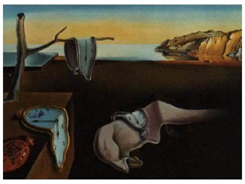

Journal of Statistical Software

MMMMMM YYYY, Volume VV, Issue II. *http://www.jstatsoft.org/*

PhaseTypeR: phase-type distributions in R with reward transformations and a view towards population genetics

Iker Rivas-González
Aarhus University

Lars Nørvang Andersen
Aarhus University

Asger Hobolth
Aarhus University

Abstract

Phase-type distributions are a general class of models that are traditionally used in actuarial sciences and queuing theory, and more recently in population genetics. A phase-type distributed random variable is the time to absorption in a discrete or continuous time Markov chain on a finite state space with an absorbing state. The R package **PhaseTypeR** contains all the key functions—mean, (co)variance, probability density function, cumulative distribution function, quantile function, random sampling and reward transformations—for both continuous (PH) and discrete (DPH) phase-type distributions. Additionally, we have also implemented the multivariate continuous case (MPH) and the multivariate discrete case (MDPH). We illustrate the usage of **PhaseTypeR** in simple examples from population genetics (e.g. the time until the most recent common ancestor or the total number of mutations in an alignment of homologous DNA sequences), and we demonstrate the power of **PhaseTypeR** in more involved applications from population genetics, such as the coalescent with recombination and the structured coalescent. The multivariate distributions and ability to reward-transform are particularly important in population genetics, and a unique feature of **PhaseTypeR**.

Keywords: Ancestral process, coalescent theory, phase-type distributions, population genetics, **PhaseTypeR**.

1. Introduction

Phase-type distributions describe the time until absorption of a continuous or discrete-time Markov chain (Bladt and Nielsen 2017). The probabilistic properties of phase-type distributions (i.e. the probability density function, cumulative distribution function, quantile function, moments and generating functions) are well-described and analytically tractable using matrix manipulations.

Here we present **PhaseTypeR**, an R (R Core Team 2021) package that provides general-use core functions for continuous and discrete phase-type distributions, both for the univariate and the multivariate cases. **PhaseTypeR** can also be used to simulate from the underlying Markov chain of the phase-type objects. Additionally, the package allows for the reward transformation of phase-type distributions. The functions and objects in the package are intuitive and of general use, which enables the users to easily adapt them to their needs. **PhaseTypeR** is available on CRAN (`https://CRAN.R-project.org/package=PhaseTypeR`), and its documentation can be accessed through `https://rivasiker.github.io/PhaseTypeR/`.

The R packages already available for phase-type distributions are mainly tailored to applications in actuarial sciences and risk theory. In these cases, failure times or lifetimes are measured, and the corresponding phase-type distribution is estimated. We briefly summarize the software tools for phase-type distributions, their status and their main purpose:

- Christophe Dutang, Vincent Goulet and Mathieu Pigeon have contributed the R package **actuar** for the actuarial sciences (Dutang, Goulet, and Pigeon 2008), and the package is still under active development. The univariate continuous phase-type distribution is covered in terms of the density, cumulative distribution, moments and moment generating function (see `https://CRAN.R-project.org/package=actuar`).

- Louis Aslett has released an R package called **PhaseType** (Aslett and Wilson 2011; Aslett 2012), which is tailored to the problem of estimating a continuous phase-type distribution from failure times, and is an extension of a Markov Chain Monte Carlo algorithm developed by Bladt, Gonzalez, and Lauritzen (2003). However, the package is not maintained anymore, and it has been removed from CRAN (`https://CRAN.R-project.org/package=PhaseType`).

- Hiroyuki Okamura's **mapfit** R package is concerned with fitting phase-type distributions of failure times in reliability systems (`https://CRAN.R-project.org/package=mapfit`, Okamura 2015; Okamura and Dohi 2015, 2016). Here, the parameters in a phase-type distribution are fitted using maximum likelihood estimation. The package can also fit a phase-type distribution from a probability density function.

- Martin Bladt and Jorge Yslas's recent R package **matrixdist** (`https://CRAN.R-project.org/package=matrixdist`, Bladt and Yslas 2021) fits inhomogeneous phase-type (IPH) distributions (Albrecher and Bladt 2019). The EM-algorithm is used to estimate the parameters in the model. In Albrecher, Bladt, and Yslas (2020, advance online publication), an IPH distribution is fitted to the lifetimes of the Danish population that died in the year 2000 at ages 50 to 100. In the special homogeneous case of the IPH distributions, the package also provides the density, cumulative distribution function, quantile function, moments and opportunity of simulating from the distribution. Our package has the same features for the PH distribution, and additionally we consider reward transformations and the multivariate phase-type (MPH) extension. Furthermore, we provide the same functionality for the class of discrete phase-type (DPH) and multivariate discrete phase-type (MDPH) distributions.

Our implementation of phase-type functions in **PhaseTypeR** is of general use and not restricted to actuarial sciences and risk theory. Moreover, unlike the packages described above, **PhaseTypeR** includes reward transformations, the multivariate extensions of phase-type distributions, and both the discrete and continuous versions.

In this paper we exemplify the applications of the **PhaseTypeR** functions using several quantities in population genetics and, in particular, coalescent theory. More specifically, the time to the most recent common ancestor T_{MRCA}, the total tree length T_{total} and the total branch lengths that give rise to e.g. singletons or doubletons are examples of continuous phase-type distributed variables, and viewed together they are continuous multivariate phase-type distributed (Hobolth, Siri-Jegousse, and Bladt 2019). Additionally, the individual elements of the site frequency spectrum are examples of discrete phase-type distributed variables, and the full site frequency spectrum is multivariate discrete phase-type distributed (Hobolth, Bladt, and Andersen 2021). These statements hold for the standard coalescent process, but also for more general time-homogeneous coalescent models such as the structured coalescent (Wakeley 2009, Section 5), the multiple merger coalescent (Tellier and Lemaire 2014), and the coalescent with recombination (Wakeley 2009, Section 7.2).

This paper presents the basic theory for phase-type distributions and demonstrates how to apply the distributions using **PhaseTypeR**. The paper is organized as follows. Section 2 is concerned with the univariate and multivariate continuous phase-type distributions. The basic phase-type object contains a subintensity matrix and an initial probability vector (potentially with a defect), and the four commonly associated functions `dPH` (probability density function), `pPH` (cumulative distribution function), `qPH` (quantile function) and `rPH` (random sampling). The function `rFullPH` provides a simulation of the full sample path from a continuous phase-type distribution. We then introduce the reward transformations and the multivariate continuous phase-type distribution. Section 3 follows the same structure and introduces the same type of functions (called `dDPH`, `pDPH`, `qDPH`, `rDPH`, `rFullDPH`) for the univariate and multivariate discrete phase-type distributions. In Section 4 we apply phase-type theory to understand the ancestral recombination graph for two loci and two samples, and in Section 5 we demonstrate how phase-type theory can be used to learn about a structured population. The paper ends with a conclusion and a discussion of future extensions and applications of our package.

2. Continuous phase-type distributions

2.1. Theory for the phase-type distribution

A continuous phase-type distribution is a sum of exponential distributions that occur sequentially until absorption. More specifically, a phase-type distribution is the time to absorption of a Markov jump process.

Following the notation in Bladt and Nielsen (2017), let $\{X_t\}_{t\geq 0}$ be a Markov jump process with p transient states and a single absorbing state. The time until absorption τ of such a process then follows a continuous phase-type distribution, where the rate matrix of the underlying Markov jump process $\boldsymbol{\Lambda}$ is given as

$$\boldsymbol{\Lambda} = \begin{pmatrix} \boldsymbol{T} & \boldsymbol{t} \\ \boldsymbol{0} & 0 \end{pmatrix}. \tag{1}$$

The $p \times p$ matrix $\boldsymbol{T}$ is the subintensity matrix, and the elements are the transition rates between the transient states. Because of the properties of rate matrices, all the rows of $\boldsymbol{\Lambda}$ sum to 0 (i.e. $\boldsymbol{\Lambda e} = \boldsymbol{0}$, where $\boldsymbol{e}$ is a vector of ones), which means that the phase-type distribution can be defined by the subintensity matrix $\boldsymbol{T}$, while the exit rate column vector $\boldsymbol{t}$

Quantity	Formula
Mean	$\mathrm{E}(\tau) = \boldsymbol{\pi}(-\boldsymbol{T})^{-1}\boldsymbol{e}$
Moments	$\mathrm{E}(\tau^n) = n!\boldsymbol{\pi}(-\boldsymbol{T})^{-n}\boldsymbol{e}$
Variance	$\mathrm{V}(\tau) = \mathrm{E}(\tau^2) - \mathrm{E}(\tau)^2$
Density	$f(x) = \boldsymbol{\pi}\exp(\boldsymbol{T}x)(-\boldsymbol{T}\boldsymbol{e}),\ x \geq 0$
Cumulative distribution	$F(x) = 1 - \boldsymbol{\pi}\exp(\boldsymbol{T}x)\boldsymbol{e},\ x \geq 0$

Table 1: Formulas for the mean, moments, variance, probability density function and cumulative distribution function of the continuous phase-type distribution. Here, $\boldsymbol{\pi}$ is the vector of initial probabilities, $\boldsymbol{T}$ is the subintensity matrix and $\boldsymbol{e}$ is a vector of ones.

is given by $\boldsymbol{t} = -\boldsymbol{T}\boldsymbol{e}$. Additionally, we have a vector of size p corresponding to the initial probabilities $\boldsymbol{\pi}$, such that $\tau \sim \mathrm{PH}(\boldsymbol{\pi}, \boldsymbol{T})$ (i.e. $P(X_0 = i) = \pi_i, i = 1, \ldots, p$). The sum of the initial probabilities ($\boldsymbol{\pi}\boldsymbol{e}$) might sum to less than 1. If this is the case, then we can define the defect as $1 - \boldsymbol{\pi}\boldsymbol{e}$, which corresponds to the probability of starting in the absorbing state without passing through any of the transient states (i.e. $P(X_0 = p+1) = 1 - \boldsymbol{\pi}\boldsymbol{e}$).

The properties of phase-type distributions can easily be calculated due to their matrix-form representation. The mean, variance, probability density function and cumulative distribution function for the continuous phase-type distribution are summarized in Table 1. For the mathematical derivations of these formulas we refer to Bladt and Nielsen (2017).

Continuous phase-type distributions can be linearly transformed via rewards (Bladt and Nielsen 2017). This is achieved by assigning a non-negative reward to each of the transient states $1, \ldots, p$. The resulting distribution is also a phase-type distribution. Let the rewards be given by the function $r(i)$, $i = 1, \ldots, p$, and summarized in the vector $\boldsymbol{r} = (r_1, \ldots, r_p)$, where $r(i) = r_i$. Consider the reward-transformed random variable

$$\tau^* = \int_0^\tau r(X_t)dt,$$

where $\{X_t\}_{t\geq 0}$ is the underlying jump Markov process for the original phase-type distribution $\tau \sim \mathrm{PH}(\boldsymbol{\pi}, \boldsymbol{T})$. We have that τ^* is phase-type distributed with $\boldsymbol{\pi}^*$ and $\boldsymbol{T^*}$ denoting the initial distribution and subintensity matrix of this distribution, respectively. If all the elements in the reward vector $\boldsymbol{r}$ are strictly positive, then $\boldsymbol{T}^* = \mathrm{diag}(\boldsymbol{1/r})\boldsymbol{T}$, where $\mathrm{diag}(\boldsymbol{1/r})$ is the $p \times p$ diagonal matrix with $1/r_i$ on the diagonal, while the initial probability vector stays the same ($\boldsymbol{\pi}^* = \boldsymbol{\pi}$). If $\boldsymbol{r}$ contains zero-valued rewards, then the states should with a reward of zero can be excluded. As a result, the transient states should be re-defined and the resulting $\boldsymbol{\pi}^*$ and $\boldsymbol{T}^*$ are lower-dimensional; we refer to Theorem 3.1.33 in Bladt and Nielsen (2017) for the mathematical details.

If several univariate continuous phase-type distributions are defined by the same subintensity matrix but different reward vectors $(\boldsymbol{r_1}, \boldsymbol{r_2}, \ldots, \boldsymbol{r_m})$, then we represent the system as a multivariate continuous phase-type distribution $\mathrm{MPH}(\boldsymbol{\pi}, \boldsymbol{T}, \boldsymbol{R})$, where $\boldsymbol{\pi}$ is the initial probability vector, $\boldsymbol{T}$ is the subintensity matrix, and $\boldsymbol{R} = (\boldsymbol{r_1}, \boldsymbol{r_2}, \ldots, \boldsymbol{r}_m)$ is the $p \times m$ reward matrix.

2.2. Defining the phase-type object

To exemplify the usage of **PhaseTypeR**, we will use phase-type representations of common summary statistics in population genetics. Perhaps the most prominent example is the time until the most recent common ancestor T_{MRCA}, which can be defined as a convolution of exponential distributions following the standard coalescent process (see Figure 1). This means that the T_{MRCA} follows a univariate continuous phase-type distribution $T_{\mathrm{MRCA}} \sim \mathrm{PH}(\boldsymbol{\pi}, \boldsymbol{T})$, with initial probabilities $\boldsymbol{\pi}$ and subintensity matrix $\boldsymbol{T}$ (Hobolth *et al.* 2019). For the standard coalescent model (Kingman 1982) and a sample size of four chromosomes, the subintensity matrix for the T_{MRCA} is given by

$$\boldsymbol{T} = \begin{pmatrix} -6 & 6 & 0 \\ 0 & -3 & 3 \\ 0 & 0 & -1 \end{pmatrix}, \tag{2}$$

with initial probability vector $\boldsymbol{\pi} = (1, 0, 0)$.

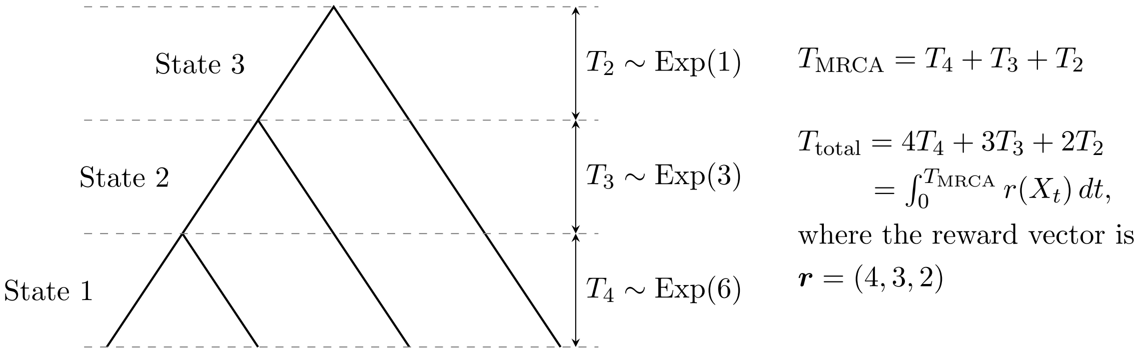
State 3
State 2
State 1
$T_2 \sim \mathrm{Exp}(1)$
$T_3 \sim \mathrm{Exp}(3)$
$T_4 \sim \mathrm{Exp}(6)$
$T_{\mathrm{MRCA}} = T_4 + T_3 + T_2$
$T_{\mathrm{total}} = 4T_4 + 3T_3 + 2T_2$
$= \int_0^{T_{\mathrm{MRCA}}} r(X_t)\, dt,$
where the reward vector is
$\boldsymbol{r} = (4, 3, 2)$

Figure 1: Coalescent process for $n = 4$ samples. The underlying Markov jump process $\{X_t\}$ is in state $X_t = 1$ for $0 \le t < T_4$, $X_t = 2$ for $T_4 \le t < T_4 + T_3$, and $X_t = 3$ for $T_4 + T_3 \le t < T_4 + T_3 + T_2$. The process is in the absorbing state for $t \ge T_4 + T_3 + T_2$. The rewards in the states correspond to the number of lineages and the reward-transformed variable T_{total} correspond to the total tree length.

We can specify the initial probabilities and the subintensity matrix for this univariate continuous phase-type distribution using the `PH()` function:

```
R> subintensity_matrix <- matrix(c(-6,  6,  0,
+                                   0, -3,  3,
+                                   0,  0, -1),
+                               ncol = 3, byrow = T)
R> initial_probabilities <- c(1, 0, 0)
R> T_MRCA <- PH(subintensity_matrix, initial_probabilities)
R> T_MRCA

$subint_mat
     [,1] [,2] [,3]
[1,]   -6    6    0
[2,]    0   -3    3
[3,]    0    0   -1

$init_probs
```

```
     [,1] [,2] [,3]
[1,]    1    0    0

$defect
[1] 0

attr(,"class")
[1] "cont_phase_type"
```

2.3. The mean and the variance of a phase-type distribution

The mean and variance of a phase-type object can be accessed by `mean()` and `var()`, respectively. For the phase-type representation of T_{MRCA} defined above, `mean(T_MRCA)` yields 1.5 and `mean(T_MRCA)` yields 1.138889, which match the well-known results from classical population genetics formulas (Wakeley 2009, Section 3.3).

2.4. Distribution functions and random sampling for a phase-type distribution

PhaseTypeR uses standard R suffixes for the probability density function (`dPH`), the cumulative distribution function (`pPH`), the quantile function (`qPH`) and the random sampling function (`rPH`) for univariate continuous phase-type distributions:

```
R> dPH(c(0.1, 0.5, 0.8), T_MRCA)

[1] 0.06482665 0.48210919 0.54651397

R> pPH(c(0.1, 0.5, 0.8), T_MRCA)

[1] 0.002348541 0.121417559 0.280279868

R> qPH(c(0.05, 0.5, 0.95), T_MRCA)

[1] 0.3302855 1.2328314 3.5830871

R> set.seed(3)
R> rPH(3, T_MRCA)

[1] 0.6459884 0.1019513 1.0577725
```

Sometimes, it is of interest to retrieve the full sample path of the Markov jump process. The user can achieve so using the `rFullPH` function, which returns a data frame containing all the visited states and the time spent in each:

```
R> set.seed(3)
R> rFullPH(T_MRCA)
```

```
  state      time
1     1 0.1025055
2     2 0.3346948
3     3 0.2087881
```

2.5. Reward transformation

The T_{MRCA} is tightly related to the total tree length, or T_{total} (Hobolth *et al.* 2019). More specifically, T_{total} is a linear transformation of T_{MRCA}, so its phase-type representation can be obtained by a reward transformation. The reward vector needed is $\boldsymbol{r} = (n, n-1, ..., 2)$, where n is the sample size (recall Figure 1).

Reward transformation in **PhaseTypeR** can be done using the `reward_phase_type()` function. For the case of $n = 4$ the reward vector is $\boldsymbol{r} = (4, 3, 2)$, and a phase-type representation of T_{total} can be obtained by reward-transforming T_{MRCA}:

```
R> reward <- c(4, 3, 2)
R> T_total <- reward_phase_type(T_MRCA, reward)
R> T_total

$subint_mat
     [,1] [,2] [,3]
[1,] -1.5  1.5  0.0
[2,]  0.0 -1.0  1.0
[3,]  0.0  0.0 -0.5

$init_probs
     [,1] [,2] [,3]
[1,]    1    0    0

$defect
[1] 0

attr(,"class")
[1] "cont_phase_type"
```

The mean and variance of the total branch length are given by

```
R> c(mean(T_total), var(T_total))

[1] 3.666667 5.444444
```

We note once again that these results match the ones derived from classical population genetic formulas (e.g. Wakeley 2009, Section 3.3).

2.6. The multivariate continuous phase-type distribution

Similar to the construction of T_{total}, we can reward-transform the phase-type representation of T_{MRCA} to get the distribution of the total branch length leading to each of the elements of the site frequency spectrum $(\xi_1, \ldots, \xi_{n-1})$, i.e. singletons (ξ_1), doubletons (ξ_2), etc. In order to do so for $n = 4$, we first need to extend the subintensity matrix $\boldsymbol{T}$ by sub-dividing state 3 into two. The reason for this is that 1/3 of the times the second coalescent will lead to the creation of two doubleton branches, while 2/3 of the times it will lead to one singleton branch and one tripleton branch (see Figure 2). The resulting subintensity matrix is given by

$$\boldsymbol{T'} = \begin{pmatrix} -6 & 6 & 0 & 0 \\ 0 & -3 & 1 & 2 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}. \tag{3}$$

Note that this is a less efficient but equal phase-type representation of T_{MRCA} if the initial probability vector is $\boldsymbol{\pi'} = (1, 0, 0, 0)$.

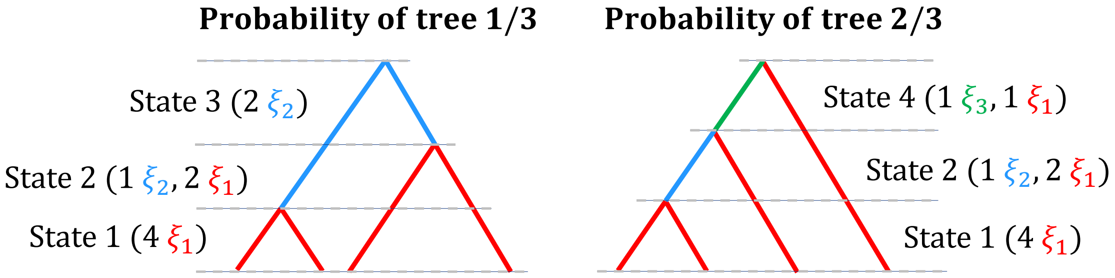

Figure 2: Coalescent process for $n = 4$ samples. State 1 contains 4 singleton branches (mutations in singleton branches are only present in one sample, represented in red), state 2 contains 2 singleton branches and 1 doubleton branch (mutations in doubleton branches are present in two samples, represented in blue), state 3 contains 2 doubleton branches, and state 4 contains 1 singleton and 1 tripleton branch (mutations in tripleton branches are present in three samples, represented in green).

Following Hobolth *et al.* (2019), if we transform T_{MRCA} with a reward of $\boldsymbol{r_1} = (4, 2, 0, 1)$, we get a phase-type representation of the total branch length leading to singletons, denoted L_1. Knowing that the reward vectors for doubletons and tripletons are $\boldsymbol{r_2} = (0, 1, 2, 0)$ and $\boldsymbol{r_3} = (0, 0, 0, 1)$, instead of reward-transforming each element of the site frequency spectrum separately, we can define a multivariate continuous phase-type distribution $\boldsymbol{L} = (L_1, L_2, L_3) \sim \mathrm{MPH}(\boldsymbol{\pi'}, \boldsymbol{T'}, \boldsymbol{R})$ with initial distribution $\boldsymbol{\pi'} = (1, 0, 0, 0)$, subintensity matrix $\boldsymbol{T'}$ and reward matrix $\boldsymbol{R} = (\boldsymbol{r_1}, \boldsymbol{r_2}, \boldsymbol{r_3})$.

Multivariate continuous phase-type distributions are implemented in **PhaseTypeR** as follows:

```
R> subintensity_matrix <- matrix(c(-6, 6,  0,  0,
+                                   0, -3,  1,  2,
+                                   0,  0, -1,  0,
+                                   0,  0,  0, -1),
+                                 ncol = 4, byrow = T)
R> initial_probabilities <- c(1, 0, 0, 0)
R> reward_matrix <- matrix(
+     c(4,2,0,1,
+       0,1,2,0,
+        0,0,0,1),
+      nrow = 4)
R> L <- MPH(subintensity_matrix, initial_probabilities, reward_matrix)
R> L

$subint_mat
     [,1] [,2] [,3] [,4]
[1,]   -6    6    0    0
[2,]    0   -3    1    2
[3,]    0    0   -1    0
[4,]    0    0    0   -1

$init_probs
     [,1] [,2] [,3] [,4]
[1,]    1    0    0    0

$reward_mat
     [,1] [,2] [,3]
[1,]    4    0    0
[2,]    2    1    0
[3,]    0    2    0
[4,]    1    0    1

$defect
[1] 0

attr(,"class")
[1] "mult_cont_phase_type"
```

This type of multivariate representation is useful to calculate the variance-covariance matrix of the variables. For the multivariate case, this can be accessed using `var()`:

```
R> var(L)

           [,1]       [,2]       [,3]
[1,]  1.7777778 -0.2222222  0.8888889
[2,] -0.2222222  2.3333333 -0.4444444
[3,]  0.8888889 -0.4444444  0.8888889
```

Here, the diagonal is the variance of each of the elements (total branch length leading to singletons, doubletons and tripletons in this case), and the off-diagonal values are the covariances between the different elements. The analytical formula for calculating the covariance can be found in Theorem 8.1.5 in Bladt and Nielsen (2017).

Moreover, **PhaseTypeR** also computes univariate quantities related to the marginal distributions of the MPH, i.e. the probability density function (`dMPH`), the cumulative distribution function (`pMPH`) and the quantile function (`qMPH`). Random draws (`rMPH`) and random draws with full path (`rFullMPH`) for the multivariate case use the same underlying sample path for the Markov jump process.

3. Discrete phase-type distributions

3.1. Theory for the discrete phase-type distribution

A discrete phase-type distribution describes a process of geometric distributions that occur sequentially until absorption. It is similar to the continuous case, but the underlying process is a discrete time absorbing Markov chain instead of a Markov jump process.

If we define the Markov chain as $\{X_n\}_{n\in\mathbb{N}}$, which has p transient states and one absorbing state, the discrete time until absorption τ follows a discrete phase-type distribution. The transition probability matrix $\boldsymbol{P}$ of the Markov chain is then defined as

$$\boldsymbol{P} = \begin{pmatrix} \boldsymbol{T} & \boldsymbol{t} \\ \boldsymbol{0} & 1 \end{pmatrix}. \tag{4}$$

The transition probabilities among the transient states are therefore contained in the $p \times p$ subtransition matrix $\boldsymbol{T}$. Similar to the continuous case, the discrete phase-type distribution can be described solely by $\boldsymbol{T}$. Since all the rows in $\boldsymbol{P}$ sum to 1 we have $\boldsymbol{Pe} = \boldsymbol{e}$ and the exit probability vector is given by $\boldsymbol{t} = \boldsymbol{e} - \boldsymbol{Te} = (\boldsymbol{I} - \boldsymbol{T})\boldsymbol{e}$, where $\boldsymbol{I}$ is a $p \times p$ identity matrix. If we let $\boldsymbol{\pi}$ be the vector of initial probabilities, then $\tau \sim \mathrm{DPH}(\boldsymbol{\pi}, \boldsymbol{T})$.

Similar to the continuous case, the mean, variance, probability density function and cumulative distribution function for the discrete phase-type distribution can be defined using matrix manipulation (Bladt and Nielsen 2017), and they are summarized in Table 2.

Quantity	Formula
Mean	$\mathrm{E}(\tau) = \boldsymbol{\pi}(\boldsymbol{I} - \boldsymbol{T})^{-1}\boldsymbol{e}$
Moments	$\mathrm{E}(\tau^n) = n!\boldsymbol{\pi}(\boldsymbol{I} - \boldsymbol{T})^{-n}\boldsymbol{e}$
Variance	$\mathrm{V}(\tau) = \mathrm{E}(\tau^2) - \mathrm{E}(\tau)^2$
Density	$f(x) = \boldsymbol{\pi}\boldsymbol{T}^{x-1}\boldsymbol{t}$, $x \geq 1$
Cumulative distribution	$F(x) = 1 - \boldsymbol{\pi}\boldsymbol{T}^{x}\boldsymbol{e}$, $x \geq 1$

Table 2: Formulas for the mean, moments, variance, probability density and cumulative distribution function of the discrete phase-type distribution. Here, $\boldsymbol{\pi}$ is the vector of initial probability, $\boldsymbol{T}$ is the subtransition matrix, $\boldsymbol{t}$ is the exit probability vector, $\boldsymbol{e}$ is a vector of ones, and $\boldsymbol{I}$ is an identity matrix.

Additionally, discrete phase-type distributions can also be transformed with non-negative integer rewards. If a reward for a certain state is set to 0, then that state is removed from the subtransition matrix. If instead it is set to a positive integer larger than 1, then the subtransition matrix is extended to "force" the Markov chain to pass through a state several times. The full mathematical construction of reward transformations for the discrete case is presented in Theorem 5.2 in Campillo Navarro (2018).

Similar to the continuous case, multivariate discrete phase-type distributions can be constructed by combining univariate distributions that share the same subtransition matrix but have different rewards $(\boldsymbol{r_1}, \boldsymbol{r_2}, \dots, \boldsymbol{r_m})$. The resulting joint distribution $\mathrm{MDPH}(\boldsymbol{\pi}, \boldsymbol{T}, \boldsymbol{R})$ contains an initial probability vector $\boldsymbol{\pi}$, a subtransition matrix $\boldsymbol{T}$ and a $p \times m$ reward matrix $\boldsymbol{R} = (\boldsymbol{r_1}, \boldsymbol{r_2}, \dots, \boldsymbol{r_m})$.

3.2. Defining the discrete phase-type object

Another summary statistic in population genetics that can be represented using phase-type theory is the number of segregating sites S_{total} (Hobolth *et al.* 2019). Consider the coalescent process with a sample size of $n = 3$, i.e. we have a state 1 with three singleton branches and state 2 with one singleton branch and one doubleton branch (Kingman 1982). The coalescent rate in state 1 is $\binom{3}{2} = 3$ and the mutation rate is $\theta/2$ on each of the three branches (so $3\theta/2$ in total). The probability that the first event in the ancestral process is a mutation before a coalescent is given by the mutation rate relative to the total rate, a situation which corresponds to two competing exponential distributions (see e.g. Wakeley 2009, equations (2.60) and (4.5)). Therefore, the number of mutations when three branches are present is geometrically distributed with probability of mutation

$$p_1 = \frac{3\theta/2}{3 + 3\theta/2} = \frac{\theta}{2+\theta}. \tag{5}$$

Similarly, when two branches are present, the coalescent rate is $\binom{2}{2} = 1$ and the total mutation rate is $2\theta/2$. Thus, the number of mutations in this case is geometrically distributed with probability of mutation

$$p_2 = \frac{2\theta/2}{1 + 2\theta/2} = \frac{\theta}{1+\theta}. \tag{6}$$

We may describe the situation using a discrete phase-type distribution with subtransition probability matrix

$$\boldsymbol{T} = \begin{pmatrix} p_1 & p_{12} \\ 0 & p_2 \end{pmatrix} = \begin{pmatrix} p_1 & (1-p_1)\,p_2 \\ 0 & p_2 \end{pmatrix} = \begin{pmatrix} \frac{\theta}{2+\theta} & \frac{2}{2+\theta}\frac{\theta}{1+\theta} \\ 0 & \frac{\theta}{1+\theta} \end{pmatrix}, \tag{7}$$

Here, a jump into state 1 corresponds to a mutation on the level of the tree with three branches, and a jump into state 2 corresponds to a mutation on the level of the tree with two branches. We can also start in a situation with no mutation, which corresponds to directly jumping to the absorbing state. In order to model this, we can work with a defective initial distribution given by $\boldsymbol{\pi} = (p_1, p_{12})$. The probability of zero jumps (mutations) is then

$$1 - p_1 - p_{12} = \frac{2}{2+\theta}\frac{1}{1+\theta},$$

which corresponds to the defect. The total number of mutations, thus, follows a univariate discrete phase-type distribution $S_{\text{total}} \sim \mathrm{DPH}(\boldsymbol{\pi}, \boldsymbol{T})$. We remark that this same distribution arises from adding Poisson mutations on the phase-type distributed total tree length (Theorem 3.5, eq. (19) in Hobolth *et al.* (2019)).

In **PhaseTypeR**, it is straightforward to specify a univariate discrete phase-type distributions with `DPH()`. For the case of S_{total} when $n = 3$ and $\theta = 3$:

```
R> tht <- 3
R> p_1 <- tht/(2+tht)
```

```
R> p_12 <- ( 2/(2+tht) )*( tht/(1+tht) )
R> p_2 <- tht/(1+tht)
R> T_mat <- matrix(c(p_1, p_12,
+                    0,    p_2),
+                  ncol = 2, byrow = T)
R> init_probs <- c(p_1, p_12)
R> S_total <- DPH(T_mat, init_probs)
R> S_total

$subint_mat
     [,1] [,2]
[1,]  0.6 0.30
[2,]  0.0 0.75

$init_probs
     [,1] [,2]
[1,]  0.6  0.3

$defect
[1] 0.1

attr(,"class")
[1] "disc_phase_type"
```

3.3. The mean and the variance of a discrete phase-type distribution

Similar to the continuous case, the mean and variance of the discrete phase-type object can be computed by `mean()` and `var()`, respectively. For the phase-type representation of S_{total} defined above, `mean(S_total)` yields 4.5 and `var(S_total)` yields 15.75. These results match the ones derived from classical population genetic formulas (e.g. Wakeley 2009, Section 4.1.1).

3.4. Distribution functions and sampling for a discrete phase-type distribution

PhaseTypeR also contains functions for the probability density function (`dDPH`), the cumulative distribution function (`pDPH`), the quantile function (`qDPH`) and random sampling (`rDPH`) of univariate discrete phase-type distributions:

```
R> dDPH(c(0, 1, 2, 10), S_total)

[1] 0.10000000 0.13500000 0.13725000 0.02573811

R> pDPH(c(0, 1, 2, 10), S_total)

[1] 0.1000000 0.2350000 0.3722500 0.9191577

R> qDPH(c(0.05, 0.5, 0.95), S_total)
```

```
[1]  0  4 12
```

```
R> set.seed(3)
R> rDPH(5, S_total)
```

```
[1] 14  3  8  8 12
```

We can also simulate a sample path from the Markov chain by using the `rFullDPH()` function. This returns a data frame with the visited states and the time spent in each of them:

```
R> set.seed(45)
R> rFullDPH(S_total)
```

```
  state time
1     1    1
2     2   13
```

3.5. Reward transformation

While S_{total} does not distinguish the different types of segregating sites, sometimes we are interested in knowing the type of mutations based on the site frequency spectrum. When $n = 3$, there are two types of segregating sites, namely singletons (ξ_1) and doubletons (ξ_2). Mutations in state 1 are always singletons, but mutations in state 2 are singletons with probability 1/2 and doubletons with probability 1/2. We therefore extend the subtransition probability matrix to

$$\boldsymbol{M_\xi} = \begin{pmatrix} p_1 & \frac{1}{2}p_{12} & \frac{1}{2}p_{12} \\ 0 & \frac{1}{2}p_2 & \frac{1}{2}p_2 \\ 0 & \frac{1}{2}p_2 & \frac{1}{2}p_2 \end{pmatrix}, \tag{8}$$

where the new state 2 corresponds to singletons in the old state 2 and the new state 3 corresponds to doubletons in the old state 2. If we define an initial probability vector of $\boldsymbol{\pi_\xi} = (p_1, \frac{1}{2}p_{12}, \frac{1}{2}p_{12})$, we can define a discrete phase-type distribution $S_{\text{total}} \sim \text{DPH}(\boldsymbol{\pi_\xi}, \boldsymbol{M_\xi})$. This way of defining S_{total} is a more inefficient though equivalent representation compared to the definition in the previous section. However, we can now transform the phase-type distribution via rewards to get discrete phase-type representations of the different elements of the site frequency spectrum. For example, $\xi_1 \sim \text{DPH}(\boldsymbol{\pi_1}, \boldsymbol{M_1})$, which can be derived by reward-transforming S_{total} with a reward vector of $\boldsymbol{r_1} = (1, 1, 0)$. This can be done in **PhaseTypeR** using the `reward_phase_type()` function:

```
R> T_mat <- matrix(c(p_1, p_12/2, p_12/2,
+                      0,   p_2/2,  p_2/2,
+                      0,   p_2/2,  p_2/2),
+                    ncol = 3, byrow = T)
R> init_probs <- c(p_1, p_12/2, p_12/2)
R> S_total <- DPH(T_mat, init_probs)
R> singletons <- reward_phase_type(S_total, c(1, 1, 0))
R> singletons
```

```
$subint_mat
     [,1] [,2]
[1,]  0.6 0.30
[2,]  0.0 0.75

$init_probs
     [,1] [,2]
[1,]  0.6  0.3

$defect
[1] 0.1

attr(,"class")
[1] "disc_phase_type"
```

And similarly, for doubletons with a reward vector of $\boldsymbol{r_2} = (0, 0, 1)$:

```
R> doubletons <- reward_phase_type(S_total, c(0, 0, 1))
R> doubletons

$subint_mat
     [,1]
[1,]  0.6

$init_probs
     [,1]
[1,]  0.6

$defect
[1] 0.4

attr(,"class")
[1] "disc_phase_type"
```

Note that when $\theta = 3$

```
R> c(mean(singletons), mean(doubletons))

[1] 3.0 1.5
```

This matches the famous result from coalescent theory (e.g. Wakeley 2009, Section 4.1.3), which states that the mean of the elements in the site frequency spectrum is $\mathrm{E}[\xi_i] = \theta/i$, $i = 1, \ldots, n-1$.

3.6. The multivariate discrete phase-type distribution

Naturally, the joint site frequency spectrum (ξ_1, ξ_2) is multivariate discrete phase-type distributed with initial distribution $\boldsymbol{\pi} = (p_1, p_{12})$, subtransition matrix $\boldsymbol{M_\xi}$, and reward vectors

$\boldsymbol{r_1} = (1, 1, 0)$ and $\boldsymbol{r_2} = (0, 0, 1)$, i.e.

$$(\xi_1, \xi_2) \sim \text{MDPH}(\boldsymbol{\pi}, \boldsymbol{M_\xi}, \boldsymbol{R_\xi}),$$

with $\boldsymbol{R_\xi} = (\boldsymbol{r_1}, \boldsymbol{r_2})$.

Using **PhaseTypeR**:

```
R> SFS <- MDPH(T_mat, init_probs, matrix(c(1, 1, 0, 0, 0, 1), nrow = 3))
R> SFS
```

```
$subint_mat
     [,1]  [,2]  [,3]
[1,]  0.6 0.150 0.150
[2,]  0.0 0.375 0.375
[3,]  0.0 0.375 0.375

$init_probs
     [,1] [,2] [,3]
[1,]  0.6 0.15 0.15

$reward_mat
     [,1] [,2]
[1,]    1    0
[2,]    1    0
[3,]    0    1

$defect
[1] 0.1

attr(,"class")
[1] "mult_disc_phase_type"
```

This construction can be extended to any sample size n with site frequency spectrum (SFS) $(\xi_1, \ldots, \xi_{n-1})$. The general situation is described in Hobolth *et al.* (2021), and the special case $n = 4$ is illustrated in detail in Section 4 in that paper.

PhaseTypeR can be used to calculate the variance-covariance matrix of a multivariate discrete phase-type distribution:

```
R> var(SFS)
```

```
     [,1] [,2]
[1,] 7.50 2.25
[2,] 2.25 3.75
```

Here, the diagonal corresponds to the variance in the number of singletons and doubletons, respectively. The covariance is provided in the off-diagonal values, whose formula can be consulted in Campillo Navarro (2018).

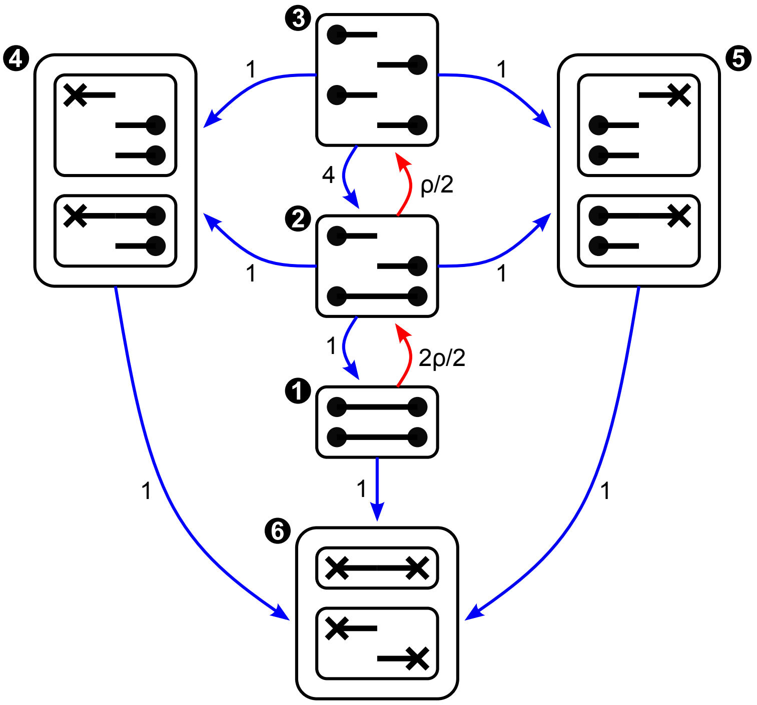

Figure 3: State space and transition rates for the two-locus ancestral recombination graph (ARG). Coalescent events are represented with blue arrows, while recombination events are marked in red. The corresponding coalescent or recombination rates are labeled next to each arrow.

Similar to the continuous case, **PhaseTypeR** also contains functions for calculating univariate quantities of the marginal distributions of MDPH. These include the probability density function (`dMDPH`), the cumulative distribution function (`pMDPH`) and the quantile function (`qMDPH`). Moreover, random draws (`rMDPH`) and random draws with full path (`rFullMDPH`) for the multivariate discrete case use the same underlying sample path for the Markov chain.

4. The coalescent with recombination

The traditional procedure for deriving the correlation between the branch lengths in two loci for a sample of size two is by a first-step analysis (e.g. Wakeley 2009, Section 7). In this section we demonstrate how to use phase-type theory to obtain the result.

The state space and transition rates for the two-locus ancestral recombination graph is shown in Figure 3. The filled circles represent material ancestral to the sample, and the crosses indicate that the most common ancestor has been found. The lines between the circles or crosses indicate if the ancestral material is present on the same chromosome. The starting state is state 1 at present day with two samples from the same chromosome.

The time τ when both loci have found their common ancestor is $\mathrm{PH}(\boldsymbol{\alpha}, \boldsymbol{S})$ distributed with

$\boldsymbol{\alpha} = (1, 0, 0, 0, 0)$ and

$$\boldsymbol{S} = \begin{pmatrix} -(1+2\rho/2) & 2\rho/2 & 0 & 0 & 0 \\ 1 & -(3+\rho/2) & \rho/2 & 1 & 1 \\ 0 & 4 & -6 & 1 & 1 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & -1 \end{pmatrix}. \tag{9}$$

The tree height T_{left} in the left locus is the first time the ancestral process $\{X(t) : t \geq 0\}$ enters state 4 or state 6 or, equivalently, the time spent in state 1, 2, 3 and 5 before absorption in state 6. We therefore have

$$T_{\text{left}} = \min\left\{t \geq 0 : X(t) \in \{4, 6\}\right\} = \int_0^{\tau} \boldsymbol{r}_{\text{left}}(X_t)dt$$

with the reward vector $\boldsymbol{r}_{\text{left}} = (1, 1, 1, 0, 1)$. Similarly, the tree height T_{right} in the right locus is the first time the ancestral process enters state 5 or state 6 or, equivalently, the time spent in state 1, 2, 3 and 4 before absorption in state 6. We therefore have

$$T_{\text{right}} = \min\left\{t \geq 0 : X(t) \in \{5, 6\}\right\} = \int_0^{\tau} \boldsymbol{r}_{\text{right}}(X_t)dt$$

with the reward vector $\boldsymbol{r}_{\text{right}} = (1, 1, 1, 1, 0)$. A classical result in population genetics gives the covariance between the two tree heights

$$\text{Cov}(T_{\text{left}}, T_{\text{right}}) = \frac{\rho + 18}{\rho^2 + 13\rho + 18},$$

and we note that for large recombination rates $\text{Cov}(T_{\text{left}}, T_{\text{right}})$ is close to zero, and for small recombination rates it is close to one. Note that T_{left} and T_{right} are both exponentially distributed with a rate of 1, so $\text{Var}(T_{\text{left}}) = \text{Var}(T_{\text{right}}) = 1$, and, consequently, $\text{Cor}(T_{\text{left}}, T_{\text{right}}) = \text{Cov}(T_{\text{left}}, T_{\text{right}})$ (see also Wakeley 2009, equation (3.10)). Moreover, as shown by a simple proof in Wilton, Carmi, and Hobolth (2015), we have that $P(T_{\text{left}} = T_{\text{right}}) = \text{Cov}(T_{\text{left}}, T_{\text{right}})$.

An implementation using **PhaseTypeR** simply consists of specifying the initial distribution, rate matrix for the ancestral process, rewards for the two tree heights, and calling the variance function for the multivariate phase-type distribution.

```
R> recomb_rate <- 0.3
R> ARG_subint_mat <- function(recomb_rate) {
+    matrix(
+      c(-(1+2*recomb_rate/2),   2*recomb_rate/2,    0,             0,  0,
+          1,                  -(3+recomb_rate/2),   recomb_rate/2, 1,  1,
+          0,                    4,                 -6,             1,  1,
+          0,                    0,                  0,            -1,  0,
+          0,                    0,                  0,             0, -1),
+    nrow=5, byrow=TRUE)
+  }
R> subintensity_matrix <- ARG_subint_mat(recomb_rate)
R> initial_probabilities <- c(1, 0, 0, 0, 0)
R> reward_left <- c(1, 1, 1, 0, 1)
```

```
R> reward_right <- c(1, 1, 1, 1, 0)
R> T_joint <- MPH(subintensity_matrix,
+                 initial_probabilities,
+                 matrix(c(reward_left, reward_right), nrow = 5))
R> c(var(T_joint)[1, 2],
+    (recomb_rate + 18) / (recomb_rate ^ 2 + 13 * recomb_rate + 18))

[1] 0.8321965 0.8321965
```

We can see that the phase-type result is equal to the classical formula provided above.

From this multivariate phase-type representation of the ARG, we can simulate, for example, 1,000 draws from the joint distribution of $(T_{\text{left}}, T_{\text{right}})$ using `rMPH(1000, T_joint)` in **PhaseTypeR**. If the recombination rate ρ is set to a small value, then most of the draws will result in $T_{\text{left}} = T_{\text{right}}$, and the joint density will concentrate along the diagonal, as shown in Figure 4, left (Simonsen and Churchill 1997). If instead ρ is large, then most of the draws will result in $T_{\text{left}} \neq T_{\text{right}}$ (Figure 4, right).

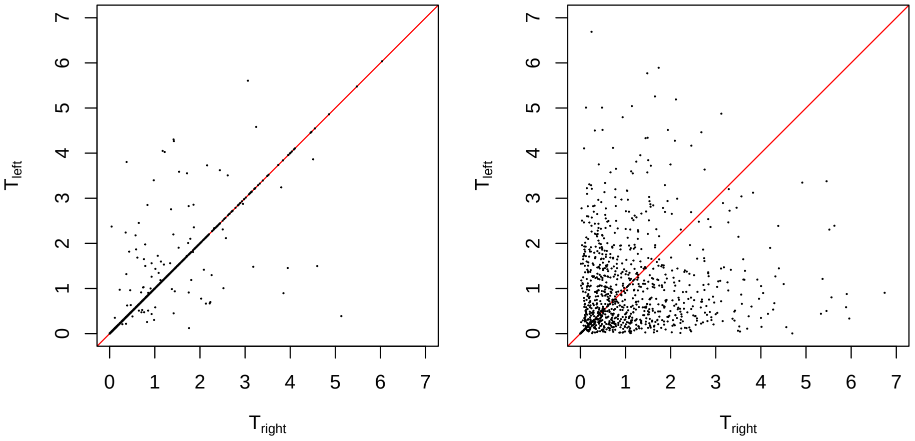

Figure 4: Scatter plot of a simulation of 1,000 draws from the joint distribution of the coalescent times $(T_{\text{left}}, T_{\text{right}})$ in the two-loci, two-sample ARG. The recombination rate ρ was set to 0.166 and 11.316 in the left and right plots, respectively, such that $P(T_{\text{left}} = T_{\text{right}})$ equals 0.9 or 0.1. The red diagonal identity line is plotted as a reference.

5. The structured coalescent

We now consider the structured coalescent, and use the notation and set-up described in Section 5.2 in Wakeley (2009). The number of demes (or sub-populations) is $D \geq 2$, and we assume that the rate of migration for a lineage is the same between any two demes and is given by $M/(2(D-1))$. We also assume that the coalescent rate for two lineages within any deme is one. We focus on moments and distributions of coalescent times for samples of size 2. Since this model is completely symmetric we only need three states: a 'within' state where the two lineages are in the same deme, a 'between' state where the lineages are in two

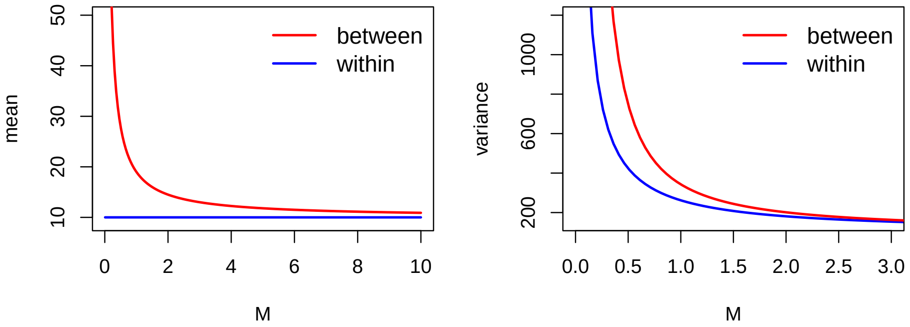

Figure 5: Mean (left) and variance (right) of the coalescent time across varying migration rates, with ten demes ($D = 10$) and two samples. The initial state was set to either within the same deme (blue, state 1) or between demes (red, state 2).

different demes, and a 'common ancestry' state where the two lineages have coalesced. The last state is an absorbing state.

The ancestral process transitions from the 'within' state to the 'between' state with rate M because we have two lineages and each lineage can migrate to $(D - 1)$ demes. The rate of coalescent is one in the 'within' state. The transition rate from the 'between' state to the 'within' state is $M/(D-1)$ because one of the two lineages has to move to exactly that deme where the other lineage is located. It is impossible to coalesce in the 'between' state because the lineages are in different demes. The subintensity matrix is thus given by

$$\boldsymbol{S} = \begin{pmatrix} -(M+1) & M \\ M/(D-1) & -M/(D-1) \end{pmatrix}, \tag{10}$$

where the order of states is first 'within' and second 'between'.

It is straight-forward to use **PhaseTypeR** to determine the mean and variance for a given number of demes and varying migration rate.

```
R> initial_within <- c(1, 0)
R> initial_between <- c(0, 1)
R> structured_subintensity_matrix <- function(deme_number, migration_rate){
+   subintensity_matrix <- matrix(
+     c(-migration_rate-1,                migration_rate,
+        migration_rate/(deme_number-1), -migration_rate/(deme_number-1)),
+     nrow=2, ncol=2, byrow=TRUE)
+   subintensity_matrix
+ }
R> n <- 200
R> mig_rate_vec <- seq(0.01, 10, len=n)
R> mean_within <- rep(0, n)
R> mean_between <- rep(0, n)
R> var_within <- rep(0, n)
R> var_between <- rep(0, n)
R> for (i in 1:n){
```

```
+     structured_subint_mat <-
+       structured_subintensity_matrix(deme_number=10, mig_rate_vec[i])
+     withinPH <- PH(structured_subint_mat, initial_within)
+     mean_within[i] <- mean(withinPH)
+     var_within[i] <- var(withinPH)
+     betweenPH <- PH(structured_subint_mat, initial_between)
+     mean_between[i] <- mean(betweenPH)
+     var_between[i] <- var(betweenPH)
+   }
```

The resulting plots are shown in Figure 5, and they reproduce Figure 5.1 in Wakeley (2009). We note that the mean coalescent time for two samples from the same deme is independent of the migration rate: the mean time $\boldsymbol{e}_1(-\boldsymbol{S})^{-1}\boldsymbol{e} = D$ (recall Table 1) is the number of demes D. We also note that the mean and variance are substantially different for the two starting states when the migration is low, but converge when the migration rate is high.

Similarly we can find the density functions for the coalescent time.

```
R> x <- seq(0, 14, length.out = 100)
R> structured_subint_mat_1 <-
+    structured_subintensity_matrix(deme_number=2, migration_rate=1.0)
R> structured_subint_mat_2 <-
+    structured_subintensity_matrix(deme_number=10, migration_rate=1.0)

R> ## Initial state within:
R> withinPH_1 <- PH(structured_subint_mat_1, initial_within)
R> withinPDF_1 <- dPH(x, withinPH_1)
R> withinPH_2 <- PH(structured_subint_mat_2, initial_within)
R> withinPDF_2 <- dPH(x, withinPH_2)

R> ## Initial state between:
R> betweenPH_1 <- PH(structured_subint_mat_1, initial_between)
R> betweenPDF_1 <- dPH(x, betweenPH_1)
R> betweenPH_2 <- PH(structured_subint_mat_2, initial_between)
R> betweenPDF_2 <- dPH(x, betweenPH_2)
```

In Figure 6 we show the densities of the coalescent times with fixed migration rate $M = 1$, deme number $D = 2$ or $D = 10$, and initial state either within (left plot) or between (right plot). Figure 6 in this article reproduces figures 5.2 and 5.3 in Wakeley (2009). Perhaps the most striking difference between the left and right plot is that the coalescence density is monotonocially decreasing when the initial sampling is within one deme, whereas the coalescence density is unimodal when the initial sampling is between two demes.

6. Conclusion, discussion and perspectives

In **PhaseTypeR** we have implemented the key characteristics and desired functions for the DPH, MDPH, PH and MPH distributions, and in this paper we have illustrated the usage

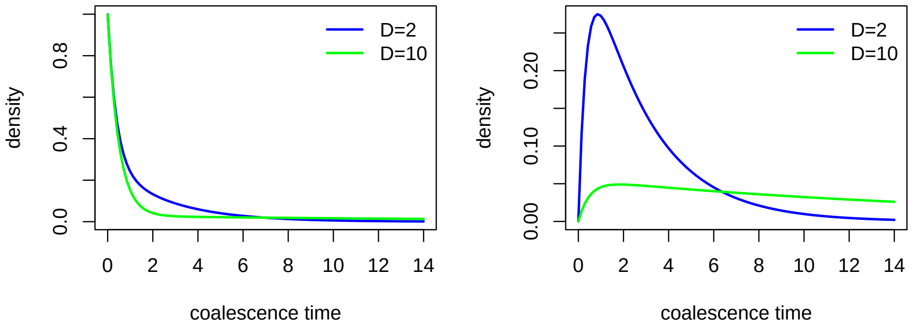

Figure 6: Density of the coalescent time between two samples, where the migration rate is fixed to $M = 1$ and the initial state is set to either within the same deme (left) or between demes (right). Blue and green indicate a number of demes of $D = 2$ or $D = 10$, respectively.

in simple examples (Section 2 and Section 3) and more involved applications (Section 4 and Section 5) from population genetics. The ability to reward-transform is particularly important in population genetics, and a unique feature of **PhaseTypeR**.

We have demonstrated that phase-type theory in general and **PhaseTypeR** in particular contains the basic foundation and implementation for obtaining insight and understanding a wide range of population genetic models. In Section 2 we concentrated on the standard Kingman's coalescent, in Section 3 on the coalescent with mutation, in Section 4 on the ancestral recombination graph, and in Section 5 on the structured coalescent. All of these models are homogeneous in time and determined by the instantaneous rate matrix and initial distribution. Other time-homogeneous population genetic models include the multiple merger coalescent (Tellier and Lemaire 2014; Freund 2021; Birkner and Blath 2021) and dormancy (Blath and Kurt 2021).

A major challenge with the coalescent models is the rapid increase in the size of the state space with the number of samples. Indeed, in the simple standard Kingman's coalescent, the size of the state space equals the partition number from number theory (Hobolth *et al.* 2021), which increases exponentially fast in the the square root of the sample size. The instantaneous rate matrices for the coalescent models are often sparse, and with a clear structure (i.e. the number of ancestral lineages always decreases except when recombination is present). The current version of **PhaseTypeR** is not taking advantage of such special structure, but it could be important for future versions because the size of population genetic data sets are often very large.

Another extension of **PhaseTypeR** could be to allow for in-homogeneity in time. For example, Arredondo, Mourato, Nguyen, Boitard, Rodríguez, Noûs, Mazet, and Chikhi (2021) consider a structured coalescent where the number of demes is constant in time, but the migration rate has different values in epochs of time in the past. Such a model requires to paste together the probabilities from the different epochs, and intermediate epochs require the calculation of the matrix exponential (see e.g. supplementary material in Zeng, Charlesworth, and Hobolth (2021)). Recent progress for calculating the matrix exponential for large rate matrices is available in Sherlock (2021).

Applications of phase-type distributions for statistical inference is still in its infancy, but we hope that this package will fuel the development. We have demonstrated how to determine the mean and co-variance matrix for the site frequency spectrum from a coalescent model given the sample size and a set of parameters. A natural procedure for estimating the parameters of a coalescent model using phase-type theory is to match the observed and expected site frequency spectrum. Birkner and Blath (2021) describe inference methods for coalescent models with highly skewed offspring distributions using the site frequency spectrum and the methods of moments for parameter estimation.

Acknowledgements

We are grateful to Emil Dare, Moisès Coll Macià, Kasper Munch and Mikkel Heide Schierup for valuable comments and suggestions that helped improve the manuscript.

References

Albrecher H, Bladt M (2019). "Inhomogeneous phase-type distributions and heavy tails." *Journal of Applied Probability*, **56**(4), 1044–1064. doi:10.1017/jpr.2019.60.

Albrecher H, Bladt M, Yslas J (2020, advance online publication). "Fitting inhomogeneous phase-type distributions to data: the univariate and the multivariate case." *Scandinavian Journal of Statistics*. doi:https://doi.org/10.1111/sjos.12505.

Arredondo A, Mourato B, Nguyen K, Boitard S, Rodríguez W, Noûs C, Mazet O, Chikhi L (2021). "Inferring number of populations and changes in connectivity under the n-island model." *Heredity*, **126**(6), 896–912. doi:https://doi.org/10.1038/s41437-021-00426-9.

Aslett LJM (2012). *MCMC for inference on phase-type and masked system lifetime models*. Ph.D. thesis, Trinity College (Dublin, Ireland). School of Computer Science & Statistics. doi:http://hdl.handle.net/2262/77559.

Aslett LJM, Wilson SP (2011). "Markov chain Monte Carlo for Inference on Phase-type Models." In *ISI 2011 Proceedings*.

Birkner M, Blath J (2021). *Probabilistic Structures in Evolution*, chapter 8: Genealogies and inference for populations with highly skewed offspring distributions, pp. 151–178. European Mathematical Society.

Bladt M, Gonzalez A, Lauritzen SL (2003). "The estimation of Phase-type related functionals using Markov chain Monte Carlo methods." *Scandinavian Actuarial Journal*, **2003**(4), 280–300. doi:https://doi.org/10.1080/03461230110106435.

Bladt M, Nielsen BF (2017). *Matrix-exponential distributions in applied probability*, volume 81 of *Probability Theory and Stochastic Modelling*. Springer. ISBN 978-1-4939-8377-3.

Bladt M, Yslas J (2021). "matrixdist: An R package for inhomogeneous phase-type distributions." ArXiv preprint arXiv:2101.07987.

Blath J, Kurt N (2021). *Probabilistic Structures in Evolution*, chapter 12: Population genetic models of dormancy, pp. 247–263. European Mathematical Society.

Campillo Navarro A (2018). *Order statistics and multivariate discrete phase-type distributions*. Ph.D. thesis, Technical University of Denmark (Copenhagen, Denmark). Department of Applied Mathematics and Computer Science. ISSN 0909-3192.

Dutang C, Goulet V, Pigeon M (2008). "actuar: An R package for actuarial science." *Journal of Statistical software*, **25**, 1–37. doi:10.18637/jss.v025.i07.

Freund F (2021). *Probabilistic Structures in Evolution*, chapter 9: Multiple-merger genealogies: Models, consequences, inference, pp. 179–202. European Mathematical Society. doi:10.4171/ECR/17.

Hobolth A, Bladt M, Andersen LN (2021). "Multivariate phase-type theory for the site frequency spectrum." *Journal of Mathematical Biology*, **83**(6), 1–28. doi:https://doi.org/10.1007/s00285-021-01689-w.

Hobolth A, Siri-Jegousse A, Bladt M (2019). "Phase-type distributions in population genetics." *Theoretical population biology*, **127**, 16–32. doi:https://doi.org/10.1016/j.tpb.2019.02.001.

Kingman JFC (1982). "The coalescent." *Stochastic processes and their applications*, **13**(3), 235–248. doi:https://doi.org/10.1016/0304-4149(82)90011-4.

Okamura H (2015). *mapfit: A Tool for PH/MAP Parameter Estimation*. R package version 0.9.7, URL https://CRAN.R-project.org/package=mapfit.

Okamura H, Dohi T (2015). "mapfit: An R-Based Tool for PH/MAP Parameter Estimation." In J Campos, BR Haverkort (eds.), *Quantitative Evaluation of Systems*, pp. 105–112. Springer International Publishing. ISBN 978-3-319-22263-9.

Okamura H, Dohi T (2016). "PH fitting algorithm and its application to reliability engineering." *Journal of the Operations Research Society of Japan*, **59**(1), 72–109. doi:https://doi.org/10.15807/jorsj.59.72.

R Core Team (2021). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Sherlock C (2021). "Direct statistical inference for finite Markov jump processes via the matrix exponential." *Computational Statistics*, **36**, 2863–2887. doi:https://doi.org/10.1007/s00180-021-01102-6.

Simonsen KL, Churchill GA (1997). "A Markov chain model of coalescence with recombination." *Theoretical population biology*, **52**(1), 43–59. doi:https://doi.org/10.1006/tpbi.1997.1307.

Tellier A, Lemaire C (2014). "Coalescence 2.0: a multiple branching of recent theoretical developments and their applications." *Molecular ecology*, **23**(11), 2637–2652. doi:https://doi.org/10.1111/mec.12755.

Wakeley J (2009). *Coalescent Theory: An Introduction.* W. H. Freeman, New York, NY. ISBN 078-0-9747077-5-4.

Wilton PR, Carmi S, Hobolth A (2015). "The SMC' is a highly accurate approximation to the ancestral recombination graph." *Genetics*, **200**(1), 343–355. doi:https://doi.org/10.1534/genetics.114.173898.

Zeng K, Charlesworth B, Hobolth A (2021). "Studying models of balancing selection using phase-type theory." *Genetics*, **218**(2). doi:https://doi.org/10.1093/genetics/iyab055.

Affiliation:

Iker Rivas-González
Bioinformatics Research Center
Aarhus University
8000 Aarhus C, Denmark
E-mail: irg@birc.au.dk
URL: https://github.com/rivasiker/

Journal of Statistical Software http://www.jstatsoft.org/
published by the American Statistical Association http://www.amstat.org/

Volume VV, Issue II *Submitted:* yyyy-mm-dd
MMMMMM YYYY *Accepted:* yyyy-mm-dd