1 Statistical alignment

1.1 Transducer composition

For our purposes, a transducer is a tuple $\mathbb{T} = (\Omega, \Phi, \phi_0, \Psi, \tau)$ where Ω is an alphabet, Φ is a set of states, $\phi_0 \in \Phi$ is the start state, $\Psi \subseteq \Phi$ are the end states, $\tau : \mathfrak{T} \to \Re$ is the transition weight function, $\mathfrak{T} = \Phi \times (\Omega \cup \{\epsilon\}) \times (\Omega \cup \{\epsilon\}) \times \Phi$ is the labeled transition relation, and ϵ is the empty string.

A transition $(i, x, y, j) \in \mathfrak{T}$ has source and destination states i, j and input and output labels x, y. For input and output sequences $\mathcal{X}, \mathcal{Y} \in \Omega^*$ define the Forward[1] weight $\mathbb{T}_{\mathcal{X}\mathcal{Y}}$ to be the sum of weights of all paths from ϕ_0 to any $\psi \in \Psi$ such that the concatenated input and output labels are, respectively, $\mathcal{X}$ and $\mathcal{Y}$, where the weight of a path is defined as the product of its transition weights. Two transducers are considered equivalent if their Forward weights are equal for all input and output sequences.

A transition is an "input" if its input label is not ϵ, and "null" if both its input and output labels are ϵ. The "outgoing transitions" of a state are all the finite-weight transitions whose source is that state. A state "waits" if all its outgoing transitions (if there are any) are inputs, and "continues" if (i) it has at least one outgoing transition and (ii) none of its outgoing transitions are inputs. A transducer is a "waiting machine" if all its states either wait or continue. Any transducer can be transformed into an equivalent waiting machine by adding an extra state and null transition to split any state that doesn't meet the requirement.

A composition of two transducers $\mathbb{A}, \mathbb{B}$ is a transducer $\mathbb{AB}$ representing the matrix product

$$(\mathbb{AB})_{\mathcal{X}\mathcal{Z}} = \sum_{\mathcal{Y} \in \Omega^*} \mathbb{A}_{\mathcal{X}\mathcal{Y}} \mathbb{B}_{\mathcal{Y}\mathcal{Z}} \ \forall \ \mathcal{X}, \mathcal{Z} \in \Omega^*$$

To establish the existence of at least one such transducer it's sufficient to consider the case where $\mathbb{B}$ is a waiting machine. Take the Cartesian product of $\mathbb{A}$'s and $\mathbb{B}$'s state spaces and synchronize transitions, so that $\mathbb{A}$'s output labels match $\mathbb{B}$'s input labels and $\mathbb{B}$ must be in a wait state before $\mathbb{A}$ can transition

$$\tau_{\mathbb{AB}}((i,i'), x, z, (j,j')) = \begin{cases} \tau_{\mathbb{A}}(i,x,\epsilon,j) + \displaystyle\sum_{y \in \Omega} \tau_{\mathbb{A}}(i,x,y,j) \tau_{\mathbb{B}}(i',y,\epsilon,i') & \text{if } i' \text{ waits, } j' = i' \text{, and } z = \epsilon \\ \displaystyle\sum_{y \in \Omega} \tau_{\mathbb{A}}(i,x,y,j) \tau_{\mathbb{B}}(i',y,z,j') & \text{if } i' \text{ waits and } j' \neq i' \\ \tau_{\mathbb{B}}(i',\epsilon,z,j') & \text{if } i' \text{ continues, } j = i \text{, and } x = \epsilon \\ 0 & \text{otherwise} \end{cases}$$

Note this composition preserves alignment transitivity: characters that are aligned[2] through both $\mathbb{A}$ and $\mathbb{B}$ are aligned by $\mathbb{AB}$ with the same weight. We will require this to be true of all transducer compositions we consider: it's how we map transitions from composite machines to simpler machines.

1.2 State-based emissions

A transducer has state-based emissions if its state space can be partitioned into match, insert, delete, and null states $\{\sigma_M, \sigma_I, \sigma_D, \sigma_N\}$ such that for all transitions (i, x, y, j) of finite weight, exactly one of the following is true

$$\begin{aligned} j \in \sigma_M : &\quad x \in \Omega, \quad y \in \Omega \\ j \in \sigma_I : &\quad x = \epsilon, \quad y \in \Omega \\ j \in \sigma_D : &\quad x \in \Omega, \quad y = \epsilon \\ j \in \sigma_N : &\quad x = \epsilon, \quad y = \epsilon \end{aligned}$$

and further $\tau(i, x, y, j) = U_{ij} E_j(x, y)$ where $\mathbf{U}$ is a transition matrix and E_j an emission weight.

Any transducer can be transformed into an equivalent state-based emitter by adding states.

[1] So-called because it can be computed by the Forward algorithm, after marginalizing any null transition cycles.

[2] In the sense of being associated with the same transition, on a given state path.

1.3 Expected transition count T_{XY}

Given a conditionally-normalized[3] state-based emitter with exactly one match state $\sigma_M = \{1\}$, we seek T_{XY}, the expected number of times that the shortest closed walk from/to state 1, with its null states removed, includes a transition from σ_X to σ_Y.

Define some indicator matrices $\mathbf{J}^{X \to Y}$, $\mathbf{J}^X$ (for $X, Y \in \{M, I, D, N\}$) and $\mathbf{J}^L$, $\mathbf{J}^R$

$$\begin{aligned}
J_{ij}^{X \to Y} &= \left\{ \begin{array}{ll} 1 & \text{if } i \in \sigma_X \text{ and } j \in \sigma_Y \\ 0 & \text{if } i \notin \sigma_X \text{ or } j \notin \sigma_Y \end{array} \right. \\
J_{ij}^{X} &= \left\{ \begin{array}{ll} 1 & \text{if } j \in \sigma_X \\ 0 & \text{if } j \notin \sigma_X \end{array} \right. \\
J_{ij}^{L} &= \left\{ \begin{array}{ll} 1 & \text{if } j \notin \sigma_M \\ 0 & \text{if } j \in \sigma_M \end{array} \right. \\
J_{ij}^{R} &= \left\{ \begin{array}{ll} 1 & \text{if } i \notin \sigma_M \\ 0 & \text{if } i \in \sigma_M \end{array} \right.
\end{aligned}$$

Let $\mathbf{U}$ be the transition matrix. Then

$$T_{XY} = \left(\left(\mathbf{I} - \mathbf{J}^L \odot \mathbf{U} \right)^{-1} \left(\mathbf{J}^{X \to Y} \odot \left(\left(\mathbf{I} - \mathbf{J}^N \odot \mathbf{U} \right)^{-1} \mathbf{U} \right) \right) \left(\mathbf{I} - \mathbf{J}^R \odot \mathbf{U} \right)^{-1} \right)_{11}$$

Here $\odot$ is the elementwise product.

1.4 Rate of change R_{XY} of expected transition count

Suppose that $\mathbb{Q}$ is the generator for an indel process, $\mathbb{G}(\delta_t) = \mathbb{I} + \mathbb{Q}\delta_t$ is the infinitesimal transducer, $\mathbb{F}(t)$ the finite-time approximator[4], and $\mathbb{F}\mathbb{G}$ the transitively-aligning composition of $\mathbb{F}$ and $\mathbb{G}$.

Let $\mathbf{U}(\theta, t, \delta_t)$ be the transition matrix of the composite machine $\mathbb{F}\mathbb{G}$ at time t, infinitesimal time increment δ_t, and parameter value θ. By the definition of $\mathbb{G}$ this is first-order in δ_t, so it has the form

$$\mathbf{U}(\theta, t, \delta_t) = \mathbf{U}_0(\theta, t) + \mathbf{U}_t(\theta, t)\delta_t$$

We are interested in

$$\begin{aligned}
\frac{\partial}{\partial t} T_{XY} &= \lim_{\delta_t \to 0} \frac{T_{XY}(\theta, t, \delta_t) - T_{XY}(\theta, t, 0)}{\delta_t} \\
\frac{\partial^2}{\partial \theta \partial t} T_{XY} &= \lim_{\delta_\theta \to 0} \lim_{\delta_t \to 0} \frac{(T_{XY}(\theta + \delta_\theta, t, \delta_t) - T_{XY}(\theta + \delta_\theta, t, 0)) - (T_{XY}(\theta, t, \delta_t) - T_{XY}(\theta, t, 0))}{\delta_\theta \delta_t}
\end{aligned}$$

We will use the time derivative to numerically integrate T_{XY}, and the cross partial derivative for parameter-fitting. For later convenience define $R_{XY} \equiv \frac{\partial}{\partial t} T_{XY}$.

Expand $\mathbf{U}$ to first order in both δ_θ and δ_t

$$\mathbf{U}(\theta + \delta_\theta, t, \delta_t) = \mathbf{U}_0 + \mathbf{U}_\theta \delta_\theta + \mathbf{U}_t \delta_t + \mathbf{U}_{\theta t} \delta_\theta \delta_t + O(\delta_\theta^2)$$

where $\mathbf{U}_0 = \mathbf{U}(\theta, t, 0)$, $\mathbf{U}_t = \frac{1}{\delta_t} (\mathbf{U}(\theta, t, \delta_t) - \mathbf{U}(\theta, t, 0))$, $\mathbf{U}_\theta = \frac{\partial}{\partial \theta} \mathbf{U}_0$, $\mathbf{U}_{\theta t} = \frac{\partial}{\partial \theta} \mathbf{U}_t$.

We can now expand the matrix inverses. In general, for $x \in \{L, N, R\}$,

$$\begin{aligned}
\left(\mathbf{I} - \mathbf{J}^x \odot \mathbf{U} \right)^{-1} &= \mathbf{V}_0^x + \mathbf{V}_\theta^x \delta_\theta + \mathbf{V}_t^x \delta_t + \mathbf{V}_{\theta t}^x \delta_\theta \delta_t + O(\delta_\theta^2) + O(\delta_t^2) \\
\mathbf{V}_0^x &= (\mathbf{I} - \mathbf{J}^x \odot \mathbf{U}_0)^{-1} \\
\mathbf{V}_\theta^x &= \mathbf{V}_0^x \left(\mathbf{J}^x \odot \mathbf{U}_\theta \right) \mathbf{V}_0^x \\
\mathbf{V}_t^x &= \mathbf{V}_0^x \left(\mathbf{J}^x \odot \mathbf{U}_t \right) \mathbf{V}_0^x \\
\mathbf{V}_{\theta t}^x &= \mathbf{V}_0^x \left(\mathbf{J}^x \odot \mathbf{U}_{\theta t} \right) \mathbf{V}_0^x
\end{aligned}$$

[3] Meaning the Forward weight is a probability whose sum over output sequences, for any given input sequence, is 1.

[4] Alternatively, $\mathbb{F}$ can be an HMM whose output approximates the equilibrium distribution of $\mathbb{Q}$ as $t \to \infty$.

Thus

$$\begin{aligned} T_{XY} &= \left(\left(\mathbf{V}_0^L + \mathbf{V}_\theta^L \delta_\theta + \mathbf{V}_t^L \delta_t + \mathbf{V}_{\theta t}^L \delta_\theta \delta_t \right) \right. \\ & \quad \times \left(\mathbf{J}^{X \to Y} \odot \left(\left(\mathbf{V}_0^N + \mathbf{V}_\theta^N \delta_\theta + \mathbf{V}_t^N \delta_t + \mathbf{V}_{\theta t}^N \delta_\theta \delta_t \right) \left(\mathbf{U}_0 + \mathbf{U}_\theta \delta_\theta + \mathbf{U}_t \delta_t + \mathbf{U}_{\theta t} \delta_\theta \delta_t \right) \right) \right) \\ & \quad \left. \times \left(\mathbf{V}_0^R + \mathbf{V}_\theta^R \delta_\theta + \mathbf{V}_t^R \delta_t + \mathbf{V}_{\theta t}^R \delta_\theta \delta_t \right) \right)_{11} + O(\delta_\theta^2) + O(\delta_t^2) \end{aligned}$$

For convenience define

$$W_{a,b,c,d} = \left(\mathbf{V}_a^L \left(\mathbf{J}^{X \to Y} \odot \left(\mathbf{V}_b^N \mathbf{U}_c \right) \right) \mathbf{V}_d^R \right)_{11}$$

Examining the coefficients of δ_t and $\delta_\theta \delta_t$, we see that

$$\begin{aligned} R_{XY} &= W_{t,0,0,0} + W_{0,t,0,0} + W_{0,0,t,0} + W_{0,0,0,t} \\ \frac{\partial}{\partial \theta} R_{XY} &= W_{t,\theta,0,0} + W_{t,0,\theta,0} + W_{t,0,0,\theta} + W_{\theta,t,0,0} + W_{0,t,\theta,0} + W_{0,t,0,\theta} + W_{\theta,0,t,0} + W_{0,\theta,t,0} + W_{0,0,t,\theta} \\ & \quad + W_{\theta,0,0,t} + W_{0,\theta,0,t} + W_{0,0,\theta,t} + W_{\theta t,0,0,0} + W_{0,\theta t,0,0} + W_{0,0,\theta t,0} + W_{0,0,0,\theta t} \end{aligned}$$

This first-order expansion allows us to "differentiate through" the matrix inverse.

1.5 Expected state count S_X

In order to parameterize $\mathbb{F}$ using $\{T_{XY}\}$, we need to normalize by S_X, the expected number of σ_X-states entered by the walk. Note that $S_x = \sum_{y \in \{M,I,D\}} T_{xy} = \sum_{y \in \{M,I,D\}} T_{yx}$ and $S_M = 1$. Thus if we have all the $\{T_{XY}\}$ then we can trivially find the $\{S_X\}$.

Sometimes it may be useful to compute the S_X separately:

$$\begin{aligned} S_X &= \left(\left(\mathbf{J}^X \odot \left(\mathbf{I} - \mathbf{J}^L \odot \mathbf{U} \right)^{-1} \right) \left(\mathbf{I} - \mathbf{J}^R \odot \mathbf{U} \right)^{-1} \right)_{11} \\ &= \left(\left(\mathbf{J}^X \odot \left(\mathbf{V}_0^L + \mathbf{V}_\theta^L \delta_\theta + \mathbf{V}_t^L \delta_t + \mathbf{V}_{\theta t}^L \delta_\theta \delta_t \right) \right) \left(\mathbf{V}_0^R + \mathbf{V}_\theta^R \delta_\theta + \mathbf{V}_t^R \delta_t + \mathbf{V}_{\theta t}^R \delta_\theta \delta_t \right) \right)_{11} + O(\delta_\theta^2) + O(\delta_t^2) \\ \frac{\partial}{\partial t} S_X &= W'_{t,0} + W'_{0,t} \\ \frac{\partial^2}{\partial \theta \partial t} S_X &= W'_{t,\theta} + W'_{\theta,t} + W'_{\theta t,0} + W'_{0,\theta t} \\ W'_{a,b} &= \left(\left(\mathbf{J}^X \odot \mathbf{V}_a^L \right) \mathbf{V}_b^R \right)_{11} \end{aligned}$$

The approach in [1] (following [2]) is to find S_X for the transition matrix of $\mathbb{F}$ using the first of the above equations, then to express $\frac{\partial}{\partial t} S_X = \sum_y R_{YX}$ in terms of this, obtaining a solvable ODE for S_X.

1.6 Runge-Kutta recursions for T_{XY} and derivatives

Let $\mathbf{R} \equiv \mathbf{R}(\Theta, \mathbf{T})$, where $\mathbb{G} \equiv \mathbb{G}(\Theta, \delta_t)$ and $\mathbb{F} \equiv \mathbb{F}(\mathbf{T}, t)$.

$\mathbf{R}$ includes all the R_{XY}, as $\mathbf{T}$ includes the T_{XY}, and Θ the infinitesimal model parameters.

We use the Runge-Kutta method (RK4) to numerically integrate $\dot{\mathbf{T}} = \mathbf{R}$ so as to estimate $T_{XY}(t)$ at timepoints $(t_0 = 0, t_1, t_2 \ldots)$. Let $\Delta_n = t_{n+1} - t_n$. Applying RK4 gives

$$\begin{aligned} \mathbf{T}^{(n+1)} &= \mathbf{T}^{(n)} + \frac{\Delta_n}{6} \left(\mathbf{K}^{(n,1)} + 2\mathbf{K}^{(n,2)} + 2\mathbf{K}^{(n,3)} + \mathbf{K}^{(n,4)} \right) \\ \mathbf{K}^{(n,1)} &= \mathbf{R} \left(\Theta, \mathbf{T}^{(n)} \right) \\ \mathbf{K}^{(n,2)} &= \mathbf{R} \left(\Theta, \mathbf{T}^{(n)} + \frac{\Delta_n}{2} \mathbf{K}^{(n,1)} \right) \\ \mathbf{K}^{(n,3)} &= \mathbf{R} \left(\Theta, \mathbf{T}^{(n)} + \frac{\Delta_n}{2} \mathbf{K}^{(n,2)} \right) \\ \mathbf{K}^{(n,4)} &= \mathbf{R} \left(\Theta, \mathbf{T}^{(n)} + \Delta_n \mathbf{K}^{(n,3)} \right) \end{aligned}$$

For any parameter $\theta \in \Theta$ we can use the chain rule to find the derivative

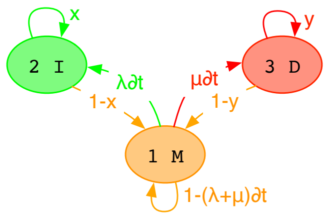
$$\begin{aligned}
\frac{\partial}{\partial\theta}\mathbf{T}^{(n+1)} &= \frac{\partial}{\partial\theta}\mathbf{T}^{(n)} + \frac{\Delta_n}{6}\left(\frac{\partial}{\partial\theta}\mathbf{K}^{(n,1)} + 2\frac{\partial}{\partial\theta}\mathbf{K}^{(n,2)} + 2\frac{\partial}{\partial\theta}\mathbf{K}^{(n,3)} + \frac{\partial}{\partial\theta}\mathbf{K}^{(n,4)}\right) \\
\frac{\partial}{\partial\theta}\mathbf{K}^{(n,1)} &= \frac{\partial}{\partial\theta}\mathbf{R}\left(\Theta, \mathbf{T}^{(n)}\right) + \sum_{X,Y} \frac{\partial}{\partial T_{XY}}\mathbf{R}\left(\Theta, \mathbf{T}^{(n)}\right) \frac{\partial}{\partial\theta} T^{(n)}_{XY} \\
\frac{\partial}{\partial\theta}\mathbf{K}^{(n,2)} &= \frac{\partial}{\partial\theta}\mathbf{R}\left(\Theta, \mathbf{T}^{(n)} + \frac{\Delta_n}{2}\mathbf{K}^{(n,1)}\right) + \sum_{X,Y} \frac{\partial}{\partial T_{XY}}\mathbf{R}\left(\Theta, \mathbf{T}^{(n)} + \frac{\Delta_n}{2}\mathbf{K}^{(n,1)}\right)\left(\frac{\partial}{\partial\theta}T^{(n)}_{XY} + \frac{\Delta_n}{2}\frac{\partial}{\partial\theta}K^{(n,1)}_{XY}\right) \\
\frac{\partial}{\partial\theta}\mathbf{K}^{(n,3)} &= \frac{\partial}{\partial\theta}\mathbf{R}\left(\Theta, \mathbf{T}^{(n)} + \frac{\Delta_n}{2}\mathbf{K}^{(n,2)}\right) + \sum_{X,Y} \frac{\partial}{\partial T_{XY}}\mathbf{R}\left(\Theta, \mathbf{T}^{(n)} + \frac{\Delta_n}{2}\mathbf{K}^{(n,2)}\right)\left(\frac{\partial}{\partial\theta}T^{(n)}_{XY} + \frac{\Delta_n}{2}\frac{\partial}{\partial\theta}K^{(n,2)}_{XY}\right) \\
\frac{\partial}{\partial\theta}\mathbf{K}^{(n,4)} &= \frac{\partial}{\partial\theta}\mathbf{R}\left(\Theta, \mathbf{T}^{(n)} + \Delta_n\mathbf{K}^{(n,3)}\right) + \sum_{X,Y} \frac{\partial}{\partial T_{XY}}\mathbf{R}\left(\Theta, \mathbf{T}^{(n)} + \Delta_n\mathbf{K}^{(n,3)}\right)\left(\frac{\partial}{\partial\theta}T^{(n)}_{XY} + \Delta_n\frac{\partial}{\partial\theta}K^{(n,3)}_{XY}\right)
\end{aligned}$$

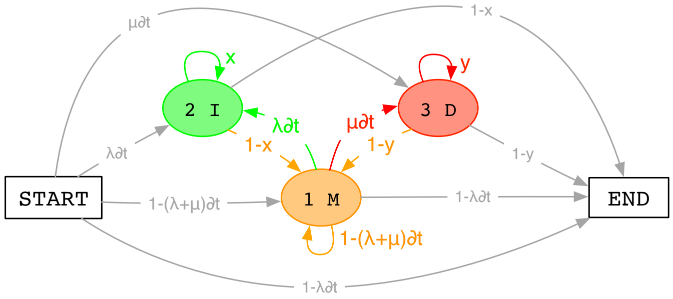
1.7 Infinitesimal transducer $\mathbb{G}$ for General Geometric Indel model (GGI)

A sequence $\mathcal{S}(t)$ evolves by substitutions (rate matrix $\mathbf{Q}$), deletions (rate μ per site, extension probability y), and insertions (rate λ, extension x, characters $\sim$ $\mathbf{Q}$'s stationary distribution π).

At equilibrium, sequence length $\sim$ Geometric(λ/μ). Character frequencies π.

If the model is required to be reversible then $\lambda y(1-x) = \mu x(1-y)$.

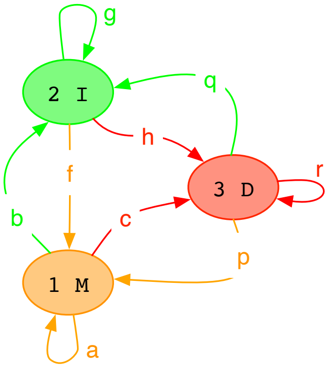


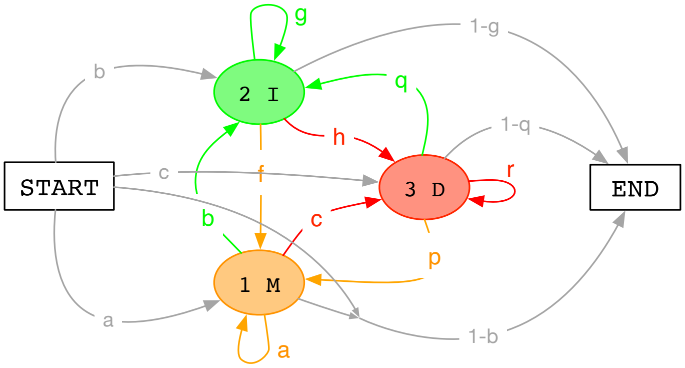


To calculate gap length distributions inside the sequence, it's convenient to ignore end effects, assuming instead that the sequence is infinitely long. We can then dispense with start and end states, as shown in the figure on the left. (For finite-length sequences, we could use the version on the right.)

1.8 Approximate finite-time transducer $\mathbb{F}$ for GGI model

A transducer whose Forward likelihood approximates $P(\mathcal{S}(t)|\mathcal{S}(0))$





Again, the version on the left is for infinite sequences; on the right, finite sequences.

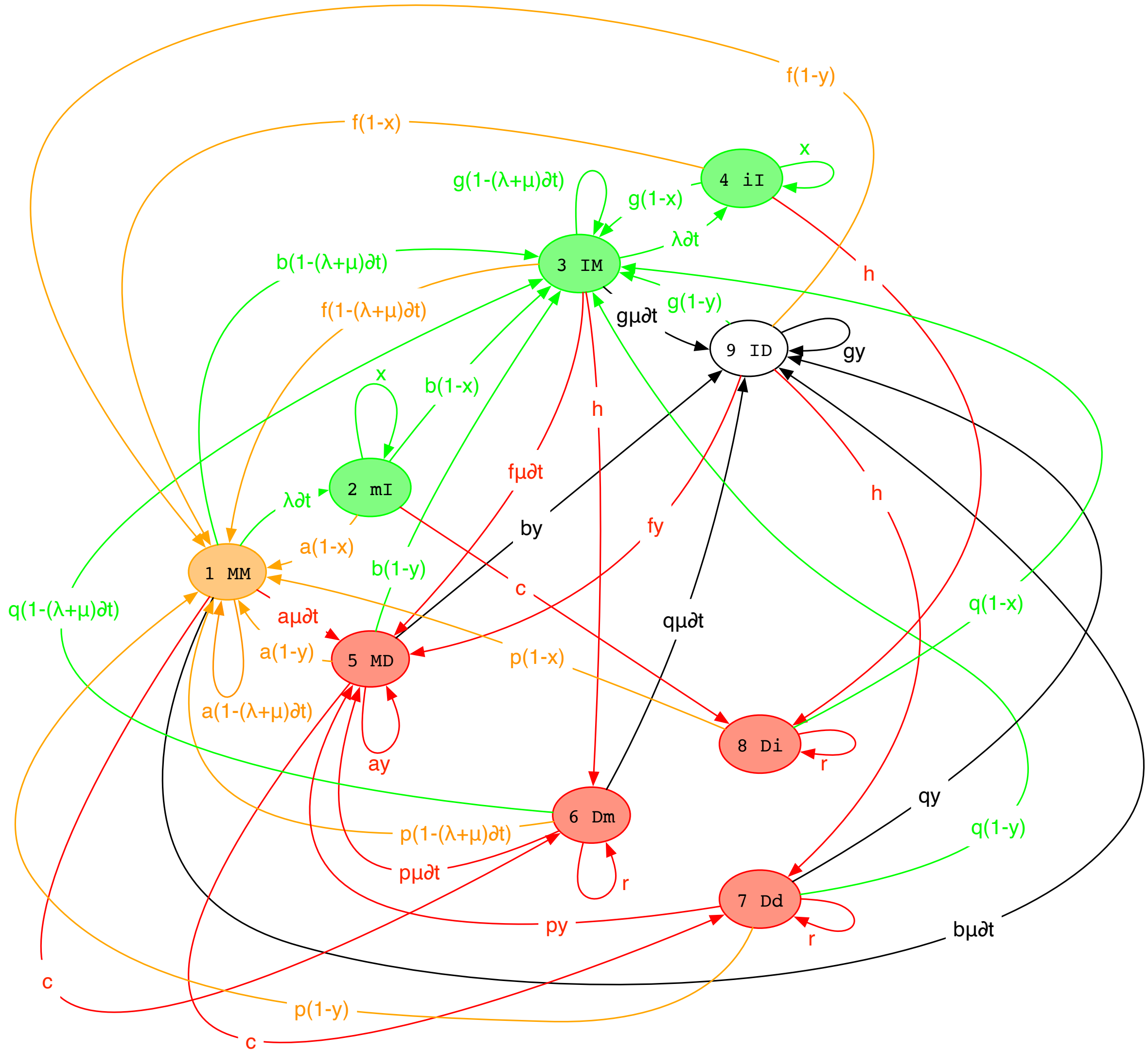
At $t = 0$: $a(0) = 1$, $f(0) = 1-x$, $g(0) = x$, $p(0) = 1-y$, $r(0) = y$, and $b(0) = c(0) = h(0) = q(0) = 0$.

For $t > 0$,

$$\begin{pmatrix} a & b & c \\ f & g & h \\ p & q & r \end{pmatrix} = \begin{pmatrix} T_{MM}/S_M & T_{MI}/S_M & T_{MD}/S_M \\ T_{IM}/S_I & T_{II}/S_I & T_{ID}/S_I \\ T_{DM}/S_D & T_{DI}/S_D & T_{DD}/S_D \end{pmatrix}$$

$$= \begin{pmatrix} T_{MM} & T_{MI} & 1 - T_{MM} - T_{MI} \\ T_{IM}/S_I & (S_I - T_{MI} - T_{DI})/S_I & (T_{MI} + T_{DI} - T_{IM})/S_I \\ (1 - T_{MM} - T_{IM})/S_D & T_{DI}/S_D & (S_D + T_{MM} + T_{IM} - T_{DI} - 1)/S_D \end{pmatrix}$$

1.9 Composite machine $\mathbb{FG}$ for GGI model



$$\mathbf{U} = \begin{pmatrix}
a(1-(\lambda+\mu)\delta_t) & \lambda\delta_t & b(1-(\lambda+\mu)\delta_t) & 0 & a\mu\delta_t & c & 0 & 0 & b\mu\delta_t \\
a(1-x) & x & b(1-x) & 0 & 0 & 0 & 0 & c & 0 \\
f(1-(\lambda+\mu)\delta_t) & 0 & g(1-(\lambda+\mu)\delta_t) & \lambda\delta_t & f\mu\delta_t & h & 0 & 0 & g\mu\delta_t \\
f(1-x) & 0 & g(1-x) & x & 0 & 0 & 0 & h & 0 \\
a(1-y) & 0 & b(1-y) & 0 & ay & 0 & c & 0 & by \\
p(1-(\lambda+\mu)\delta_t) & 0 & q(1-(\lambda+\mu)\delta_t) & 0 & p\mu\delta_t & r & 0 & 0 & q\mu\delta_t \\
p(1-y) & 0 & q(1-y) & 0 & py & 0 & r & 0 & qy \\
p(1-x) & 0 & q(1-x) & 0 & 0 & 0 & 0 & r & 0 \\
f(1-y) & 0 & g(1-y) & 0 & fy & 0 & h & 0 & gy
\end{pmatrix}$$

State sets are $\sigma_M = \{1\}$ (orange), $\sigma_I = \{2, 3, 4\}$ (green), $\sigma_D = \{5, 6, 7, 8\}$ (red), $\sigma_N = \{9\}$ (white).

1.10 Algebraic ODEs for T_{XY} under GGI model

Derived in [1].

$$
\begin{aligned}
S_I(t) &= \exp\left(\frac{\lambda t}{1-x}\right) - 1 \\
S_D(t) &= \exp\left(\frac{\mu t}{1-y}\right) - 1 \\
T_{ij}(0) &= 1 \text{ if } i = j = M,\ 0 \text{ otherwise} \\
\frac{d}{dt} T_{MM}(t) &= \mu \frac{bf(1-y)}{1-gy} - (\lambda+\mu)a \\
\frac{d}{dt} T_{MI}(t) &= -\mu \frac{b(1-g)}{1-gy} + \lambda(1-b) \\
\frac{d}{dt} T_{IM}(t) &= \lambda a - \mu \frac{f(1-g)(b(1-r)+cq)}{(1-gy)(f(1-r)+hp)} \\
\frac{d}{dt} T_{DI}(t) &= \mu \frac{(1-g)(b(1-r-hq)+cgq)}{(1-gy)(f(1-r)+hp)}
\end{aligned}
$$

with $T_{MM}(0) = 1$ and $T_{MI}(0) = T_{IM}(0) = T_{DI}(0) = 0$.

The substitution matrix $E_M(x,y)$ for the match state is the matrix exponential $\exp(\mathbf{Q}t)_{xy}$, for which a Padé approximant or other power series expansion can be used in order to remain automatically differentiable. For the insert states the emission weights are $E_I(\epsilon, y) = \pi_y$, and for the delete states $E_D(x, \epsilon) = 1$.

1.10.1 A simpler form for the GGI ODEs

Rewrite the GGI ODEs with $a \equiv T_{MM}, b \equiv T_{MI}, u \equiv T_{IM}, v \equiv T_{DI}, S \equiv S_I$:

$$
\begin{aligned}
\frac{da}{dt} &= \frac{\mu(1-y)bu}{S - y(S-b-v)} - (\lambda+\mu)a \\
\frac{db}{dt} &= -\frac{\mu(b+v)b}{S - y(S-b-v)} + \lambda(1-b) \\
\frac{du}{dt} &= -\frac{\mu(b+v)u}{S - y(S-b-v)} + \lambda a \\
\frac{dv}{dt} &= \frac{\mu(b+v)(S-v)}{S - y(S-b-v)} \\
S &= \exp\left(\frac{\lambda t}{1-x}\right) - 1
\end{aligned}
$$

with $a(0) = 1$, $b(0) = u(0) = v(0) = 0$, $a'(0) = -\lambda - \mu$, $b'(0) = u'(0) = \lambda$, and $v'(0) = 0$.

Note that the denominator vanishes at $t = 0$, so these formulae are undefined there. However, in the limit $t \to 0$, the counts are obtainable to first order in t by inspection of $\mathbb{G}(\delta t)$, yielding the expressions for the derivatives at the boundary. It is straightforward to use L'Hôpital's rule to verify that these first-order approximations satisfy the ODEs as $t \to 0$.

The variables (a, b, u, v) represent expected counts and are nonnegative always. Further, for $t > 0$ (and assuming $\lambda, \mu > 0$), we know that $a < \exp(-\mu t)$, $a + b < 1$, $a + u < 1$, $b + v < S$, and $v < D - (1 - a - u)$ where $D \equiv S_D = \exp\left(\frac{\mu t}{1-y}\right) - 1$.

1.10.2 A more stable form for the GGI ODEs

As $t \to \infty$ we have $S \simeq \exp\left(\frac{\lambda t}{1-x}\right)$ and (because ancestral residues are inevitably deleted) $a \to 0$. Note that S and D (and possibly v) also grow exponentially. This is a problem for numerical integration: it leads to overflow. We can stabilize the ODEs by transforming to variables (a, b, u, q, L, M) where $L = 1/(S+1)$, $M = 1/(D+1)$, $q = v/D$, and $q' = (v/D)' = (v' - qD')/D$ with $0 \leq q, L, M \leq 1$. Then

$$
\begin{aligned}
\frac{da}{dt} &= \frac{\mu(1-y)buLM}{M(1-y)+Lqy+LM(y(1+b-q)-1)} - (\lambda+\mu)a \\
\frac{db}{dt} &= -\frac{\mu(bM+q(1-M))bL}{M(1-y)+Lqy+LM(y(1+b-q)-1)} + \lambda(1-b) \\
\frac{du}{dt} &= -\frac{\mu(bM+q(1-M))uL}{M(1-y)+Lqy+LM(y(1+b-q)-1)} + \lambda a \\
\frac{dq}{dt} &= \frac{1}{1-M}\left(\frac{\mu(bM+q(1-M))(M(1-L)-qL(1-M))}{M(1-y)+Lqy+LM(y(1+b-q)-1)} - \frac{q\lambda}{1-y}\right) \\
L &= \exp\left(-\frac{\lambda t}{1-x}\right) \\
M &= \exp\left(-\frac{\mu t}{1-y}\right)
\end{aligned}
$$

with $a(0)=1$, $b(0)=u(0)=q(0)=0$, $a'(0)=-\lambda-\mu$, $b'(0)=u'(0)=\lambda$, and $q'(0)=0$.

Note the relationship between the variables in these ODEs and the transition probabilities of $\mathbb{F}(t)$:

$$
\begin{aligned}
\begin{pmatrix} a & b & c \\ f & g & h \\ p & q & r \end{pmatrix} &= \begin{pmatrix} a & b & 1-a-b \\ u/S & (S-b-v)/S & (b+v-u)/S \\ (1-a-u)/D & v/D & (D+a+u-v-1)/D \end{pmatrix} \\
&= \begin{pmatrix} a & b & 1-a-b \\ \frac{L}{1-L}u & 1-\frac{L}{1-L}\left(b+\frac{1-M}{M}q\right) & \frac{L}{1-L}\left(b+\frac{1-M}{M}q-u\right) \\ \frac{M}{1-M}(1-a-u) & q & 1-q-\frac{M}{1-M}(1-a-u) \end{pmatrix}
\end{aligned}
$$

Conversely, $u=fS$ and $v=qD$, which gives closed-form solutions in the special case of TKF91 (see below).

1.10.3 The long-time limit

Although it is possible to analyze the asymptotic behavior of these ODEs as t gets large, in general for large enough t we do not need to bother with the Pair HMM at all, since any alignment signal will be undetectable. At what value of t does this occur? A heuristic argument is as follows: for an alphabet of size[5] A, neglecting the effect of substitutions, we will no longer be able to detect the correct alignment when $(S_I+1)(S_D+1) > \kappa A^{1/(1-a)}$ where the exponent is the mean length of a run of matches and $\kappa > 1$ is some precautionary constant factor (e.g. $\kappa = 2$). An upper bound for a is $\exp(-\mu t)$. This is enough to determine whether any given timepoint is past the threshold of being worth simulating. Beyond that point, we can use the following transition matrix

$$
\begin{pmatrix} a & b & c \\ f & g & h \\ p & q & r \end{pmatrix} = \begin{pmatrix} (1-g)(1-r) & g & (1-g)r \\ (1-g)(1-r) & g & (1-g)r \\ (1-r) & 0 & r \end{pmatrix}
$$

with $g = 1-1/(S_I+1)$ and $r = 1-1/(S_D+1)$.

For an estimate of where the threshold occurs we can solve by iteration: the maximum time T to which we should simulate is $T = \lim_{n\to\infty} T_n$ where

$$
T_{n+1} = \frac{1}{\frac{\lambda}{1-x}+\frac{\mu}{1-y}} \log\left(1+\kappa A^{1/\left(1-e^{-\mu T_n}\right)}\right)
$$

which we can initialize with the even cruder estimate $T_0 = \max\left(\frac{1-x}{\lambda}, \frac{1-y}{\mu}\right)\log(1+A)$ which corresponds to $\max\left(S_I(T_0), S_D(T_0)\right) = A$. This converges reasonably quickly ($n \sim 10$).

[5] Or, more accurately, perplexity.

1.11 TKF91 as exactly-solved special case of GGI model

When $x = y = 0$ the model is TKF91 [3] with solution $\begin{pmatrix} a & b & c \\ f & g & h \\ p & q & r \end{pmatrix} = \begin{pmatrix} (1-\beta)\alpha & \beta & (1-\beta)(1-\alpha) \\ (1-\beta)\alpha & \beta & (1-\beta)(1-\alpha) \\ (1-\gamma)\alpha & \gamma & (1-\gamma)(1-\alpha) \end{pmatrix}$ where $\alpha = \exp(-\mu t)$, $\beta = \frac{\lambda(\exp(-\lambda t) - \exp(-\mu t))}{\mu \exp(-\lambda t) - \lambda \exp(-\mu t)}$ and $\gamma = 1 - \frac{\mu\beta}{\lambda(1-\alpha)}$.

In terms of our ODE variables (a, b, u, v)

$$\begin{aligned} a &= \frac{\mu - \lambda}{\mu L - \lambda M} LM \\ b &= \frac{\lambda}{\mu L - \lambda M} (L - M) \\ u &= \frac{\mu - \lambda}{\mu L - \lambda M} M(1 - L) \\ v &= \frac{\mu - \lambda}{\mu L - \lambda M} - 1 \end{aligned}$$

with $L = \exp(-\lambda t)$ and $M = \exp(-\mu t)$.

1.12 GGI model transition likelihood

The following recipe refers to the case where the alignment is specified. If it's unspecified, use the Forward algorithm to sum over alignments.

The pairwise alignment of ancestor i and descendant j can be summarized by two lists:

- The list A_M of pairs of aligned characters (ω, ω') from ancestor $\mathcal{S}(0)$ and descendant $\mathcal{S}(t)$;
- The list A_{ID} of pairs of unaligned gap sequences (δ, δ') that were deleted and inserted in between.

The alignment probability, computable in time $O(L)$ (and in practice very fast), is

$$P(A_M, A_{ID}, \mathcal{S}(t) | \mathcal{S}(0)) = \prod_{(\omega,\omega') \in A_M} \exp(\mathbf{Q}t)_{\omega\omega'} \prod_{(\delta,\delta') \in A_{ID}} G(|\delta|, |\delta'|, t) \prod_{\omega' \in \delta'} \pi_{\omega'}$$

where $G(i, j, t)$ is the probability that between the next pair of aligned characters in $\mathcal{S}(0)$ and $\mathcal{S}(t)$, there were i characters deleted from $\mathcal{S}(0)$ and j characters inserted into $\mathcal{S}(t)$, with no homology

$$\begin{aligned} G(i,j,t) &= \begin{cases} a & i = j = 0 \\ cr^{i-1}p & i > 0, j = 0 \\ bg^{j-1}f & i = 0, j > 0 \\ g^{j-1}r^{i-1} \left(bhp + cqf + \sum_{k=1}^{\min(i,j)} C(i,j,k,t) \right) & i > 0, j > 0 \end{cases} \\ C(i,j,k,t) &= \left(\frac{hq}{gr} \right)^k \binom{i-1}{k-1} \binom{j-1}{k-1} \frac{1}{k^2} \left(b(j-k)(rfk + hp(i-k)) + c(i-k)(gpk + qf(j-k)) \right) \end{aligned}$$

1.13 Phylogenetic alignment under GGI model

Both forms of the pairwise likelihood (alignment specified, or unspecified) can be extended to multiple sequences on a phylogeny. Upgrade $\exp(\mathbf{Q}t)$ to Felsenstein pruning.

For the alignment-unspecified form, use transducer composition to obtain N-sequence HMMs [4]. The resulting Forward algorithm is $O(L^N)$. To ameliorate this, discard all but a few representative sample paths [5], or use MCMC [6].

References

[1] I. Holmes. A model of indel evolution by finite-state, continuous-time machines. *Genetics*, 216(4):1187–1204, 12 2020.

[2] N. De Maio. The Cumulative Indel Model: fast and accurate statistical evolutionary alignment. *Systematic Biology*, Jul 2020.

[3] J. L. Thorne, H. Kishino, and J. Felsenstein. An evolutionary model for maximum likelihood alignment of DNA sequences. *Journal of Molecular Evolution*, 33:114–124, 1991.

[4] J. Silvestre-Ryan, Y. Wang, M. Sharma, S. Lin, Y. Shen, S. Dider, and I. Holmes. Machine Boss: rapid prototyping of bioinformatic automata. *Bioinformatics*, Jul 2020.

[5] O. Westesson, G. Lunter, B. Paten, and I. Holmes. Accurate reconstruction of insertion-deletion histories by statistical phylogenetics. *PLoS One*, 7(4), 2012.

[6] B. D. Redelings and M. A. Suchard. Incorporating indel information into phylogeny estimation for rapidly emerging pathogens. *BMC Evolutionary Biology*, 7:40, 2007.