

# Conjugate Interaction - Dispersal Matrices that Optimizes the Community Stability

Greg & Burcu

CSSS 17

- **Motivation :** Complexity versus Stability question : Weak interactions stabilize food webs.
- **Aim #1:** Comparing the stability of uniform interaction / dispersal matrices with exponentially distributed interaction / dispersal matrices, as a function of  $\alpha_a = \alpha_m$ .

Community (A)	Dispersal (M)
$\sim U(-1, 1)$	$\sim U(-1, 1)$
$\sim U(-1, 1)$	$\sim \exp(\alpha_m)$
$\sim \exp(\alpha_a)$	$\sim U(-1, 1)$
$\sim \exp(\alpha_a)$	$\sim \exp(\alpha_m)$

- **Aim #2:** Stability as a linear or a non-linear function of parameters : When both matrices are exponentially distributed with  $\alpha_a$  and  $\alpha_m$ , how does stability change as a multiplicative / additive function of the  $\{\alpha_a, \alpha_m\}$  pair?
- **Aim #3:** Generalization : Find conjugate interaction-dispersal matrix pairs that optimizes the stability of a food web.
- **Aim #4:** Validation : Can we use real data sets to validate the stability of the conjugate pairs?
- **Aim #3:** Future work : Abundance vector, source and sink added to the system, perturbation in abundance.

## Parameter Definitions

- $N$  : Number of species.
- $P_N$  : Number of patches (habitats).
- $X_i^l$  : Abundance of species  $i$  in patch  $l$ .
- $a_{ij}^l$  : Interaction coefficient between species  $i$  and species  $j$  in patch  $l$ .

- $m_i^{l \rightarrow k}$  : Migration coefficient of species  $i$  from patch  $l$  to patch  $k$ .

### Simulation Rules

- Dispersal is symmetric, i.e.,  $m_i^{l \rightarrow k} = m_i^{k \rightarrow l}$
- Same species have the same dispersal rate, i.e.,  $m_i^{l \rightarrow k} = m_i \forall k, l$ .
- Initial Conditions : Each patch is identical, and abundance of species are randomly drawn.
- **Governing ODE**

$$\frac{dX_i^l}{dt} = X_i^l \sum_{j=1}^N a_{ij}^l X_j^l - \sum_{k=1(l \neq k)}^{P_N} m_i^{l \rightarrow k} X_i^l + \sum_{k=1(k \neq l)}^{P_N} m_i^{k \rightarrow l} X_i^k \quad (1)$$

- **Jacobian Matrix** : Can be automatized via MATLAB. (Or can be symbolically hardcoded as well)