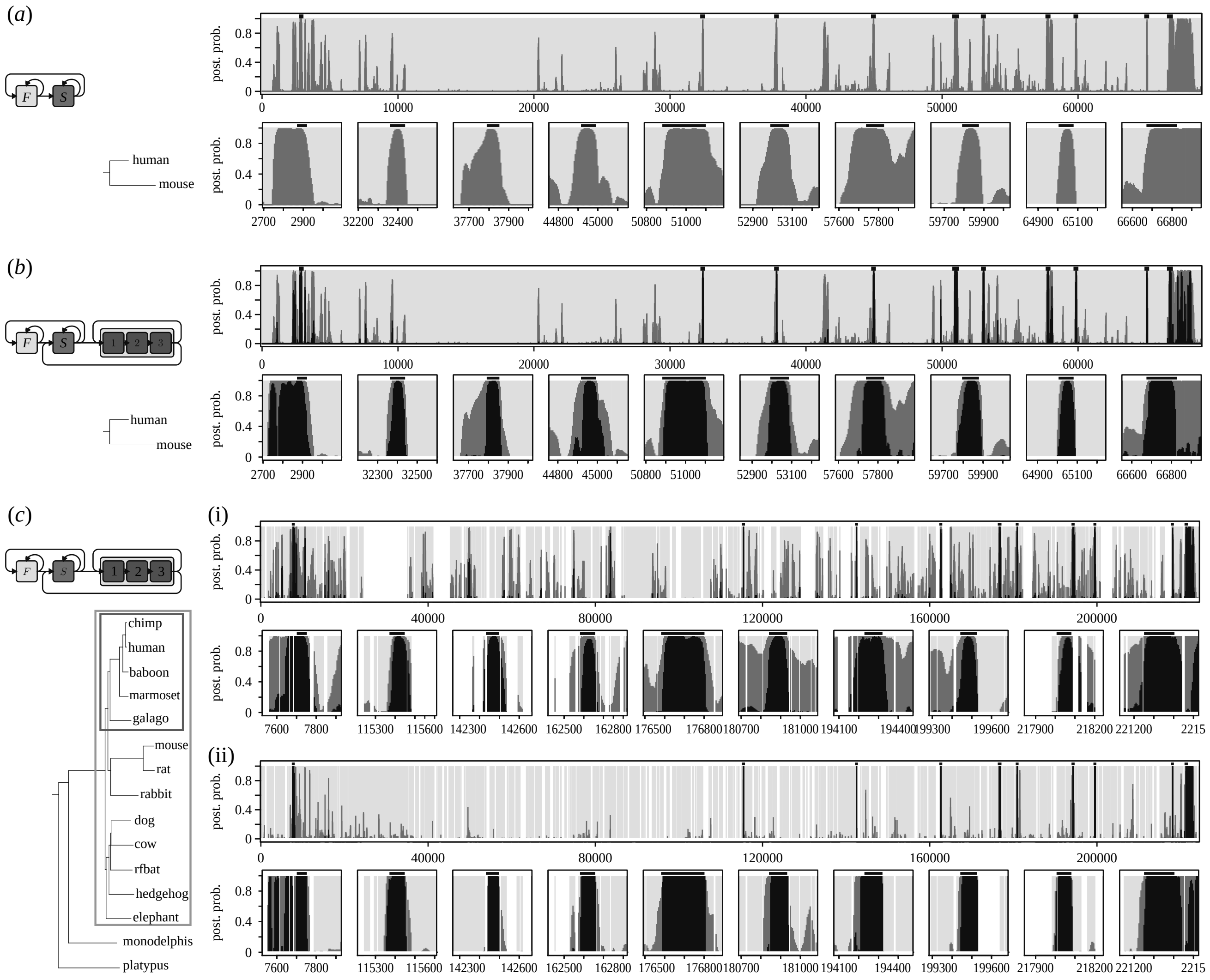

Figure 3. The panels in (*a*)–(*c*) show the posterior probability of different structure classes (top) across the full alignment and (bottom) around the known protein-coding exons. In (*a*) and (*b*), the models FAST/SLOW and CODON are used to align the human and mouse sequences; in (*c*), the model CODON to align fifteen mammalian sequences. Light grey, dark grey and black represent the structure states modelling fast and slowly evolving sites and protein-coding regions, respectively. In (*c*), the addition of more distantly related sequences (dark grey and light grey frames in the tree correspond to panels in (i) and (ii) respectively) increases the evolutionary information and the high posterior probability for the codon states (in black) more accurately matches the locations of known exons. The known locations of the coding exons are marked with black bars (top). The empty gaps in the plots indicate insertions in some other part of the tree.

provide information of the spatial variation of evolutionary processes and help the more difficult alignment of distantly related sequences. Second, multiple sequences provide more information of the sequence structure than two sequences only, and multiple closely related sequences can provide information on features that do not exist in a more distantly related sequence. As the method is progressive, information is generated for each internal node and can be used to study e.g. lineage-specific differences.

As expected, the alignment of very close sequences, such as human and chimpanzee, does not provide information on the sequence structure and, with the exception of long gaps, the posterior probabilities of different structure classes roughly reflect their background frequencies (not shown). On the other hand, the posterior probabilities for the codon classes in the alignment of five primate species rather accurately match the known protein-coding exons and provide an exon annotation comparable with that of the human–mouse pairwise alignment—with the difference that the former would potentially identify novel exons only existing in primates (figure 3*c* (i)).

The addition of the rest of the eutherian mammals (figure 3*c* (ii)) further sharpens the posterior probability curve at the exon boundaries but does not fully resolve the over-prediction of coding sequence in the beginning and end of the gene. Interestingly, the exon seven is consistently predicted to start 50 bases earlier than the true splice site (figure 3*c*). The upstream region is nearly identical all the way until monodelphis but a one-base insertion in mouse and rat suggests non-protein-coding function (not shown). The inclusion of monodelphis sequence would have only a marginal effect on the exon annotation, and the platypus sequence is incomplete and lacks the first exon.

Using alignment anchoring, the pairwise alignments of human and mouse sequences took approximately 600 and 1500 s on an AMD Opteron workstation when using the models FAST/SLOW and CODON, respectively. In multiple alignment, long insertion–deletions and missing data slow down the alignment significantly.

4. DISCUSSION

We have developed an alignment method that allows for incorporating sequence structure information into the alignment process while still taking into account the evolutionary relatedness among the sequences. In contrast to an earlier approach extending the profile alignment (Edgar & Sjölander 2003), we base our method on progressive alignment and model the structural regions/sites with distinct evolutionary processes. Although our approach is not based on a full evolutionary model such as that of Arribas-Gil *et al.* (2007) and Satija *et al.* (2008), it is computationally less demanding and can be easily extended to describe large numbers of processes and biologically realistic sequence structures. The computational complexity of our approach naturally grows with the number of processes described, but our preliminary analyses have shown that in many cases even a moderate number of structure classes is able to capture a significant proportion of the evolutionary signal, such as nucleotide sequences' codon structure and more variable third positions. Also, the complexity reduces significantly when the structure classes are only sparsely connected, and we have successfully tested models with few tens of different classes.

We thank Simon Whelan for his help with modelling multiple evolutionary processes and many ideas and suggestions during this work.

APPENDIX A

Algorithm A.1. An algorithm for pairwise alignment of sequences with structure.

Initialization:

$v_h^{\cdot}(i,-1), v_h^{\cdot}(-1,j)$ are set to 0; $\quad v_h^X(0,0) = v_h^Y(0,0) = \beta_{Bh}\delta_h; \quad v_h^M(0,0) = \beta_{Bh}(1-2\delta_h)$.

Recursion:

$i = 0,\ldots,n, \quad j = 0,\ldots,m, \quad$ except $(0,0)$; $\quad g = 1,\ldots,r, \quad h = 1,\ldots,r$;

$$v_h^X(i,j) = d^h_{x_i,-} \times \max\begin{cases} v_g^X(i-1,j)h^{gh}_{XX} \\ v_g^Y(i-1,j)h^{gh}_{YX} \\ v_g^M(i-1,j)h^{gh}_{MX} \end{cases}$$

$$v_h^Y(i,j) = d^h_{-,y_j} \times \max\begin{cases} v_g^X(i,j-1)h^{gh}_{YX} \\ v_g^Y(i,j-1)h^{gh}_{YY} \\ v_g^M(i,j-1)h^{gh}_{MY} \end{cases}$$

$$v_h^M(i,j) = d^h_{x_i y_j} \times \max\begin{cases} v_g^X(i-1,j-1)h^{gh}_{XM} \\ v_g^Y(i-1,j-1)h^{gh}_{YM} \\ v_g^M(i-1,j-1)h^{gh}_{MM} \end{cases}$$

if $(g = h)$

$$h^{gh}_{XX} = h^{gh}_{YY} = (\varepsilon_g + (1-\varepsilon_g)\beta_{gh}\delta_h)$$
$$h^{gh}_{MM} = \gamma_g + (1-\gamma_g)\beta_{gh}(1-2\delta_h)$$

if $(g \neq h)$

$$h^{gh}_{XX} = h^{gh}_{YY} = (1-\varepsilon_g)\beta_{gh}\delta_h$$
$$h^{gh}_{MM} = (1-\gamma_g)\beta_{gh}(1-2\delta_h)$$

always

$$h^{gh}_{YX} = h^{gh}_{XY} = (1-\varepsilon_g)\beta_{gh}\delta_h$$
$$h^{gh}_{MX} = h^{gh}_{MY} = (1-\gamma_g)\beta_{gh}\delta_h$$
$$h^{gh}_{XM} = h^{gh}_{YM} = (1-\varepsilon_g)\beta_{gh}(1-2\delta_h)$$

Termination:

$$v_h^E = \max(v_h^X(n,m)(1-\varepsilon_h), v_h^Y(n,m)(1-\varepsilon_h), v_h^M(n,m)(1-\gamma_h))\beta_{hE}.$$

REFERENCES

Arribas-Gil, A., Metzler, D. & Plouhinec, J.-L. 2007 Statistical alignment with a sequence evolution model allowing rate heterogeneity along the sequence, *IEEE/ACM Trans. Comput. Biol. Bioinform.* 29 Aug 2007, IEEE Computer Society Digital Library. (doi:10.1109/TCBB.2007.70246)

Durbin, R., Eddy, S., Krogh, A. & Mitchison, G. 1998 *Biological sequence analysis: probabilistic models of proteins and nucleic acids*. Cambridge, UK: Cambridge University Press.

Eddy, S. 1998 Profile hidden Markov models. *Bioinformatics* **14**, 755–763. (doi:10.1093/bioinformatics/14.9.755)

Edgar, R. & Sjölander, K. 2003 SATCHMO: sequence alignment and tree construction using hidden Markov models. *Bioinformatics* **19**, 1404–1411. (doi:10.1093/bioinformatics/btg158)

Gotoh, O. 1982 An improved algorithm for matching biological sequences. *J. Mol. Biol.* **162**, 705–708. (doi:10.1016/0022-2836(82)90398-9)

Hasegawa, M., Kishino, H. & Yano, T. 1985 Dating of the human-ape splitting by a molecular clock of mitochondrial DNA. *J. Mol. Evol.* **22**, 160–174. (doi:10.1007/BF02101694)

Hein, J., Jensen, J. & Pedersen, C. 2003 Recursions for statistical multiple alignment. *Proc. Natl Acad. Sci. USA* **100**, 14 960–14 965. (doi:10.1073/pnas.2036252100)

Hogeweg, P. & Hesper, B. 1984 The alignment of sets of sequences and the construction of phyletic trees: an integrated method. *J. Mol. Evol.* **20**, 175–186. (doi:10.1007/BF02257378)

Holmes, I. 2003 Using guide trees to construct multiple-sequence evolutionary HMMs. *Bioinformatics* **19**, i147–i157. (doi:10.1093/bioinformatics/btg1019)

Jukes, T. & Cantor, C. 1969 *Evolution of protein molecules*, pp. 21–132. New York, NY: Academic Press.

Karplus, K., Barrett, C. & Hughey, R. 1998 Hidden Markov models for detecting remote protein homologies. *Bioinformatics* **14**, 846–856. (doi:10.1093/bioinformatics/14.10.846)

Löytynoja, A. & Goldman, N. 2005 An algorithm for progressive multiple alignment of sequences with insertions. *Proc. Natl Acad. Sci. USA* **102**, 10 557–10 562. (doi:10.1073/pnas.0409137102)

Nielsen, R. & Yang, Z. 1998 Likelihood models for detecting positively selected amino acid sites and applications to the HIV-1 envelope gene. *Genetics* **148**, 929–936.

Pollard, D., Bergman, C., Stoye, J., Celniker, S. & Eisen, M. 2004 Benchmarking tools for the alignment of functional noncoding DNA. *BMC Bioinformatics* **5**, 6. (doi:10.1186/1471-2105-5-6)

Rabiner, L. 1989 A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE* **77**, 257–286. (doi:10.1109/5.18626)

Satija, R., Pachter, L. & Hein, J. 2008 Combining statistical alignment and phylogenetic footprinting to detect regulatory elements. *Bioinformatics* **24**, 1236–1242. (doi:10.1093/bioinformatics/btn104)

The ENCODE Project Consortium 2007 Identification and analysis of functional elements in 1% of the human genome by the ENCODE pilot project. *Nature* **447**, 799–816. (doi:10.1038/nature05874)

Thompson, J., Higgins, D. & Gibson, T. 1994 CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res.* **22**, 4673–4680. (doi:10.1093/nar/22.22.4673)

Thorne, J., Kishino, H. & Felsenstein, J. 1991 An evolutionary model for maximum likelihood alignment of DNA sequences. *J. Mol. Evol.* **33**, 114–124. (doi:10.1007/BF02193625)