Manual for RecHMM

RecHMM is implemented in R and can be run on a command line as:

```
Rscript RecHMM.R <file.SNP> <file.coreGenome> <file.output> <initial R/M rate (0.3) or read model> <Border of recomb region (0.5)> <Peak of recomb region (0.5)> <Maximum no. of iterations (100)> <no. of sites between reference sites (50)>
```

Or you can replace related variables in the RecHMM script with desired values and copy the script into the interactive command interface of R to run it in a window.

Parameters to set:

Interactive interface variables	Command line parameters	Default value	Description
file.SNP	file.SNP	input.SNP	An input file containing the SNP/indel information
file.core	file.coreGenome	input.region	An input file containing the core genomic regions
file.output	file.output	output.RecHMM	The output file containing both the maximum likelihood models and the inferred recombiant stretches
no.initialRate / file.model	initial R/M rate or Read model	0.3	An arbitary initial cut-off values. Or load a pre-calculated model to identify the recombinant stretches
region.border	Border of recomb region	0.5	The minimum likelihood of a site to be assigned as 'recombinant' to establish a recombinant stretch
region.peak	Peak of recomb region	0.5	The minimum requirement of the highest likelihood of 'recombinant' for a stretch to be reported
no.maxIteration	Maximum no. of iterations	100	Stop the E-M algorithm after *no.maxIteration* times even if the E-M algorithm does not converge
no.interval	no. of sites between reference sites	50	intervals between the reference sites

The format of input/output files:

file.SNP: A file with the coordinates and the lineages of SNPs/indels. Homoplastic SNPs/indels can be put in the same line or multiple lines. For example:

```
35 N001.N002
1763 N039.97-7358 N131.A68-37
41510 N018.N023
41510 N001.CIS-1131-72
…
```

File.coreGenome: a file with coordinates in the reference genome of the start and end sites of each region in the core genome. For example:

1 193

247 47725

48916 63473

…

file.output: a file that contains both the maximum likelihood HMM model and the inferred recombinant stretches. For example:

Model initial: # R/M rate to initialize the E-M algorithm

0.3

Model LL, Diff: # The final ln(Likelihood) of model and its difference with the previous run

-1017.87035198385 0.00000918393412181

theta: # global mutation rate

61.4470512340238

R: # global recombination rate

22.4541085374652

delta: # parameter of the geometry distribution for the length of recombinant streches

0.0059812176664259

nu: # density of mutations in the recombinant streches

0.0281736023453083

Branch Names: # name of each branch in the input file

$0 $12 $13 $14 $15 $16 $2 $3 $4 $5 $7 $9

Branch Length: # estimates of branch lengthes

0.178826857948185 0.450025699796207 0.0573265045011334 0.0192653319148043 0.0384290639679409 0.407197915152274
0.019233431128903 0.0190454402759558 0.0771200360304676 0.019149153190227 0.0985880150833512 0.688137866114893

Initial: # estimates of the assignment in the first bases in branches

0.918101002683807 0.953386061451781 0.998191643717514 0.999652029983747 0.998820520528238 0.846898668962749
0.999517123260534 0.999500260877615 0.930066617809411 0.999559340952492 0.93441934698615 0.853553493447001

Recombinant stretches: # estimates of recombinant stretches (start end highest_likelihood length branch)

3584 4102 0.999737015840685 519 $0

2641 2722 0.82974515239418 82 $12

9093 9099 0.52073540174754 7 $12

9719 9738 0.613026134561714 20 $12

94 356 0.990567033974415 263 $16

3336 3518 0.999996277240251 183 $16

4062 4233 0.999997054790003 172 $16

6802 7219 0.999996725932052 418 $4

2085 2550 0.999999906366431 466 $7

979 1037 0.825641806560888 59 $9

5992 6012 0.519679648945266 21 $9

6161 6180 0.529878935925302 20 $9

6870 7365 0.99980775309543 496 $9

Algorithm for RecHMM

Comparison between RecHMM and ClonalFrame

We inherited most of the algorithms in CLONALFRAME but made three major changes. Firstly, RecHMM works with a fixed topology. This accelerates the estimation significantly, because the searching of topologies is the most time consuming process in CLONALFRAME.

Secondly, we applied a simpler Hidden Markov Model for each branch. CLONALFRAME applied a Markov structure with 8 hidden states, which allow estimating variations in three branches connecting to a node instantly. We assumed that most SNPs can be correctly located at branches by "Ancestral state reconstruction" process and thus applied a Markov structure with only 2 hidden states, "recombination" and "non-recombination", for one branch, rather than 8 states in CLONALFRAME, which estimates variations in all three branches connecting to a node instantly. Since the time complexities of most of the algorithms handling the HMM model are $O(M^2L)$, reducing the number of hidden states accelerate the calculation dramatically.

Finally, RecHMM estimates parameters with the Expectation Maximization (EM) algorithm, which searches all possible samples from the hidden states and requires much less iterations than MCMC process. Normally we can find the parameters with maximum likelihood within 100 iterations.

Models for genealogy and hidden states

We selected a subset of the core genome that can be called as "reference sites", as described in the CLONALFRAME, to accelerate the calculation. These reference sites include all the polymorphic sites, the sites at the beginning or end of the core genome and additional sites at intervals of 50 bps. For a total number of M reference sites in the core genome and a genealogy with N branches, the status of all

reference sites in all branches can be represented as an observation matrix $O = \begin{bmatrix} o_{1,1} & \cdots & o_{M,1} \\ \vdots & \ddots & \vdots \\ o_{1,N} & \cdots & o_{M,N} \end{bmatrix}$. Let i refer to the ith reference site and j to the jth branch, $o_{i,j} = 0$ unless it corresponds to a SNP/indel, in which case $o_{i,j} = 1$. O are generated from a serial sampling $H = \begin{bmatrix} h_{1,1} & \cdots & h_{L,1} \\ \vdots & \ddots & \vdots \\ h_{1,N} & \cdots & h_{L,N} \end{bmatrix}$ from one of two hidden states, where $h_{i,j} = S_r$ if this site was imported by recombination, or $h_{i,j} = S_n$ if it was in a non-recombinant region.

Emission Matrix

We assumed that mutations and recombinations happen in each branch as Poisson processes of rate $\theta/2$ and $\rho/2$, respectively. For a jth branch with length l_j, the total number of mutation and recombination events are Poisson distributed with mean $\theta l_j/2$ and $\rho l_j/2$. Thus, the average rates of mutation and recombination per site are $m = (\theta l_j/2)/C$ and $r = (\rho l_j/2)/C$. Furthermore, we assumed that the recombinant stretches have a characteristic density of nucleotide variants designated v, and was distributed geometrically with parameter δ. We also assigned r_j' to be the probability that the first site at the jth branch to be assigned as S_r. Then the emission matrix E, which represents the probability of $o_{i,j}$ given $h_{i,j}$ at the ith reference site in the jth branch is:

	$h_{i,j} = S_n$	$h_{i,j} = S_r$
$P(o_{i,j} = 1 \mid h_{i,j})$	$(\theta l_j/2)/C$	v
$P(o_{i,j} = 0 \mid h_{i,j})$	$1 - (\theta l_j/2)/C$	$1 - v$

Transition Matrix

The calculation of transition matrix is calculated by the same approach as in CLONALFRAME. A standard transition matrix T when neighboring sites are continuous can be represented as:

	$h_{i,j} = S_n$	$h_{i,j} = S_r$
$P(h_{i+1,j} = S_n \mid h_{i,j})$	$1 - (\rho l_j/2)/C$	δ
$P(h_{i+1,j} = S_r \mid h_{i,j})$	$(\rho l_j/2)/C$	$1 - \delta$

Let S_x and S_y be two independent picks of $\{S_n, S_r\}$, t_{S_x,S_y} indicates the transition rate from S_x to S_y. Let $P = \{p_1, \dots, p_M\}$ be the coordinates of reference sites along the core genome, the distances between neighboring sites D can be calculated as: $d_i = p_{i+1} - p_i$. when $d = 1$, the transition matrix $Q(S_x, S_y, j, 1) = t_{S_x,S_y}$. And when $d > 1$, for $S_z \in \{S_r, S_n\}$, Q can be calculated recursively as: $Q(S_x, S_y, j, d) = \sum_{S_z} Q(S_x, S_z, j, d-1 | h_{i,j} = S_x) * t_{S_z,S_y} * E(o_{i,j} = 0 | h_{i,j} = S_z)$. Q do not depend on the positions of reference sites but only the distances between neighboring sites. Thus we need only calculate Q once when all other parameters are given.

E-M algorithm

For each branch b_j, $Pr(b_j) \equiv Pr(h_{*,j} | S_n, S_r; l_j, u_j, \rho, \theta, v, \delta)$, a summation of all possible combination of hidden states $h_{*,j}$ can be estimated efficiently using the forward algorithm. Let the whole tree to be D, the summarized probability of all the branches in D can be calculated directly using $Pr(D) = \prod_{j=1}^{N} Pr(b_j)$.

We can use the Expectation-Maximization (E-M) algorithm to find the local maxima of parameters l_j, ρ, v, δ. Let $\alpha_{i,j,S_x} = Pr(h_{1,j}, \dots, h_{i,j} | h_{i,j} = S_x)$ and $\beta_{i,j,S_x} = Pr(h_{i+1,j}, \dots, h_{L,j} | h_{i,j} = S_x)$, in which $S_x \in \{S_r, S_n\}$. Then

$$\theta = \frac{\sum_{j=1}^{N} (\sum_{i=1}^{M} \alpha_{i,j,S_n} \times \beta_{i,j,S_n} \times o_{i,j} / Pr(b_j))}{\sum_{j=1}^{N} l_j} \quad (1);$$

$$v = \frac{\sum_{j=1}^{N} (\sum_{i=1}^{M} \alpha_{i,j,S_r} \times \beta_{i,j,S_r} \times o_{i,j} / Pr(b_j))}{M \times N} \quad (2);$$

$$r_j' = \frac{\sum_{i=1}^{M} \alpha_{i,j,S_r} \times \beta_{i,j,S_r}}{Pr(b_j)} \quad (3);$$

To calculate the parameter ρ and δ, we introduced another matrix V, in which $V(S_x, S_y, j, 1) = t_{S_x,S_y}$, and $V(S_x, S_y, j, d) = \sum_{S_z} V(S_x, S_z, j, d-1 | h_{i,j} = S_x) * t_{S_z,S_y}$ when $d > 1$. Then

$$\rho = \frac{\sum_{j=1}^{N} \left(\sum_{i=1}^{M-1} \alpha_{i,j,S_n} \times \beta_{i+1,j,S_r} \times \frac{Q(S_n, S_r, j, d_i)}{V(S_n, S_r, j, d_i)} \times t_{S_n,S_r} \times d_i \times Pr(o_{i+1,j} | h_{i+1,j} = S_r) / Pr(b_j) \right)}{\sum_{j=1}^{N} l_j} \quad (4);$$

$$\delta = \frac{\sum_{j=1}^{N} \left(\sum_{i=1}^{M-1} \alpha_{i,j,S_r} \times \beta_{i+1,j,S_n} \times \frac{Q(S_r, S_n, j, d_i)}{V(S_r, S_n, j, d_i)} \times t_{S_r,S_n} \times d_i \times Pr(o_{i+1,j} | h_{i+1,j} = S_n) / Pr(b_j) \right)}{(M-1) \times N} \quad (5);$$

And the length for each branch b_j is

$$l_j = 2 \times \frac{\sum_{i=1}^{M} \alpha_{i,j,S_n} \times \beta_{i,j,S_n} \times \frac{c_{i,j}}{Pr(b_j)} + \sum_{i=1}^{M-1} \alpha_{i,j,S_n} \times \beta_{i+1,j,S_r} \times Pr(o_{i+1,j} | h_{i+1,j} = S_r) / Pr(b_j)}{\theta + R} \quad (6).$$

Initialization and finalization of EM algorithm

For these analyses, nucleotides were assigned to the recombinant state if they fell between the closest pairs of SNPs/indels for each of 10 arbitrary cut-off values drawn from [0.01, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9] of all SNPs/indels. Global parameters ρ, δ, θ and v, as well as parameters l_j and r_j' for each branches can be estimated according to formulates 1-6. We then iterated the E-M algorithm until successive likelihoods differed by <0.00001.