Bioinformatics, 34(4), 2018, 576–584
doi: 10.1093/bioinformatics/btx643
Advance Access Publication Date: 12 October 2017
Original Paper

Sequence analysis

Beyond similarity assessment: selecting the optimal model for sequence alignment via the Factorized Asymptotic Bayesian algorithm

Taikai Takeda[1,†] and Michiaki Hamada[1,2,3,4,5,*]

[1]Department of Electrical Engineering and Bioscience, Faculty of Science and Engineering, Waseda University, Tokyo 169–8555, Japan, [2]Computational Bio Big-Data Open Innovation Laboratory (CBBD-OIL), National Institute of Advanced Industrial Science and Technology (AIST), Tokyo 169–8555, Japan, [3]Artificial Intelligence Research Center (AIRC), National Institute of Advanced Industrial Science and Technology (AIST), Tokyo 135–0064, Japan, [4]Institute for Medical-Oriented Structural Biology, Waseda University, Tokyo 162–8480, Japan and [5]Graduate School of Medicine, Nippon Medical School, Tokyo 113–8602, Japan

*To whom correspondence should be addressed.
†Present address: Indeed, Recruit Holdings Co., Ltd, Tokyo, Japan
Associate Editor: John Hancock

Received on May 19, 2017; revised on October 7, 2017; editorial decision on October 9, 2017; accepted on October 10, 2017

Abstract

Motivation: Pair Hidden Markov Models (PHMMs) are probabilistic models used for pairwise sequence alignment, a quintessential problem in bioinformatics. PHMMs include three types of hidden states: match, insertion and deletion. Most previous studies have used one or two hidden states for each PHMM state type. However, few studies have examined the number of states suitable for representing sequence data or improving alignment accuracy.
Results: We developed a novel method to select superior models (including the number of hidden states) for PHMM. Our method selects models with the highest posterior probability using Factorized Information Criterion, which is widely utilized in model selection for probabilistic models with hidden variables. Our simulations indicated that this method has excellent model selection capabilities with slightly improved alignment accuracy. We applied our method to DNA datasets from 5 and 28 species, ultimately selecting more complex models than those used in previous studies.
Availability and implementation: The software is available at https://github.com/bigsea-t/fab-phmm.
Contact: mhamada@waseda.jp
Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

The alignment of biological sequences (e.g. DNA, RNA and proteins) is one of the most classical and important problems in the field of bioinformatics. Sequence alignment permits the assessment of the functional relationships among biological sequences by quantifying sequence similarity. Since similar nucleotides or amino acids sequences are often functionally related, the development of quantitative evaluations of sequence similarity has been of great interest. This high demand for similarity evaluations has driven the development of a variety of alignment programs (Altschul *et al.*, 1990; Frith *et al.*, 2010; Thompson *et al.*, 1994). Moreover, sequence alignments are essential for analyzing the huge amounts of sequence data produced by high-throughput sequencers in computational tasks such as mapping read sequences onto reference genomes (Hamada *et al.*, 2017; Li and Homer, 2010).

For this alignment task, probabilistic approaches are widely recognized. These probabilistic approaches include Pair Hidden Markov Models (PHMMs) (Durbin *et al.*, 1998), which handle indels and substitutions that occur throughout molecular evolution by using sequentially dependent unobserved hidden states, specifically the match, insertion and deletion states as well as their corresponding probabilistic symbol emissions (cf. Fig. 1).

There have been several attempts to construct slightly more complex PHMMs. Bradley *et al.* (2009), Lunter *et al.* (2008) and Paten *et al.* (2008) used PHMMs with two insertion and two deletion states, and Cartwright (2009) proposed a general version of PHMMs that employs a zeta power-law model of indel lengths. Additionally, a few generalizations of PHMMs have been proposed (such as Pachter *et al.*, 2002), which introduced generalized PHMMs for DNA–DNA, DNA–cDNA and DNA–protein alignments. However, to the best of our knowledge, no previous study has focused on determining the suitable number of states for representing the biological models that describe sequence evolution or for achieving better alignment accuracy.

Bayesian model selection provides a sophisticated approach for selecting the best model by maximizing model evidence. In this, some parameters are marginalized out, and so a preference for simpler models is inherent to the method. When the model prior is uniform, maximizing model evidence is equivalent to maximizing the posterior probability of model given data, so we can choose the model with the largest posterior by maximizing the model evidence.

The well-known difficulty of Bayesian model selection is that the model evidence is analytically intractable in general, including for PHMMs. Markov Chain Monte Carlo (MCMC) (Hastings, 1970) and Variational Inference (VI) (Beal, 2003; Blei *et al.*, 2016; Jordan *et al.*, 1999) enable approximation of the difficult-to-compute model evidence, but both approaches have a drawback: high computational cost. In contrast, the Factorized Asymptotic Bayesian (FAB) algorithm (Fujimaki and Hayashi, 2012; Fujimaki and Morinaga, 2012; Hayashi *et al.*, 2015) is a promising alternative model selection technique based on the Factorized Information Criterion (FIC). One advantage of the FAB algorithm is its simultaneous optimization of the model structure and the parameters, which makes the FAB algorithm more scalable than VI and MCMC. The advantages are further discussed in Section 2.2.

The contributions of this study are summarized as follows:

1. We developed a novel FIC-based model selection algorithm for PHMMs and demonstrate the reasonably good accuracy in model selection using a synthetic dataset. To the best of our knowledge, this is the first attempt in the literature to apply a model selection method to PHMMs.
2. The model selection method slightly improved evaluation metrics on the same synthetic dataset.
3. We conducted experiments on real DNA sequences and found that our method selects a more complex probabilistic structure than the ones that have been traditionally used for pairwise alignment of these species.

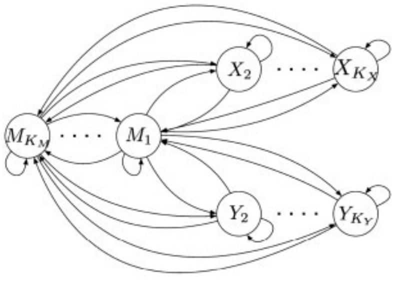

Fig. 1. Transition diagram of hidden states in a PHMM. The match states, which emit a pair of characters ($M_1 \dots M_{K_M}$), are connected to all the other states, whereas the X- and Y-insertion states, which emit a pair of a character and a gap symbol '-' ($X_1 \dots X_{K_X}$ and $Y_1 \dots Y_{K_Y}$), are only connected to the match states

2 Materials and methods

PHMMs are a type of probabilistic generative model for sequence alignment (Durbin *et al.*, 1998) with three types of hidden states: a match-type state M, an X-insertion-type state X and a Y-insertion-type state Y (Fig. 1). The insertion states model the molecular evolution of indels, and the emission probability of the match states characterizes the substitution rates.

In this study, we employed the FAB algorithm (Fujimaki and Morinaga, 2012; Hayashi *et al.*, 2015) to select the best model structure for a PHMM. The FAB algorithm is an information criterion-based technique that enables the simultaneous optimization of both the parameters and the model structure. The properties of the FAB algorithm are explained in Section 2.2 in more detail. Note that we modified the standard formalization of PHMM (e.g. Durbin *et al.*, 1998) because it does not use hidden variables explicitly, which is inappropriate for the FAB algorithm. In this section, we introduce our formalization of the PHMM with explicit hidden variables (Section 2.1) and then develop a proposed model selection method using the FAB algorithm (Section 2.2).

In the following, we denote the number of match-type states, X-insertion-type states and Y-insertion-type states as K_M, K_X and K_Y, respectively. Additionally, K represents the total number of hidden states, that is $K = K_M + K_X + K_Y$. Formally, we regard model selection as selecting the number of hidden states (K_M, K_X, K_Y).

2.1 Pair Hidden Markov Model

Let observed sequences be $\boldsymbol{x} = \{\boldsymbol{x}^n\}_{n\in[1,N]}$ and $\boldsymbol{y} = \{\boldsymbol{y}^n\}_{n\in[1,N]}$, where N is the number of sequence pairs. The nth sequences are $\boldsymbol{x}^n = \{x_t^n\}_{t\in[1,T_X^n]}$ and $\boldsymbol{y}^n = \{y_u^n\}_{u\in[1,T_Y^n]}$, where T_X^n and T_Y^n are the lengths of $\boldsymbol{x}^n$ and $\boldsymbol{y}^n$, respectively. We abbreviate all the observed sequences as $\boldsymbol{X} = \{\boldsymbol{x}, \boldsymbol{y}\}$. Unlike normal HMMs, PHMMs have hidden variables $\boldsymbol{Z} = \{\boldsymbol{z}^n\}_{n\in[1,N]}$, which are *two-dimensional* and the nth of which is $\boldsymbol{z}^n = \{z_{tu}^n\}_{t\in[0,T_X^n],u\in[0,T_Y^n]}$ (Fig. 2). Note that these two-dimensional hidden variables are not a common formalization and include a *zero-state*, introduced below. The value $z_{t,u}^n$ corresponds to the hidden state where, for match states, x_t^n and y_u^n are matched. For insertion states, z_{tu}^n represents that x_t corresponds to the gap '–' and that the last used symbol in $\boldsymbol{y}^n$ is y_u^n and vice versa for Y-insertion states. The hidden state $z_{tu}^n = \{z_{t,u,k}^n\}_{k\in[1,K]}$ is a 1-of-K representation, but slightly modified to allow a *zero-state*, where $z_{t,u,k}^n$ for all k is zero and does not emit any symbols from that variable (an example is shown in Fig. 3). This is because of the unique characteristics of PHMMs (in comparison with conventional HMMs); only a subset of the hidden variables emit symbols, that is only the hidden variables corresponding to aligned positions emit symbols (Fig. 3). The set $\Pi = \{\boldsymbol{\alpha}, \boldsymbol{\beta}, \phi\}$ is a parameter set, where $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ and ϕ represent the initial probability, transition probability and emission probability parameters, respectively. Also, each hidden state k corresponds to one of the state types $\{M, X, Y\}$, which is given by a function $\mathcal{S}$ where $\mathcal{S}(k) \in \{M, X, Y\}$.

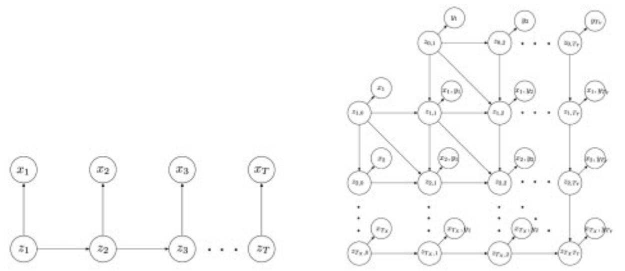

Fig. 2. Graphical model representation of (**a**) HMMs and (**b**) PHMMs. The hidden states, denoted by z_t, are one-dimensional in the normal HMM, whereas they, denoted by $(z_{t,u})$, are two-dimensional in the PHMM. In the PHMM, a pair of symbol emissions (x_t, y_u) is an emission from the hidden state $z_{t,u}$, describing a pair of (aligned) nucleotides in the case of DNA alignments, for example

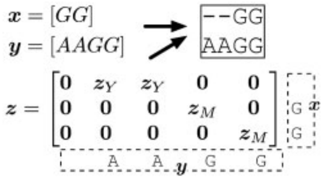
$\boldsymbol{x} = [GG]$
$\boldsymbol{y} = [AAGG]$
--GG
AAGG
$\boldsymbol{z} = \begin{bmatrix} \mathbf{0} & \boldsymbol{z}_Y & \boldsymbol{z}_Y & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{z}_M & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{z}_M \end{bmatrix}$
G
G
$\boldsymbol{x}$
A
A
$\boldsymbol{y}$
G
G

Fig. 3. An alignment example for a pair of DNA sequences *x* and *y* as well as the corresponding *two-dimensional* hidden states, where $(K_M, K_X, K_Y) = (1, 1, 1)$, illustrating how hidden states are encoded. $0 = (0, 0, 0)$ is the zero-state that does not emit any symbols, and $z_M = (1, 0, 0)$, $z_X = (0, 1, 0)$ and $z_Y = (0, 0, 1)$ are the 1-of-*K* coding corresponding to the *M*-type, *X*-type and *Y*-type states, respectively. This hidden state encoding allows us to generate the alignment $\frac{--GG}{AAGG}$ from the sequence pair *GG* and *AAGG*

Now we can write the complete log-likelihood of PHMM as

$$\ln p(\boldsymbol{X}, \boldsymbol{Z} \,|\, \Pi) = \sum_{n=1}^{N} \Bigg[\ln p(z_{\text{in}}^{n} \,|\, \boldsymbol{\alpha}) + \sum_{t=0}^{\mathrm{T}_{\mathrm{X}}^{\mathrm{n}}} \sum_{u=0}^{\mathrm{T}_{\mathrm{Y}}^{\mathrm{n}}} \Big(\ln \mathrm{p}(z_{\mathrm{tu}}^{\mathrm{n}} \,|\, \mathrm{pa}(z_{\mathrm{tu}}^{\mathrm{n}}), \boldsymbol{\beta}) + \ln \mathrm{p}(\mathrm{x}_{\mathrm{t}}^{\mathrm{n}}, \mathrm{y}_{\mathrm{u}}^{\mathrm{n}} \,|\, z_{\mathrm{tu}}^{\mathrm{n}}, \phi) \Big) \Bigg] \tag{1}$$

where z_{in}^{n} is a set of hidden variables corresponding to the initial states. The initial hidden variable varies with the type of hidden state because each hidden state's type uses a different number of original sequences; an M-type state uses both x_1 and y_1, whereas X-type and Y-type states each use one of them. For this reason, the initial hidden variable of an M-type state is $z_{1,1}$, whereas it is $z_{1,0}$ and $z_{0,1}$ for X-type and Y-type states, respectively. For the transition probability, we use $pa(z_{tu}^{n}) = \{z_{t'u'k}\}_{k \in [1,K]}$ as a set of (previous) hidden variables from which this can transit to $z_{t,u}$. We further denote the emission probability as a categorical distribution using new variables $\{\psi_{tuk}^{n}\}$ as

$$p(x_t, y_u \,|\, z_{tuk}^{n} = 1) = \psi_{tuk}^{n} = \begin{cases} \phi_k(x_t, y_u) & \text{if} \quad \mathcal{S}(k) = M \\ \phi_k(x_t, -) & \text{if} \quad \mathcal{S}(k) = X \\ \phi_k(-, y_u) & \text{if} \quad \mathcal{S}(k) = Y \end{cases} \tag{2}$$

where ϕ_k represents the categorical emission probability of the kth hidden state. Note that the X-type and Y-type states emit the gap '–' instead of the normal symbols (e.g. A, T, G or C in the case of DNA alignments). Thus, the dimensionality of the parameter of the emission probability differs with the type of hidden states, namely, $L^2 - 1$ for the match states and $L - 1$ for the insertion states, where L is the number of symbols ($L = 4$ in the case of DNA sequences). Using this notation, we can rewrite the complete likelihood in an explicit form.

$$\begin{aligned} \ln p(\boldsymbol{X}, \boldsymbol{Z} \,|\, \Pi) &= \sum_{n=1}^{N} \Bigg[\ln p(z_{\text{in}}^{n} \,|\, \boldsymbol{\alpha}) \\ &+ \sum_{t=0}^{\mathrm{T_X}} \sum_{u=0}^{\mathrm{T_Y}} \sum_{k=1}^{\mathrm{K}} \Big(\mathrm{pa}(z_{\mathrm{tu}}^{\mathrm{n}})_{\mathrm{k}} \ln \mathrm{p_k}(z_{\mathrm{tu}}^{\mathrm{n}} \,|\, \boldsymbol{\beta}_{\mathrm{k}}) + \mathrm{z_{tuk}^{n}} \ln \mathrm{p}(\mathrm{x_t^n}, \mathrm{y_u^n} \,|\, z_{\mathrm{tu}}^{\mathrm{n}}, \phi_{\mathrm{k}}) \Big) \Bigg] \\ &= \sum_{\mathrm{n=1}}^{\mathrm{N}} \sum_{\mathrm{k=1}}^{\mathrm{K}} \Bigg[\mathrm{z^n_{d_{xk}, d_{yk}, k}} \ln \alpha_{\mathrm{k}} \\ &+ \sum_{\mathrm{t=0}}^{\mathrm{T_X^n}} \sum_{\mathrm{u=0}}^{\mathrm{T_Y^n}} \Big(\sum_{\mathrm{l=1}}^{\mathrm{K}} \mathrm{z^n_{(t-d_{xk}),(u-d_{yk}),k}} \mathrm{z^n_{tul}} \ln \beta_{\mathrm{kl}} + \mathrm{z^n_{tuk}} \ln \psi^{\mathrm{n}}_{\mathrm{tuk}} \Big) \Bigg] \end{aligned} \tag{3}$$

where $p_k(z_{tu} \,|\, \boldsymbol{\beta}_{\mathrm{k}}) = \prod_{l=1}^{K} p(z_{tul} = 1 \,|\, z_{(t-d_{xl}),(u-d_{yl}),k} = 1)^{z_{tul}}$ and (d_{xk}, d_{yk}) is a transition direction defined as

$$(d_{xk}, d_{yk}) = \begin{cases} (1, 1) & \text{if} \quad \mathcal{S}(k) = M \\ (1, 0) & \text{if} \quad \mathcal{S}(k) = X \\ (0, 1) & \text{if} \quad \mathcal{S}(k) = Y. \end{cases}$$

Again, it should be noted that the representation in Equation (3) is essential for a derivation of our model selection algorithm in the following section.

2.2 PHMM model selection algorithm: FAB-PHMM

We formalize the model selection problem for PHMM as a maximization of the model evidence.

$$\mathcal{M}^{*} = \underset{\mathcal{M}}{\arg\max} \ln p(\boldsymbol{X} \,|\, \mathcal{M}) \tag{4}$$

where the evidence is given by $p(\boldsymbol{X} \,|\, \mathcal{M}) = \int \sum_{Z} p(\boldsymbol{X}, \boldsymbol{Z}, \Pi \,|\, \mathcal{M}) d\Pi$. Note that the model size $\mathcal{M} = \{K_M, K_X, K_Y\}$ is parameterized by the number of hidden states of each state type. However, the model evidence is difficult to compute; thus, we generally need approximations. In this study, we use FIC as an asymptotically accurate approximation.

FIC has following three appealing properties:

1. **Asymptotic equivalence to marginal likelihood.** Although Bayesian Information Criterion (BIC) is a widely used and simple information criterion, it lacks theoretical justification because of the non-regularity of the latent variable models (Watanabe, 2009). PHMM is not an exception to this, so the BIC's approximation is invalid for PHMMs. Unlike BIC, FIC is consistent with the marginal likelihood for latent variable models. Practically speaking, Fujimaki and Hayashi (2012) empirically showed that BIC-HMM tends to choose overly complicated models, while FIC-HMM chooses optimal models more often.
2. **Simultaneous optimization of model and parameters.** VI is closely related to FIC. Both of them perform similar approximation using variational distribution. One advantage of FIC is that it can optimize parameters and models simultaneously. This makes FIC-based optimization computationally more efficient.
3. **Prior free.** Unlike VI, FIC does not require prior distributions because it treats priors as $\mathcal{O}(1)$. Thus, FIC is hyper-parameter tuning free and easier to optimize.

In the following, we will start with the derivation of FIC_{PHMM} (Section 2.2.1), then take a lower bound to derive *FICLB* (Section

2.2.2) for optimization via expectation maximization (EM). We iteratively optimize the target function *FICLB* with respect to a variational distribution q and parameters Π in the E step (Section 2.2.3) and M step (Section 2.2.4), respectively. The model $\mathcal{M}$ is tuned via *model pruning* (Section 2.2.5).

2.2.1 Factorized information criterion

Let Ξ be a set of local (component-dependent) parameters and Θ be a set of global (component-independent) parameters. Additionally, we denote all the parameters as $\Pi = \{\Xi, \Theta\}$ (in the case of PHMM, local parameters $\Xi = \{\boldsymbol{\beta}, \phi\}$ and global parameters $\Theta = \{\alpha\}$). Hayashi *et al.* (2015) have shown that the model evidence can be approximated as an asymptotically accurate information criterion, FIC, which can be expressed as

$$FIC(\mathcal{M}) = \mathbb{E}_{q^{**}}\left[\ln p(\boldsymbol{X}, \boldsymbol{Z} \mid \bar{\Pi}, \mathcal{M}) - \frac{1}{2}\ln \mid F_{\bar{\Xi}} \mid\right] - \frac{1}{2}D_{\Pi} \ \ln N + \mathcal{H}(q^{**}).$$

where D_{Π} is the number of free parameters in Π, $\bar{\Pi} = \{\bar{\Xi}, \bar{\Theta}\}$ is a maximum joint likelihood estimators (MJLEs), $\boldsymbol{F}_{\bar{\Xi}}$ is the Hessian matrix of $-\ln p(\boldsymbol{X}, \boldsymbol{Z} \mid \Pi)$ with respect to $\bar{\Xi}$, $q^{**}(\boldsymbol{Z}) = p(\boldsymbol{Z} \mid \boldsymbol{X}, \mathcal{M})$ is the marginal posterior and $\mathcal{H}(q^{**})$ is the entropy of q^{**}. In FIC, the penalty term is given by the volume of the Fisher information matrix $\mid F_{\bar{\Xi}} \mid$, which penalizes complexity in the model.

Here we derive FIC for PHMM, FIC_{PHMM}. Since the local parameters $\Xi = \{\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_{\mathrm{K}}, \phi_1, \ldots, \phi_K\}$ do not interact with each other, the Fisher information matrix F_{Ξ} is a block diagonal matrix whose blocks are $\{F_{\boldsymbol{\beta}_1}, \ldots, F_{\boldsymbol{\beta}_{\mathrm{K}}}, F_{\phi_1}, \ldots, F_{\phi_K}\}$, thus $\ln \mid F_{\Xi} \mid = \sum_k \left(\ln \mid F_{\xi_{\mathrm{k}}} \mid + \ln \mid F_{\boldsymbol{\beta}_{\mathrm{k}}} \mid\right)$. Here, using the Equation (3), we can write these Fisher information matrices as

$$F_{\boldsymbol{\beta}_{\mathrm{k}}} = \frac{1}{N}\sum_{n=1}^{N}\sum_{t=0}^{T_X}\sum_{u=0}^{T_Y} pa\left(z^n_{tu}\right)_k \nabla^2_{\boldsymbol{\beta}} \ \ln p_k\left(z^n_{tu} \mid \boldsymbol{\beta}_{\mathrm{k}}\right)$$

and

$$F_{\phi_k} = \frac{1}{N}\sum_{n=1}^{N}\sum_{t=0}^{T_X}\sum_{u=0}^{T_Y} z^n_{tuk} \nabla^2_{\phi} \ln p\left(x^n_t, y^n_u \mid z^n_{tu}, \phi_k\right),$$

where both $\ln p_k\left(z^n_{tu} \mid \boldsymbol{\beta}_{\mathrm{k}}\right)$ and $\ln p\left(x^n_t, y^n_u \mid z^n_{tu}, \phi_k\right)$ are $\mathcal{O}(1)$ with respect to the number of samples N. Thus, the penalty term is

$$\ln \mid F_{\boldsymbol{\beta}_{\mathrm{k}}} \mid = D_{\boldsymbol{\beta}_{\mathrm{k}}} \ln \frac{\zeta^{\mathrm{trans}}_k(\boldsymbol{Z})}{N} + \mathcal{O}(1)$$

$$\ln \mid F_{\phi_k} \mid = D_{\phi_k} \ln \frac{\zeta^{\mathrm{emit}}_k(\boldsymbol{Z})}{N} + \mathcal{O}(1)$$

$$\text{where} \quad \begin{cases} \zeta^{\mathrm{trans}}_k(\boldsymbol{Z}) & = \displaystyle\sum_{n=1}^{N}\sum_{t=0}^{T_X}\sum_{u=0}^{T_Y} pa(z^n_{tu})_k \\ & = \displaystyle\sum_{n=1}^{N}\sum_{t=0,u=0}^{T^n_x, T^n_y} z^n_{tuk} - \sum_{n=1}^{N} z^n_{T^n_x, T^n_y}. \\ \zeta^{\mathrm{emit}}_k(\boldsymbol{Z}) & = \displaystyle\sum_{n=1}^{N}\sum_{t=0,u=0}^{T^n_x, T^n_y} z^n_{tuk}. \end{cases}$$

The newly introduced symbols ζ^{emit}_k and $\zeta^{\mathrm{trains}}_k$ are *effective samples* of, respectively, transition and emission probability for the kth latent variable. The values $D_{\boldsymbol{\beta}_{\mathrm{k}}}$ and D_{ϕ_k} are the dimensionalities of parameters $\boldsymbol{\beta}_{\mathrm{k}}$ and ϕ_k, respectively.

Finally, ignoring the $\mathcal{O}(1)$ term, we derive FIC for PHMM as

$$FIC_{PHMM}(\mathcal{M}) = \mathbb{E}_{q^{**}}\left[\ln p(\boldsymbol{X}, \boldsymbol{Z} \mid \bar{\Pi}) - \sum_{k=1}^{K}\frac{D_{\beta_k}}{2}\ln \frac{\zeta^{\mathrm{trans}}_k(\boldsymbol{Z})}{N} - \sum_{k=1}^{K}\frac{D_{\phi_k}}{2}\ln \frac{\zeta^{\mathrm{emit}}_k(\boldsymbol{Z})}{N}\right] - \frac{D_{\Pi}}{2}\log N + \mathcal{H}(q^{**}).$$

The penalty terms are now sums of parameter dimensionality weighted by the corresponding effective samples. For example, the dimensionality of the kth emission probability D_{ϕ_k} is weighted by $\zeta^{\mathrm{emit}}_k(\boldsymbol{Z})$. When the effective sample of the kth component is small, the penalty term for the kth latent variable also becomes small. In this case, $\boldsymbol{Z}$ is *degenerate* and we can safely prune the kth latent component. This model pruning is further discussed in Section 2.2.5.

2.2.2 FIC lower bound

We employ an EM algorithm to optimize the parameters. To make the EM algorithm tractable, we further take the lower bound of *FI C*$_{\mathrm{PHMM}}$ and derive *FICLB*. We use three approximations to construct the lower bound. (i) Since the MJLEs $\bar{\Pi}$ is unavailable in practice, we replace it by the arbitrary parameter Π, which is optimized in the M step. (ii) Instead of the marginal posterior q^{**}, we use a variational distribution q, which is optimized in the E step. (iii) We take a lower bound of the negative logarithm as $-\log\left(\sum_{ntu} z^n_{tuk}\right) \geq -L\left(\sum_{ntu} z^n_{tuk}, \sum_{ntu}\tilde{q}\left(z^n_{tuk}\right)\right)$, where L is linear approximation of the logarithm function $L(a, b) = \log b + (a - b)/b$ and $\tilde{q}$ is any distribution over $\boldsymbol{Z}$. During the optimization procedure, $\tilde{q}$ is set to be the variational distribution q of the previous time step. Using these approximations, we now get the lower bound

$$\begin{aligned} FIC_{\mathrm{PHMM}}(\mathcal{M}) &\geq FICLB(\mathcal{M}, q, \tilde{q}, \Pi) \\ &= \mathbb{E}_q\left[\log p(\mathbf{X}, \mathbf{Z} \mid \Pi) + \sum_{n=1}^{N}\sum_{t=0,u=0}^{T^n_x, T^n_y} z^n_{tuk} \log \delta_{tuk}\right] \\ &\quad - \frac{D_\alpha}{2}\log N - \sum_{k=1}^{K}\frac{D_{\beta_k}}{2}\log\left(\sum_{n,t,u}\zeta^{\mathrm{trans}}_k(\tilde{\mathbf{Z}}) - 1\right) \\ &\quad - \sum_{k=1}^{K}\frac{D_{\phi_k}}{2}\log\left(\zeta^{\mathrm{emit}}_k(\tilde{\mathbf{Z}}) - 1\right) + \mathcal{H}(q) \end{aligned}$$

$$\text{where} \quad \delta_{tuk} = \begin{cases} \exp\left(-\dfrac{D_{\phi_k}}{2\zeta^{\mathrm{emit}}_k(\tilde{\mathbf{Z}})}\right) \text{if } t = T_X \text{ and } u = T_Y \\ \exp\left(-\dfrac{D_{\phi_k}}{2\zeta^{\mathrm{emit}}_k(\tilde{\mathbf{Z}})} - \dfrac{D_{\beta_k}}{2\zeta^{\mathrm{trans}}_k(\tilde{\mathbf{Z}})}\right) \text{otherwise.} \end{cases}$$

Here, we introduced the auxiliary variable $\tilde{\mathbf{Z}} = \{\tilde{q}\left(z^n_{tuk}\right)\}_{t,u,k,n}$ for simplicity. The full algorithm including *model pruning* (the model selection mechanism) is explained in Section 2.2.5.

2.2.3 E-step updates

We need to obtain the distribution q^* that maximizes *FICLB* (see Algorithm 2.1 for details). This can be done using a modified forward–backward algorithm as follows:

$$f^n_{tuk} = \begin{cases} 0 & \text{if } t < 0 \text{ or } u < 0 \text{ or } (t, u) = (0, 0) \\ \alpha_k \psi^n_{tuk} \delta_{tuk} & \text{if initial position} \\ \psi^n_{tuk} \delta_{tuk} \displaystyle\sum_{j=1}^{K} f^n_{t-d_{xk}, u-d_{yk}, k} \beta_{j,k} & \text{otherwise} \end{cases} \tag{5}$$

$$b^n_{tuk} = \begin{cases} 0 & \text{if } t > T^n_x \text{ or } u > T^n_y \\ 1 & \text{if } t = T^n_x \text{ and } u = T^n_y \\ \sum_{l=1}^{K} \psi^n_{tul} \delta_{tul} b^n_{t+d_{xl},u+d_{yl},l} \beta_{k,l} & \text{otherwise} \end{cases} \tag{6}$$

Using these forward–backward variables, the optimal variational distribution q^* is obtained as follows:

$$q^*\left(z^n_{tuk}\right) = \frac{f^n_{tuk} b^n_{tuk}}{\sum_l f^n_{T^n_x,T^n_y,l}} \tag{7}$$

$$q^*\left(z^n_{t-d_{xk},u-d_{xy},j}, z^n_{tuk}\right) = \frac{f^n_{t-d_{xk},u-d_{xy},j} \beta_{jk} \phi^n_{tuk} b^n_{tuk}}{\sum_l f^n_{T^n_x,T^n_y,l}} \tag{8}$$

2.2.4 M-step updates

Now, we want to find the Π that maximizes *FICLB* for fixed q (see Algorithm 2.1 for details). For those parameters, we have the update function:

$$\alpha_k \propto \sum_n q\left(z^n_{d_{xk},d_{yk},k}\right) \tag{9}$$

$$\beta_{jk} \propto \textstyle\sum_{n,t,u} q\left(z^n_{t-d_{xk},u-d_{xy},j}, z^n_{tuk}\right)$$

$$\phi_k(x,y) \propto \begin{cases} \sum_{ntu} q\left(z^n_{tuk}\right) \mathbb{I}(x = x_t \wedge y = y_u) & \text{if } \mathcal{S}(k) = M \\ \sum_{ntu} q\left(z^n_{tuk}\right) \mathbb{I}(x = x_t) & \text{if } \mathcal{S}(k) = X \\ \sum_{ntu} q\left(z^n_{tuk}\right) \mathbb{I}(y = y_u) & \text{if } \mathcal{S}(k) = Y \end{cases} . \tag{10}$$

For calculation of β_{jk}, out-of-range indexing is treated as zero, that is $q(z^n_{d_{xk},d_{yk},k}) = 0$ if $t - d_{xk} < 0$ or $u - d_{yk} < 0$.

2.2.5 Pruning degenerated components

In contrast to VI, the FAB algorithm enables simultaneous optimization of a model $\mathcal{M}$ and its parameters Π via *model pruning* (Hayashi *et al.*, 2015). Let us call $\boldsymbol{Z}$ *degenerated* when there exists an equivalent likelihood for a smaller model $\tilde{\mathcal{M}}$, that is $p(\boldsymbol{X}, \boldsymbol{Z} \mid \Pi, \mathcal{M}) = p(\boldsymbol{X}, \tilde{\boldsymbol{Z}} \mid \tilde{\Pi}, \tilde{\mathcal{M}})$, where $\mathcal{M} > \tilde{\mathcal{M}}$. In such cases, $\mathcal{M}$ is overcomplete so we can transform the model $(\boldsymbol{Z}, \Pi) \to (\tilde{\boldsymbol{Z}}, \tilde{\Pi})$ to obtain a new smaller and equivalent model. This transformation is called *model pruning*. In the case of FAB-PHMM, we can prune the components k with *effective samples* $\sum_{ntu} q(z^n_{tuk})/N$, which are beneath some threshold ϵ. Starting from a sufficiently large model, we can prune redundant components while optimizing parameters. This model pruning algorithm is shown in Algorithm 2.1.

We observed that this model pruning algorithm sometimes fails by being captured within poor local optima. In such cases, degenerated components are not pruned. To avoid this problem, we incorporate *greedy pruning* (Algorithm 2.2). When the algorithm converged, we append the current model $(\Pi, \mathcal{M}, q)$ to model candidates. Then, delete the component with the fewest effective samples (greedy_pruning) and restart the algorithm. After finding the smallest possible model $(K_M, K_X, K_Y) = (1, 1, 1)$, we choose the model with the largest FIC from among the model candidates.

2.2.6 Computational complexity

For each iteration in Algorithm 2.1, the computational complexity is $\mathcal{O}\left(N \max\left(T^n_X\right) \max\left(T^n_Y\right) K^2\right)$ for the E and M steps, and $\mathcal{O}\left(N \max\left(T^n_X\right) \max\left(T^n_Y\right) K\right)$ for model pruning. Therefore, the overall complexity for each step is $\mathcal{O}\left(N \max\left(T^n_X\right) \max\left(T^n_Y\right) K^2\right)$. Note that this complexity is exactly the same as ordinary parameter learning for PHMM using the Baum–Welch algorithm (Durbin *et al.*, 1998).

Algorithm 2.1 The FAB-PHMM algorithm

```
Input: data $\boldsymbol{X}$, initial model $\mathcal{M} = (K_M, K_X, K_Y)$, initial variational distribution $q$, initial parameter $\Pi$, stopping threshold $\eta$ and pruning threshold $\epsilon$
$FICLB_{\text{prev}} = \infty$
loop
  $\tilde{q} \leftarrow q$
  $q \leftarrow \arg\max_q FICLB(\mathcal{M}, q, \tilde{q}, \Pi)$    ▷ E-step
  for all $k$ that satisfy $\sum_{n,t,u} q\left(z^n_{tuk}\right) \le \epsilon$ do
    delete the $k$th hidden state of model $\mathcal{M}$    ▷ Pruning
  end for
  $\Pi \leftarrow \arg\max_{\tilde{\Pi}} FICLB\left(\mathcal{M}, q, \tilde{q}, \tilde{\Pi}\right)$    ▷ M-step
  if $\mid FICLB(\mathcal{M}, q, \tilde{q}, \Pi) - FICLB_{\text{prev}} \mid < \eta$ then
    end loop
  end if
  $FICLB_{\text{prev}} \leftarrow FICLB\left(\mathcal{M}, q, \tilde{q}, \tilde{\Pi}\right)$
end loop
```

Algorithm 2.2 The greedy FAB-PHMM algorithm

```
Input: data $\boldsymbol{X}$, initial model $\mathcal{M} = (K_M, K_X, K_Y)$, initial variational distribution $q$, initial parameter $\Pi$, stopping threshold $\eta$ and pruning threshold $\epsilon$
Initialize candidates with an empty list
loop
$\left(\bar{\Pi}, \bar{\mathcal{M}}, \bar{q}\right) \leftarrow$ FABPHMM-algorithm($\boldsymbol{X}, \Pi, \mathcal{M}, q, \eta, \epsilon$) ▷ Alg. 2.1
  append $\left(\bar{\Pi}, \bar{\mathcal{M}}, \bar{q}\right)$ to candidates
  if $\bar{\mathcal{M}} = (1, 1, 1)$ then
    end loop
  end if
  $(\Pi, \mathcal{M}, q) \leftarrow$ greedy_pruning($\bar{\Pi}, \bar{\mathcal{M}}, \bar{q}$)
      ▷ Delete the component with fewest effective samples
end loop
choose model with highest FIC from candidates
```

3 Experiments

We performed three types of experiments to answer the following three questions. First, how accurate is the proposed method in selecting the optimal model (Section 3.1)? Next, how much does the proposed method contribute to alignment accuracy relative to fixed-size models (Section 3.2)? Finally, what kinds of models are selected when we train our proposed method against real DNA data (Section 3.3)?

In this study, because of the high computational cost of parameter learning, we concentrated on short alignments (up to 200 bp). Additionally, we considered only global alignments. Extensions for the alignment of longer sequences and local alignment will be discussed in Section 4. Moreover, following previous research (Bradley

Table 1. Model sizes of hand-crafted models

small	med	large	imb	imb_large	huge	imb_huge
(1, 1, 1)	(1, 2, 2)	(1, 4, 4)	(1, 2, 1)	(1, 4, 2)	(1, 6, 6)	(1, 6, 3)

Note: The first, second and third values in each triplet are the numbers of match states (K_M), X-insertion states (K_X) and Y-insertion states (K_Y), respectively. Model names including the term 'imb' indicate an imbalanced model with unequal values of K_X and K_Y. See Supplementary Figures S1–S7 for details of the parameters.

Table 2. Fraction of precise model selections from multiple estimations for data produced with different model sizes

No. of samples	Precise model selection for different model sizes						
	small	med	large	imb	imb_large	huge	imb_huge
100	**10/10**	0/10	0/10	**10/10**	0/10	0/10	0/10
200	**10/10**	0/10	0/10	9/10	0/10	0/10	0/10
300	**10/10**	0/10	0/10	**4/10**	7/10	0/10	0/10
400	**10/10**	0/10	0/10	**9/10**	**10/10**	0/10	**10/10**
500	**10/10**	0/10	0/10	**7/10**	**9/10**	0/10	5/10
600	**10/10**	**5/10**	0/10	**9/10**	**10/10**	**5/10**	**9/10**
700	**10/10**	**10/10**	3/10	**5/10**	**10/10**	**10/10**	**10/10**
800	**10/10**	**10/10**	**10/10**	7/10	**10/10**	**10/10**	**10/10**
900	**9/10**	**10/10**	**10/10**	8/10	**10/10**	**10/10**	**10/10**
1000	**9/10**	**8/10**	**10/10**	7/10	**10/10**	**10/10**	**10/10**

Note: We ran the model selection algorithm 10 times for each combination of sample number and model size (shown in Table 1) with random initial parameters. Each table entry indicates the fraction of runs in which the correct optimal model was selected. The bold-face font indicates precise prediction of the model size, that is when the method selected the model with the largest FIC correctly.

et al., 2009; Lunter *et al.*, 2008), we here concentrate on models with a single match state and multiple insertion states, although our method is potentially applicable to multiple match states (cf. Fig. 1 and Section 4). For all the experiments, we set the stopping threshold $\eta = 10^{-5}$ and pruning threshold $\epsilon = 10^{-4}$.

3.1 Model selection capability

We first investigated the model selection capability of the proposed method. We used synthetic data because the *true model* is not available for real dataset. We defined parameters of PHMMs of different sizes manually and generated alignments from them. The true model size $\mathcal{M}_{\text{true}} = (K_M^*, K_X^*, K_Y^*)$ and their names are provided in Table 1. From the manual models, we generated N alignments of fixed length 100. After removing the gaps from each alignment, we fed the sequences to Algorithm 2.2 and estimated the optimal model only from the data (pairs of sequences). We then determined if it can recover the true model. We set the initial model size to be $(K_M, K_X, K_Y) = (1, 10, 10)$. We ran experiments for sample sizes $N = 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000$. Parameters for each model can be found in Supplementary Section S3 and Figures S1–S7.

Table 2 shows the fraction of models that were correctly predicted—prediction is correct when the true model size and predicted model size are exactly the same. As the number of samples N grew, the approximation became more precise. Also, some models tend to require more samples for precise model selection. With sufficient samples, FAB-PHMM successfully recovered models almost perfectly except in some cases [i.e. in experiments (med, $N = 1000$) and (small, $N = 1000$)]. Even in these cases, the inaccurately selected models have smaller FIC values than accurately selected models, and we assumed that the algorithm was trapped in a poor local optimum. However, we can easily avoid this problem by using multiple runs and choosing the model with the largest FIC. Indeed, when we picked such a model out of 10 replicates generated with a different random seed, the predictions were always correct when the sample size was greater than or equal to 700 (see bold-face text in Table 2; the bold-face font indicates that the model with the highest FIC value successfully predicted the correct model).

Note that this experimental setting is much simpler than the real setting, where the true model is not in the PHMM class (i.e. the real DNA sequences are not PHMM generated). However, we assume that FAB-PHMM can select the optimal model, in the sense of choosing the closest possible models.

3.2 Alignment accuracy

We also explored the alignment accuracy of the proposed method. Since it is difficult to obtain true genome alignments, we reused the generated data from the manual models in the model selection experiment. For alignment accuracy assessments, we had five alignment datasets for each different model size. We first trained multiple models on each of those datasets and then performed alignments. For training, we used PHMM (with a fixed model size) of those eight different model sizes with random parameter initializations in addition to FAB-PHMM, which automatically chose the optimal model size. In total, we used nine PHMMs, including one with the true model sizes as well as one FAB-PHMM. In order to avoid poor local optima, we ran five trainings for each setting and selected the model with the best score (FIC for FAB-PHMM and likelihood for PHMM).

As a measure of performance in terms of accuracy, we used the f1 score, which is the harmonic mean of precision and recall:

$$\text{f1} = 2\frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{11}$$

$$\text{precision} = \frac{\#\ \text{correctly inferred positions}}{\#\ \text{inferred positions}} \tag{12}$$

$$\text{recall} = \frac{\#\ \text{correctly inferred positions}}{\#\ \text{true positions}}. \tag{13}$$

For example, when the true alignment is $\begin{smallmatrix} x_1x_2x_3x_4 \\ -y_1y_2y_3 \end{smallmatrix}$ and the inferred alignment is $\begin{smallmatrix} x_1x_2x_3x_4 \\ y_1-y_2y_3 \end{smallmatrix}$, the true and inferred positions are $\{(x_2, y_1), (x_3, y_2), (x_4, y_3)\}$ and $\{(x_1, y_1), (x_3, y_2), (x_4, y_3)\}$, respectively. The correctly inferred position is simply the intersection of the true positions and inferred positions: $\{(x_3, y_2), (x_4, y_3)\}$. In this case, precision is 2/3 and recall is 2/3; thus, the f1 score is 2/3.

We report the result of the number of sequences where $N = 1000$ in Table 3. For every dataset, the proposed method performed on par with the true model while all the other fixed-size models performed relatively poorly in some cases: for example for the large model dataset experiment, FAB-PHMM and the large model (i.e. the same model size as the one that produced dataset) performed better than others, whereas the large model performed relatively poorly for smaller datasets.

Although this alignment accuracy measure is widely used (e.g. Rivas and Eddy, 2015), this approach only considers aligned bases that correspond to match states and ignores all those corresponding to insertion states. For this reason, we also evaluated the insertion

Table 3. Alignment f1 score

Simulated models	Training models							
	`small`	`med`	`large`	`imb`	`imb_large`	`huge`	`imb_huge`	`fab`
`small`	*0.9286*	0.9286	0.9255	0.9283	0.9281	0.9247	0.9262	**0.9287**
`med`	0.8337	*0.8327*	0.8318	**0.8342**	0.8324	0.8258	0.8279	0.8328
`large`	0.8196	0.8255	*0.8309*	0.8238	0.8278	0.8277	0.8313	**0.8315**
`imb`	0.8882	0.8963	0.8927	*0.8965*	0.8925	0.8919	0.8928	**0.8968**
`imb_large`	0.8602	**0.8694**	0.8681	0.8659	*0.8686*	0.8654	0.8678	0.8688
`huge`	0.9419	0.9473	0.9506	0.9449	0.9490	*0.9513*	0.9510	**0.9515**
`imb_huge`	0.9681	0.9688	**0.9717**	0.9690	0.9715	**0.9717**	*0.9719*	**0.9717**
Average	0.9035	0.9066	0.9071	0.9061	0.9069	0.9052	0.9068	**0.9083**

Note: For each model of data generation (i.e. the simulated models shown in Table 1), we trained models of fixed sizes including the true model and proposed model (`fab`) before the f1 score evaluation. The italic and bold values indicate the result of training with the true model size and the best score obtained without the true model size, respectively. See Table 1 for the details of the models.

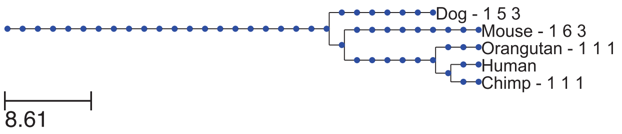

Fig. 4. Trained model sizes for the LAST dataset shown on a phylogenetic tree generated using phyloT and the ETE Toolkit. The model sizes are to the right of the species names, for example 'Mouse—1 6 3' means the selected model for the human–mouse alignment is $(K_M, K_X, K_Y) = (1, 6, 3)$

counterparts of the f1 score (see Supplementary Section S1 for further details).

In addition to assessing alignment accuracy, we also calculated the perplexity of each trained model in order to show how well the models explain the data. Refer to Supplementary Section S2 for further details.

3.3 Model selection from real DNA sequences

We used real DNA data to explore the resulting models selected by FAB-PHMM. Since this study concentrates on global alignments of short sequences, proper analyses require homologous DNA sequence pairs. For this reason, we used locally aligned DNA data produced by Frith and Kawaguchi (2015) (https://zenodo.org/record/17436##.WA3REpOLQYM) and multiple-aligned DNA data produced by MULTIZ (http://hgdownload-test.cse.ucsc.edu/goldenPath/hg38/multiz20way/).

3.3.1 LAST dataset

The LAST dataset (Frith and Kawaguchi, 2015) contains pairwise alignments of human sequences to those of four other species (dog, orangutan, mouse and chimpanzee). For our purposes, we selected the alignments with lengths between 100 and 200. Additionally, we removed the alignments with a 'missmap' probability (an alignment ambiguity measure provided with the dataset) of less than 10^{-5} because we wanted to use only highly reliable homologous pairs. This resulted in 1640 to 94 904 remaining alignments for each of the alignments between the human sequences and those of the four species. We extracted overlapping regions from across all four species and cropped alignments to include only those overlapping regions. Then, we randomly sampled 1000 alignments and removed gaps from them in order to input them into FAB-PHMM. We ran 10 trainings with different random seeds (which determine the initial parameters in PHMM) and selected the best model with the highest FIC value. We set the initial model size to be $(K_M, K_X, K_Y) = (1, 12, 12)$.

Figure 4 shows the selected model size for each species. For species more closely related to humans (i.e. orangutan and chimpanzee), the algorithm selected a similar model and vice versa for species more distantly related to humans (i.e. dog and mouse). Specifically, in the case of chimpanzee and orangutan alignments, the simplest model $(K_M, K_X, K_Y) = (1, 1, 1)$ was chosen, whereas many more X- and Y-insertion states were estimated in the case of dog and mouse alignments. Additionally, it was observed that more X-insertion states were predicted than Y-insertion states for dog and mouse alignments.

Figures 5 and 6, Supplementary Figures S8 and S9 provide the detailed parameters of the selected models from human–chimpanzee, human–mouse, human–orangutan and human–dog sequence alignments, respectively. Overall, all the trained substitution matrixes are almost symmetric. We have the following observations about the human–mouse alignment: (i) two X-insertion states, X3 and X4, with similar and relatively high transition probabilities from the match state (M) were predicted and (ii) X3 had higher emission probabilities of A and C while X4 has higher emission probabilities of T and G. In contrast, the human–dog sequence alignment provides the following observations: (i) the self-transition probabilities of X1, X3 and X5 were similar (about 0.8), though the emission probability of X1 had an almost uniform distribution while those of X3 and X5 were skewed with different profiles; (ii) the self-transition probability of X2 and X4 were smaller (about 0.4) than those of the others, meaning the states corresponded to shorter gaps; (iii) we obtained two long insertion states, Y1 and Y3, and one short insertion state, Y2, while Y2 and Y3 had similar emission profiles.

3.3.2 MULTIZ dataset

We also used the multiz20way dataset, which consists of multiple alignments of 19 different species' genome assemblies to the human genome. As with the previous experiment, we randomly selected 1000 alignments of human sequences to sequences from each of the 19 species where each was restricted to have lengths of 90–110 bp. These alignments were then used as a training set in FAB-PHMM. We set the initial model size to be $(K_M, K_X, K_Y) = (1, 12, 12)$.

The selected model sizes are shown on a phylogenetic tree in Figure 7. In general, we can see species that are more distantly related to humans tend to have alignments with humans that are described by `large` models, for example the human–bushbaby alignment model has a model size of (1, 9, 7) and the human–dog alignment model is (1, 9, 9). However, for species that are more closely related to humans, somewhat random model sizes were selected to describe alignments, for example the human–gorilla

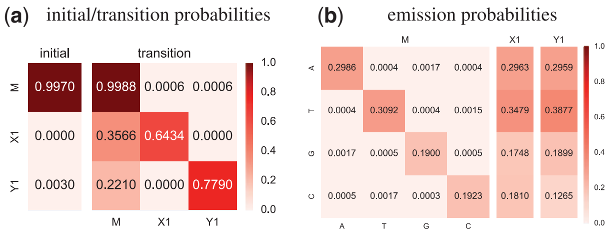

Fig. 5. Trained PHMM for human–chimpanzee alignments for sequences from the LAST dataset using the proposed method. The resulting model is the simplest one, $(K_M, K_X, K_Y) = (1, 1, 1)$. (**a**) Trained initial and transition probabilities and (**b**) emission probabilities

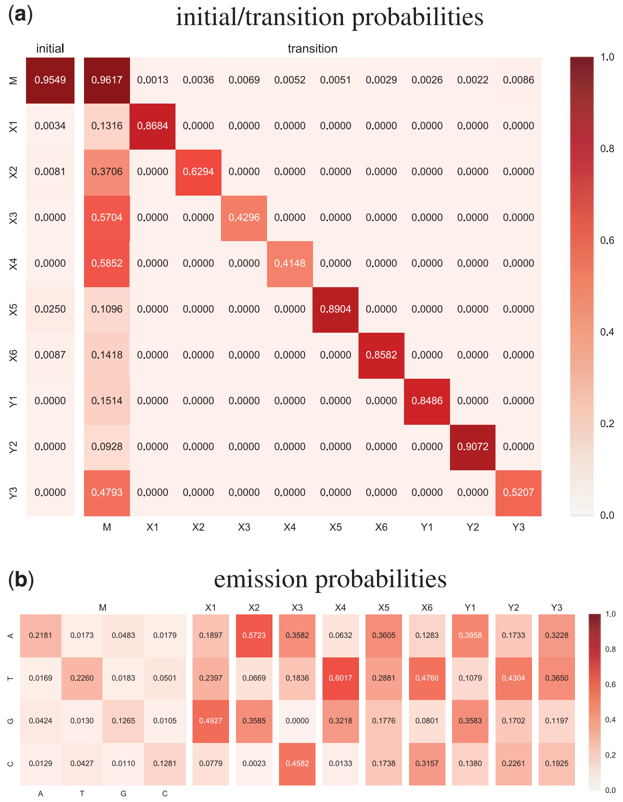

Fig. 6. Trained PHMM and its parameters for alignments between human and mouse sequences. Panel (**a**) shows X1–X5 and Y1–Y3 have five X-insertion states and three Y-insertion states, respectively, where each value is an estimated initial/transition probability. For example, the value of 0.4793 in cell (M , $Y3$) is equal to the transition probability from $Y3$ to M. Panel (**b**) shows emission probabilities of each hidden states. For example, the value 0.0173 in cell (T, A) in the left most panel is equal to the probability of a match state emitting a nucleotide pair ($x = A, y = T$)

alignment model size is (1, 3, 3), whereas the human–chimpanzee model (a closer species' alignment model) has a model size of (1, 5, 4).

4 Discussion

Although our proposed method may be potentially adapted to have multiple match states (cf. Fig. 1), we have concentrated on a method with a single match state and multiple X- and Y-insertion states. This is because some previous studies focused only on multiple insertion states, and learning multiple match states in addition to multiple insertion states would lead to more complex models that are not interpretable. (Also, the use of multiple match states incurs more computational cost due to the increase in trained parameters.) Still, it might be interesting to investigate multiple match states for further improvement of alignment accuracy or making novel inferences about sequence evolution, because multiple match states could correspond to the substitution rate of regions.

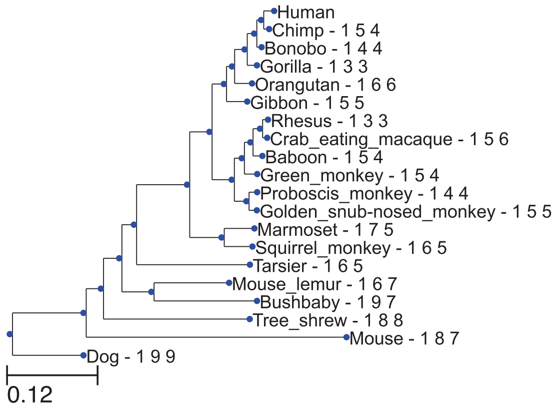

Fig. 7. Trained model sizes for the MULTIZ dataset using the proposed method with an inferred phylogenetic tree. See the caption of Figure 4

Our experiments in Section 3.2 show that models trained by our proposed method achieved better alignment accuracy than other models, but the improvement was only marginal. We assume that this is because maximizing model evidence not always result in maximizing alignment accuracy: higher model evidence might contribute to better sequence modelling, but it might not affect alignment accuracy metric. This result is consistent with previous research by Lunter *et al.* (2008) in which the authors indicated that modifications of insertion states can result in only small improvements in alignment accuracy.

In our analyses using real data (Section 3.3), we utilized limited datasets owing to the high computational cost of our proposed method. Even with these limited datasets, we observed that much more complex models than traditionally used models are selected as optimal. This result implies a possibility that much more complex probabilistic structures exist behind the probabilistic alignments than previously believed.

As well as improving alignment accuracy, the selected model structure may provide interesting insight about biological sequences because the selected probabilistic model structure contains latent information of input data. In other words, the selected model may provide insights into the biological functions of sequences. Similarly, decoded hidden states of PHMMs may reveal (e.g. in DNA alignments) that regions with different hidden states correspond to different functions, such as exon versus intron sequences, non-coding RNAs and regulatory elements. This information can be useful for inferring novel biological insights from sequences.

For the best model selection for real data, however, a comprehensive dataset (including, e.g. coding, non-coding and repetitive regions) is required because the trained model structures depend on input data. Indeed, our analyses led to different model selections based on the LAST and MULTIZ datasets. This could be because the homologous pairs taken for the LAST dataset were more similar than those taken for the MULTIZ dataset due to the protocols of generating input sequences (cf. Section 3.3). In our future work, we will utilize larger and unbiased datasets in order to select more reliable models. The main bottleneck is the high computational cost of our algorithm. To address this, we will attempt to accelerate the training process, such as by parallelization of the algorithms, stochastic optimization (Hoffman *et al.*, 2013; Liu *et al.*, 2015; Robbins and Monro, 1951) and seed-extension heuristics in forward and backward algorithms (Hamada *et al.*, 2017).

In this study, we focused on genomic DNA sequences, but our method is also applicable to RNA or protein sequences. The number of characters in protein sequences greatly exceeds those in DNA sequences, which would lead to more complex models that may provide interesting biological insights. These applications will be included in our future research as well.

5 Conclusions

In this study, we proposed a novel method to develop PHMMs based on FIC and demonstrated the model selection capability of the proposed model using a synthetic dataset. We believe this is the first study that focuses on model selection of PHMM. On the same synthetic dataset, we observed slight improvement of evaluation metrics of sequence alignments. Additionally, we conducted experiments on real DNA sequences and found that they are best handled with a more complex probabilistic structure than the ones that have been traditionally used for pairwise alignment of these species. This result implies a possibility that more complex probabilistic structures exist behind probabilistic alignments than previously believed.

Acknowledgements

The authors are grateful to Kohei Hayashi (AIST), Tsukasa Fukunaga (Waseda University) and members of the Hamada Laboratory at Waseda University. Computation for this study was partially performed on the NIG supercomputer at ROIS National Institute of Genetics.

Funding

This work was supported by MEXT KAKENHI [grant numbers JP24680031, JP16H05879 and JP25240044 to M.H.], in part.

Conflict of Interest: none declared.

References

Altschul,S.F. *et al.* (1990) Basic local alignment search tool. *J. Mol. Biol.*, **215**, 403–410.

Beal,M.J. (2003) Variational algorithms for approximate Bayesian inference. *PhD Thesis*, June, 1–281.

Blei,D.M. *et al.* (2016) Variational inference: a review for statisticians. *J. Am. Stat. Assoc.*, **112**, 859–877.

Bradley,R.K. *et al.* (2009) Fast statistical alignment. *PLoS Comput. Biol.*, **5**, e1000392.

Cartwright,R.A. (2009) Problems and solutions for estimating indel rates and length distributions. *Mol. Biol. Evol.*, **26**, 473–480.

Durbin,R. *et al.* (1998) *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press.

Frith,M.C. and Kawaguchi,R. (2015) Split-alignment of genomes finds orthologies more accurately. *Genome Biol.*, **16**, 106.

Frith,M.C. *et al.* (2010) Parameters for accurate genome alignment. *BMC Bioinformatics*, **11**, 80.

Fujimaki,R. and Hayashi,K. (2012) Factorized asymptotic Bayesian hidden Markov models. In *Proceedings of the 29th International Conference on Machine Learning (ICML)* Edinburgh, Scotland, GB, pp. 799–806. Omnipress, New York, NY, USA.

Fujimaki,R. and Morinaga,S. (2012) Factorized asymptotic Bayesian inference for mixture modeling. In *Proceedings of the 15th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pp. 400–408.

Hamada,M. *et al.* (2017) Training alignment parameters for arbitrary sequencers with LAST-TRAIN. *Bioinformatics*, **33**, 926–928.

Hastings,W.K. (1970) Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, **57**, 97–109.

Hayashi,K. *et al.* (2015) Rebuilding factorized information criterion: asymptotically accurate marginal likelihood. In *Proceedings of the 32nd International Conference on Machine Learning*, **37**.

Hoffman,M.D. *et al.* (2013) Stochastic variational inference. *J. Mach. Learn. Res.*, **14**, 1303–1347.

Jordan,M.I. *et al.* (1999) Introduction to variational methods for graphical models. *Mach. Learn.*, **37**, 183–233.

Li,H. and Homer,N. (2010) A survey of sequence alignment algorithms for next-generation sequencing. *Brief. Bioinformatics*, **11**, 473–483.

Liu,C. *et al.* (2015) Scalable model selection for large-scale factorial relational models. In: Bach,F. and Blei,D. (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, Vol. 37. PMLR, Lille, France, pp. 1227–1235.

Lunter,G. *et al.* (2008) Uncertainty in homology inferences: assessing and improving genomic sequence alignment. *Genome Res.*, **18**, 298–309.

Pachter,L. *et al.* (2002) Applications of generalized pair hidden Markov models to alignment and gene finding problems. *J. Comput. Biol.*, **9**, 389–399.

Paten,B. *et al.* (2008) Enredo and Pecan: genome-wide mammalian consistency-based multiple alignment with paralogs. *Genome Res.*, **18**, 1814–1828.

Rivas,E. and Eddy,S.R. (2015) Parameterizing sequence alignment with an explicit evolutionary model. *BMC Bioinformatics*, **16**, 406.

Robbins,H. and Monro,S. (1951) A stochastic approximation method. *Ann. Math. Stat.*, **22**, 400–407.

Thompson,J.D. *et al.* (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.

Watanabe,S. (2009) *Algebraic Geometry and Statistical Learning Theory*. Cambridge University Press, New York, NY.