# Bayesian estimation of time-trees:

## A journey through a strange land

Luiz Max Carvalho
lm.carvalho@ed.ac.uk

Institute of Evolutionary Biology
Maxwell Institute seminar series 2017

THE UNIVERSITY
*of* EDINBURGH

# Acknowledgements

Andrew Rambaut
UoE

Marc Suchard
UCLA

Guy Baele
KU Leuven

# Plan for today

## Problem

What are trees and why are interested in them?

## Parameter space

What does the space we are trying to sample look like?

## MCMC in tree space

A journey through a strange land

## Preliminary results and perspectives

Performance analyses and open problems.

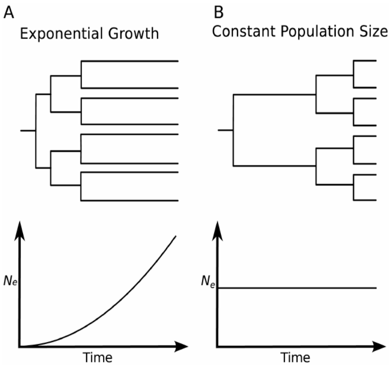
# Motivation

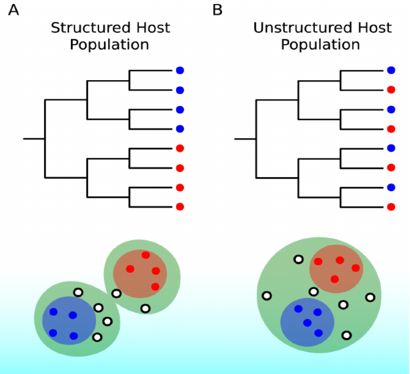
## Phylodynamics of fast-evolving viruses

Inferring spatial and temporal dynamics from genomic data:

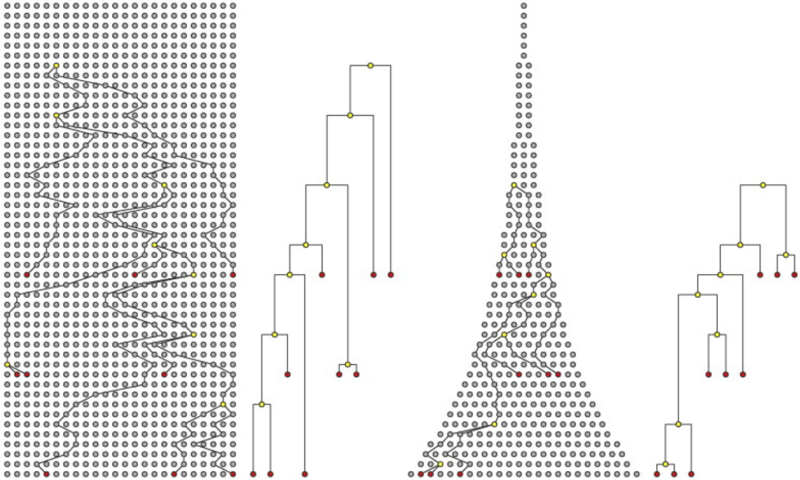
**Phylogenies***!

* plus complicated models





# Trees and the coalescent



# Central object: time-calibrated trees

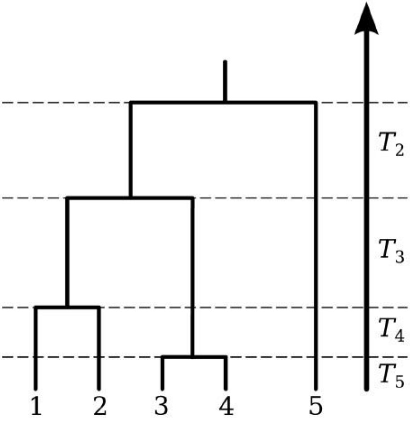


Figure: Figure 4 from Volz et al. (2013).

Let $T_n$ denote the time for $n$ lineages to *coalesce*, i.e., merge into one ancestral lineage, in a population of size $N_e$. Then:

$$Pr(T_n = t) = \lambda_n e^{-\lambda_n t}$$
$$\lambda_n = \binom{n}{2}\frac{1}{N_e} = \binom{n}{2}\frac{1}{N_e \tau}$$

where $N_e$ is the effective population size and $\tau$ is the generation time. Let $T_{\text{mrca}}$ denote the age of the most recent common ancestor:

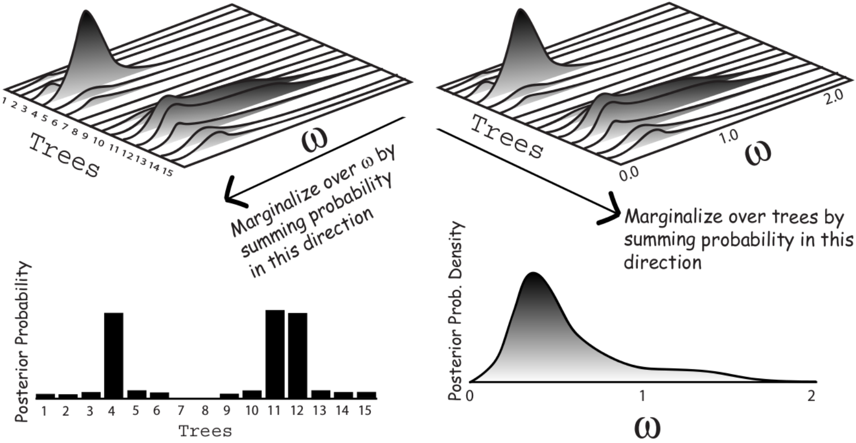
$$\begin{aligned}\mathbb{E}[T_{\text{mrca}}] &= \mathbb{E}[T_n] + \mathbb{E}[T_{n-1}] + \ldots + \mathbb{E}[T_2] \\ &= 1/\lambda_n + 1/\lambda_{n-1} + \ldots + 1/\lambda_2 \\ &= 2N_e(1 - \frac{1}{n})\end{aligned}$$

# Target

$$p(t, \boldsymbol{b}, \boldsymbol{\omega}|D) = \frac{f(D|t, \boldsymbol{b}, \boldsymbol{\omega})\pi(t, \boldsymbol{b}, \boldsymbol{\omega})}{\sum_{t_i \in \boldsymbol{T}_n} \int_{\boldsymbol{B}} \int_{\boldsymbol{\Omega}} f(D|t_i, \boldsymbol{b}_i, \boldsymbol{\omega})\pi(t_i, \boldsymbol{b}_i, \boldsymbol{\omega}) d\boldsymbol{\omega} d\boldsymbol{b}_i} \quad (1)$$

- $D$: observed sequence (DNA) data;
- $\boldsymbol{T}_n$: set of all binary ranked trees ($\mathbb{G}^{(2n-3)!!}$);
- $\boldsymbol{b}_k$: set of branch lengths of $t_k \in \boldsymbol{T}_n$ ($\mathbb{R}_+^{2n-2}$, kind of) ;
- $\boldsymbol{\omega}$: set of parameters of interest such as substitution model parameters, migration rates, heritability coefficients, etc.

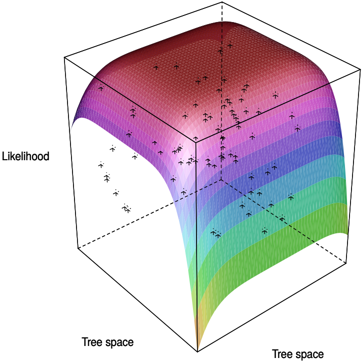
# Bayesian phylogenetics

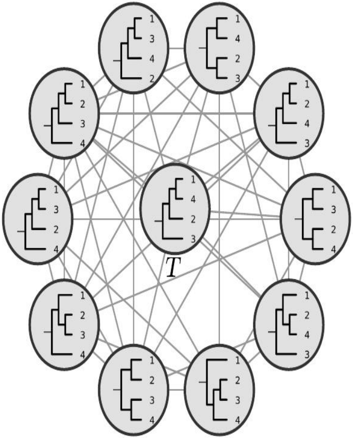


Marginalize over ω by summing probability in this direction

Marginalize over trees by summing probability in this direction

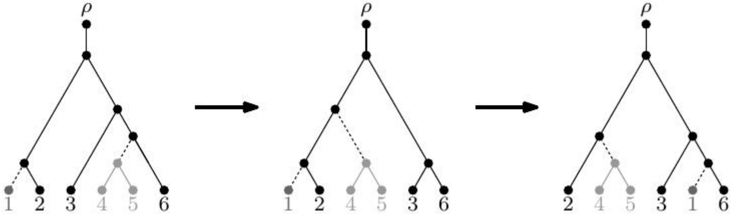
# Tree space: a strange land



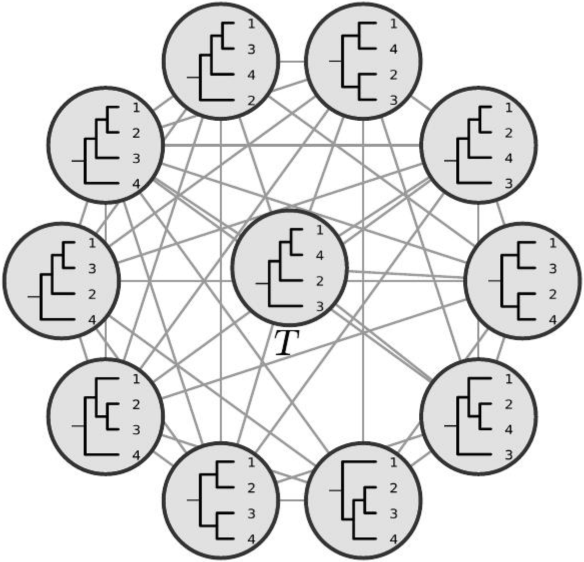


# Discrete tree space: tree surgery

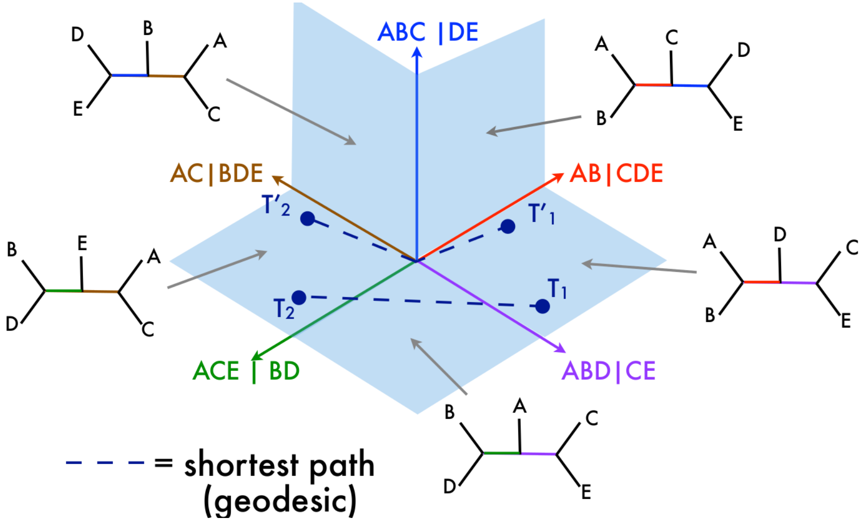
**Subtree prune-and-regraft (SPR)**:



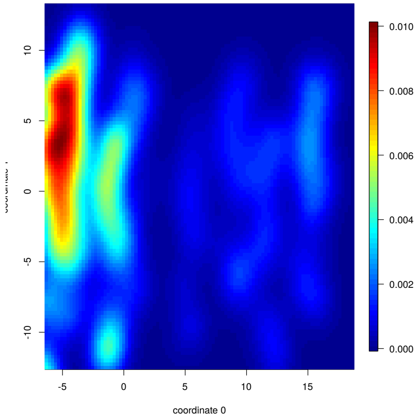
# Discrete tree space: SPR graph



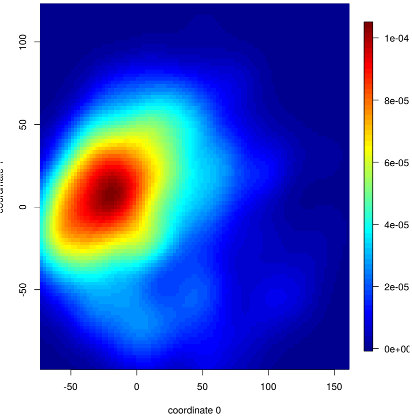
# Continuous tree space: BHV



# Multi-dimensional scaling



Topology



Topology + branches

# Thus

- Non-standard, **huge** parameter space;
- No canonical representation
- Tip (leaf) heights impose constraints.

Open problems:

- Random walks on the SPR graph (and others);
- Useful representation for time-trees;

# Metropolis-Hastings for trees

General MH setup.

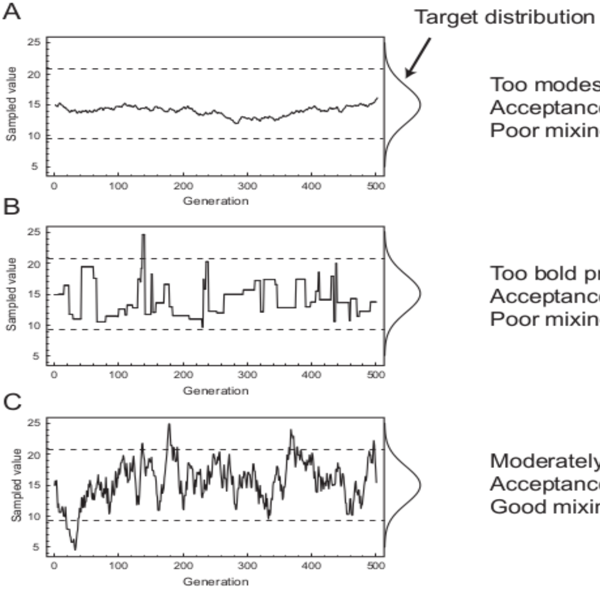
Let $\tau = (t, \boldsymbol{b})$ denote a tree with topology $t$ and branch lengths $\boldsymbol{b}$. For two trees $\tau$ and $\tau'$, denote the transition kernel by $q_\gamma(\tau|\tau') := Pr(\tau' \to \tau|\gamma)$.

Accepting with probability

$$A_\gamma(\tau|\tau') = min\left(1, \frac{p(\tau', \boldsymbol{\omega}|D)q_\gamma(\tau|\tau')}{p(\tau, \boldsymbol{\omega}|D)q_\gamma(\tau'|\tau)}\right)$$

leads to the desired target.

# Exploring parameter space: **mixing**



Too modest proposals
Acceptance rate too high
Poor mixing

Too bold proposals
Acceptance rate too low
Poor mixing

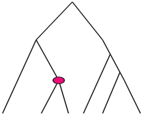
Moderately bold proposals
Acceptance rate intermediate
Good mixing

# Height-constrained kernels: SubTreeLeap (STL)

1. Excluding the root, pick a node $i$ in $\tau$ uniformly at random, i.e., with probability $1/(2n-3)$;
2. Draw a patristic distance $\delta$ from the distance kernel $k(\delta|\sigma)$;
3. Find the set of destination nodes ${\mathbf{D_i}}^{\delta}$ that are within distance $\delta$ **and** whose heights are not less than $h(i)-\delta$;
   If ${\mathbf{D_i}}^{\delta} = \varnothing$:
   - prune $p_i$ and regraft it at height $h_b = h(p_i) - \delta$ or $h_a = h(p_i) + \delta$ with probability $1/2$, creating a new tree $\tau'$, else
   - pick a node $j \in {\mathbf{D_i}}^{\delta}$ with probability $Pr(i \rightarrow j) = 1/|{\mathbf{D_i}}^{\delta}|$, prune the tree at $p_i$ and regraft it at $p_j$, creating a new tree $\tau'$;

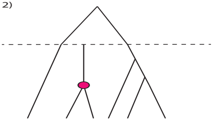
# STL – illustration

1)



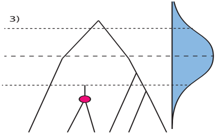
Pick a node

2)



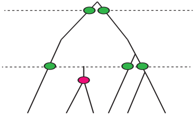
Disconnect its parent

3)



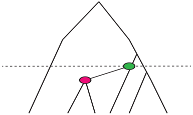
Draw a new height from a normal centred on old height of parent. Also consider the symmetrical height above or below the old height.

4)



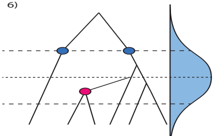
Pick uniformally from branches subtending that height and the symmetrical height above or below (in this case 5).

5)



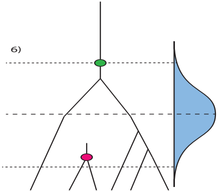
Attach parent to the chosen location.

6)



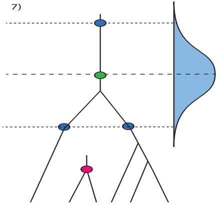
Hastings ratio: ratio of reverse probability (1 / number of reverse locations, i.e., 1/2) to forwards probability (i.e., 1/5).
Hastings ratio = 5 / 2

6)



There is always at least 1 target location (above the root).

7)

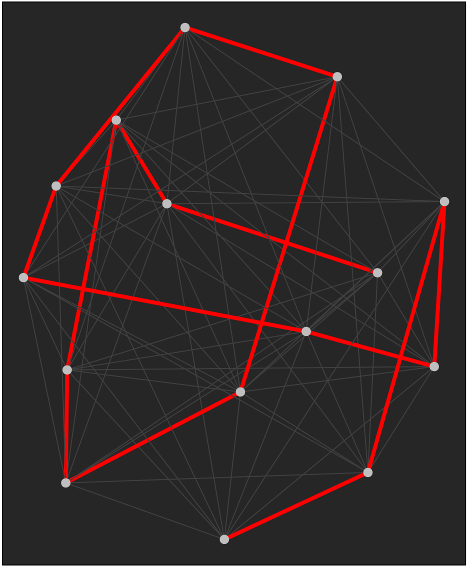


In this case the HR would be 1/3

# STL – properties

- Adaptive;
- Height-constrained;
- Changes topology and branch lengths **simultaneously**;
- Inherits cool properties from SPR.

# Conjecture: SubTreeLeap is Hamiltonian on $T_n$



# Quantifying exploration

- MDS;
- Clade – aka subtree – frequencies;
- Clade switching;
- Effective sample size (ESS) of continuous parameters.

# Clade “space”

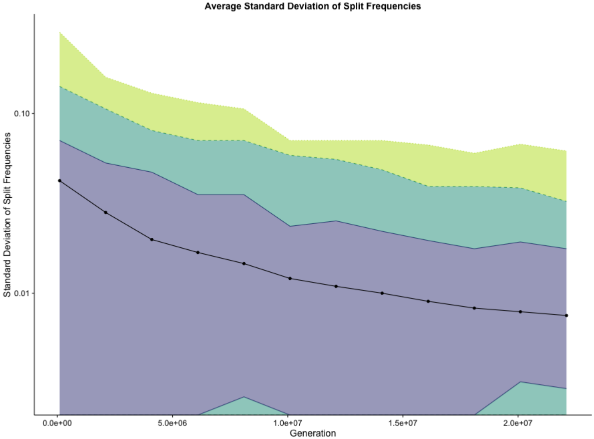
A clade $c$ is any collection of leaves $s_1, s_2, \ldots, s_n$ such that they share a common ancestor in the tree. For $n$ taxa (leaves) there are $A(n) = 2^{n-1} - 1$ possible clades.

Let $\boldsymbol{X}_i = \{X_i^{(1)}, X_i^{(2)}, \ldots, X_i^{(n)}\} \in [0, 1]^n$ be a collection of samples from a Markov chain such that $X_i^{(j)} = 1$ if clade $i$ was sampled in the $j$-th iteration and 0 otherwise. Also, for $s_i = \sum_k X_i^{(k)}$ we call $f_i^c = s_i/n$ the *frequency* of clade $i$.

# Clade frequencies – deviation

$$d := \max_{1 \leq i \leq A(n)} |f_i^c - r_i^c|,$$

where $\boldsymbol{f}^c$ and $\boldsymbol{r}^c$ are the observed and true clade frequencies.



# Clade switching

Let $m_i = \min(n - s_i, s_i)$, it can be shown that the maximum number of transitions that can be observed from $\boldsymbol{X}_i$ is either $J_i = 2m_i$.

Let $\delta_i = \Delta(\boldsymbol{X}_i)$, where $\Delta(\cdot)$ a function that counts the number of state transitions in $\boldsymbol{X}_i$. Then $\sigma_i = \delta_i / J_i \in [0, 1]$ is a score that measures the relative efficiency of sampling by comparing how how many transitions happened compared to the theoretical maximum.

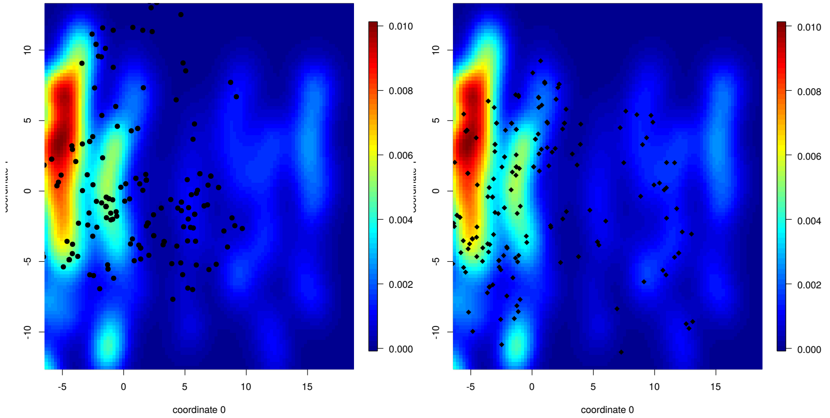
# Experimental setup

All MCMC implemented in the JAVA open-source software BEAST (http://beast.community/);

- Default kernels:
  - SubTreeSlide – adaptive, rarely moves topology;
  - Narrow exchange – non-adaptive, local moves;
  - Wide exchange – non-adaptive, bold moves;
  - NodeHeights – scale all node heights by a factor (within their bounds);
- SubTreeLeap;

– Most results will be shown for 100 MCMC runs.

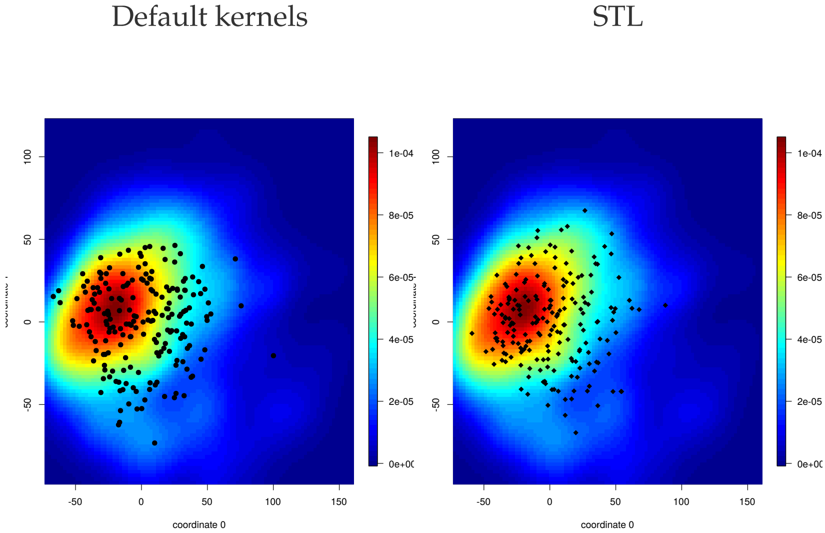
# Traversing tree space – Topology



Default kernels

STL

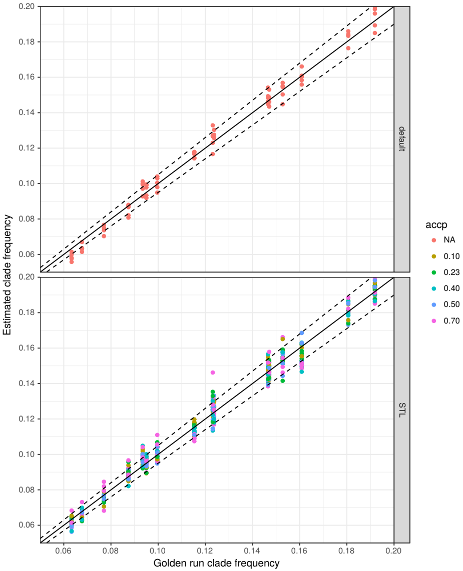
# Traversing tree space – Topology + branch lengths



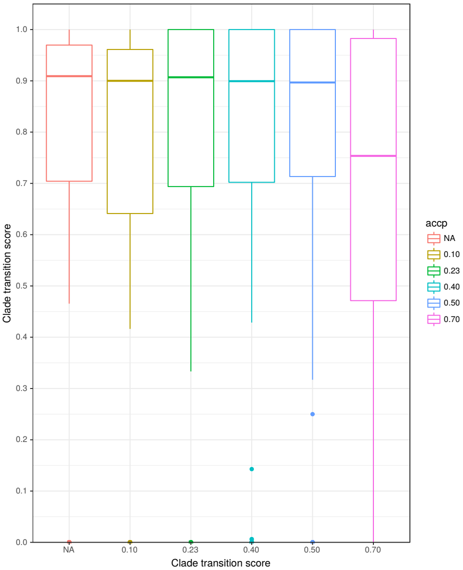
Default kernels

STL

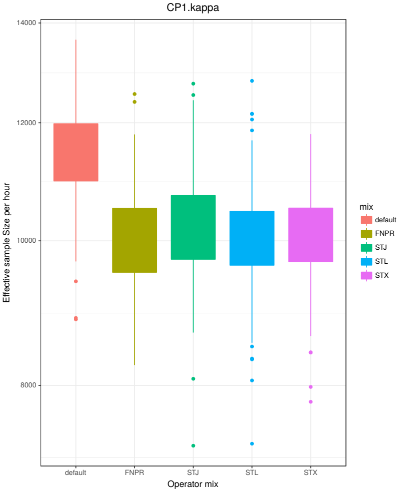
# Clade frequencies – example results

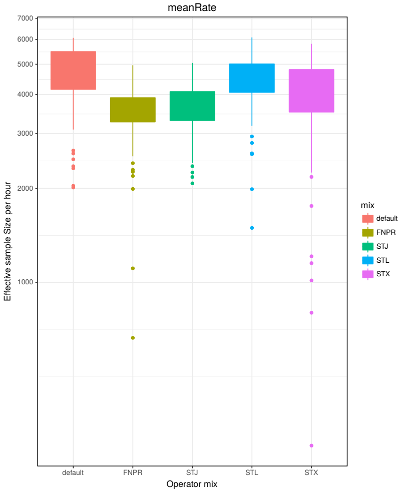


# Clade switching – example results

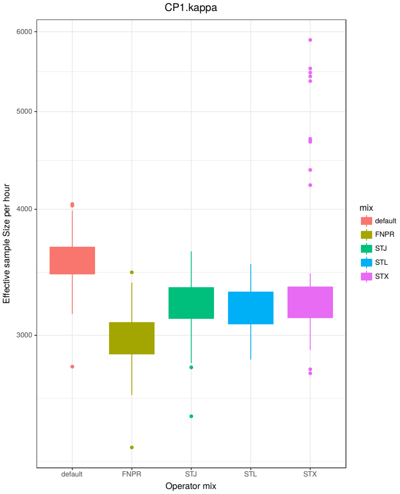


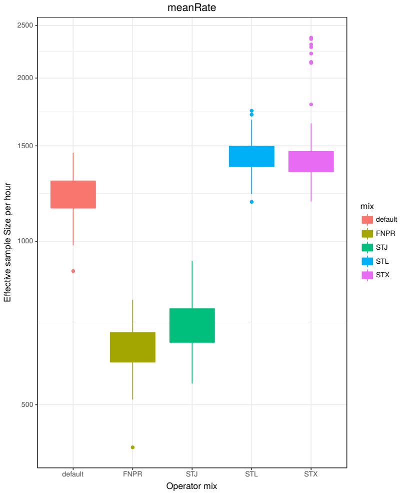
# Dengue 4 *env* (17 taxa, 1485 NT sites)



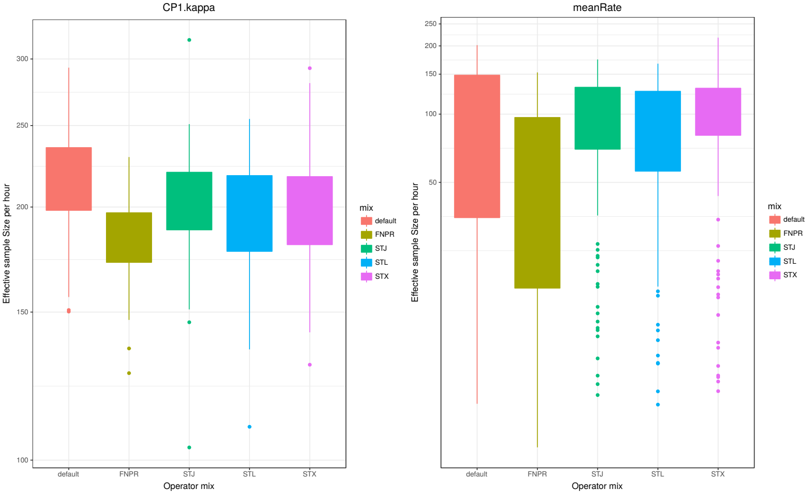


# RSVA G protein (35 taxa, 629 NT sites)

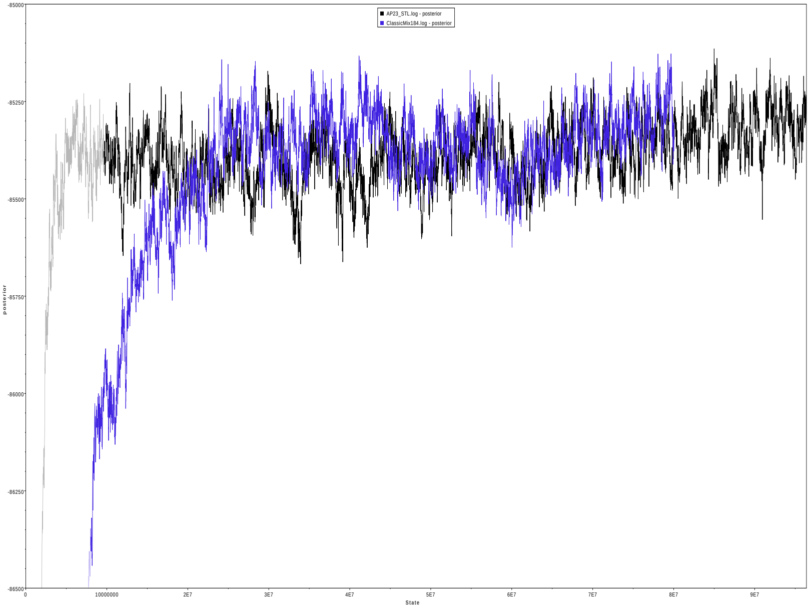




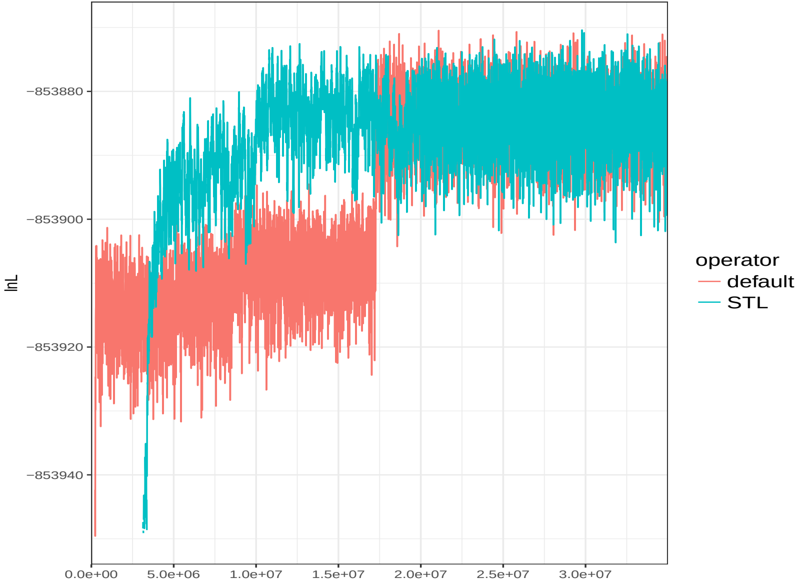
# YFV *prM/E* gene (71 taxa, 654 NT sites)



# Ebola virus full genome (1610 taxa (!), 18990 NT sites)



# Metazoans (contemporaneous, 55 taxa, 30257 AA sites)



# Hence

SubTreeLeap seems to

- ◎ explore topology space more throughly (and mix better in clade space);
- ◎ facilitate sampling other parameters of interest **conditional** on the tree;

Open problems:

- Can we construct even more efficient proposals? How to exploit structure?
- Different distance kernels (currently Gaussian);
- Different weighting (currently uniform);
- Optimal scaling: what's the optimal acceptance probability?

# Take home[1]

## Searching trees is **hard**

Complex, discrete and **HUGE** parameter space

[1] this talk is available online

# Take home[1]

## Searching trees is **hard**

Complex, discrete and **HUGE** parameter space

## Height-preserving tree rearrangements are **good**

Use the extra information provided by the tip dates

[1] this talk is available online

# Take home[1]

## Searching trees is **hard**

Complex, discrete and **HUGE** parameter space

## Height-preserving tree rearrangements are **good**

Use the extra information provided by the tip dates

## Adaptive moves are more efficient

Avoid wasting computing power

[1] this talk is available online

# Take home[1]

## Searching trees is **hard**

Complex, discrete and **HUGE** parameter space

## Height-preserving tree rearrangements are **good**

Use the extra information provided by the tip dates

## Adaptive moves are more efficient

Avoid wasting computing power

## Much more work is needed

We should prepare for an era of plenty

[1] this talk is available online

THE
END