Local adaptation in dispersal in multi-resource landscapes.

This document provides basic instructions for running the simulation and provides a description of the parameters used. The included example code can be used to run a single iteration or multiple replicated landscapes equivalent to any of those in the manuscript, or new ones using unexplored parameter ranges.

The simulation is coded entirely in R and pieces controlling different components of the life cycle and landscape production can be found within the files in the src directory. Prior to running, the packages abind and parallel will need to be installed. Many scripts source other scripts to load necessary functions - be sure to modify the file path names at the top of each script before attempting to run any of them on your own machine.

The script example_main has two examples of how to run either a single iteration of this simulation (L2-L33) or multiple iterations with different parameter combinations (L39-74). To generate a single landscape, modify the parameters in make.prms.ls (as defined in Table 1). The functions make.prms and make.prms.sim will generate the inputs necessary to run a single simulation with the function run.sim; the arguments for these functions are defined below in Table 2. When running a single simulation, you may select the option to print updates on the population state (print=T or F) and/or to plot the population distribution on the landscape (plot=T or F).

Running a set of replicated landscapes across a range of parameter values takes more arguments (Table 1) that allow one resource type to be varied in spatial autocorrelation and abundance. This function, var.acl.frac.suitable.h2.only, will generate two files: prms.array.RData (a dataframe of landscape parameter values) and prms.ls.list (a list containing all landscapes and landscape objects necessary to run a full simulation). Simulations on all of these landscapes can be run in parallel using the function run.prms.ls.list, which takes the arguments shown in Table 2. This will generate an output file for each unique landscape. These files can be summarized using the script make_summary, which contains the lists described in Table 3. These summary files contain the information necessary to generate all of the figures in the manuscript, but other information could certainly be extracted from the raw data files.

In addition to scripts necessary to run the simulation, we have also included two figure files to visualize simulation output. We have also included summary files for cases to generate Figs 4 and 5 from the published manuscript. To generate these figures, the packages plotrix, grid, and gridBase will need to be installed. The figure files are currently coded to plot the output from the example simulations in the example_main file; you will need to modify the file and path names to generate these figures with the larger included summary files.

Name	Description
ltc.size	The number of grid cells on the landscape; will generate a landscape of ltc.size x ltc.size.
acl	In a single model iteration, a vector containing the spatial autocorrelation for resources A and B of the form c(σ_A, σ_B)
frac.suitable	In a single model iteration, a vector containing the fraction of the landscape occupied by resources A and B of the form c(f_A, f_B)
acl.h1	When running multiple replicates in parallel, the degree of spatial autocorrelation in resource B (σ_B); this value will not be varied between replicates.
frac.suitable.h1	When running multiple replicates in parallel, the fraction of the landscape containing resource B (f_B); this value will not be varied between replicates.
range.acl.h2	When running multiple replicates in parallel, the range of values for spatial autocorrelation in resource A (σ_A; this can either be a single value or a string of values.
range.frac.suitable.h2	When running multiple replicates in parallel, range of resource abundances for resource A (f_A); this can either be a single value or a string of values.
n.reps	When running multiple replicates in parallel, the number of unique landscapes to generate for each combination of landscape parameters.
save.path	File path where landscape output will be saved.
remove.files	Whether or not to remove previously existing files from the save.path location. Use TRUE to begin a new simulation and FALSE to re-start a partially completed set of simulations.
n.cores	The number of computer cores to run in parallel.

Table 1: Descriptions of landscape generation parameters.

Name	Description
save.path	File path where simulation output will be saved. This must be the location where relevant landscape files are stored.
save.dir	Name of simulation output file.
remove.files	Whether or not to remove previously existing files from the save.path location. Use TRUE to begin a new simulation and FALSE to re-start a partially completed set of simulations.
load.dir	Directory to load previous population state from a file.
load.strat	Whether or not to load strategies from previous run; if TRUE, loads from load.dir, if FALSE, generates new dispersal strategy.
n.cores	The number of computer cores to run in parallel.
n.gens	Number of generations the simulation will run.
save.freq	Frequency at which the population state will be saved, in generations. E.g., save.freqs=10 will save the population state at generations $0, 10, 20, 30...$
K	Carrying capacity per patch.
r	maximum growth rate.
initial.strat	The number of bins over which dispersal probability will be distributed when the simulation is initiated. E.g., for $initial.strat = 5$, the first 5 bins of the dispersal kernel will each be set to $1/5$.
n.bins	Number of bins in the dispersal kernel.
max.dist	Maximum dispersal distance; note that the size of each bin is max.dist/n.bins.
num.strategies	Maximum number of segregating dispersal strategies in any given generation.
disturb.r	Per generation probability that each patch will experience a disturbance event.
ws	String of survival probabilities for each ecological fit allele. A.h1, a.h1, A.h2, and a.h2 are the probabilities that A and a will survival to adulthood in patches containing resource B and resource A, respectively.
mr.A	Mutation rate at the ecological fit locus.
recomb	Recombination rate between ecological fit and dispersal loci.

Table 2: Descriptions of simulation parameters.

Name	Description
a.freqs.list	List containing the total number of each ecological specialization allele on the landscape at each save point.
a.space.freqs.list	List containing the per patch abundance of each ecological specialization in each habitat type at each save point. $H1$ refers to resource B, $H2$ to resource A, and $H3$ to patches containing both resources together.
deltags.list	Matrices of changes in abundance during the growth phase, separated by ecological fit allele, at each save point.
geno.freqs.list	List of total abundances of all genotypes (ecological fit x dispersal strategy combinations) at each save point.
m.by.h.list	List of arrays showing the average dispersal strategy in each habitat type at each save point, with dimensions (n.bins+1, n.gens/save.freq, 3).
m.freqs.list	List of vectors, each showing the total abundance of each dispersal strategy at each save point.
mds.list	List of matrices of dispersal mortality rates, separated by ecological fit allele, at each save point.
mss.list	List of matrices of mortality due to selection at the ecological fit locus, separated by ecological fit allele, at each save point.
strats.list	List of matrices of all unique dispersal strategies present in the population at each time point.

Table 3: Descriptions of model summary output.