# Supporting Information: Evolutionary Inference via the Poisson Indel Process

Alexandre Bouchard-Côté Michael I. Jordan

## 1 Proofs for the Main PIP Properties

In this section, we prove Theorem 1 and Proposition 2. We begin by stating and proving two lemmas.

**Lemma 1** *Let $U \sim \mathrm{Unif}(0,t)$ and $W \sim \mathrm{Exp}(\mu)$ be independent for fixed $t, \mu > 0$. Then*

$$\mathbb{P}(W + U > t) = \frac{1 - \exp(-t\mu)}{t\mu}.$$

**Proof:** By conditioning:

$$\begin{aligned}\mathbb{P}(W + U > t) &= \mathbb{E}\big[\mathbb{P}(W + U > t|U)\big] \\ &= \int_0^t \frac{\exp(-x\mu)}{t}\,\mathrm{d}x \\ &= \frac{1 - \exp(-t\mu)}{t\mu}.\end{aligned}$$

■

**Lemma 2** *Let $\tau_0$ denote a degenerate topology consisting of a root $\Omega$ connected to a single leaf $v_0$ by an edge of length $t$. Let $H_i$ be a homology path as defined in the main paper, with $\tau = \tau_0$. For all $x \in \tau$, define $I(x) = \{i : H_i(x) \neq \varepsilon, 1 \leq i \leq I\}$ and:*

$$\begin{aligned}N &= |I(\Omega)| \\ N' &= |I(v_0)|.\end{aligned}$$

*Then $N \sim \mathrm{Poi}(\lambda/\mu)$ implies $N' \sim \mathrm{Poi}(\lambda/\mu)$.*

**Proof:** To prove the result, we decompose $N$ and $N'$ as follows (see Figure S.1):

$$\begin{aligned} N_1 &= |I(\Omega)\backslash I(v_0)| \\ N_2 &= |I(\Omega) \cap I(v_0)| \\ N_3 &= |I(v_0)\backslash I(\Omega)| \\ N_4 &= |I\backslash I(\Omega)\backslash I(v_0)| \\ N &= N_1 + N_2 \\ N' &= N_2 + N_3. \end{aligned}$$

By the Coloring Theorem [1],

$$N_2 \sim \text{Poi}\left(\nu(\{\Omega\})\mathbb{P}(W > t)\right),$$

where $W$ is a rate $\mu$ exponential random variable, and $\nu$ is as in the condition of Theorem 1. Therefore $N_2 \sim \text{Poi}(\lambda \exp(-t\mu)/\mu)$. Similarly,

$$N_3 \sim \text{Poi}\left(\nu(\tau\backslash\{\Omega\})\mathbb{P}(W + U > t)\right),$$

where $U \sim \text{Unif}(0, t)$, and therefore from Lemma 1, $N_3 \sim \text{Poi}(\lambda(1-\exp(-t\mu))/\mu)$. It follows that:

$$\begin{aligned} N' &= N_2 + N_3 \\ &\sim \text{Poi}\left(\frac{\lambda}{\mu}e^{-\mu} + \frac{\lambda}{\mu}\left(1 - e^{-\mu}\right)\right) \\ &= \text{Poi}\left(\frac{\lambda}{\mu}\right), \end{aligned}$$

which concludes the proof of the lemma. ■

We can now prove Theorem 1:

**Proof:** In order to establish the equivalence, it is enough to show that for all edges $e = (v \to v')$ in the tree, the following two properties hold:

1. The distribution of the string length at the ancestral endpoint, $|Y(v)|$, is identical in the local and global descriptions: a Poisson distribution with rate $\lambda/\mu$.

2. The distribution of the number and locations of mutations that fall on $e\backslash\{v, v'\}$ are also identical in the local and global descriptions.

We will enumerate the edges in the tree in preorder, using induction to establish these two hypotheses on this list of edges.

In the base case, hypothesis 1 is satisfied by construction: the local description is initialized with a $\text{Poi}(\lambda/\mu)$-distributed number of characters, and in the global description, the intensity measure $\nu$ of the Poisson process $\mathbf{X}$ assigns a point mass $\lambda/\mu$ to $v = \Omega$.

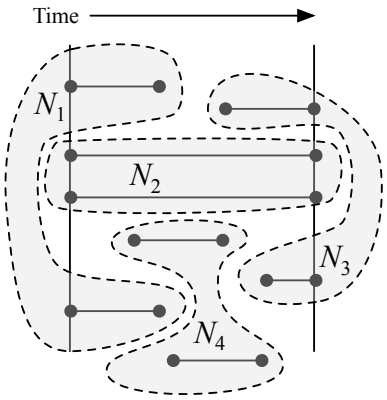


Figure S.1: Notation used in the appendix. The horizontal lines denote the times where each character is present in the sequence. The vertical line on the left denotes the sequence at $\Omega$, and the vertical line on the right, the sequence at $v_0$. The sites are decomposed depending on whether they are present at each of two points $\Omega, v_0$ in $\tau_0$.

To establish hypothesis 1 in the inductive case, let $e' = (v'' \to v)$ denote the parent edge. By hypothesis 1 on $e'$, $|Y(v'')| \sim \text{Poi}(\lambda/\mu)$, therefore by Lemma 2 and hypothesis 2 on $e'$, hypothesis 1 is satisfied on $e$ as well.

To establish hypothesis 2, it is enough to show that for all $x \in e \backslash \{v, v'\}$ the waiting time for each type of mutation given $Y(x)$ is exponential, with rates:

(a) $\lambda$ for insertion,

(b) $\mu \cdot |Y(x)|$ for deletion, and

(c) $\sum_{\sigma \neq \varepsilon} \theta_{\sigma,\sigma'} |Y(x)|_\sigma$ for substitutions to $\sigma' \neq \varepsilon$, where $|s|_\sigma$ denotes the number of characters of type $\sigma \in \Sigma$ in the string $s \in \Sigma^*$.

Item (a) follows from the Poisson Interval Theorem [1]. Items (b) and (c) follow from the standard Doob-Gillespie characterization of CTMCs: if $X_t$ is a CTMC with rate matrix $Q = (q_{i,j})$ and $Z_{i,j}$ are independent exponential random variables with rate $q_{i,j}$, then

$$(\Delta, J) | (X_0 = i) \stackrel{d}{=} (\min_{j \neq i} Z_{i,j}, \operatorname*{argmin}_{j \neq i} Z_{i,j}),$$

where $\Delta = \inf\{t : X_t \neq i\}$, $J = X_\Delta$. ■

We now turn to Proposition 2 and establish reversibility.

**Proof:** Let $h(n_1, n_2, n_3, n_4) = \mathbb{P}(N_i = n_i, i \in \{1, 2, 3, 4\})$. Using reversibility of $\theta$, it is enough to show that $h$ is invariant under the permutation (1 3); i.e., $h(n_1, n_2, n_3, n_4) = h(n_3, n_2, n_1, n_4)$.

We have that $h(n_1, n_2, n_3, n_4)$ is equal to:

$$\begin{aligned}
&\mathbb{P}\Big(N_i = n_i, \sum_i N_i = \sum_i n_i, N_1 + N_2 = n_1 + n_2, N_3 + N_4 = n_3 + n_4\Big) \\
&\quad = \mathbb{P}\Big(\sum_i N_i = \sum_i n_i\Big) \times \\
&\qquad \mathbb{P}\Big(N_1 + N_2 = n_1 + n_2, N_3 + N_4 = n_3 + n_4 \Big| \sum_i N_i = \sum_i n_i\Big) \times \\
&\qquad \mathbb{P}(N_1 = n_1, N_2 = n_2 | N_1 + N_2 = n_1 + n_2) \times \\
&\qquad \mathbb{P}(N_3 = n_3, N_4 = n_4 | N_3 + N_4 = n_3 + n_4) \\
&\quad = f_1(n_1 + n_2 + n_3 + n_4) \times \\
&\qquad \left(\frac{1/\mu}{1/\mu + t}\right)^{n_1+n_2} \left(\frac{t}{1/\mu + t}\right)^{n_3+n_4} \times \\
&\qquad \left(1 - e^{-\mu t}\right)^{n_1} f_2(n_2) \times \\
&\qquad \left(\frac{1 - e^{-\mu t}}{t\mu}\right)^{n_3} f_3(n_4),
\end{aligned}$$

where only the dependencies of the functions $f_1, f_2$ and $f_3$ is important in this argument, not their exact form. By inspection, it is clear that $h$ is invariant under the permutation (1 3). ■

# 2 Proofs for the Likelihood Computation

First, we show how the function $\varphi$, defined in the main paper, simplifies the computation of $p_\tau(m)$:

$$\begin{aligned}
p_\tau(m) &= \mathbb{E}\big[\mathbb{P}(M = m||\mathbf{X}|)\big] \\
&= \sum_{n=|m|}^{\infty} \mathbb{P}(|\mathbf{X}| = n) \cdot \binom{n}{|m|} \cdot (p(c_\emptyset))^{n-|m|} \prod_{c \in m} p(c) \\
&= \frac{e^{\|\nu\|} \prod_{c \in m} p(c)}{|m|!(p(c_\emptyset))^{|m|}} \sum_{n=|m|}^{\infty} \frac{(\|\nu\| p(c_\emptyset))^n}{(n - |m|)!} \\
&= \frac{e^{\|\nu\|} (\|\nu\| p(c_\emptyset))^{|m|} \prod_{c \in m} p(c)}{|m|!(p(c_\emptyset))^{|m|}} \sum_{k=0}^{\infty} \frac{(\|\nu\| p(c_\emptyset))^k}{k!} \\
&= \frac{e^{\|\nu\|} (\|\nu\| p(c_\emptyset))^{|m|} \prod_{c \in m} p(c)}{|m|!(p(c_\emptyset))^{|m|}} \exp\left(\|\nu\| p(c_\emptyset)\right) \\
&= \varphi(p(c_\emptyset), |m|) \prod_{c \in m} p(c).
\end{aligned}$$

Next, we show how to compute $f_v = \mathbb{P}(C = c | V = v)$ for all $v \in \mathscr{V}$. The recursions for $f_v$ are similar to those found in stochastic Dollo models [2]. Note first that $f_v$ can be zero for some vertices. To see where and why, consider the subset of leaves $S$ that that have an extant nucleotide in the current column $c$, $S = \{v \in \mathscr{L} : H(v) \neq \varepsilon\}$. Then $f_v$ will be non-zero only for the vertices ancestral to all the leaves in $S$. Let us call this set of vertices $A$ (see Figure S.2).

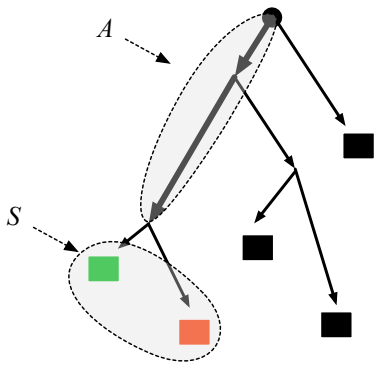


Figure S.2: Given a set $S$ of leaves $v$ with $H(v) \neq \varepsilon$, we define the set $A$ of vertices with nonzero modified Felsenstein peeling weight to be those ancestral to the leaves in $S$. In this example, $A$ contains three vertices.

To compute $f_v$ on the remaining vertices, we introduce an intermediate variable, $\tilde{f}_v = \mathbb{P}(C = c|V = v, H(v) \neq \varepsilon)$. This variable can be computed using the standard Felsenstein peeling recursion (dynamic programming) as follows:

$$\tilde{f}_v(\sigma) = \begin{cases} \mathbf{1}(c(v) = \sigma) & \text{if } v \in \mathscr{L} \\ \sum_{\sigma' \in \Sigma_\varepsilon} \exp(b(v)Q)_{\sigma,\sigma'} \prod_{w \in \text{child}(v)} \tilde{f}_w(\sigma') & \text{o.w.} \end{cases} \tag{1}$$

$$\tilde{f}_v = \sum_{\sigma \in \Sigma} \pi_\sigma \tilde{f}_v(\sigma). \tag{2}$$

From Lemma 1, we have an expression for the survival probability at $v$ given an insertion on the edge $(\text{pa}(v) \to v)$:

$$\begin{aligned} \beta(v) &= \mathbb{P}(H(v) \neq \varepsilon | V = v) \\ &= \frac{1}{b(v)} \frac{1}{\mu} \left(1 - e^{-\mu b(v)}\right). \end{aligned} \tag{3}$$

Finally, for $c \neq c_\emptyset$, we have:

$$\begin{aligned} f_v &= \mathbb{P}(C = c | V = v) \\ &= \mathbb{E}[\mathbb{P}(C = c | V = v, H(v))] \\ &= \begin{cases} \tilde{f}_v & \text{if } v = \Omega \\ \mathbf{1}[v \in A] \beta(v) \tilde{f}_v & \text{o.w.,} \end{cases} \end{aligned} \tag{4}$$

and for $c = c_\emptyset$:

$$f_v = \begin{cases} \tilde{f}_v & \text{if } v = \Omega \\ 1 + \beta(v)(\tilde{f}_v - 1) & \text{o.w.} \end{cases} \tag{5}$$

# 3 Proposal distributions

To perform full joint inference over trees and alignments using Markov chain Monte Carlo, several objects need to be resampled: the tree topology, the branch lengths, the MSA, and the parameters.

For trees and branch lengths, we use standard proposal mechanisms [3]. Our MSA proposal is inspired by the proposal of [4], avoiding the mixing problems of auxiliary variables [5, 6, 7]. Our proposal distribution consists of two steps. First, we partition the leaves into two sets $A, B$. Given a current MSA $m_0$, the support of the proposal is the set $S$ of MSAs $m$ satisfying the following constraints:

1. If $e$ has both endpoints in $A$ (or both in $B$), then $e \in m \iff e \in m_0$.
2. If $e, e'$ have both endpoints in $A$ (or both in $B$), then $e \prec_m e' \iff e \prec_{m_0} e'$.

The notation $\prec_m$ is based on the concept of posets over the columns (and edges) of an MSA [8].

We propose an element $m^* \in S$ with probability proportional to $\prod_{c \in m^*} p(c)$. The set $S$ has exponential size, but can be sampled efficiently using standard pairwise alignment dynamic programming. A Metropolis-Hastings ratio is then computed to correct for $\varphi$. Note that the proposal induces an irreducible chain: one possible outcome of the move is to remove all links between two groups of sequences. The chain can therefore move to the empty MSA and then construct any MSA incrementally.

For the parameters, we used multiplicative proposals in the $(\lambda, \mu)$ parameterization [3].

# 4 Computational Aspects

In this section, we provide a brief discussion of the role that the marginal likelihood plays in both frequentist and Bayesian inference methods.

## 4.1 Maximum likelihood

In the case of maximum likelihood, the overall inference problem involves optimizing over the marginal likelihood:

$$\sup_{\tau \in \mathcal{T}(\mathscr{L}), m \in \mathcal{M}(y)} \log p_\tau(m),$$

where $\tau$ ranges over phylogenies on the leaves $\mathscr{L}$, and $m$ ranges over the alignments consistent with the observed sequences $y$. This optimization problem can be approached using simulated annealing, where a candidate phylogeny and MSA pair $(\tau', m')$ is proposed at each step $i$, and is accepted (meaning that it replaces the previous candidate $(\tau, m)$) according to a sequence of

acceptance functions $f^{(i)}(p, p')$ depending only on the marginal probabilities $p = p_\tau(m), p' = p_{\tau'}(m')$. Provided $\lim_{i\to\infty} f^{(i)}(p, p') = \mathbf{1}[p' > p]$ sufficiently slowly, this algorithm converges to the maximum likelihood phylogeny and MSA [9].

### 4.2 Bayes estimators

In order to define a Bayes estimator, one typically specifies a decision space $D$ (for example the space of MSAs, or the space of multifurcating tree topologies, or both), a projection into this space, $(\tau, m) \mapsto \rho(\tau, m) \in D$, and a loss function $l : D \to [0, \infty)$ on $D$ (for example, for tree topologies, the symmetric clade difference, or partition metric [10]; and for alignments, $1-$ the edge recall or Sum-of-Pairs (SP) score [11]).

Given these objects, the optimal decision in the Bayesian framework (also known as the consensus tree or alignment), is obtained by minimizing over $d \in D$ the risk $\mathbb{E}[l(d, \rho(T, M))|\mathcal{Y}]$. This expectation is intractable, so it is usually approximated with the empirical distribution of the output $(\tau^{(i)}, m^{(i)})$ of an Markov chain Monte Carlo (MCMC) algorithm. Producing MCMC samples boils down to computing acceptance ratios of the form:

$$\frac{p(\tau')p_{\tau'}(m')}{p(\tau)p_\tau(m)} \cdot \frac{q_{(\tau',m')}(\tau, m)}{q_{(\tau,m)}(\tau', m')},$$

for some proposal having density $q$ with respect to a shared reference measure on $\mathcal{T}(\mathscr{L})\times\mathcal{M}(y)$. We thus see that for both maximum likelihood and joint Bayesian inference of the MSA and phylogeny the key problem is that of computing the marginal likelihood $p_\tau(m)$.

## 5 Pseudocode and Example

In this section, we summarize the likelihood computation. We also give a concrete numerical example to illustrate the calculation.

1. Inputs:
   (a) PIP parameter values $(\lambda, \mu)$, substitution matrix $\theta$ over $\Sigma$.
   *Example:* $(\lambda, \mu) = (2.0, 1.0), \Sigma = \{a\}$
   (b) Rooted phylogenetic tree $\tau$
   *Example:* $\tau = ((v_2 : 1.0, v_3 : 1.0)v_0 : 1.0, v_4 : 2.0)v_1$;
   (c) Multiple sequence alignment $m$
   *Example:* $m =$

   ```
   v_2|-a
   v_3|aa
   v_4|a-
   ```

2. Computing modified Felsenstein recursion:

(a) For each site, compute $\tilde{f}_v(\sigma)$ in post-order using Equation (1), and from each $\tilde{f}_v(\sigma)$, compute $\tilde{f}_v$ using Equation (2)
*Example:*
*for site 1,* $(\tilde{f}_{v_2}, \tilde{f}_{v_3}, \tilde{f}_{v_0}, \tilde{f}_{v_4}, \tilde{f}_{v_1}) = (0.0, 1.0, 0.23, 1.0, 0.012)$*;*
*for site 2,* $(\tilde{f}_{v_2}, \tilde{f}_{v_3}, \tilde{f}_{v_0}, \tilde{f}_{v_4}, \tilde{f}_{v_1}) = (1.0, 1.0, 0.14, 0.0, 0.043)$*;*

(b) Do the same for an artificial site or column $c_\emptyset$ where all leaves have a gap
*Example:*
*for site 3,* $(\tilde{f}_{v_2}, \tilde{f}_{v_3}, \tilde{f}_{v_0}, \tilde{f}_{v_4}, \tilde{f}_{v_1}) = (0.0, 0.0, 0.40, 0.0, 0.67)$*;*

3. For each node $v$ in the tree, compute the survival probability $\beta(v)$ using Equation (3) (setting it to 1 at the root for convenience)
*Example:*
$(\beta(v_2), \beta(v_3), \beta(v_0), \beta(v_4), \beta(v_1)) = (0.63, 0.63, 0.63, 0.43, 1.0)$

4. For each site, compute the set of nodes $A$ ancestral to all extant characters, as described in the caption of Figure S.2
*Example:*
*for site 1,* $A = \{v_1\}$
*for site 2,* $A = \{v_0, v_1\}$

5. Computing $f_v$:

(a) For each site, compute $f_v$ using Equation (4)
*Example:*
*for site 1,* $(f_{v_2}, f_{v_3}, f_{v_0}, f_{v_4}, f_{v_1}) = (0.0, 0.0, 0.0, 0.0, 0.012)$*;*
*for site 2,* $(f_{v_2}, f_{v_3}, f_{v_0}, f_{v_4}, f_{v_1}) = (0.0, 0.0, 0.086, 0.0, 0.043)$*;*

(b) For $c_\emptyset$, use Equation (5)
*Example:*
*for site 3,* $(f_{v_2}, f_{v_3}, f_{v_0}, f_{v_4}, f_{v_1}) = (0.37, 0.37, 0.62, 0.57, 0.67)$*;*

6. For each node $v$ in the tree, compute $\iota_v = \mathbb{P}(V = v)$ as shown in Section 3 of the main paper
*Example:*
$(\iota(v_2), \iota(v_3), \iota(v_0), \iota(v_4), \iota(v_1)) = (0.17, 0.17, 0.17, 0.33, 0.17)$

7. Compute $p_\tau(m)$ from the $\iota_v$'s, $f_v$'s as shown in Section 3 of the main paper
*Example:* $\log p_\tau(m) = -11$

# References


[1] Kingman JFC (1993) *Poisson Processes* (Oxford Studies in Probabilities).

[2] Alekseyenko A, Lee C, Suchard MA (2008) Wagner and Dollo: a stochastic duet by composing two parsimonious solos. *Systematic Biology* 57 (5):772–784.

[3] Lakner C, van der Mark P, Huelsenbeck JP, Larget B, Ronquist F (2008) Efficiency of Markov Chain Monte Carlo Tree Proposals in Bayesian Phylogenetics. *Systematic Biology* 57:86–103.

[4] Lunter G, Miklós I, Drummond A, Jensen J, Hein J (2005) Bayesian coestimation of phylogeny and sequence alignment. *BMC Bioinformatics* 6(83).

[5] Holmes I, Bruno WJ (2001) Evolutionary HMM: A Bayesian approach to multiple alignment. *Bioinformatics* 17:803–820.

[6] Jensen J, Hein J (2002) Gibbs sampler for statistical multiple alignment., (Dept of Theor Stat, U Aarhus), Technical report.

[7] Bouchard-Côté A, Jordan MI, Klein D (2009) Efficient inference in phylogenetic InDel trees. *In Proceedings of Advances in Neural Information Processing Systems* 21:177–184.

[8] Schwartz A, Pachter L (2006) Multiple alignment by sequence annealing. *Bioinformatics* 23:e24–e29.

[9] Delyon B (1988) Convergence of the simulated annealing algorithm., (Massachusetts Institute of Technology), Technical report.

[10] Bourque M (1978) Ph.D. thesis (Université de Montréal).

[11] Robert CP (2001) *The Bayesian Choice: From Decision-Theoretic Foundations to Computational Implementation* (Springer).