Simulator of evolution with landscape variation (SELVa)

Manual

Overview.

1. The simulator is distributed as a Java jar file.
2. Information about the simulation is given in a config file.
3. To run the simulator, open the command line prompt, go to the directory where the above-mentioned files are and type
   % java –jar Selva.jar config.txt
   Currently, the jar is built using Java 1.7, so you have to have the corresponding JDK on your system.
4. You will also need to provide a file with your phylogenetic tree in the Newick format and provide its location in the config file (TREE_FILE parameter).

The config file

The config file consists of whitespace-separated pairs of parameters and their values, given on separate lines. Lines that begin with "#" are comments and ignored by the program.

The config file contains the following fields. Default values (if such exist) are shown in **bold** in the Example(s) column. If an option does not have default values, its values must be specified, either at all times, or depending on the values of other parameters (the latter case is mentioned in the Description column).

Parameter	Description	Example(s)
ALPHABET	String that specifies the alphabet that your sequence uses (no spaces)	ARNDCEQGHILKMFPSTWYV
LENGTH	Length of a sequence that uses a particular landscape.	1 (for single-site evolution simulation) 1000 (for simulating a longer sequence that shares a landscape
NUM_INSTANCES	The number of independent landscapes that are going to be simulated in parallel. The length of the sequence that evolves according to this landscape is given in the LENGTH parameter	**1** (default)
NUM_THREADS	The number of threads that are going to be used by the simulator. It should be **no greater than NUM_RUNS,** as there currently is no functionality for multithreading a single run (simulation of a single landscape)	**1** (default)
TREE_FILE	Absolute or relative path to the file containing the phylogenetic tree in Newick format	/path/to/tree/file.tre
Fields related to landscape		
INITIAL_FITNESS	How is the initial fitness specified? file – if the fitness is given in a file (see below for file description). If this is set, the FITNESS_FILE must be supplied	file lognorm gamma

	`lognorm` – if each character's fitness is randomly generated from lognormal distribution with mu = 0 and sigma = `DIST_PARAM` (described below) `gamma` – if each character's fitness is randomly generated from the gamma distribution with alpha=beta=`DIST_PARAM` (described below)	
`FITNESS_FILE`	**Required if INITIAL_FITNESS is `file`, ignored otherwise**. This is the name of the file containing the fitness vector as a space-separated list of numbers. The length of the list should be the same as that of the `ALPHABET` string	`/path/to/fitness/file.txt` whose contents may be: 1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1
`DIST_PARAM`	**Required if INITIAL_FITNESS is GAMMA or LOGNORM, ignored otherwise.** This is the parameter of the distribution (sigma for lognormal, alpha=beta for gamma)	`0.1`
The following fields deal with landscape changes		
Rules determining when and how the landscape changes		

`NEW_FITNESS_RULE`	How do we obtain the new fitness landscape?
	`iid`: sample it from the same distribution as the initial fitness (in which case `INITIAL_FITNESS` **must** come from a distribution, not a file)
	`shuffle:` new fitness vector is produced by randomly shuffling the previous fitness vector
	`current_allele_dependent:` the fitness of the current letter is increased every `LANDSCAPE_CHANGE_INTERVAL` time units by `AGE_DEPENDENCE_COEFFICIENT`. **This option only works with landscape change timing is at deterministic fixed intervals and sequence of LENGTH 1**
		`iid`
		`shuffle`
		`current_allele_dependent`

`LANDSCAPE_CHANGE_TIMING`	How do we determine when to change the landscape?
	`stochastic:` landscape change is a Poisson process with parameter equal to `LANDSCAPE_CHANGE_PARAMETER` (time is in terms of tree length units).
	`fixed_num_changes:` the landscape is changed at fixed-length
		`stochastic`
		`fixed_num_changes`
		`fixed_interval_length`

	intervals whose length is such that there will be `LANDSCAPE_CHANGE_PARAMETER` changes along the longest path from the root to a leaf in a tree. `fixed_interval_length:` the landscape is changed every `LANDSCAPE_CHANGE_PARAMETER` time (tree length) units	
`SHARED_LANDSCAPE`	Do parallel tree branches share the same landscape? If `true`, then there is one copy of the landscape that is evolving with time and that is shared by all branches of the phylogenetic tree. If `false`, then parallel branches of the tree have separate independently evolving landscapes. **This option is incompatible with the current allele-dependent landscape change (see below).**	`true` **`false`** (default)
Parameters detailing the rules of landscape change		
`LANDSCAPE_CHANGE_PARAMETER`	The single parameter governing the times of landscape change whose interpretation depends on the choice of `LANDSCAPE_CHANGE_TIMING`. If `LANDSCAPE_CHANGE_TIMING = stochastic`, this is the parameter of the Poisson process (mean rate, lambda) If `LANDSCAPE_CHANGE_TIMING = fixed_num_changes`, then this is the number of landscape changes along the longest root-to-leaf path in the tree If `LANDSCAPE_CHANGE_TIMING = fixed_interval_length`, it is the length of the interval between time changes (in tree time units)	`0.1`
`AGE_DEPENDENCE_COEFFICIENT`	The number added to the current character's fitness value every time unit. **Required if and only if `NEW_FITNESS_RULE` is `current_allele_dependent`**	`1`
`Q_NORMALIZATION`	How is the Q matrix normalized? `constant_rate:` $\sum_i -q_{ii}*p_i = 1$, i.e., the mean rate of change per time unit is 1. This is the approach taken by Yang's evolver `constant_for_flat:` each q_{ij} is divided by the alphabet length – 1, so that $\sum_i -q_{ii} * pi_i = 1$ if all $q_{ij} =1$. Thus, if the	**`constant_rate`** (default)

	landscape is flat, the mean rate of change per time unit is 1 `none:` no normalization	
`SCALE_LANDSCAPCE_CHANGE_TO _SUBSTITUTION_RATE`	Only relevant if `LANDSCAPE_CHANGE_TIMING` is `stochastic` and `Q_NORMALIZATION` is not constant_rate. If `true,` the rate of landscape change is obtained by multiplying the `LANDSCAPE_CHANGE_PARAMETER` by the allele substitution rate. WARNING: since this option requires recomputing the stationary distribution vector π at every landscape change, it will likely slow down execution.	**false** (default)
Additional I/O options		
`PRINT_LANDSCAPE_INFO`	If `true,` then information about every landscape change time and the corresponding new fitness vectors are printed to files `changetimes.merged.fasta` and `fitnesses.merged.fasta,` respectively. WARNING: this can **greatly** increase your memory usage and execution time, so set it to true only if you are sure you need the information (or if you are using small datasets)	**false** (default)