ATNr: Allometric trophic models in R

Benoit Gauzens*[1,2], Ulrich Brose[1,2], Eva Delmas[3], and Emilio Berti[1,2]

[1]EcoNetLab, German Centre for Integrative Biodiversity Research (iDiv)
Halle-Jena-Leipzig 04103 Germany
[2]Institute of Biodiversity, Friedrich Schiller University Jena 07743 Germany
[3]Institut des Sciences de la Forêt Tempérée, Université du Québec en Outaouais

* Benoit Gauzens, Puschstraße 4, 04103 Leipzig. benoit.gauzens@idiv.de

Abstract

1. Understanding and predicting how densities of interacting species change over time has been one of the main goals of community ecology, which has become a pressing challenge in the context of global change.

2. We present the R package *ATNr*, which provides an implementation of different versions of Allometric Trophic Network models (Yodzis and Innes (1992)) that simulate the biomass dynamics of trophically interacting species.

3. Relying on *C++* routines, the *ATNr* proposes an efficient and standardized implementation of the different ATNs models.

4. By proposing a set of built in functions ready to use in a language widely used in the community of ecologists, the *ATNr* package offers an easy access to ATN models.

Keywords

Allometric Trophic Network, Bio-energetic, Community ecology, Food web, Modelling, Population dynamics

Introduction

Understanding and predicting how densities of interacting species change over time has been one of the main goals of community ecology and that has become a pressing challenge in the context of global change, which asks

for predictions of how anthropogenic and natural changes in environmental conditions will influence the functioning of future communities. Indeed, species biomass density influences ecological processes (e.g. energy fluxes), species persistence, and ecosystem stability (Kramer et al. (2018); Gauzens et al. (2019)). Therefore, understanding how species' biomass changes over time is necessary for basic ecological research as well as applied management and conservation. However, tackling this issue with experimental approaches faces several limitations, as collecting temporal datasets on the dynamics of species from complex communities over long period of times is nearly impossible.

Due to such empirical limitations, predictions on dynamic of species densities are usually achieved using theoretical models that implement ordinary differential equations (ODEs) describing the rate of change in species' densities depending on their interactions with the other species in the community (Volterra, 1927; Sauve and Barraquand, 2020; Boit et al., 2012). Parametrising such models, however, can easily become unmanageable for large communities, due to the required quantification of species' biological rates and interspecific interaction strengths. To solve this problem, Yodzis & Innes (1992) proposed a "bio-energetic" model that quantifies the key parameters driving species dynamics using allometric relationships with body mass (Williams et al., 2007). This greatly simplified the problem of model parametrisation, allowing to use body mass in order to derive the other parameters.

The general approach of the bio-energetic model, extended by the development of metabolic theory (Brown et al., 2004), led to a new family of models, called Allometric Trophic Network models (ATNs, Martinez 2020),

as their parametrisation heavily relies on the body mass of species. In practice, allometric relationship are used to relate the body mass of species to several of the biological processes driving feeding rates and species dynamics (for instance, attack rate, handling time, and metabolic rates are all function of species' body mass) and, in some cases, to estimate the presence or absence of trophic interactions Schneider et al. (2016). Interestingly, ATNs have been shown to have the potential to represent ecosystem complexity realistically, predicting accurately community dynamics (Boit et al., 2012; Curtsdotter et al., 2019) and responses to environmental gradients (Fussmann et al., 2014). ATN models provide thus a practical compromise between biological complexity (their parametrisation is eased by the use of few easily measurable traits, i.e. species' body mass) and realism in order to model density dynamics of interacting species.

Three major ATN models are currently used today, hereafter referred as *unscaled*, *scaled*, and *unscaled with nutrients*. These models differ in the possibility of including nutrient dynamics or in the way the different biological rates underlying species growth rate are determined: biological rates can be either estimated from empirical observation without further transformations or scaled to a desired level for the sake of comparability. More formally, the *unscaled* model is based on untransformed biological rates (such as attack rate and handling time) derived using allometries. The *unscaled with nutrients* model relies on the same set of assumptions as the *scaled* version, but it explicitly includes the dynamics of nutrients and their interactions with plant species. The *scaled* model, on the other hand, scales the biological rates to the growth rate of the smallest basal species). Together, these models per-

mit to simulate species' density dynamics in a variety of cases, e.g. when nutrients are explicitly accounted for and when the interest is in the untransformed biological rates or in their relative quantities. Currently, however, the application of ATN models is hindered by a lack of a standard framework, with studies using different implementations for the same underlying mathematical model.

An important step towards a common framework was recently achieved by Delmas et al. (2017), who developed a standard implementation for ATNs in the Julia programming language. Yet, this implementation focuses on a specific class of ATNs (*scaled*) and in a programming language that is unfamiliar to many ecologists. Here, we present the new R package *ATNr*, which provides a standard implementation for the three major classes of ATN models. Importantly, in *ATNr*, the core of the calculations are evaluated by *C++* routines called from within the R environment, providing a standard framework that is familiar to many ecologists while having a high computational performance.

1 ATN models

All ATN models available in *ATNr* describe the biomass dynamic of species by estimating their net growth rates over time. *ATNr* relies on two general assumptions: i) biomass production of species are positively affected by what they consume, i.e. the models are based on energetic transfers between resources and their consumers (plant species can either consume nutrients or have intrinsic logistic growth rates); ii) loss rates of species are driven by their

consumers and by metabolic costs. In this section we cover the main aspects of the three ATN models contained in *ATNr*, highlighting the major differences between them. More details and the units of the different variables can be found in the package vignette (`vignette("model_descriptions", package = "ATNr")`).

1.1 Species growth rate

Growth rate of species i is defined as the change of its biomass (B_i) over time: $\frac{dB_i}{dt}$. For basal species, the growth rate depends on their net intrinsic growth rate minus their metabolic losses and the negative effects of their consumers:

$$\frac{dB_i}{dt} = B_i r_i G_i - X_i B_i - \sum_j B_j F_{ij}, \tag{1}$$

where r_i is the maximum, mass-specific growth rate, G_i defines the proportion of r_i that is realised such as $B_i r_i G_i$ corresponds the net growth rate. X_i is the per-biomass metabolic losses, and F_{ij} is the per-biomass feeding rate of consumer j for resource i ($F_{ij} = 0$ if j does not feed on i).

For non-basal species, net intrinsic growth rate is substituted in eq. 1 by the biomass gained from resources multiplied by the assimilation efficiency of the resource j (e_j):

$$\frac{dB_i}{dt} = B_i \sum_j F_{ji} e_j - X_i B_i - \sum_j B_j F_{ij}, \tag{2}$$

These equations define allometric models as the terms involved are parametrized using scaling relationships with body mass of species (M). For instance, r_i and X_i both follow a power law that is a function of the body mass M_i of

species i:

$$r_i \sim r_0 M_i^{b_r}, \tag{3}$$

$$X_i \sim X_0 M_i^{b_x}, \tag{4}$$

where r_0 and X_0 are normalisation constants and b_r and b_x the scaling exponents.

The three different versions of ATN models proposed in the package all derive from this set of equations, and only differ in how the feeding rates F_{ij} (also called functional response) are specified and how the net growth rates of basal species are calculated. For instance, a major difference in the calculation of G_i among models is whether or not the dynamic of the nutrient pool is considered.

Here we present the formulation of feeding rate and net growth rate used by the different models and depict how the different variables are usually defined by proposing a default parametrisation based on what is used in the literature. As a convention, for all parameters that depend on both resources and consumers, like F_{ij}, the first index refers to the resource and the second to the consumer. We provide the default parametrisations of model terms commonly used in literature in the package vignette.

1.2 Unscaled ATN

This ATN implements the model as in Binzer et al. (2016). In *ATNr* it is called *Unscaled* because the biological rates are used in their untransformed form (contrary to the *Scaled* ATN, see below). The functional response and

growth rate are defined as:

$$F_{ij} = \frac{a_{ij}B_i^q}{1 + c_jB_j + \sum_k h_{kj}a_{kj}B_k^q} \quad (5)$$

$$G_i = 1 - \frac{s_i}{K_i} \quad (6)$$

where a_{ij} is the attack rate of j on i, q is the Hill exponent determining the shape of the functional response (for type II, $q = 1$; for type III, $q \in]1, 2]$), c_j defines the intra-specific interference competition, h_{ij} is the handling time of j on i, K_i is basal species i carrying capacity, and s_i is a parameter defining the ratio between inter- and intra-specific competitions for resources among basal species.

1.3 Scaled ATN

This ATN implements the model as in Delmas et al. (2017). It is called *Scaled* because the biological rates of species are scaled to the growth rate of the smallest basal species. The consequence of this scaling is that time is now defined as the inverse of the growth rate of the smallest producer species and the outcome of the model is to be interpreted on a relative time scale. While this may seem unnecessary complicated, it allows to work at a scale adapted to the study system and to directly compare processes that happen on very different time scales (e.g. phytoplankton bloom and forest growth; Williams et al. 2007). In particular, r_i and x_i from eq. 1 and 2 are divided by the intrinsic growth rate and metabolic cost of the basal species with the smallest body mass. When species are sorted by body mass, the first one is the smallest species and: $r_i = \frac{r_i}{r_1}$, $x_i = \frac{X_i}{r_1}$. Then, it introduces

the maximum feeding rate of species relative to their metabolic rate y_i, i.e. the ratio between maximum feeding rate and metabolic rate.

This redefinition of the variables implies a redefinition of the equations describing the dynamics of species:

$$\frac{dB_i}{dt} = B_i r_i G_i - x_i B_i - \sum_j x_j y_j B_j F_{ij} \tag{7}$$

$$\frac{dB_i}{dt} = x_i y_i B_i \sum_j F_{ji} e_j - x_i B_i - \sum_j x_j y_j B_j F_{ij}, \tag{8}$$

The functional response and growth rate are then defined as:

$$F_{ij} = \frac{w_{ij} B_i^q}{B_0^q + c_j B_j + \sum_k w_{kj} B_k^q} \tag{9}$$

$$G_i = 1 - \frac{s_i}{K_i} \tag{10}$$

where w_{ij} is the relative consumption rate of j on i, such that $\sum_i w_{ij} = 1$.

1.4 Unscaled ATN with nutrients

This ATN implements the model as in Schneider et al. (2016). It differs from the *Unscaled* in how the attack rate is defined and, more importantly, by allowing growth rate of plants to be determined by a nutrient pool which dynamic is explicitly modeled. The functional response and growth rate are defined as:

$$F_{ij} = \frac{w_{ij} b_{ij} B_i^q}{m_i \left(1 + c_j B_j + \sum_k w_{kj} h_{kj} b_{kj} B_k^q\right)} \tag{11}$$

$$G_i = \min_{\forall\, n \,\in\, \text{nutrients}} \left(\frac{N_n}{K_{ni} + N_n} \right) \tag{12}$$

where b_{ij} is the resource-specific capture coefficient, m_i is body mass of i, N_n is the concentration of nutrient n, and K_{ni} is the nutrient uptake efficiency of basal species i on nutrient n, smaller for higher efficiencies. The change of nutrient concentration over time is described by another set of differential equations:

$$\frac{dN_n}{dt} = D(S_n - N_n) - v_{ni} \sum_i r_i G_i B_i. \tag{13}$$

Here, D is the global turnover rate that determines the rate by which the nutrients are refreshed, S_n is the maximal concentration of nutrient n, and v_{ni} sets the relative content of nutrient n in plant i.

2 Using the ATNr package

The *ATNr* package consists of compiled *C++* routines, which define the ATN models, together with R functions to interface with the instances of the ATNs and utility functions to solve the ordinary differential equations (ODE) and plot results. The framework of *ATNr* can be divided into three steps (Table 2): 1) create an instance of an ATN model; 2) initialise parameter values (potentially using the default parametrisation proposed); and 3) solve the ODE using an ODE solver.

Step	*ATNr* function	required input
Create an instance of an ATN model	`create_model_Scaled()`	nb_s, nb_b, BM and fw
	`create_model_Unscaled()`	nb_s, nb_b, BM and fw
	`create_model_Scaled_nuts()`	nb_s, nb_b, nb_n, BM and fw
Initialize parameters to their default values	`initialise_default_Scaled()`	*model*
	`initialise_default_Unscaled()`	*model* and *temperature*
	`initialise_default_Scaled_nuts()`	*model*, *L.mat* and *temperature*
Solve ODE and obtain biomass dynamics	`lsoda_wrapper()`	t, y, *model*
Plot biomass dynamics	`plot_odeweb()`	x, nb_s, *model*

Table 1: Step and *ATNr* functions to create ATN models, specify default parametrisations, and solve the biomass dynamics. nb_s is the number of species, nb_b the number of basal species, nb_n the number of nutrients, BM species body mass, fw the food web matrix, *model* an instance of ATN model, *temperature* the environmental temperature in Celsius, *L.mat* the L matrix, t the integration time steps, y the starting biomasses of species and nutrients, and x is the matrix with ODE solutions as returned by `lsoda_wrapper()`.

To create an ATN model, an adjacency matrix of the food web is needed to define consumer-resource interactions together with species body masses to parametrise the ATN. The adjacency matrix can be derived from empirical data or simulated using synthetic food webs. *ATNr* contains two functions to generate synthetic food webs. The first is based on the niche model (Williams and Martinez, 2000; Allesina et al., 2008): `create_niche_model(S, C)`, where S is the number of species and C the connectance of the network. The second is based on the L matrix approach, which uses the consumer-resource body mass ratio (Schneider et al., 2016): `create_Lmatrix(BM, nb_b, Ropt = 100, gamma = 2, th = 0.01)`, where BM is a vector of species body masses, nb_n the number of basal species, $Ropt$ the optimal predator-prey body mass ratio, γ the width of the feeding kernel, and th is the threshold below which interactions are considered to not be realized.

From species body masses and trophic network, ATN models can be created with the functions `create_model_Scaled()` for *scaled* models, `create_model_Unscaled()` for *unscaled* models, and `create_model_Unscaled_nuts()` for *unscaled with nutrients* models. All three functions generate a *C++* object (formally an instance of the *C++* class) accessible within the *R* environment as an object of class *S4*. Once the model is created, it can be initialised to the default values used in literature (Delmas et al., 2017; Binzer et al., 2016; Schneider et al., 2016) by calling `initialise_default_Scaled()`, `initialise_default_Unscaled()`, and `initialise_default_Unscaled_nuts()`.

Once the ATN models are properly created and initialised, we can solve the ODEs by specifying the integration steps and biomass of species at starting conditions as R vectors and calling an ODE solver. In *ATNr*, the wrapper

function `lsoda_wrapper(steps, biomass, atn.model)` calls the lsoda solver from package *deSolve*.

As an example to summarize the general framework of *ATNr*, we compute biomass dynamics for a food web of 20 species generated using the niche model and implementing the *unscaled* ATN model. We start by creating a food web of 20 species using the niche model and specifying a connectance of 0.3:

```
library(ATNr)
set.seed(123)
n_species <- 20 #number of species
conn <- 0.3 #connectance
fw <- create_niche_model(n_species, conn)
```

We then calculate the number of basal species and species' trophic levels, which are used to obtain their body masses:

```
n_basal <- sum(colSums(fw) == 0) #number basal species
TL = TroLev(fw) #trophic levels
masses <- 1e-2 * 10 ^ (TL - 1)
```

We can now initialise an instance of the *unscaled* ATN model, setting the parameters to their default values:

```
atn.unscaled <- create_model_Unscaled(
  n_species,
  n_basal,
  masses,
  fw
)
initialise_default_Unscaled(atn.unscaled)
```

Once we define starting biomasses (here randomly drawn $\in [2, 3]$) and the integration steps, the ODE can be solved:

```
biomasses <- runif(n_species, 2, 3)
times <- seq(1, 2e8, 1e6)
sol <- lsoda_wrapper(times, biomasses, atn.unscaled)
```

The variable *sol* is a *deSolve* matrix, with *n_species + 1* columns and as many rows as the integration steps. The first column contains the integration steps and the remaining column the biomass of species at each time step. The biomass dynamics can be plotted using *ATNr* function `plot_odeweb()` (Fig. 1):

```
plot_odeweb(sol, n_species)
```

From *sol*, it is straightforward to obtain several metrics with key importance for community ecology, e.g. the number of species that went extinct (i.e. species for which biomass goes below a certain threshold, here 10^{-6}):

```
limit.extinction <- 1e-6
final.biomasses <- sol[nrow(sol), -1] #removing time column
sum(final.biomasses < limit.extinction) #number of species extinct
[1] 0
```

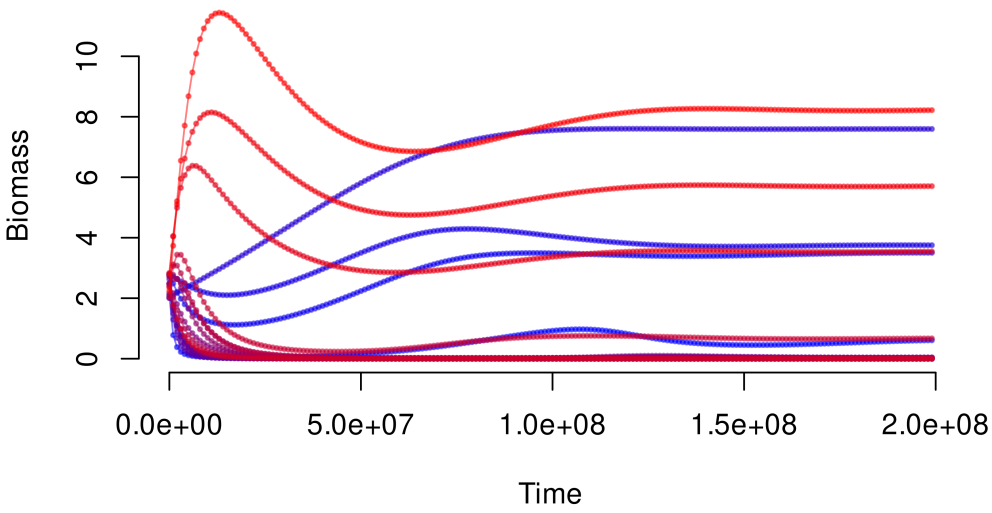

Figure 1: Biomass dynamics as evaluated in the example code. Colors show different species, with dots being the solved dynamics at the pre-defined integration steps.

3 Applications

To show the potential applications of *ATNr*, we explored two cases: 1) the effects of temperature on species persistence and 2) the paradox of enrichment. These examples are also shown in the package vignette (`vignette("ATNr")`).

3.1 Effect of temperature on species persistence

ATNr makes it relatively easy to vary one parameter to assess its effect on the population dynamics. As example, we focus here on environmental temperature, which has been shown to influence biological rates of organism and to play an important role in their persistence (Gauzens et al., 2020). In *ATNr*, temperature can be specified in the *unscaled* and *unscaled with nutrients* models at initialisation: `initialise_default_Unscaled(model, temperature)` and `initialise_default_Unscaled_nuts(model, L.mat, temperature)`. We first create a food web using the L matrix approach:

```
set.seed(12)
n_species <- 50 #number of species
n_basal <- 20 #number of basal species
n_nut <- 2 #number of nutrients
masses <- 10 ^ c(  #body mass of species
  sort(runif(n_basal, 1, 3)),
  sort(runif(n_species - n_basal, 2, 9))
)
L <- create_Lmatrix(masses, n_basal, # L matrix
                    Ropt = 50, gamma = 2)
fw <- L
fw[fw > 0] <- 1 #binarize L matrix
```

As we are interested in modelling nutrients explicitly, we use here the ATN model including nutrients. We thus create and initialise an instance of the *unscaled with nutrients* ATN model:

```
model <- create_model_Unscaled_nuts(
  n_species, n_basal, n_nut,
  masses, fw
)
```

To study the effect of temperature on species persistence, it is sufficient to loop over a temperature range and re-initialise the model instance at every iteration (Fig. 2):

```
temperatures <- seq(4, 22, by = 2) #temperature gradient
extinctions <- rep(NA, length(temperatures)) #store results
biomasses <- runif(n_species + n_nut, 2, 3) #starting biomasses
times <- seq(0, 100000, 100) #time steps

# loop over temperatures
i <- 0
for (t in temperatures){
  model <- initialise_default_Unscaled_nuts(model, L, temperature = t)
  model$q <- 1.2 # constant q
  model$S <- rep(10, n_nut) # constant S
  # running simulations for the Schneider model:
  sol <- lsoda_wrapper(times, biomasses, model, verbose = FALSE)
  i <- i + 1
  # first three columns are: time, nutrient 1 and nutrient 2,
  # which are not of interest: starting from column 4.
  extinctions[i] <- sum(sol[nrow(sol), 4:ncol(sol)] < 1e-6)
}
```

Here, q and S were fixed to constant values, instead of being randomly sampled, in order to make runs at different temperature comparable.

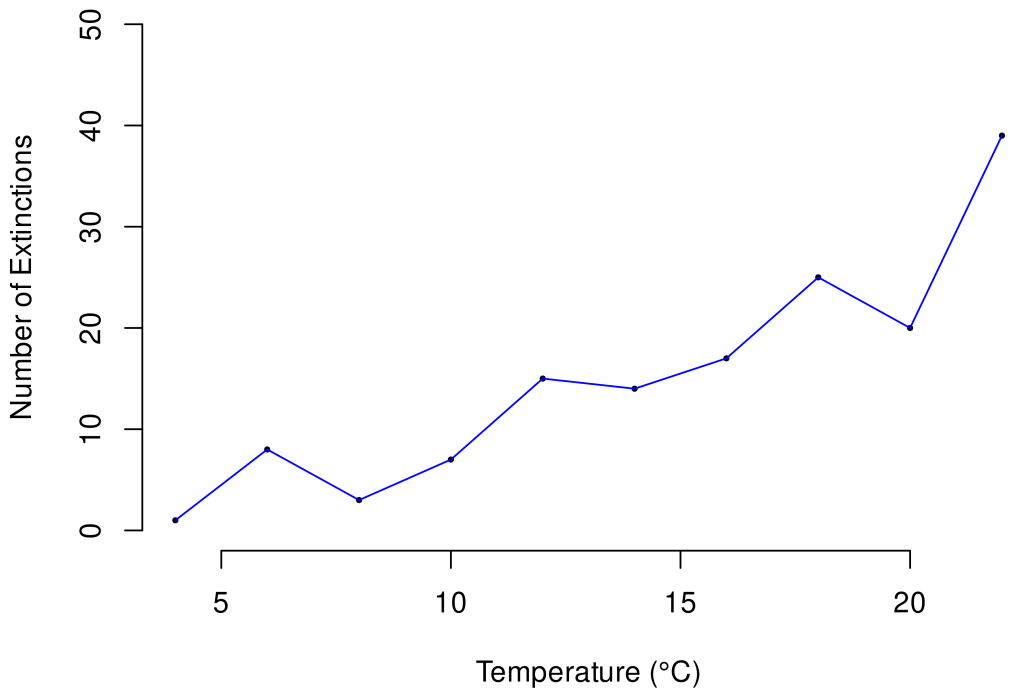

Figure 2: Effect of temperature on species persistence using the *unscaled with nutrients* ATN model. For increasing temperatures, more species go extinct (total species richness = 50).

As a note, we want to point out that in the above example there is no need to create separate instances of the ATN model, as internal parameters (temperature in this case) are changed only through `initialise_default_Unscaled_nuts()`. When this is not the case (e.g. changing number of nutrients or species) a new instance of the ATN model is needed for each simulation.

3.2 Paradox of enrichment

Here, we want to test the paradox of enrichment, i.e. that increasing the carrying capacity of basal species may destabilize population dynamics, by using the *scaled* model. First, we create a food web with 10 species and initialise the model:

```
set.seed(1234)
S <- 10
fw <- create_niche_model(S, C = 0.15)
TL <- TroLev(fw) #to calculate masses
masses <- 0.01 * 100 ^ (TL - 1) #body mass of species
mod <- create_model_Scaled(
  nb_s = S,
  nb_b = sum(colSums(fw) == 0),
  BM = masses,
  fw = fw
)
mod <- initialise_default_Scaled(mod)
times <- seq(0, 300, by = 2) #time steps
biomasses <- runif(S, 2, 3) #starting biomasses
```

Then, we solve the system specifying the carrying capacity of basal species equal to one (`mod$K <- 1`) and then increased this to 10 (`mod$K <- 10`; Fig. 3):

```
mod$K <- 1
sol1 <- lsoda_wrapper(times, biomasses, mod, verbose = FALSE)
mod$K <- 10
sol10 <- lsoda_wrapper(times, biomasses, mod, verbose = FALSE)
```

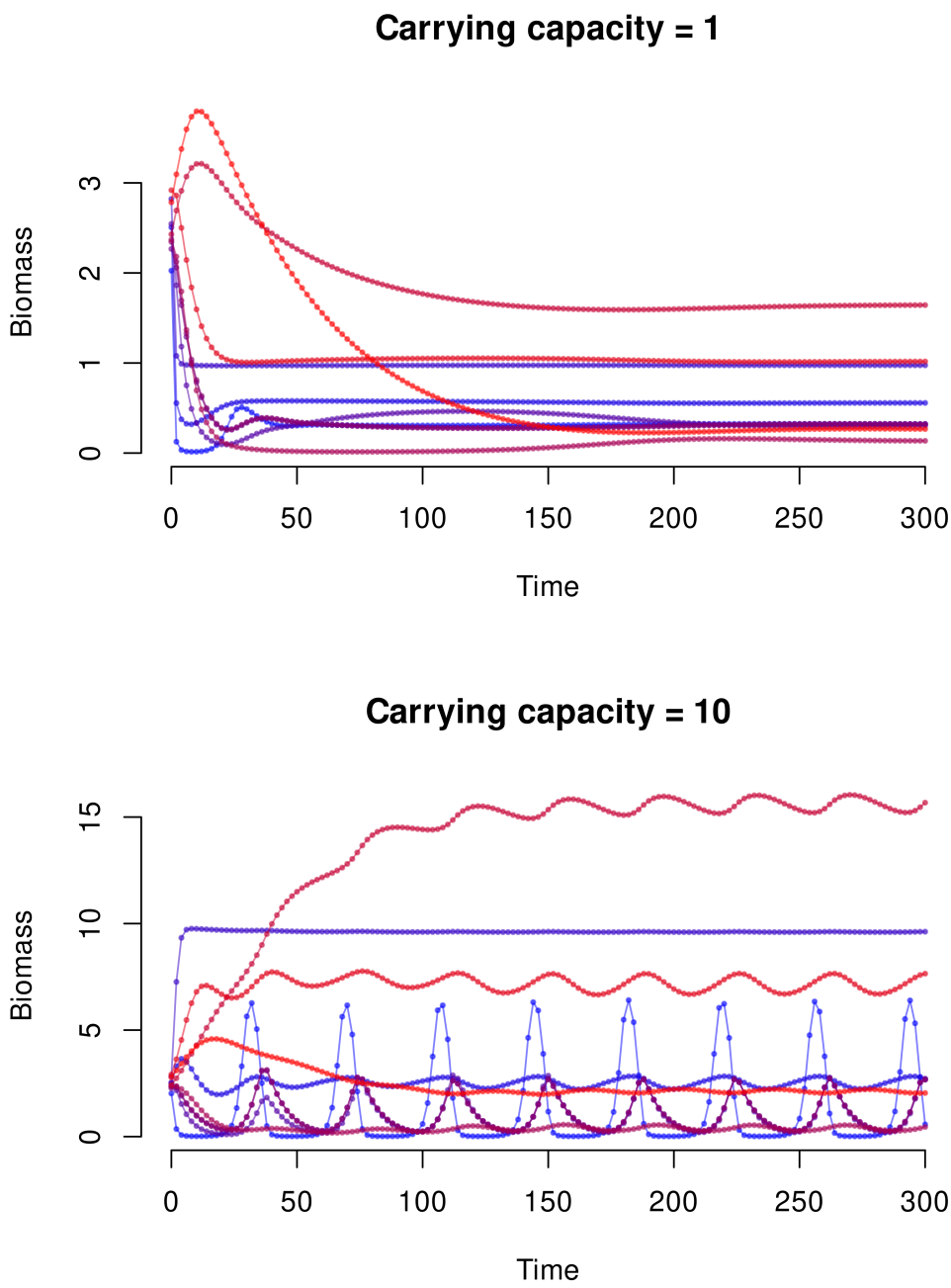

Figure 3: Paradox of enrichment using the *unscaled* ATN model. For higher carrying capacity of basal species ($K = 10$), dynamical oscillations arise, destabilizing the community.

4 Conclusion

The R package *ATNr* provides a standardized implementation of different versions of Allometric Trophic Network models, describing the dynamics over time of trophically interacting populations using allometric relationships. *ATNr* provides four major advantages: First, as our implementation relies on *C++* routines, it allows to efficiently use ATN models in order to simulate the dynamics of interacting populations, up to several hundreds of species. Second, *ATNr* unifies the three major ATN models into a single library, providing a standardized implementation that enhances comparisons of different ATNs within a study and compatibility across several projects. Third, it provides an interface to ATN models in the R language, i.e. suited for a variety of users, especially in ecology. Finally, *ATNr* is ready-to-use and makes ATN models available also for novices, while allowing more expert users some degree of flexibility.

Acknowledgements

BG, EB and UB cknowledge the support of the German Centre for Integrative Biodiversity Research (iDiv) Halle-Jena-Leipzig funded by the German Research Foundation (FZT 118). We thank Daniel Reuman for his helpful comments on the package.

Conflict of Interest statement

All authors declare that they have no conflicts of interest.

Author contributions

BG and EB designed the study and developed the package. BG wrote the first draft of the manuscript. All authors made substantial revisions and comments to the manuscript.

Data accessibility

A beta version of *ATNr* is deposited on CRAN: https://cran.r-project.org/package=ATNr. A development version can be accessed on GitHub: https://github.com/gauzens/ATNr. Should our manuscript be positively evaluated by reviewers, a finalised version will be pushed on CRAN

References

Allesina, S., Alonso, D., and Pascual, M. (2008). A general model for food web structure. *science*, 320(5876):658–661.

Binzer, A., Guill, C., Rall, B. C., and Brose, U. (2016). Interactive effects of warming, eutrophication and size structure: impacts on biodiversity and food-web structure. *Global Change Biology*, 22(1):220–227.

Boit, A., Martinez, N. D., Williams, R. J., and Gaedke, U. (2012). Mechanistic theory and modelling of complex food-web dynamics in lake constance. *Ecology Letters*, 15(6):594–602.

Brown, J. H., Gillooly, J. F., Allen, A. P., Savage, V. M., and West, G. B. (2004). Toward a metabolic theory of ecology. *Ecology*, 85(7):1771–1789.

Curtsdotter, A., Banks, H. T., Banks, J. E., Jonsson, M., Jonsson, T., Laubmeier, A. N., Traugott, M., and Bommarco, R. (2019). Ecosystem function in predator–prey food webs—confronting dynamic models with empirical data. *Journal of Animal Ecology*, 88(2):196–210.

Delmas, E., Brose, U., Gravel, D., Stouffer, D. B., and Poisot, T. (2017). Simulations of biomass dynamics in community food webs. *Methods in Ecology and Evolution*, 8(7):881–886.

Fussmann, K. E., Schwarzmüller, F., Brose, U., Jousset, A., and Rall, B. C. (2014). Ecological stability in response to warming. *Nature Climate Change*, 4(3):206–210.

Gauzens, B., Barnes, A., Giling, D. P., Hines, J., Jochum, M., Lefcheck, J. S., Rosenbaum, B., Wang, S., and Brose, U. (2019). fluxweb: An r package to easily estimate energy fluxes in food webs. *Methods in Ecology and Evolution*, 10(2):270–279.

Gauzens, B., Rall, B. C., Mendonça, V., Vinagre, C., and Brose, U. (2020). Biodiversity of intertidal food webs in response to warming across latitudes. *Nature Climate Change*, 10(3):264–269.

Kramer, A. M., Berec, L., and Drake, J. M. (2018). Editorial: Allee effects in ecology and evolution. *Journal of Animal Ecology*, 87(1):7–10.

Martinez, N. D. (2020). Allometric trophic networks from individuals to socio-ecosystems: Consumer–resource theory of the ecological elephant in the room. *Frontiers in Ecology and Evolution*, 8.

Sauve, A. M. C. and Barraquand, F. (2020). From winter to summer and back: Lessons from the parameterization of a seasonal food web model for the białowieża forest. *Journal of Animal Ecology*, 89(7):1628–1644.

Schneider, F. D., Brose, U., Rall, B. C., and Guill, C. (2016). Animal diversity and ecosystem functioning in dynamic food webs. *Nature Communications*, 7(1):1–8.

Volterra, V. (1927). Variazioni e fluttuazioni del numero d'individui in specie animali conviventi.

Williams, R. J., Brose, U., and Martinez, N. D. (2007). Homage to yodzis and innes 1992: scaling up feeding-based population dynamics to complex ecological networks. In *From energetics to ecosystems: the dynamics and structure of ecological systems*, pages 37–51. Springer.

Williams, R. J. and Martinez, N. D. (2000). Simple rules yield complex food webs. *Nature*, 404(6774):180–183.

Yodzis, P. and Innes, S. (1992). Body size and consumer-resource dynamics. *The American Naturalist*, 139(6):1151–1175.