Package 'TreeSimGM'

September 25, 2017

Type Package

Title Simulating Phylogenetic Trees under General Bellman Harris and
Lineage Shift Model

Version 2.3

Date 2017-09-20

Author Oskar Hagen, Tanja Stadler

Maintainer Oskar Hagen <oskar.hagen@outlook.com>

Depends TreeSim, ape

Description Provides a flexible simulation tool for phylogenetic trees under a general model for speci-
ation and extinction. Trees with a user-specified number of extant tips, or a user-
specified stem age are simulated. It is possible to assume any probability distribu-
tion for the waiting time until speciation and extinction. Furthermore, the waiting times to speci-
ation / extinction may be scaled in different parts of the tree, meaning we can simu-
late trees with clade-dependent diversification processes. At a specia-
tion event, one species splits into two. We allow for two different modes at these splits: (i) sym-
metric, where for every speciation event new waiting times until speciation and extinc-
tion are drawn for both daughter lineages; and (ii) asymmetric, where a speciation event re-
sults in one species with new waiting times, and another that carries the extinc-
tion time and age of its ancestor. The symmetric mode can be seen as an vicariant or al-
lopatric process where divided populations suffer equal evolutionary forces while the asymmet-
ric mode could be seen as a peripatric speciation where a mother lineage continues to exist.

License GPL-2

LazyLoad yes

Suggests knitr, rmarkdown

VignetteBuilder knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2017-09-25 18:40:02 UTC

R topics documented:

TreeSimGM-package   *Simulating Phylogenetic Trees under General Bellman Harris and Lineage Shift Model*

Description

Provides a flexible simulation tool for phylogenetic trees under a general model for speciation and extinction. Trees with a user-specified number of extant tips, or a user-specified stem age are simulated. It is possible to assume any probability distribution for the waiting time until speciation and extinction. Furthermore, the waiting times to speciation / extinction may be scaled in different parts of the tree, meaning we can simulate trees with clade-dependent diversification processes. At a speciation event, one species splits into two. We allow for two different modes at these splits: (i) symmetric, where for every speciation event new waiting times until speciation and extinction are drawn for both daughter lineages; and (ii) asymmetric, where a speciation event results in one species with new waiting times, and another that carries the extinction time and age of its ancestor. The symmetric mode can be seen as an vicariant or allopatric process where divided populations suffer equal evolutionary forces while the asymmetric mode could be seen as a peripatric speciation where a mother lineage continues to exist.

Details

Package:	TreeSimGM
Type:	Package
Version:	2.3
Date:	2017-09-20
License:	GPL-2
LazyLoad:	yes

TreeSimGM returns a list of phylogenetic trees. The trees can be simulated in different ways: (i) by age 'sim.age' (i.e. the simulation is started with one species and the user specifies a time, which is the stem age after which the simulation should stop); (ii) by taxa 'sim.taxa' (i.e. the user specifies the number of extant tips in the final tree). The user always specifies a distribution and parameters for the waiting time until speciation / extinction. Moreover, the user needs to specify the mode (i.e. symmetric or asymmetric), thus, there are four major possible combinations: (a) sim.age with parameter symmetric = TRUE; (b) sim.age with parameter symmetric = FALSE (i.e. asymmetric); (c) sim.taxa with parameter symmetric = TRUE; (d) sim.taxa with parameter symmetric = FALSE (i.e. asymmetric). Additionally, the user can choose if lineage-specific changes shall occur, through

scaling in the waiting times to speciation or extinction events. The scaling can happen for (1) speciation, (2) extinction or (3) both. Setting the probability for a new species to have a scaled speciation or extinction times ('shiftspprob' and 'shiftextprob') to a value between]0,1] will automatically activate the lineage-specific change model. If a change happens, a value is taken from the speciation / extinction waiting time distribution and is multiplied by a stochastic scaling factor. By default, both speciation and extinction shift frequencies are set to zero, meaning that if not stated otherwise, no shifts are considered.

Author(s)

Oskar Hagen, Tanja Stadler

Maintainer: Oskar Hagen <oskar.hagen@outlook.com>

References

O. Hagen and T. Stadler (2017). TreeSimGM: Simulating phylogenetic trees under general Bellman Harris models with lineage-specific shifts of speciation and extinction in R. Methods in Ecology and Evolution.

O. Hagen, K. Hartmann, M. Steel and T. Stadler (2015). Age-dependent Speciation can Explain the Shape of Empirical Trees. Systematic Biology. 64, v.3, p.432-440.

T. Stadler: Simulating trees on a fixed number of extant species. Syst. Biol. (2011) 60: 676-684.

See Also

TreeSim, ape, sim.age, sim.taxa, track.shift

Examples

```
##plots the first tree of a list of two simulated trees
#simulation based on age, this case = 3.
#Note that by default, symmetric = TRUE
#i.e. we simulate under symmetric speciation and under no extinction.
library("TreeSimGM")
trialtrees <- sim.age(3,4,"rweibull(0.4,3)")
plot(trialtrees[[1]])

##obtain 5 trees with 40 tips under asymmetric speciation and with extinction
#simulation, without extinction
mytree <-  sim.taxa(numbsim=5, n=40, m=40,  waitsp="rweibull(0.1,1)",
waitext="rexp(0.5)", symmetric = FALSE)

#see sim.age and sim.taxa help pages for further details
```

sim.age	*Simulating General Model Trees on Age*

Description

sim.age simulates stochastic trees conditioned on the time since origin (i.e. the stem age) of the extant tips under a general model (i.e. any given distribution or function for speciation and extinction). For simulations stopped when a fixed number of extant species is reached see sim.taxa. The method allows for symmetric and asymmetric speciation mode. To keep all extinct tips in the tree use complete=TRUE, if complete=FALSE all extinct lineages without extant descendants are pruned, resulting in a so called reconstructed tree. The user needs to specify a distribution or function for the waiting time to speciation and extinction. To account for changes in the speciation and extinction process which are inherited by the descendants, the change probability needs to be non-zero, "shiftsp$prob" and "shiftext$prob" respectively. Since it is the probability of changes in new species of age 0, this value has to range between 0 and 1. It is possible to have changes only for the speciation or the extinction process as well as for both at the same time. By default, both speciation and extinction change probabilities are set to zero, meaning that if not stated otherwise, no changes are considered. If a change happens, a scaling factor referred to "$strength", will be multiplied to the waiting time drawn from the speciation / extinction waiting time distribution. Therefore, values smaller than 1 will shorten and values bigger than one will prolong waiting times (no negative values should be used). For every node, speciation and extinction changes are stored at "shiftsp" and "shiftext" respectively. Extant and extinct tips with changes on speciation are marked with 1 and 0 for changes or no-changes under "shifted.sp.living" and "shifted.sp.extinct" respectively. "shifted.ext.living" and "shifted.ext.extinct" follow the same order but for changes on the extinction process. Note that shifts are not allowded at the root of the tree.

Usage

```
sim.age(age, numbsim, waitsp, waitext="rexp(0)", symmetric = TRUE, complete = TRUE,
tiplabel=c("sp.", "ext.", "Ss", "Se"), shiftsp=list(prob=0, strength="runif(0.5,0.9)"),
shiftext=list(prob=0, strength="runif(0.1,0.2)"), sampling=list(frac=1, branchprop=FALSE))
```

Arguments

age	is the total age until each tree will be simulated (e.g. age=3), starting from one ancestral species.
numbsim	is the number of simulated trees.
waitsp	refers to the waiting time until speciation and can be informed as a (1) function that generates a single waiting time number per call or a (2) string containing the name of the probability function, i.e. random generation function and its parameters except for the first: e.g. "rexp(1.5)" or "rweibull(0.4,3)". For the last input method (2), any probability density function available in R can be used. The random number generator function has always as first parameter n (which is the number of observations). As we need only one randomly generated waiting time per draw, this first parameter of the random number generator function should be omitted from the input string. HINT: see the help of the specified distribution function for more details on its specific parameters (e.g. ?rexp).

waitext	is the same as the "waitsp" but for the probability of extinction. By default, extinction is set to ZERO (e.g. waitext="rexp(0)"), i.e. no extinction process is accounted for.
symmetric	s defines which speciation mode should be used. If symmetric=TRUE the symmetric mode will be used; if FALSE, the asymmetric model will be used. By default symmetric=TRUE.
complete	if complete=TRUE, the tree with the extinct and non-sampled lineages is returned. If complete=FALSE, the extinct and non-sampled lineages are suppressed. complete=TRUE is default.
tiplabel	is a vector of 4 strings/labels that will be given for each tip that [1] survived until the present; [2] got extinct [3] experienced speciation changes or [4] experienced extinction changes along the way to the root. An automatic sequential number is added to the chosen label. tiplabel=c("sp.", "ext.", "Ss", "Se") is default.
shiftsp	a list containing [[1]] the probability $prob by which a change in the waiting time to speciation happens and the [[2]] distribution or function $strength from which a scaling factor is drawn (as in waitsp), and multiplied to the drawn speciation waiting time. shiftsp$prob should range from 0 (no change) to 1 (all new species are changed). shiftsp=list(prob=0, strength="runif(0.5,0.9)") is default.
shiftext	similar to shiftsp but for the shifts probabilities and strength of scaling factor of the changes in the waiting time to extinction. shiftext=list(prob=0, strength="runif(0.1,0.2)") is default
sampling	a list containing [[1]] the sampling fraction $frac and [[2]] a boolean $branchprop defining if the sampling should be proportional to the pendant branch lengths, in this case, longer branches would be more likely to be sampled. If $frac is smaller than 1, one sampled tree is returned with n tips as specified by the user, and the full tree prior to sampling is returned inside the tree object as $beforesampling. In the sampled tree, shift information can only be visualized though the tip labels, a complete shift history can be retrieved for the full tree $beforesampling. sampling=list(frac=1, branchprop=FALSE) is default.

Value

treearray	Array of numbsim trees with a fixed time since origin. If tree goes extinct, 0 is returned. If only one extant and no extinct tips are present, 1 is returned. For every node, including the root, speciation and extinction changes applied (i.e. the scaling factor) are stored at $shiftsp and $shiftext respectively. Extant and extinct tips with changes on speciation are marked with 1 and 0 for change or no-change under $shifted.sp.living and $shifted.sp.extinct respectively. The $shifted.ext.living and $shifted.ext.extinct follow the same logic but store shifts in the extinction process. In the case of incomplete extant species sampling, the complete simulated tree is returned via $beforesampling

Author(s)

Oskar Hagen, Tanja Stadler

References

O. Hagen and T. Stadler (2017). TreeSimGM: Simulating phylogenetic trees under general Bellman Harris models with lineage-specific shifts of speciation and extinction in R. Methods in Ecology and Evolution.

O. Hagen, K. Hartmann, M. Steel and T. Stadler (2015). Age-dependent Speciation can Explain the Shape of Empirical Trees. Systematic Biology. 64, v.3, p.432-440.

T. Stadler: Simulating trees on a fixed number of extant species. Syst. Biol. (2011) 60: 676-684.

See Also

sim.taxa, track.shift

Examples

```
## example 1 ##

# Simulation of one tree with age 3 with a symmetric speciation mode,
# a Weibull disribution for the waiting time until speciation,
# and without extinction

sim.age(3,1,"rweibull(0.4,3)")


## example 2 ##

# Plot one of ten simulated trees under symmetric speciation mode, with
# a uniform distribution for the waiting time to speciation and
# an exponential distribution for the waiting time to extinction

my3s <- sim.age(age=3,numbsim=10,"runif(0.5,2)", "rexp(0.1)")

# note that for runif, the first argument is omitted and 0 stands for the
# minimum parameter of this specific function and 2 stands for the maximum

plot(my3s[[1]])


## example 3 ##

# Simulating trees with changes in the waiting time to speciation and extinction.
# The probability for a change of speciation / extinction in a new species
# is 0.1. Upon a change in speciation, the speciation time is scaled by a factor
# drawn from a normal distribution with mean 0.5 and sd of 0.05. This implies that
# changes shorten the time to speciation.
# Changes in extinction are normally distributed with mean of 1.5,
# leading to extended time to extinction

shif_spext_sym <- sim.age(age=2, numbsim=5,  waitsp="rexp(1)", waitext="rexp(0.5)",
symmetric = TRUE, shiftsp=list(prob=0.1, strength="rnorm(0.5,0.05)"),
shiftext=list(prob=0.1, strength="rnorm(1.5,0.05)"))
```

```
## example 4 ##

# Simulating trees under an asymmetric speciation mode with changes on extinction waiting times

set.seed(10)
shif_ext_asym <- sim.age(age=2, numbsim=3, waitsp="rexp(0.8)", waitext="rexp(0.5)",
symmetric = FALSE, shiftsp=list(prob=0.1, strength="rnorm(0.7,0.9)"),
shiftext=list(prob=0.05, strength="runif(0.4,0.5)"))


## example 5 ##

# Simulating trees using own functions instead of strings as waiting times input
# first simulating a similar tree to example 4

set.seed(10)
shif_ext_asym_funk <- sim.age(age=2, numbsim=3, waitsp=function()rexp(1,0.8),
waitext=function()rexp(1,0.5),
symmetric = FALSE, shiftsp=list(prob=0.1, strength=function()rnorm(1,0.7,0.9)),
shiftext=list(prob=0.05, strength=function()runif(1,0.4,0.5)))

# test if results are identical...
identical(shif_ext_asym, shif_ext_asym_funk)

# Now we will generate a tree based on own defined waiting time rules! only one shift
# strength for speciation.
# In our waiting time function, we choose have exponentialy distributed waiting times
# until speciation that are limit to be at least 0.5!
# if they are smaller than 0.5, the waiting time will be 0.5
# remember that this function need to generate one single number.
waitspfunk <- function() {
  wt=rexp(1,1.5)
  if(wt<0.5){wt=0.5}
  return(wt)
}
set.seed(86)
# now we plug in our function or define it directly...
funk_tree <- sim.age(age=4, numbsim=3, waitsp=waitspfunk,
waitext=function()rexp(1,0.9), shiftsp=list(prob=0.1, strength=function()0.5))
plot(funk_tree[[1]])


## example 6 ##

# Validation of sim.age using TreePar:
# estimating parameters for the speciation and extinction distribution
# based on simulated trees using exponential waiting times to speciation and gamma distributed
# (i.e. age-dependent) waiting times to extinction under an asymmetric speciation mode
## Not run:
sp_la <- 3
ext_shape <- 3
```

```
ext_scale <- 2
treesTreeSimGM <- sim.age(2, 10, waitsp=paste0("rexp(",sp_la,")"),
waitext=paste0("rgamma(",ext_shape,",",ext_scale,")"), complete=TRUE )
sptimes <- lapply(treesTreeSimGM, function(x) if  (class(x)=="phylo") getx(x))
require(TreePar) #please read TreePar documentation for installation instructions.
** This requires Matlab or at
** least Matlab runtime installation.
setwd("C:/YourPathToTreePar/TreePar-Matlab")
yourpath to matlab runtime
math_run_path <- 'C:/Program Files/MATLAB/MATLAB Runtime/v91/runtime/win64/'
out <- create.mat(sptimes[[1]],path=math_run_path)
bd_out <- bd.age.optim.matlab(sptimes[[1]],path=math_run_path, sampling=1,
lambdainit=0.5,kinit=3,thetainit=0.7,
numgridpts=500)
Lcond <- "C"
lambdainit = 1
kinit = 1
thetainit = 0.5
sampling = 1
model = "G"
precision = 4
matfilename = "setup"
Param <- paste("'", lambdainit, kinit, thetainit, "'")

runCmd <- paste("sh ./run_MaxLFcn.sh ", math_run_path, matfilename, "outputML",
                as.character(precision), Lcond, model, Param, as.character(sampling),
                sep = " ")
system(runCmd)

## End(Not run)

## example 7 ##

# Validation of sim.age using TreeSim:
# trees under exponentially distributed waiting times to speciation and extinction are simulated

la=1
mu=0.5

library(TreeSim)
treesTreeSim <- sim.bd.age(2, numbsim=1000, lambda=la, mu=mu, mrca = FALSE, complete = TRUE, K = 0)
library(TreeSimGM)
treesTreeSimGM <- sim.age(2, numbsim=1000, waitsp=paste0("rexp(",la,")"),
waitext=paste0("rexp(",mu,")") )
# treesTreeSim and treesTreeSimGM have the same underlying assunptions

## compare for number of tips
# get number of tips
tipsSimTree <- unlist(lapply(treesTreeSim, function(x) if  (class(x)=="phylo") length(x[[2]])))
tipsSimTreeGM <- unlist(lapply(treesTreeSimGM, function(x) if  (class(x)=="phylo") length(x[[2]])))
# make final list
finallist <- list(SimTree=tipsSimTree, SimTreeGM=tipsSimTreeGM)
# plot
```

```
boxplot(finallist, ylab="# of tips")

## compare for oldest branching events
# get oldest branching events
branchSimTree <- unlist(lapply(treesTreeSim, function(x) if (class(x)=="phylo") max(x[[3]])))
branchSimTreeGM <- unlist(lapply(treesTreeSimGM, function(x) if (class(x)=="phylo") max(x[[3]])))
# make final list
finallist <- list(SimTree=branchSimTree, SimTreeGM=branchSimTreeGM)
# plot
boxplot(finallist, ylab="oldest branching events")
```

---

sim.taxa	*Simulating General Model Trees on a fixed number of extant Taxa*

Description

sim.taxa simulates trees on n extant species under a general model (i.e. any given distribution for speciation and extinction). The method allows the simulation to be stopped right after reaching a given number of co-existing tips/taxa or to simulate a bigger tree (m-tips) and then uniformly sample one of the trees with n extant tips that existed in the past prior to reaching m tips. For the later, the gsa code from the R package TreeSim is used. All other settings and options are equivalent to sim.age, please consult the manual for that function for details.

Usage

```
sim.taxa(numbsim, n, m = n, waitsp, waitext="rexp(0)",symmetric = TRUE, complete = TRUE,
tiplabel=c("sp.", "ext.", "Ss", "Se"), shiftsp=list(prob=0, strength="runif(0.5,0.9)"),
shiftext=list(prob=0, strength="runif(0.1,0.2)"), sampling=list(frac=1, branchprop=FALSE),
sampling.gsa=1, gsa=FALSE)
```

Arguments

numbsim	is the number of simulated trees.
n	is the number of tips in the sampled trees (number of extant sampled leaves).
m	is the number of standing taxa that will exist on the first generated trees, to then be sampled for n number of tips. In case gsa=FALSE, m is set equal to n.
waitsp	refers to the waiting time until speciation and can be informed as a (1) function that generates a single waiting time number per call or a (2) string containing the name of the probability function, i.e. random generation function and its parameters except for the first: e.g. "rexp(1.5)" or "rweibull(0.4,3)". For the last input method (2), any probability density function available in R can be used. The random number generator function has always as first parameter n (which is the number of observations). As we need only one randomly generated waiting time per draw, this first parameter of the random number generator function should be omitted from the input string. HINT: see the help of the specified distribution function for more details on its specific parameters (e.g. ?rexp).

waitext   is the same as the "waitsp" but for the probability of extinction. By default,
               extinction is set to ZERO (e.g. waitext="rexp(0)"), i.e. no extinction process is
               accounted for.

symmetric   defines which speciation mode should be used. If symmetric=TRUE the sym-
               metric mode will be used; if FALSE, the asymmetric model will be used. By
               default symmetric=TRUE.

complete   if complete=TRUE, the tree with the extinct and non-sampled lineages is re-
               turned. If complete=FALSE, the extinct and non-sampled lineages are sup-
               pressed. complete=TRUE is default.

tiplabel   is a vector of 4 strings/labels that will be given for each tip that [1] survived until
               the present; [2] got extinct [3] experienced speciation changes or [4] experienced
               extinction changes along the way to the root. An automatic sequential number
               is added to the chosen label. tiplabel=c("sp.", "ext.", "Ss", "Se") is default.

shiftsp   a list containing [[1]] the probability $prob by which a change in the waiting
               time to speciation happens and the [[2]] distribution or function $strength from
               which a scaling factor is drawn (as in waitsp), and multiplied to the drawn spe-
               ciation waiting time. shiftsp$prob should range from 0 (no change) to 1 (all new
               species are changed). shiftsp=list(prob=0, strength="runif(0.5,0.9)") is default.

shiftext   similar to shiftsp but for the shifts probabilities and strength of scaling factor of
               the changes in the waiting time to extinction. shiftext=list(prob=0, strength="runif(0.1,0.2)")
               is default

sampling   a list containing [[1]] the sampling fraction $frac and [[2]] a boolean $branch-
               prop defining if the sampling should be proportional to the pendant branch
               lengths, in this case, longer branches would be more likely to be sampled. If
               $frac is smaller than 1, one sampled tree is returned with n tips as specified by
               the user, and the full tree is returned inside the tree object as $beforesampling.
               In the sampled tree, shift information can only be visualized though the tip la-
               bels, a complete shift history can be retrieved at the full tree $beforesampling.
               sampling=list(frac=1, branchprop=FALSE) is default.

sampling.gsa   Parameter determining how close the returned trees in treearray are to the "true"
               distribution. The higher 'sampling', the closer the output trees to the 'true' dis-
               tribution.Higher values of sampling return fewer output trees meaning a larger
               input treearray is needed. See TreeSim::sim.gsa.taxa for more details.

gsa   gsa=TRUE indicates that the sim.gsa.taxa will be used and that the n parameter
               will dictate the final number of tips on the tree. Note that m needs to be always
               bigger then n. If gsa = FALSE, there is no need of specifying m, once the final
               trees will be of size n.

Value

treearray   array of numbsim trees with a fixed number of living tips. For every node, in-
               cluding the root, speciation and extinction changes applied (i.e. the scaling fac-
               tor) are stored at $shiftsp and $shiftext respectively. Extant and extinct tips with
               changes on speciation are marked with 1 and 0 for change or no-change under
               $shifted.sp.living and $shifted.sp.extinct respectively. The $shifted.ext.living
               and $shifted.ext.extinct follow the same logic but store shifts in the extinction

process. In the case of incomplete extant species sampling, the complete simu-
lated tree is returned via $beforesampling

Author(s)

Oskar Hagen, Tanja Stadler

References

O. Hagen and T. Stadler (2017). TreeSimGM: Simulating phylogenetic trees under general Bellman Harris models with lineage-specific shifts of speciation and extinction in R. Methods in Ecology and Evolution.

O. Hagen, K. Hartmann, M. Steel and T. Stadler (2015). Age-dependent Speciation can Explain the Shape of Empirical Trees. Systematic Biology. 64, v.3, p.432-440.

T. Stadler: Simulating trees on a fixed number of extant species. Syst. Biol. (2011) 60: 676-684.

See Also

sim.age, sim.gsa.taxa, track.shift

Examples

```
## example 1 ##

# Simulating trees under a Weibull distributed time to speciation, no extinction,
# and using a symmetric speciation mode. The simulation is stopped once 10 co-existing species
# exist for the first time (gsa = FALSE).

my3 <- sim.taxa(1, n=10, waitsp="rweibull(1.5,1)",
symmetric=TRUE, complete=TRUE, tiplabel=c("tip", "tip" ,"", ""))

plot(my3[[1]])


## example 2 ##

# Trees are simulated using gsa=TRUE with an exponential distribution for
# speciation and extinction

## Not run:
** long runing timce since gsa=TRUE **
mytree <-  sim.taxa(numbsim=10, n=10, m=15,  waitsp="rexp(1.5)",
waitext="rexp(0.1)",symmetric = TRUE, complete=TRUE,
tiplabel=c("sp.", "ext.", NA, NA), sampling.gsa=2, gsa=TRUE)

## End(Not run)

## example 3 ##

# Now changes in speciation waiting times occur in new species with
# probability 0.1, again under a symmetric mode
```

```
shift_sp_sym <- sim.taxa(numbsim=10, n=10, waitsp="rexp(1)",
waitext="rexp(0.5)", symmetric = TRUE, shiftsp=list(prob=0.1, strength="runif(0.5,0.9)"))


## example 4 ##

# Simulations under an asymmetric speciation mode with changes in
# speciation and extinction waiting times

shif_spext_asym <- sim.taxa(numbsim=10, n=10, waitsp="rexp(1)",
waitext="rexp(0.5)", symmetric = FALSE, shiftsp=list(prob=0.1, strength="runif(0.5,0.9)"),
shiftext=list(prob=0.05, strength="runif(0,0.5)"))

## example 5 ##

# Waiting times as functions instead of strings, allowing for more flexibility
t1 <- sim.taxa(1,8, waitsp="rnorm(0.5,0)")
plot(t1[[1]])
t2 <- sim.taxa(1,8, waitsp=function() rnorm(1,0.5,0))
plot(t2[[1]])
identical(t1,t2)
t3 <- sim.taxa(1,8, waitsp=function() 0.5)
plot(t3[[1]])
identical(t1,t3)
# all implementations are identical!

# Creating own function for waitsp and shiftsp
waitspfunk <- function() {
  wt=rexp(1,1.5)
  if(wt>1.5){wt=20}
  return(wt)
}
# here we force all values bigger than 1.5 to be very large, i.e. 20
set.seed(13)
tshittfunk <- sim.taxa(1,10, waitsp=waitspfunk,
shiftsp=list(prob=0.2, strength=function()sample(c(0.05,0.5,0.9), 1) ))
plot(tshittfunk [[1]])
```

---

track.shift	*Tracking back shifts (speciation or extinction) on TreeSimGM simulated tree*

Description

track.shift returns all changes in speciation or extinction between the root of a tree and a specific node or tip in a phylogenetic tree simulated with TreeSimGM. It can track back changes on the

speciation process (shift='sp') or on the extinction process (shift='ext'). Changes are the multipliers of the drawn waiting times.

Usage

```
track.shift(shift="sp",tree, node)
```

Arguments

shift	specifies which macroevolutionary process shall be tracked. Can be "sp" or "ext" for tracking speciation or extinction shifts repectively (default is "sp").
tree	is the simulated tree object.
node	is the node number or tip label that should be tracked. The numbers are the number of the attributes tip.label or as specified in '$edge' on the phylo tree object.

Value

out	is a vector of changes, i.e. scaling factors of the waiting times process, from MRCA until the desired node. Each reported scaling factor is named with a node number (as in $edge). The scaling factor change happened at the start of the branch ancestral to that named node.

Author(s)

Oskar Hagen, Tanja Stadler

References

O. Hagen and T. Stadler (2017). TreeSimGM: Simulating phylogenetic trees under general Bellman Harris models with lineage-specific shifts of speciation and extinction in R. Methods in Ecology and Evolution.

O. Hagen, K. Hartmann, M. Steel and T. Stadler (2015). Age-dependent Speciation can Explain the Shape of Empirical Trees. Systematic Biology. 64, v.3, p.432-440.

T. Stadler: Simulating trees on a fixed number of extant species. Syst. Biol. (2011) 60: 676-684.

See Also

sim.age, sim.taxa

Examples

```
# setting seed for reproductibility

set.seed(2142)

# Simulating a tree under an asymmetric speciation mode with frequent changes in
# speciation and extinction (repectively 50% and 30% probability)
```

```
shif_spext_asym <- sim.taxa(numbsim=1, n=5,  waitsp="rexp(1)",
waitext="rexp(0.5)", symmetric = FALSE,
shiftsp=list(prob=0.5, strength="rnorm(0.5,0.01)"),
shiftext=list(prob=0.3, strength="runif(0,0.5)"))

# visualiying the generated stochastic tree

plot(shif_spext_asym[[1]])


# tracking speciation shifts for species 3
# note that this function returns the shifts on the branches
# starting from the mrca

track.shift(shift="sp", tree=shif_spext_asym[[1]], node=3)

# tracking speciation shifts for species 7

track.shift(shift="sp", tree=shif_spext_asym[[1]], node=7)

# tracking extinction shifts for species 7

track.shift(shift="ext", tree=shif_spext_asym[[1]], node=7)
```

Index