

Evolution in a Food Web part 2

As in the “Food Web” section, we provide a directed graph representing the food web. Node labels are species names, and arrow labels are strength of predation. Let’s say the conversion factor from prey to predator is constant k .

Then species i ’s dynamics is

$$\frac{dX_i}{dt} = \left(r_i + k \sum_{j \rightarrow i} f_{ji} X_j - \sum_{i \rightarrow j} f_{ij} X_j \right) X_i$$

where

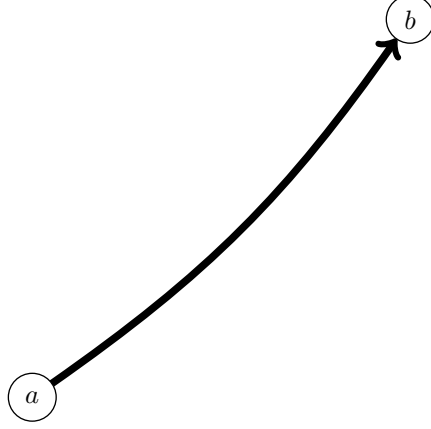
$f_{ij} = f(u_i, u_j)$ is some function of the two phenotypes controlling how well j eats i ;

u_i is the phenotype of species i ; and

$r_i = (0 \text{ if } i \text{ is a predator, } 1 \text{ else})$.

This will induce the usual dynamics of apparent competition, and adaptive dynamics of all the u_i follows.

- UPDATE: the above forces a fixed ratio of $-k$ between paired a values. We can allow that to drift a little by giving each a ‘conversion efficiency’ value, maybe? Use a planar $s_i = (u_i, \gamma_i)$, and instead of writing $kf(u_i, u_j)$, we use $k\gamma_i\gamma_j f(u_i, u_j)$. On the prey side just $f(u_i, u_j)$. Then the ratio is $k\gamma_i\gamma_j$. Let the γ values vary more slowly than f .
- UPDATE: Yes, but that way selection is only on the predator’s γ because they only affect the predatory’s fitness. Predatory’s γ_i can be how much it gets from eating a prey, but what does it mean for the prey to change the conversion rate? A predation event kills less than one prey animal? I mean, maybe if it’s a fungus or something. That might work for me - let $f(u_i, u_j)$ be encounter rate, predator’s γ_j is how much it gets from encounter, prey’s σ_i is how much it loses from encounter, so we can say $-k\gamma_j f(u_i, u_j)$ and $\sigma_i f(u_i, u_j)$ respectively, which will produce the selection I’m looking for, and ought to lead to marginal benefit for both.

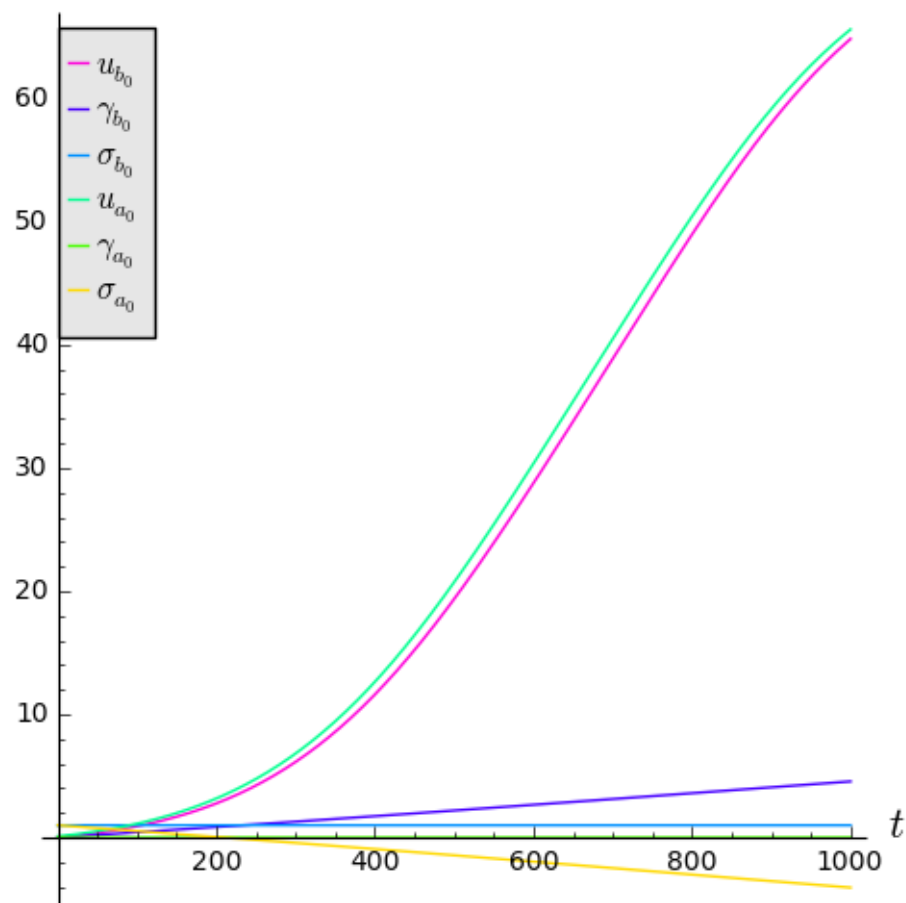


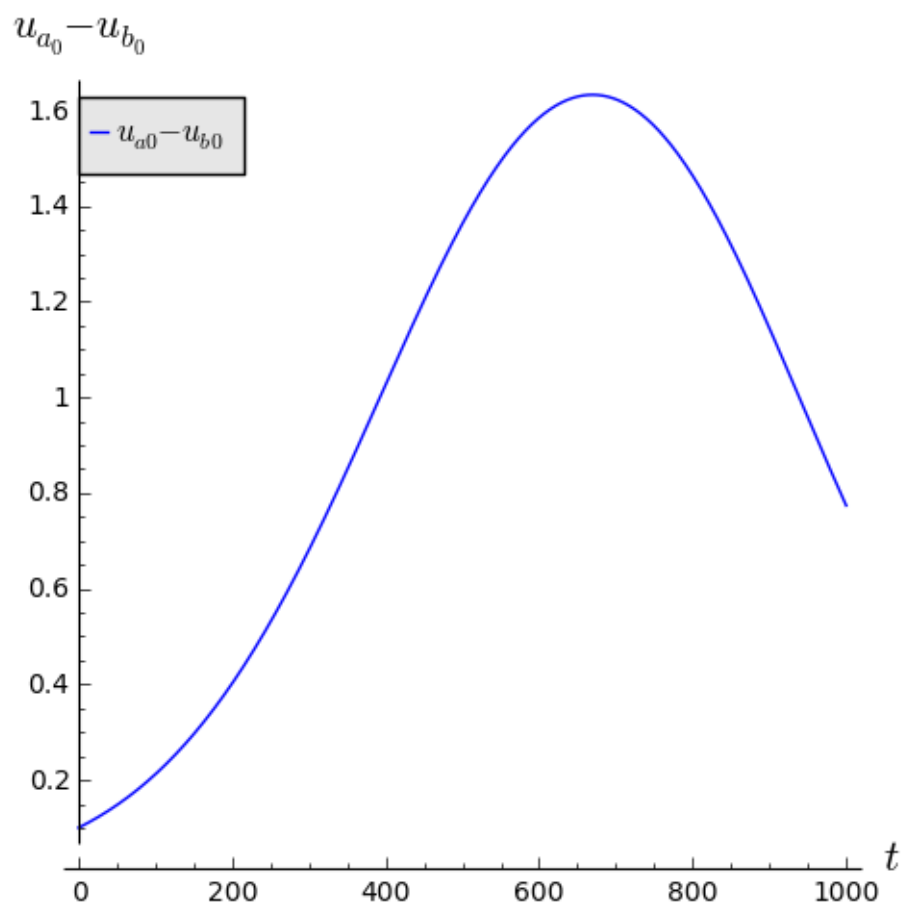
The foodweb model:

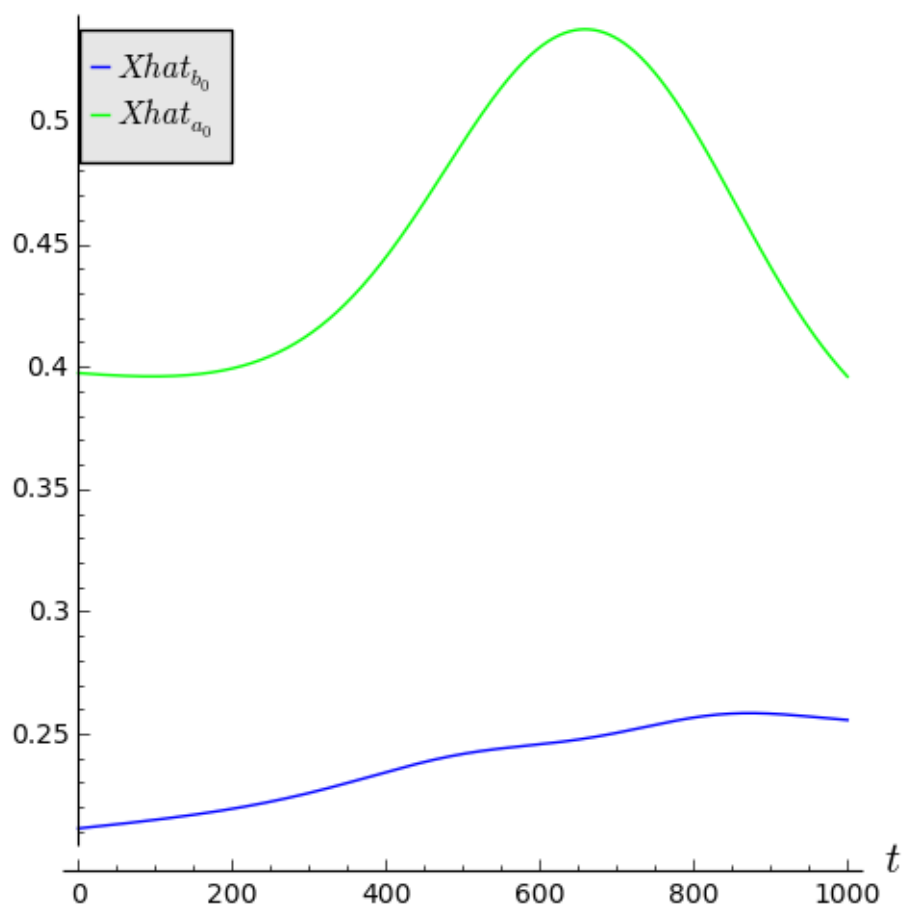
$$\begin{aligned}\frac{dX_{b0}}{dt} &= \frac{9}{1250} X_{a0} X_{b0} (\gamma_{b0} + 50) (2 \cos(-u_{a0} + u_{b0}) + 5) - X_{b0} \\ \frac{dX_{a0}}{dt} &= -\frac{1}{125} X_{a0} X_{b0} (\sigma_{a0} + 50) (2 \cos(-u_{a0} + u_{b0}) + 5) - X_{a0}^2 + X_{a0}\end{aligned}$$

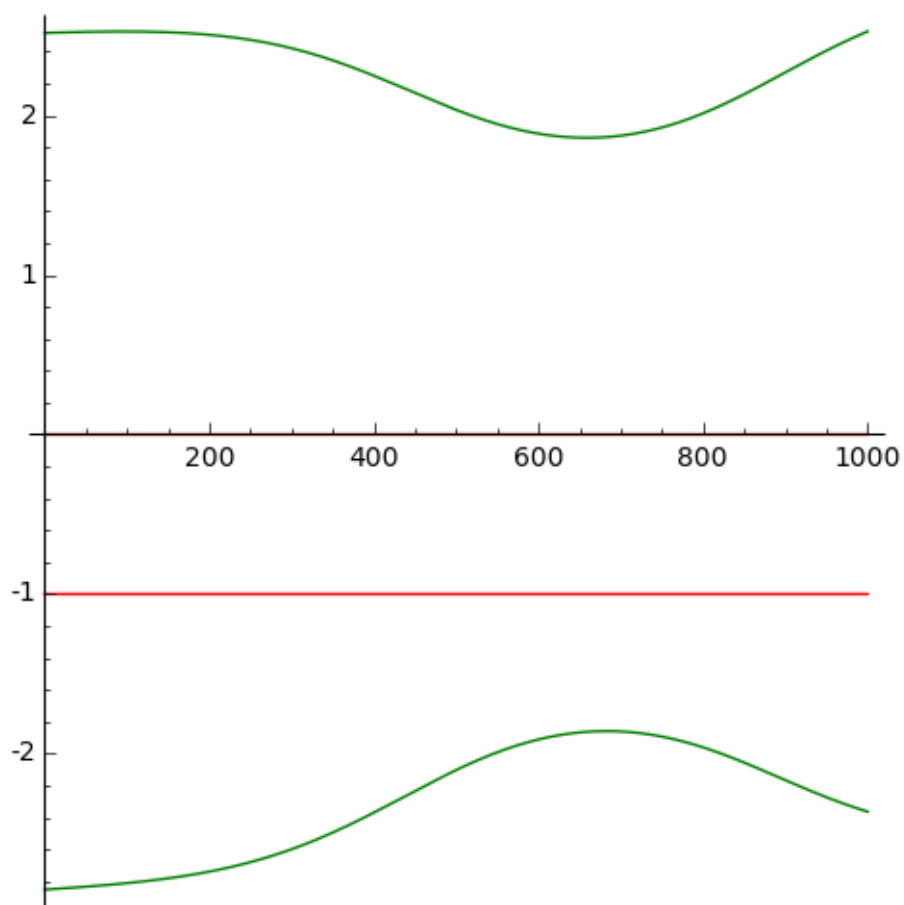
The foodweb adaptive dynamics:

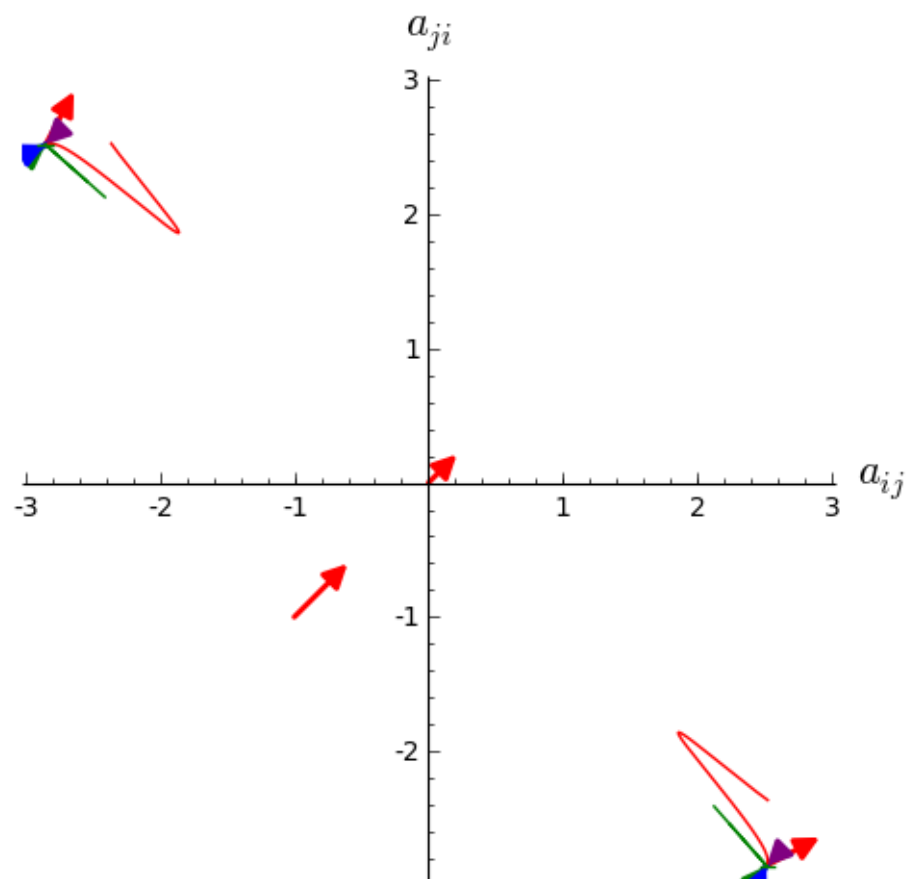
$$\begin{aligned}\frac{du_{b0}}{dt} &= -\frac{9}{625} \left(\hat{X}_{a0} \gamma_{b0} \sin(-u_{a0} + u_{b0}) + 50 \hat{X}_{a0} \sin(-u_{a0} + u_{b0}) \right) \hat{X}_{b0} \\ \frac{d\gamma_{b0}}{dt} &= \frac{9}{1250} \left(2 \hat{X}_{a0} \cos(-u_{a0} + u_{b0}) + 5 \hat{X}_{a0} \right) \hat{X}_{b0} \\ \frac{d\sigma_{b0}}{dt} &= 0 \\ \frac{du_{a0}}{dt} &= -\frac{2}{125} \left(\hat{X}_{b0} \sigma_{a0} \sin(-u_{a0} + u_{b0}) + 50 \hat{X}_{b0} \sin(-u_{a0} + u_{b0}) \right) \hat{X}_{a0} \\ \frac{d\gamma_{a0}}{dt} &= 0 \\ \frac{d\sigma_{a0}}{dt} &= -\frac{1}{125} \left(2 \hat{X}_{b0} \cos(-u_{a0} + u_{b0}) + 5 \hat{X}_{b0} \right) \hat{X}_{a0}\end{aligned}$$

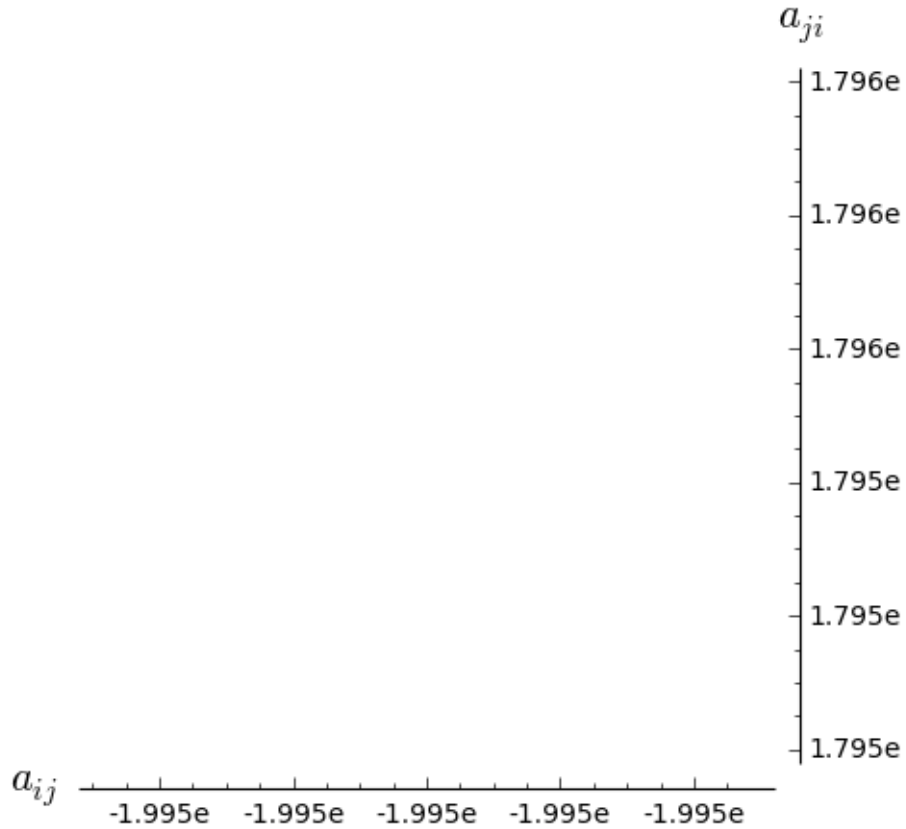












```
[WorkingWiki encountered errors: Error: <ww-make-failed: foodweb2-2-2.tikz.crop.pdf,
/Selection_Gradients/foodweb2-2-2.tikz.crop.pdf.make.log> ]

[WorkingWiki encountered errors: Error: <ww-make-failed: foodweb2-2-2.sage.out.tex-inline
/Selection_Gradients/foodweb2-2-2.sage.out.tex-inline.make.log> ]

[WorkingWiki encountered errors: Error: <ww-make-failed: foodweb2-2-2.plot.sage.out.tex-1
/Selection_Gradients/foodweb2-2-2.plot.sage.out.tex-inline.make.log>
]

[WorkingWiki encountered errors: Error: <ww-make-failed: foodweb2-2-2-adap.png,
/Selection_Gradients/foodweb2-2-2-adap.png.make.log> ]

[WorkingWiki encountered errors: Error: <ww-make-failed: foodweb2-2-2-x-vs-t.png,
/Selection_Gradients/foodweb2-2-2-x-vs-t.png.make.log> ]

[WorkingWiki encountered errors: Error: <ww-make-failed: foodweb2-2-2-a-vs-a.png,
/Selection_Gradients/foodweb2-2-2-a-vs-a.png.make.log> ]

[WorkingWiki encountered errors: Error: <ww-make-failed: foodweb2-2-2-a-vs-a-detail.png,
/Selection_Gradients/foodweb2-2-2-a-vs-a-detail.png.make.log> ]
```


[WorkingWiki encountered errors: Error: <ww-make-failed: foodweb2-2-2-a-vs-t.png,
/Selection_Gradients/foodweb2-2-2-a-vs-t.png.make.log>]