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 modeling 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. Saitja 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 different 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’.

3913 This journal is © 2008 The Royal Society
gap, a gap against a character and two characters matching, respectively, and a silent linker state ($W_p$) 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 $\beta_{ab}$ 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 $h$, $\delta_h$ is the probability of moving to one of the gap states, and $1 - 2\delta_h$ to a match state; $\epsilon_g$ and $\gamma_g$ of staying in a gap state or in a match state, respectively; and $1 - \epsilon_g$ and $1 - \gamma_g$ of moving back to the wait state $W_p$, $\epsilon_g$ and $\gamma_g$ are fixed ($\gamma_g = 0$ makes sites independent), whereas $\delta_h$ is jointly defined by a structure-specific insertion–deletion rate $r_h$ and the evolutionary time $\delta_h = 1 - e^{-r_h|x|+|y|}$, (2.1)

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

(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, $p_i^h(x_i)$, 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, $p_i^h(x_i)$ 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 their immediate ancestors, and an evolutionary substitution model in structure class $h$ (and similarly for $\gamma_j(y)$). As $z_h$ cannot be known, the probabilities are summed over all possible character assignments $a$ at the parent site, and an evolutionary score, $d^a_{h,y}$, 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

$$d^a_{h,y} = \frac{\sum_a q_h^L_{z_h}^a(a)}{\sum_a q_h^L_{z_h}^a(a)}.$$

where $q_h^L$ denotes the superscript $h$ 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 $y$, $d^a_{y,-}$, is given by

$$d^a_{y,-} = \frac{\sum_a q_h^{L_{z_h}^a(a)}}{\sum_a q_h^{L_{z_h}^a(a)}},$$

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, $d^a_{h,-}$ and $d^a_{y,-}$ are close to 1, and the alignment is dominated by $d^a_{h,y}$ 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 2b), 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.
Evolution and structure for alignment

A. Löytynoja & N. Goldman

Pairwise and progressive multiple alignment

Given the probability that 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 $x_0 \ldots x_n$ and $y_0 \ldots y_m$, but also over the structure classes $1 \ldots r$ such that the state transition probability $\beta_{bh}$ contributes to the probability of each move, and (ii) we correctly allow for the moves between states $X_h$ and $Y_b$ 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 $p_b(x_0)$ for parent site $x_0$ as the conditional probabilities of characters $a$, given the child sites in the pairwise alignment. Given equation (2.2) and defining $L_{X_1}^b(a)$ for the single child $x_1$ as

$$L_{X_1}^b(a) = \sum_b \sum_b f_a(b) \beta_{bb}$$

and similarly $L_{Y_1}^b(a)$ for the single child $y_1$), in an internal node $p_b(z_k) = L_{X_1}^b(a)$, where $\cdot$ denotes $M_{ih} X_h$ or $Y_b$ 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.

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 $1 \ldots l$, the probability of observing site $z_{lk}$ an ancestor for column $h$ in the alignment, that either matches two sites or creates a gap using structure class $h$ is given by

$$L_{X_1}^h(z_k) = \sum_a \sum_a f_a(z_k) \beta_{hh}$$

where $\cdot$ denotes $M_{ih} X_h$ or $Y_b$ depending on which is the most probable character-matching event and $L_{X_1}^h(a)$ is given by equations (2.2) and (2.5). Then, forward moves from the site $z_{l-1}$ to a matching site $z_k$ and to a site $z_{l+1}$ that aligns $x_i$ against a gap, in structure class $h$ are defined as

$$f_{X_1}^h(z_k) = \sum \sum f_{X_1}^h(z_{l-1}) h_{M_{ih}} M_{ih}(z_k)$$

and

$$f_{X_1}^h(z_{l+1}) = \sum \sum f_{X_1}^h(z_{l-1}) h_{M_{ih}} M_{ih}(z_{l+1})$$

respectively, where $\cdot$ denotes either $X_i$ or $Y_i$ depending on which 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 $f_{X_1}^h(z_k)$ and $f_{X_1}^h(z_{l+1})$, respectively. For the backward computation, the initialization and termination

Phil. Trans. R. Soc. B (2008)
terms, $b^h(z_{k+1})$ 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

$$b^h(z_k) = \sum_a b^h_a(z_{k+1})u^h_a(z_{k+1})b^h(z_{k+1}).$$

(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

$$P^h(z_k) = \frac{f^h(z_k)b^h(z_k)}{f^h(z_k)b^h(z_k)}$$

(2.9)

where $f^h$ 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 $\beta_{gh}$ and parameters $q^h_{1,2,3}(x)$, $\delta_h$ and $\epsilon_f$ 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 $q^h_{1,2,3}$ are defined by the empirical estimate $\pi$ in both classes, whereas the instantaneous rate matrices (to define the substitution rate matrices $s^h_{gh}(x)$ are based on $Q$: in class $S$, $Q$ 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, $q^h_{1,2,3}$ and the instantaneous rate matrices are defined by first computing the parameters for a codon with the selection parameter $\omega$ value 0.25 (Nielsen & Yang 1998), and then collapsing the parameters $\pi$ and $Q$ for the three distinct sites.

In both cases, the $q^h_{1,2,3}$ equals the empirical $\pi$.

(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 3a). 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 3b). 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
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 3c (i)).

The addition of the rest of the eutherian mammals (figure 3c (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 3c). 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.
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:

\[ v_h(i, -1), v_h(-1, j) \text{ are set to } 0; \quad v_h^X(0, 0) = v_h^Y(0, 0) = \beta_{BB}\delta_h; \quad v_h^{M}(0, 0) = \beta_{BB}(1 - 2\delta_h). \]

Recursion:

\[ i = 0, \ldots, n, \quad j = 0, \ldots, m, \quad \text{except } (0, 0); \quad g = 1, \ldots, r, \quad h = 1, \ldots, r; \]

\[
\begin{align*}
v_h^X(i, j) &= d_{h}^{x} \times \max \left\{ v_h^X(i - 1, j)h_{XX}^{bh}, \quad \text{if } (g = h) \right. \\
v_h^Y(i, j) &= d_{h}^{y} \times \max \left\{ v_h^Y(i, j - 1)h_{YY}^{bh}, \quad h_{XX}^{bh} = h_{YY}^{bh} = (\varepsilon_g + (1 - \varepsilon_g)\delta_{gh}\delta_h) \right. \\
v_h^{M}(i, j) &= d_{h}^{m} \times \max \left\{ v_h^X(i - 1, j - 1)h_{MM}^{bh}, \quad h_{XX}^{bh} = h_{YY}^{bh} = (1 - \varepsilon_g)\delta_{gh}\delta_h \right. \\
\end{align*}
\]

Termination:

\[ v_h^b = \max(v_h^b(n, m)(1 - \varepsilon_h), v_h^Y(n, m)(1 - \varepsilon_h), v_h^{M}(n, m)(1 - \gamma_h)\delta_{bh}). \]

REFERENCES


Hogeweg, P. & Hesper, B. 1984 The alignment of sets of sequences and the construction of phyletic trees: an


