Provisional Title

Authors[1,2,3] and Álvaro Sánchez[1,2,†]

[1] *Department of Ecology & Evolutionary Biology, Yale University, New Haven, CT, USA*
[2] *Microbial Sciences Institute, Yale University, New Haven, CT, USA*
[3] *Other affiliations...*
[†] *To whom correspondence should be addressed:* alvaro.sanchez@yale.edu

Abstract

The abstract goes here.

1 Introduction

This is an example cite [1, 2]

2 Results

3 Discussion

4 Methods

4.1 Simulations

We used the Community Simulator package [3] and included new features for our simulations. In the package, species are characterized by their resource uptake rates ($c_{i\alpha}$ for species i and resource α), and they all share a common metabolic matrix **D**. The element $D_{\alpha\beta}$ of this matrix represents the fraction of energy in the form of resource α secreted when resource β is consumed. Here we implemented a new operation mode in which species can secrete different metabolites (and/or in different abundances) when consuming a same resource. Experimental observations support the idea of distinct species producing different sets of byproducts when growing in the same primary resource **[missing ref(s)]**. We let $D_{i\alpha\beta}$ denote the fraction of energy in the form of resource α secreted *by species i* when consuming resource β —note that now $D_{i\alpha\beta}$ need not be equal to $D_{j\alpha\beta}$ if $i \neq j$, unlike in the original Community Simulator. In the package's underlying Microbial Consumer Resource Model [4, 5], this just means that the energy flux $J_{i\beta}^{out}$ now takes the form

$$J_{i\beta}^{out} = \sum_{\alpha} D_{i\beta\alpha} l_\alpha J_{i_\alpha}^{in} \tag{1}$$

The documentation for the Community Simulator package contains a detailed description of the model, parameters and package use. For the updated package with this new functionality, see Data & code availability.

For the simulations here, we first generate a library of 660 species (divided into three specialist families of 200 memebers each and a generalist family of 50 members) and 30 resources (divided into three classes of 10 members each). We randomly sample 50 species from each pool in equal ratios to seed 100 resident and 100 invasive communities respectively. We then grow and dilute the communities serially, replenishing the primary resource after each dilution. We repeat the process 20 times to ensure generational equilibrium is achieved.

5 Data & code availability

Experimental data and code for the analysis, as well as code for the simulations and the updated Community Simulator package with instructions to use the new features can be found in `github.com/jdiazc9/coalescence`.

6 Figures

7 References

1. Větrovský T and Baldrian P (2013). The Variability of the 16S rRNA Gene in Bacterial Genomes and Its Consequences for Bacterial Community Analyses. *PLoS ONE* **8(2)**:e57923

2. Johnson SG. The NLopt nonlinear-optimization package

3. Marsland R, Cui W, Goldford J and Mehta P (2020). The Community Simulator: A Python package for microbial ecology. *PLoS ONE* **15(3)**:e0230430

4. Goldford JE, Lu N, Bajić D, Estrela S, Tikhonov M, Sanchez-Gorostiaga A, Segrè D, Mehta P and Sanchez A (2018). Emergent simplicity in microbial community assembly. *Science* **361(6401)**:469–474

5. III RM, Cui W, Goldford J, Sanchez A, Korolev K and Mehta P (2019). Available energy fluxes drive a transition in the diversity, stability, and functional structure of microbial communities. *PLoS Computational Biology* **15(2)**:e1006793