SaGePhy Manual

1 Overview

SaGePhy is a probabilistic simulation framework for subgene and gene phylogeny evolution. The software is based on the GenPhyloData simulation framework and is comprised of different programs for generating species trees, gene trees, and domain trees. Trees are generated by birth-death processes that take as input the rates of birth and death events, among other arguments that are specified in detail in this manual. While the species tree is generated from scratch, the evolution of the gene tree is guided by a species tree which must be provided as input. Similarly, the evolution of the domain tree is also guided by a species tree and one or more gene trees, all of which must be provided as input. Generated trees are presented in Newick format along with auxiliary files that provide more information about the specific details of the simulation instance that produced a particular tree. SaGePhy supports the simulation of both additive and replacing horizontal gene and domain transfers, distance-biased transfers, gene and domain birth that doesn't happen at the root of the species and gene tree, respectively, and subgene family evolution.

2 Description of the software

SaGePhy contains four main programs. Three of the them concern phylogeny evolution while the fourth one is a tool for adjusting the branch lengths of a given tree to simulate non-uniform rates of evolution on the branches of that tree. These four programs are briefly described below:

- **HostTreeGen:** Takes as input a height and rates of speciation and loss, and produces a species tree of specified height using the given rates.
- **GuestTreeGen:** Takes as input a species tree and rates of duplication, transfer, and loss, and produces a gene tree that is mapped on to the given species tree and has been evolved through duplication, transfer, and loss events occurring with the given rates.
- **DomainTreeGen:** Takes as input a species tree, a directory of gene trees, and rates of duplication, transfer, and loss, and produces a domain tree that is mapped on to the gene trees within which it has evolved through duplication, transfer, and loss events occurring with the given rates.
- **BranchRelaxer:** Takes as input a species, gene, or domain tree, a probability distribution for scaling the lengths of the branches, and the appropriate arguments for the given probability distribution, and produces the given tree with branch lengths scaled according to the given distribution and its parameters.

3 Using the main programs

3.1 HostTreeGen

Usage:

java -jar sagephy-1.0.0.jar HostTreeGen [options] [time interval] [birth rate] [death rate] [out prefix]

Required Arguments:

[time interval] [birth rate] [death rate] [output prefix]

The time interval, birth rate, and death rate must be positive real numbers. A higher value for the time interval will result in a tree with a greater height. Higher rates for birth and death together contribute to more frequent sampling of evolutionary events. Individually,

a higher birth rate results in more frequent speciations while a higher death rate results in more frequent losses.

Options:

- -h, --help : Display the help menu with all of the required and optional arguments.
- -q, --quiet : Write pruned tree directly to standard output and suppress creation of any auxiliary files.
- -s, --seed : Specify a seed for the pseudorandom number generator.
- -min, --min-leaves : Enforce a minimum number of extant leaves on the pruned tree. Default: 2
- -max, --max-leaves : Enforce a maximum number of extant leaves on the pruned tree. Default: 1000
- -nox, --no-auxiliary-tags : Exclude auxiliary meta data tags in output trees.
- -p, --leaf-sampling-probability : Set the probability of observing a leaf. Leaves that are unobserved will yield to their lineages being pruned away from the pruned tree. Default: 1.0
- -bi, --start-with-bifurcation : Force the simulation process to start with a bifurcation in the tree.
- -a, --max-attempts : Set the maximum number of attempts that can be made to generate a tree that meets the requirements. Default: 10000
- -vp, --vertex-prefix : Set the prefix for the vertex label. Default: H

Example:

java -jar sagephy-1.0.0.jar HostTreeGen -min 100 -max 100 1.0 5.0 0.05 species

Output Files:

HostTreeGen produces four output files by default, two for the unpruned tree and two for the pruned tree. The unpruned tree produced by the simulation process contains those lineages which have not reached the leaves while the pruned tree has lineages ending in loss events removed from the final tree such that all lineages in the final tree end at the leaves. For each tree, a file with a ".tree" extension is produced which contains the simulated tree in Newick format. In addition, for each tree, a file with a ".info" extension is produced which specifies the details regarding the size of the tree and the number and relative frequency of each type of evolutionary event in the tree.

3.2 GuestTreeGen

Usage:

java -jar sagephy-1.0.0.jar GuestTreeGen [options] [species tree] [dup rate] [loss rate] [trans rate] [out prefix]

Required Arguments:

[species tree file or string] [duplication rate] [loss rate] [transfer rate] [output prefix]

A file with a pruned species tree in Newick format is required to guide the evolution of the gene tree; the user may alternatively pass the same tree as a string directly on the command line. The pruned species tree produced by HostTreeGen can be directly used as input for GuestTreeGen. The duplication rate, loss rate, and transfer rate must be positive real numbers. Higher rates for duplication, loss and transfer together contribute to more frequent sampling of evolutionary events. Individually, a higher duplication rate results in more frequent duplications, a higher loss rate results in more frequent losses, while a higher death rate results in more frequent losses.

Options:

- -h, --help : Display the help menu with all of the required and optional arguments.
- -q, --quiet : Write pruned tree directly to standard output and suppress creation of any auxiliary files.
- -s, --seed : Specify a seed for the pseudorandom number generator.
- -min, --min-leaves : Enforce a minimum number of extant leaves on the pruned tree. Default: 2
- -max, --max-leaves : Enforce a maximum number of extant leaves on the pruned tree. Default: 1000
- -minper, --min-leaves-per-host-leaf : Enforce a minimum number of extant guest leaves per host leaf. Default: 0
- -maxper, --max-leaves-per-host-leaf : Enforce a maximum number of extant guest leaves per host leaf. Default: 10
- -nox, --no-auxiliary-tags : Exclude auxiliary meta data tags in output trees.
- -p, --leaf-sampling-probability : Set the probability of observing a leaf. Leaves that are unobserved will yield to their lineages being pruned away from the pruned tree. Default: 1.0
- -a, --max-attempts : Set the maximum number of attempts that can be made to generate a tree that meets the requirements. Default: 10000
- -vp, --vertex-prefix : Set the prefix for the vertex label. Default: G
- -rt, --replacing-transfers : Set the probability of a horizontal gene transfer event being a replacing transfer. Default: 0.5

- -db, --distance-bias : Select the type of distance-bias to use when sampling transfer recipients. The three options are: none, simple, exponential. Default: none

  None disables distance-bias and samples transfer recipients uniformly at random.

  Simple implements a distance-bias that scales proportionally to the inverse of the phylogenetic distance.

  Exponential implements a distance-bias that scales exponentially in relation to the phylogenetic distance.

- -dbr, --distance-bias-rate : When using the exponential distance-bias, set the rate parameter of the exponential distribution to be used. Default: 1.0

  A higher value more strongly biases the selection of the transfer recipient towards one that is closer to the origin of the transfer.

- -gb, --gene-birth-sampling : Randomly sample the location of gene birth on the species tree.

- -gbc, --gene-birth-coefficient : Set level of bias towards the root of the species tree for gene tree birth location. Default: 1.0

  A higher value more strongly biases the location of gene birth towards the root of the species tree.

Example:

java -jar sagephy-1.0.0.jar GuestTreeGen -rt 0.6 -db exponential -dbr 1.5 -gb -gbc 2.1 species.pruned.tree 0.2 0.1 0.3 gene

Output Files:

GuestTreeGen produces eight output files by default, four for the unpruned tree and four for the pruned tree. The unpruned tree produced by the simulation process contains those lineages which have not reached the leaves while the pruned tree has lineages ending

in loss events removed from the final tree such that all lineages in the final tree end at the leaves. For each tree, a file with a ".tree" extension is produced which contains the simulated tree in Newick format. Similarly, for each tree, a file with a ".info" extension is produced which specifies the details regarding the size of the tree and the number and relative frequency of each type of evolutionary event in the tree. In addition, for each tree, a file with a ".guest2host" extension that specifies the details regarding the mapping of the gene tree nodes to the species tree nodes and a file with a ".leafmap" extension that lists the mapping of the gene tree leaves to those of the species tree in a simple two-column tab-delimited file are produced.

3.3 DomainTreeGen

Usage:

java -jar sagephy-1.0.0.jar DomainTreeGen [options] [species tree] [gene tree directory] [dup rate] [loss rate] [trans rate] [out prefix]

Required Arguments:

[species tree file or string] [gene tree directory] [duplication rate] [loss rate] [transfer rate] [output prefix]

A file with a pruned species tree in Newick format and a directory containing one or more files, each with a pruned gene tree in Newick format, are required to guide the evolution of the domain tree; the user may alternatively pass the same species tree as a string directly on the command line. The pruned species tree produced by HostTreeGen and the pruned gene trees produced by GuestTreeGen can be directly used as input for DomainTreeGen. The duplication rate, loss rate, and transfer rate must be positive real numbers. Higher rates for duplication, loss and transfer together contribute to more frequent sampling of evolutionary events. Individually, a higher duplication rate results in more frequent duplications, a higher loss rate results in more frequent losses while a higher death rate results in more frequent losses.

Options:

- -h, --help : Display the help menu with all of the required and optional arguments.
- -q, --quiet : Write pruned tree directly to standard output and suppress creation of any auxiliary files.
- -s, --seed : Specify a seed for the pseudorandom number generator.
- -min, --min-leaves : Enforce a minimum number of extant leaves on the pruned tree. Default: 2
- -max, --max-leaves : Enforce a maximum number of extant leaves on the pruned tree. Default: 1000
- -minper, --min-leaves-per-host-leaf : Enforce a minimum number of extant domain leaves per host leaf. Default: 0
- -maxper, --max-leaves-per-host-leaf : Enforce a maximum number of extant domain leaves per host leaf. Default: 10
- -nox, --no-auxiliary-tags : Exclude auxiliary meta data tags in output trees.
- -p, --leaf-sampling-probability : Set the probability of observing a leaf. Leaves that are unobserved will yield to their lineages being pruned away from the pruned tree. Default: 1.0
- -a, --max-attempts : Set the maximum number of attempts that can be made to generate a tree that meets the requirements. Default: 10000
- -vp, --vertex-prefix : Set the prefix for the vertex label. Default: D
- -rt, --replacing-transfers : Set the probability of a horizontal domain transfer event being a replacing transfer. Default: 0.5

- -db, --distance-bias : Select the type of distance-bias to use when sampling transfer recipients. The three options are: none, simple, exponential. Default: none

  None disables distance-bias and samples transfer recipients uniformly at random.

  Simple implements a distance-bias that scales proportionally to the inverse of the phylogenetic distance.

  Exponential implements a distance-bias that scales exponentially in relation to the phylogenetic distance.

- -dbr, --distance-bias-rate : When using the exponential distance-bias, set the rate parameter of the exponential distribution to be used. Default: 1.0

  A higher value more strongly biases the selection of the transfer recipient towards one that is closer to the origin of the transfer.

- -dbs, --domain-birth-sampling : Randomly sample the location of domain birth on the first gene tree.

- -dbc, --domain-birth-coefficient : Set level of bias towards the root of the gene tree for domain tree birth location. Default: 1.0

  A higher value more strongly biases the location of domain birth towards the root of the gene tree.

- -ig, --inter-gene-transfers : Set the probability of a horizontal domain transfer to occur across gene trees, if more than one gene tree is provided. Default: 0.5

- -is, --inter-species-transfers : Set the probability of a horizontal domain transfer to occur across species. Default: 0.5

- -all, --all-gene-trees : Enforce the evolution of the domain tree within all given gene trees.

Example:

java -jar sagephy-1.0.0.jar DomainTreeGen -rt 0.6 -db exponential -dbr 1.5 -dbs -dbc 2.1 -ig 0.9 -is 0.7 -all species.pruned.tree genes-directory 0.2 0.1 0.4 domain

Output Files:

DomainTreeGen produces eight output files by default, four for the unpruned tree and four for the pruned tree. The unpruned tree produced by the simulation process contains those lineages which have not reached the leaves while the pruned tree has lineages ending in loss events removed from the final tree such that all lineages in the final tree end at the leaves. For each tree, a file with a ".tree" extension is produced which contains the simulated tree in Newick format. Similarly, for each tree, a file with a ".info" extension is produced which specifies the details regarding the size of the tree and the number and relative frequency of each type of evolutionary event in the tree. In addition, for each tree, a file with a ".guest2host" extension that specifies the details regarding the mapping of the domain tree nodes to the gene tree nodes and a file with a ".leafmap" extension that lists the mapping of the domain tree leaves to those of the gene trees in a simple three-column tab-delimited file are produced.

3.4 BranchRelaxer

Usage:

java -jar sagephy-1.0.0.jar BranchRelaxer [options] [tree] [model] [args]

Required Arguments:

[tree file or string] [model] [arguments]

A file with a tree in Newick format is required; the user may alternatively pass the same tree as a string directly on the command line. The trees produced by HostTreeGen, GuestTreeGen, and DomainTreeGen can all be directly used as input for BranchRelaxer. In addition, the user must choose a model using which they wish to sample the relaxed branch lengths. Finally,

they must also provide the arguments required for that specific model.

Options:

- -h, --help : Display the help menu with all of the required and optional arguments.
- -o, --output-file : Output relaxed tree to a file. Also writes used model parameters to a file with a ".info" extension.
- -s, --seed : Specify a seed for the pseudorandom number generator.
- -x, --auxiliary-tags : Include auxiliary meta data tags in output tree.
- -innms, --keep-interior-names : Keep interior vertex names in the output tree. These are otherwise cleared.
- -min, --min-rate : Set the minimum rate allowed when sampling rates using a model. Default: 1e-64
- -max, --max-rate : Set the maximum rate allowed when sampling rates using a model. Default: 1e64
- -a, --max-attempts : Set the maximum number of attempts that can be made to create random rates that meet the requirements. Default: 10000

Supported Models:

- Constant [rate] : Constant rates (i.e., strict molecular clock).
- IIDGamma [k] [θ] : IID rates from Gamma(k, θ).
- IIDLogNormal [μ] [σ^2] : IID rates from ln $N(\mu, \sigma^2)$.
- IIDNormal [μ] [σ^2] : IID rates from $N(\mu, \sigma^2)$.
- IIDUniform [a] [b] : IID rates from Unif([a,b]).

- IIDExponential [λ] : IID rates from Exp(λ).
- IIDSamplesFromFile [filename] : IID rates drawn uniformly (with replacement) from a file with a column of samples.
- ACTK98 [start rate] [v] : Autocorrelated lognormal rates in accordance with Thorne-Kishino '98 but corrected to not yield increasing average rates in root-to-leaf direction.
- ACRY07 [start rate] [σ^2] : Autocorrelated lognormal rate in accordance with Rannala-Yang '07. The start rate refers to the tip of the tree in case there is a stem edge.
- ACABY02 [start rate] : Autocorrelated exponential rates in accordance with Aris-Brosou-Yang '02.
- ACLBPL07 [start rate] [μ] [θ] [σ] : Autocorrelated CIR rates in accordance with Lepage-Bryant-Phillipe-Lartillot '07. The process is simulated using a discretisation across every branch. The start rate refers to the tip of the tree in case there is a stem edge.
- IIDRK11 [host tree] [guest-to-host map] [scale factor] : IID gamma rates governed by host tree in accordance w. Rasmussen-Kellis '11. Every guest branch rate is created from a gamma distribution specific for each host edge that the branch passes over. The scale factor is then applied to all relaxed lengths. Parameters are stored in the host tree thus: (A:0.4[PARAMS=([k],[θ])],... Guest and host tree must be temporally compatible and have no lateral transfer events.

Example:

java -jar sagephy-1.0.0.jar BranchRelaxer -x -innms -o domain.relaxed.tree domain.pruned.tree ACTK98 1.00 0.10

Output Files:

By default, BranchRelaxer does not produce any output files. Instead, it prints the tree with the relaxed branch lengths onto the standard output. However, if an output file is

specified in the arguments, then it produces two output files. The first file contains the tree with the relaxed branch lengths. The second file, which contains a ".info" extension in its filename, contains details such as all of the original and relaxed branch lengths.

4 Supplementary Program: partial.py

4.1 Overview

This software simulates the subgene level evolutionary event of partial gene transfer. The python script evolves nucleotide sequences down the given gene and domain trees by starting with a random base sequence at the root of each tree and then using a substitution model to sample changes in the sequences at each node in the tree. Once these sequences have evolved and reached the leaves of the trees, the domain leaf sequences are inserted into the sequences at the leaves of their corresponding genes. Since the domains are subgene level units and they undergo transfer events, this process effectively simulates partial gene transfers.

In order to evolve the required sequences, we use the tool Seq-Gen. Seq-Gen is a program that evolves nucleotide sequences along a given phylogeny using common substitution models. Our script takes as input a directory of domain trees, a directory of gene trees, and a directory of domain to gene leaf mappings and produces directories of sequences at the leaves of the domain and gene trees and a directory of gene sequences with the domain sequence insertions.

4.2 Using the program

Dependency:

Seq-Gen executable must be either in the search path or in the same directory as partial.py

Usage:

python3 partial.py [options] [domain directory] [gene directory] [leafmap directory]

Required Arguments:

[domain tree directory] [gene tree directory] [leaf mapping directory]

The script requires as input a directory of domain trees, a directory of gene trees, and a directory of files that specify the mapping of the domain tree leaves to their corresponding gene tree leaves. The domain and gene trees must be in Newick format, while the leaf mapping files must be three-column tab-delimited files that list the mappings in a format such that the first column contains the domain tree leaf labels, the second column contains the corresponding gene tree leaf labels, and the third column contains the corresponding gene tree file name. The domain trees, gene trees, and leaf mapping files generated by the SaGePhy simulation framework can be directly used as input for this script.

Options:

- -h, --help : Display the help menu with all of the required and optional arguments.
- -dl, --domain-length : Set the length of domain sequences. Default: 100
- -gl, --gene-length : Set the length of gene sequences. Default: 1000
- -ap, --append-domain : Append the domain sequences to the corresponding gene sequences instead of inserting the domain sequences at a random position within the gene sequence, which is done by default.
- -s, --branch-scaling : Set the branch length scaling factor by which current branch lengths are multiplied to produce modified branch lengths. Default: 1.0
- -m, --model : Select the substitution model to be used for generating the sequences. Options include HKY, F84, GTR, JTT, WAG, PAM, BLOSUM, MTREV, CPREV45, MTART, LG, and GENERAL. HKY, F84, and GTR are for nucleotides, while the rest are for amino acids. Default: GTR
- -a, --alpha : Select the shape (alpha) for the gamma rate heterogeneity. Default: 1.0

- -g, --gamma-cats : Set the number of gamma rate categories. Default: continuous
- -i, --invariable-sites : Set the proportion of invariable sites. Default: 0.0
- -c, --codon : Enter three numbers separated by spaces in the format #1 #2 #3 to set the rates for codon position heterogeneity. Default: none
- -t, --transition-transversion : Set the transition-transversion ratio. Default: equal rate
- -r, --rate-matrix : Enter numbers separated by spaces, such as #1 #2 #3 #4 #5 #6 for nucleotides, to set the relative rates for substitutions. Default: all 1.0
- -f, --char-frequencies : Enter numbers #A #C #G #T for nucleotides with default of all equal rates or enter numbers #1 ... #20 for amino acids with default of matrix frequencies to set the relative frequencies, or enter e to set all equal rates.
- -z, --seed : Specify the seed for the random number generator. Default: system generated

Consult the Seq-Gen manual by following this link here to obtain more detailed information about these options.

Example:

python3 partial.py -dl 120 -gl 1380 -s 1.2 domain_dir gene_dir leafmap_dir

Output Files:

The script creates a directory called "seqs". Within that, a directory for the domain sequences and another for the gene sequences, named accordingly, are created. Within the "domains" directory, a multiple sequence FASTA file containing the generated domain sequences is created for each domain tree. Within the "genes" directory, two further directories are created: "pre-transfer" and "post-transfer". Within the pre-transfer directory, a multiple sequence FASTA file containing the original generated gene sequences is created for each gene tree. Within the post-transfer directory, a multiple sequence FASTA file containing the gene

sequences produced from inserting the domain sequences into the original gene sequences is created for each gene tree.