Model code, general instructions

The model presented in the main text of the Scientific Reports paper "**Ecological opportunity and adaptive radiations reveal eco-evolutionary perspectives on community structure in competitive communities**" is implemented in its basic form as MATLAB (version R2017b) code. To run the model "main.m" should be executed in the same directory as the other m-files. The code is commented within each m-file, below we present general features and key components of the implementation.

main.m

This is the main function that is executed to run the model. Model parameters and initial conditions are initiated at lines 17-61. The function requires two input arguments, the argument "sigma_a_ind" determines the sigma_a parameter of the model (lines 17-18) and "rep_ind" determines the replicate identifier of the given realization (line 15). For example, executing "main(1,1)" will run the model with the first and lowest value in sigma_a (sigma_a = 0.1) analyzed in the study. On the other extreme, executing "main(7,1)" will run the model with the last and largest value in sigma_a (sigma_a = 0.7). Such initiation also allows for the possibility of atomized model output for different parts of parameter space or different simulation realizations (e.g. replicates). On lines 21-61 constants and initial conditions including carrying capacity, resource width, intrinsic growth, mutation rate population size, trait value, mutation, etc., etc. are initiated (see commented code). Note also that predator parameters are initiated on lines 43-55, these provide the possibility to include predators in the model but this is not relevant for the current study and thus predator abundances are set to zero.

At execution, the code computes equilibrium population sizes for the initiated populations (line 151) and removes populations that do not have positive population size at equilibrium. Thereafter, the code moves into the evolutionary part, looping over evolutionary time. Each evolutionary time step: 1) Populations mutate (phenotypic trait change). 2) Invasion fitness for mutants is computed. 3) Mutants with positive invasion fitness are allowed to invade. 4) Mutual invisibility between mutant and resident morphs is computed. 5) Mutants are allowed to invade alongside the resident (if mutual invisibility exists) or replace the resident (if mutual invisibility does not exist). 6) Equilibrium population size is re-computed, species are defined and extinct populations are removed. Thereafter the algorithm moves into the next evolutionary step. The main script outputs (see lines 63-87) trait distribution data, abundance data, phylogenetic data, and ecological variables (e.g. interaction strengths).

Below I explain the functions that are called by main.m

mutfit_and_popequi_func.m

Depending on input arguments (e.g. fitcompflag) this function either computes ecological equilibrium for the resident community or computes mutant fitness of a rare mutant in a residentl environment. In the case of computing ecological equilibrium, the function

computes the population dynamics and population abundances at equilibrium. This is thus where the equations of the ecological model are implemented. The function outputs 1) time series of the dynamics and 2) ecological variables (e.g. ecological interaction strengths and carrying capacities) that are conditions of the current system (e.g. traits of the populations). Ecological variables are arranged in matrix form according to the form of the general Generalized Lotka-Volterra (defined in **ode_sys.m**) and population dynamics are then solved using an ODE solver provided by MATLAB.

In the case of computing mutant fitness, the function sets up the model for consumer mutant dynamics in the context of the full resident community and then computes the initial growth (fitness) of the mutant while rare. This is done by solving population dynamics for the initial growth of the mutant, using the function **sim_equilibrium.m**. The ecological equations presented in the main text are implemented here. If the mutant grows initially, fitness is positive else it is negative.

update_sys_func.m

Takes equilibrium population size data as input and removes extinct populations, trait vectors, and other data variables are also cleared from extinct population data.