

## PhyDyn: Epidemiological Modelling in BEAST

Igor Siveroni and Erik Volz

MRC Centre for Outbreak Modelling and Analysis, Department of Infectious Disease Epidemiology, Imperial College London.

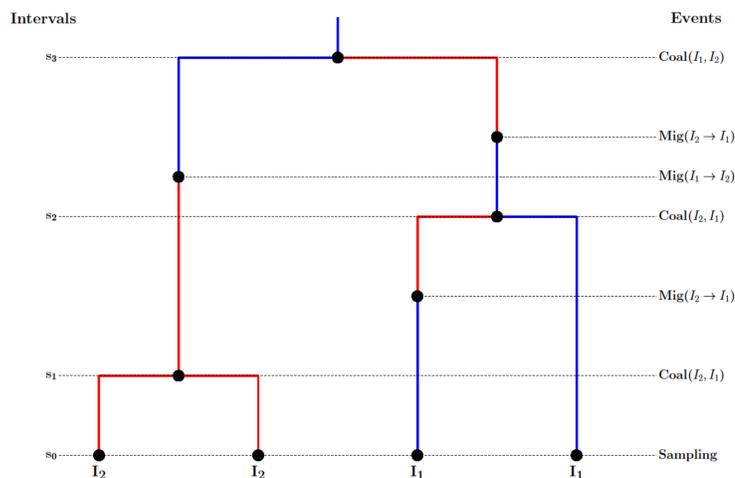
### INTRODUCTION

PhyDyn is a BEAST2 [2] package for performing Bayesian phylogenetic inference under models that deal with structured populations with complex population dynamics. This package enables simultaneous estimation of phylogeny and epidemic parameters given pathogen sequence data. The package implements a coalescent model [1] for a large class of epidemic processes specified by a deterministic nonlinear dynamical system and computes the log-likelihood of a gene genealogy conditional on a complex demographic history.

### THE STRUCTURED COALESCENT AND POPULATION DYNAMICS

The coalescent is a stochastic mathematical model that describes the ancestry of a sample of non-recombining gene copies. In modelling the structured coalescent, it is useful to think of epidemiological models as birth-death-migration processes. If we randomly sample individuals from a population and trace their ancestry back in time, coalescent events correspond to birth events in the population or, in an epidemiological setting, to transmission of infection between hosts. Deaths in the population correspond to recovery or mortality of infected hosts. For structured populations (more than one class of infected host), migration events represent individuals transitioning between different subpopulations, or compartments in SIR-type models.

The figure below shows a genealogy that might have been generated by a two-stage epidemic model with a structured population denoted by states  $I_1$  and  $I_2$ . The red branches represent infected hosts in the first stage of infection, and blue branches represent hosts in the second stage. The genealogy also shows the migration and coalescence events that took place. In this example, all taxa was sampled at time  $s_0$ .



The structured coalescent model developed in [1] allows birth and migration rates to be a function of time and the underlying population dynamics. The model framework provides a mathematical description of the probability that a branch occupies each state at some time  $s$  in the past and the rate at which a pair of branches coalesce. Above all, the model provides a way to compute the probability of a given genealogy under complex population dynamics.

### IMPLEMENTATION

PhyDyn implements a structured coalescent model [1] for a large class of epidemic processes specified by a deterministic nonlinear dynamical system. Genealogies are specified as timed phylogenetic trees in which lineages are associated with the distinct subpopulation in which they are sampled. Epidemic models are defined by a series of ordinary differential equations (ODEs) specifying the rates that new lineages introduced in the population (birth matrix  $F$ ) and the rates at which migrations, or transition between states occur (migration matrix  $G$ ). The package computes:

- Population trajectory: a time series of population sizes and rate matrices ( $F, G$ ) generated by solving the ODEs that define the epidemic model.
- The state probabilities and coalescent rates along the tree intervals. In particular, the package computes these values at branching points e.g. the root. State probabilities are calculated by solving a set of ODEs along the tree intervals backwards in time.
- Tree log-likelihood: calculated as the sum of the probabilities of the occurrence of the coalescent events at the branching points, and the probabilities that no coalescence occurs along the branches i.e. between sampling and coalescence events. In the general case, the latter is computed within the same set of ODEs used to calculate state probabilities.

PhyDyn is implemented as a BEAST2 package. Ordinary differential equations are solved using Apache Commons.

### ACKNOWLEDGEMENTS

The authors would like to acknowledge support from UK MRC Centre for Outbreak Analysis and Modelling, and the NIH MIDAS U01 GM110749 grant.

### EXAMPLE: A simple two-deme model

The framework can be applied to models with spatial structure, multiple stages of infection and models of vector-borne diseases and other multi-host pathogens. Variations on this framework can be useful for epidemics such as HIV, where the transmission probability per contact changes dramatically over the course of infection. As a running example, let's consider a two-deme model where  $S$  denotes the number of hosts that are susceptible, and  $I_1$  and  $I_2$  denote early and chronic infection stages, respectively. Upon infection, hosts enter the state  $I_1$  of average duration  $1/\gamma_1$  and transmit infection at a rate  $\beta_1$ . Infected hosts in  $I_1$  progress to state  $I_2$  of average duration  $1/\gamma_2$  with transmission rate  $\beta_2$ . There is no birth or natural mortality. The equations that describe the system are as follows:

$$\frac{d}{dt}I_1 = \frac{S}{N}(\beta_1 I_1 + \beta_2 I_2) - \gamma_1 I_1 \quad \frac{d}{dt}I_2 = \gamma_1 I_1 - \gamma_2 I_2$$

The birth ( $F$ ) and migration ( $G$ ), death and natural-birth rate matrices for this model are:

$$F(t) = \begin{pmatrix} \beta_1 I_1(t) \frac{S(t)}{N} & 0 \\ \beta_2 I_2(t) \frac{S(t)}{N} & 0 \end{pmatrix} \quad G(t) = \begin{pmatrix} 0 & \gamma_1 I_1(t) \\ 0 & 0 \end{pmatrix} \quad \mu(t) = \begin{pmatrix} 0 \\ \gamma_2 I_2(t) \end{pmatrix} \quad \eta(t) = (0 \ 0)$$

### BEAST INPUT

PhyDyn provides XML-ready classes for the specification of complex population models, generation of structured tree intervals and calculation of the log-likelihood of a structured dated tree conditional on a complex demographic history.

The example defined above can be specified as follows:

```
<model spec='PopModelODE' id='twodeme' popParams='@initialValues'
  modelParams='@rates' evaluator='compiled'>
  <definition spec='Definition'> N = S + I1 + I2 </definition>
  <matrixeq type='birth' origin='I1' destination='I1'> beta1*I1*S/N </matrixeq>
  <matrixeq type='birth' origin='I2' destination='I1'> beta2*I2*S/N </matrixeq>
  <matrixeq type='migration' origin='I1' destination='I2'> gamma1*I1 </matrixeq>
  <matrixeq type='death' origin='I1'> gamma1*I1 </matrixeq>
  <matrixeq type='death' origin='I2'> gamma2*I2 </matrixeq>
  <matrixeq type='nondeme' origin='S'>
    b*S - (beta1*I1 + beta2*I2)*S/N
  </matrixeq>
</model>
```

The population model ODE syntax allows the user to define and use complex mathematical expressions such as polynomials and trigonometric functions, as well as the use of time variables and mathematical constants. The population model is then instantiated by providing initial values to the population variables ( $S, I_1, I_2$ ) and assigning values to the free variables in the ODE expressions (rates  $\beta_1, \beta_2, \gamma_1, \gamma_2$ ).

```
<rates spec='ModelParameters' id='rates'>
  <param spec='ParamValue' names='beta1 beta2' values='0.001 0.0001'></param>
  <param spec='ParamValue' names='gamma1' values='@migRate'></param>
  <param spec='ParamValue' names='gamma2' values='0.1111'></param>
</rates>
<pop spec='TrajectoryParameters' id='initialValues' method='classick'
  integrationSteps='1001' rTol='0.01' aTol='1.0e-4' order='3' t0='-0.1'>
  <initialValue spec='ParamValue' names='I1' values='@initI1' />
  <initialValue spec='ParamValue' names='I2 S' values='1 999.0'>
</pop>
```

The implementation makes it possible to sample on both population initial values ( $I_1, I_2$ ) and model parameters ( $\gamma_1, \gamma_2$ ).

### FUTURE WORK AND AVAILABILITY

Currently, PhyDyn works with deterministic demographic models. Future versions may incorporate other population models and methods for simulating trees conditional on an epidemic process, and include semi-parametric epidemiological models.

A beta version of PhyDyn is available at <https://github.com/igococha/phydyn>

### REFERENCES

- [1] Volz EM (2012). Complex population dynamics and the coalescent under neutrality. *Genetics*, Vol: 190, pages: 187-201.
- [2] Bouckaert R et.al. BEAST 2: A Software Platform for Bayesian Evolutionary Analysis. *PLoS Computational Biology*, 10(4).