

1 Supplementary Information Appendix S1 - Behavior of the Rosenzweig-MacArthur and similar predator-prey models

In the RM model, the killing rate of prey ($g(N, P)$ in eqs. 1 and 2 of the main text) is a Holling type II functional response:

$$\frac{dN}{dt} = rN - \alpha N^2 - \frac{cNP}{D + N}, \quad (1)$$

$$\frac{dP}{dt} = e \frac{cNP}{D + N} - \mu P. \quad (2)$$

Here, r is the prey maximum population growth rate, α the per capita effect of an additional prey competitor (the carrying capacity is $K = r/\alpha$), c the maximum prey attack rate, D the half-saturation constant accounting for a saturation of prey killing rates at high prey densities, and e the conversion efficiency, assuming a linear numerical response. Depending upon parameter values, the model exhibits either damped oscillations converging to a fixed point or limit cycle oscillations (provided both species persist). If the prey growth rate is progressively increased from an initially small to a sufficiently large value, the limit cycle emerges from the fixed point (Rosenzweig, 1971), a mathematical behavior known as a Hopf bifurcation (Fig. S1.1). A simplified, heuristic illustration of the cycle mechanism is provided in Fig. S1.2.

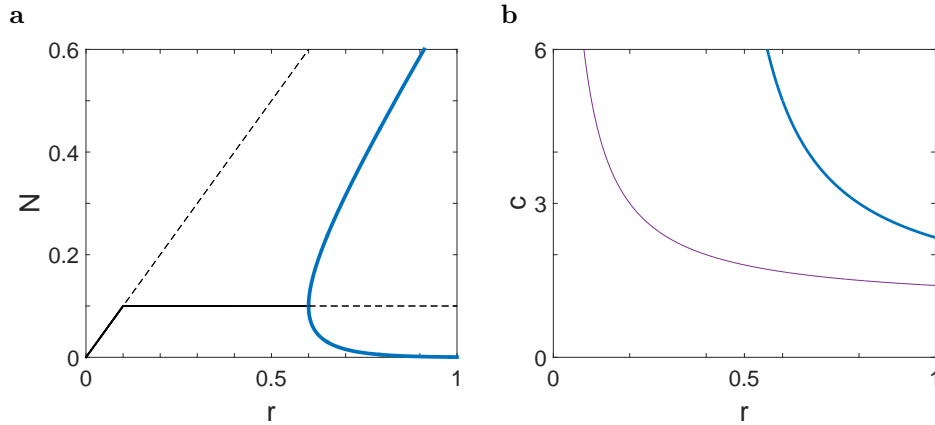


Fig. S1.1: In the one parameter diagram (a), the solid black and dashed black curves show the equilibrium value of N at the steady state as a function of r . The solid curve denotes a stable equilibrium and the dashed an unstable equilibrium. The transcritical bifurcation occurs as r increases through $r_{TC} = 0.1$ and r becomes large enough for prey to allow the survival of the predator in a stable coexistence steady state. The thick blue curve shows the maximum and minimum value of N on the stable attracting period orbit that is born due to the Hopf bifurcation when the coexistence loses stability as r increases through $r_{HB} = 0.6$. In the two parameter diagram (b), the thick blue curve shows the values of (r, c) at the Hopf bifurcation, and the thin purple curve shows the values at the transcritical bifurcation. Above the thick blue curve there is oscillatory coexistence. Between the thin purple curve and the thick blue curve there is coexistence at steady state. Below the thin purple curve there is extinction of the predator. In the one parameter diagram, $c = 5$. All other parameter values are the same as in Fig. 2 of the main text.

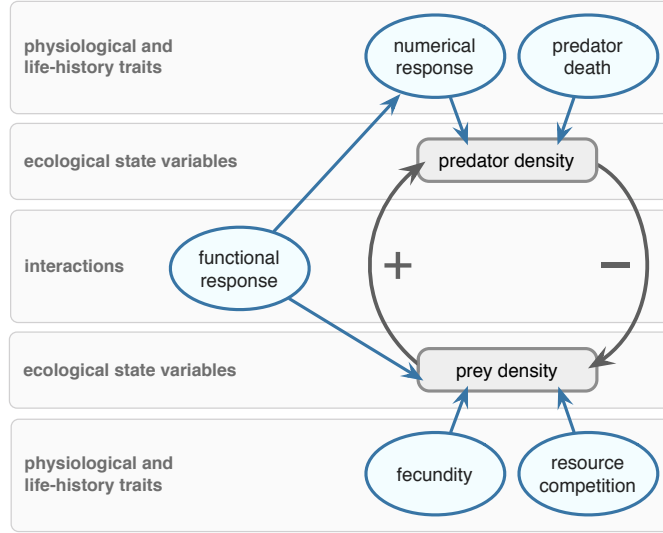


Fig. S1.2: Mechanism of a predator-prey cycle (eqs. (1)-(2)), like that of the Rosenzweig-MacArthur model. Negative effects of predators on prey density and positive effects of prey on predator density (gray arrows), create a delayed negative feedback loop (a *structural* feature) with potentially cyclic dynamics. The actual occurrence of cycles depends on *functional forms* such as the functional response $\frac{cNP}{D+N}$, as well as life history traits (for example, high prey fecundity resulting in high r ; blue ovals).

The RM model and other, similar mathematical models can exhibit not only limit cycles, but also cycles partially generated by external forcing such as environmental noise (i.e., noise-sustained oscillations). A forced version of the Bazykin model, a variant of the RM model with a self-regulating predator population (eq. 3) has been used to produce the noise-generated oscillations in Fig. 3c in the main text:

$$\begin{aligned} \frac{dN}{dt} &= rN - \alpha N^2 - \frac{cNP}{D+N}, \\ \frac{dP}{dt} &= e \frac{cNP}{D+N} - \mu P - \underbrace{(ce - \mu) \left(\frac{P}{K_p} \right) P}_{\text{predator regulation}} + \underbrace{\sigma \frac{P}{K_p} \frac{dW}{dt}}_{\text{noise term}}. \end{aligned} \quad (3)$$

Here, $\frac{dW}{dt}$ is a stochastic process that represents white (i.e., temporally uncorrelated) Gaussian noise, σ is a noise scaling factor and K_p is the carrying capacity of the predator population when prey is very abundant ($N \rightarrow \infty$). The stochastic differential equations are most often expressed in the more mathematically rigorous differential form

$$\begin{aligned} dN &= \left(rN - \alpha N^2 - \frac{cNP}{D+N} \right) dt, \\ dP &= \left(e \frac{cNP}{D+N} - \mu P - \underbrace{(ce - \mu) \left(\frac{P}{K_p} \right) P}_{\text{predator regulation}} \right) dt + \underbrace{\sigma \frac{P}{K_p} dW}_{\text{noise term}}. \end{aligned} \quad (4)$$

2 Supplementary Information Appendix S2 - Model fitting tutorial with code

Fitting stochastic parametric models for cycles - in the form of stochastic difference equations - to multi-species time series data is usually performed using log-linear models (Moran, 1953), sometimes in a multivariate setting (Hampton *et al.*, 2013). These have great appeal as their statistical machinery rests on the powerful framework of linear time series modeling.

However, nonlinear and more mechanistic models can be fitted as well (Ives *et al.*, 2008). The classical approach is maximum likelihood, but with the advent of the BUGS language (Bayesian inference Using Gibbs Sampling, Lunn *et al.*, 2000) and its derivatives such as JAGS (Just Another Gibbs Sampler, Plummer, 2003), fitting nonlinear dynamical models has become easy in a Bayesian framework - perhaps easier than maximum likelihood for rather complex models (e.g., New *et al.*, 2009; Kéry & Schaub, 2012). The relative parsimony of the model can then be compared using model selection metrics and predictive criteria such as cross-validation (Hooten & Hobbs, 2015), while absolute measures of fit can be obtained through Bayesian P-values (Kéry & Schaub, 2012).

For illustration, we considered and simulated a stochastic version of the May-Hassell host-parasitoid model (as formulated in Ives & Jansen, 1998):

$$\begin{aligned} N_{t+1} &= N_t \exp(r_N + \epsilon_N) F(P_t), \quad \epsilon_N \sim N(0, \sigma_N^2), \\ P_{t+1} &= N_t \exp(r_P + \epsilon_P) (1 - F(P_t)), \quad \epsilon_P \sim N(0, \sigma_P^2), \end{aligned} \tag{5}$$

with $F(P_t) = (1 + bP_t/k)^{-k}$ the fraction of hosts surviving parasitism and k an aggregation parameter, N_t the number of hosts, P_t the number of parasitoids, and r_N and r_P their respective intrinsic growth rates. The noise terms ϵ_N and ϵ_P are independent Gaussian random variables, thus noise is log-normally distributed.

We then assessed our ability to identify true parameter values and reproduce the host-parasitoid dynamics in Fig. S2.1 below. We also provide commented R and JAGS code for simulating and fitting the model.

Frequentist estimation through maximum likelihood (e.g., Ives *et al.*, 2008) would be likewise doable here, because our simulated model is of moderate complexity. Both frequentist and Bayesian methods rest on the specification of the likelihood for the dynamical model, which is the starting point of any model fitting procedure. In BUGS, the likelihood is written in an iterative manner (see code below), rather similar to the model simulation algorithm, and the estimation method - the Gibbs sampler, which belongs to the family of Markov Chain Monte Carlo methods - uses conditional probabilities to iteratively find the parameters. In a frequentist framework, to find the parameter set that maximizes the likelihood, the likelihood function often has to be expressed mathematically. Writing down the likelihood is made easier by remarking that the model can be written in a logarithmic scale, where multiplicative growth processes become additive and the noise is Gaussian, rather than log-normal:

$$\begin{aligned} \ln(N_{t+1}) &= \ln(N_t) + r_N + \ln(F(P_t)) + \epsilon_N, \quad \epsilon_N \sim N(0, \sigma_N^2), \\ \ln(P_{t+1}) &= \ln(N_t) + r_P + \ln(1 - F(P_t)) + \epsilon_P, \quad \epsilon_P \sim N(0, \sigma_P^2), \end{aligned} \tag{6}$$

This formulation allows to specify a Gaussian conditional probability distribution for $\ln(N_{t+1})$ and $\ln(P_{t+1})$, given the previous values of N_t and P_t . The approach is very general and can be extended to a n -dimensional stochastic difference equation with log-normal noise. Denoting the vector of log-densities $X_t = (\ln(N_t), \ln(P_t))$, we can then write down $\Pr(X_{i,t+1} = x_{i,t+1} | X_t = x_t) = \phi(f_i(x_t), \sigma_i)$ using a Gaussian distribution whose mean is a function of x_t for the species at hand i , $f_i(x_t)$. In our two-dimensional example, for the prey ($i = 1$), we have $f_1(x_t) = x_{1,t} + r_N + \ln(F(e^{x_{2,t}}))$. Finally, we can write down the likelihood of the full dynamic, multi-species and nonlinear population model

$$\mathcal{L}(X) = \prod_{t=1}^{t=t_{\max}-1} \Pr(X_{t+1} = x_{t+1} | X_t = x_t) \Pr(X_1 = x_1)$$

The log-likelihood can then be maximized using classic optimization techniques (e.g., those implemented in `optim()` in R). More complex probability distributions (non-Gaussian) as well as increased nonlinearities or increased dimensionality makes the search for the optimal parameters more difficult.

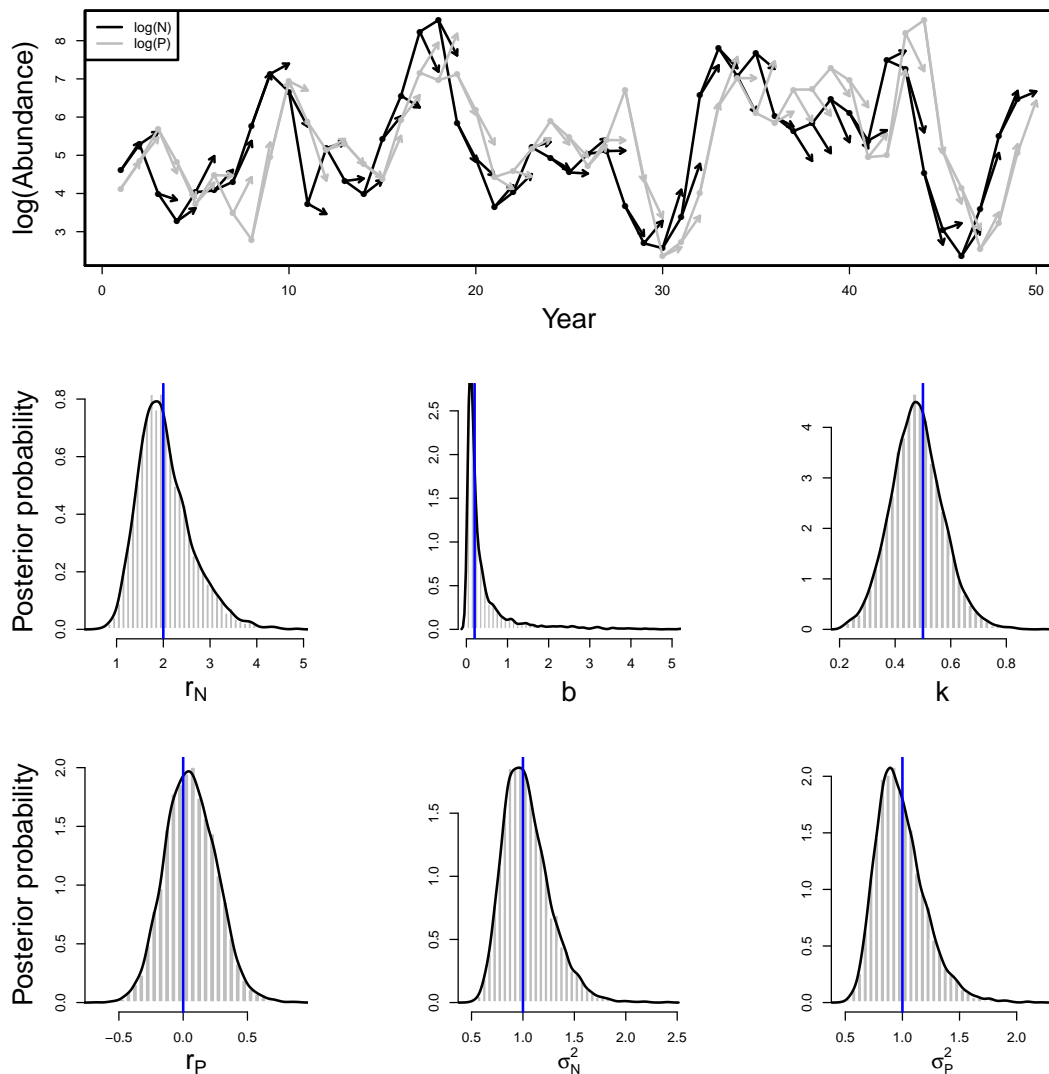


Fig. S2.1: Simulation and model fitting of the May-Hassell host-parasitoid model. Time series of population densities for the host (black) and the parasitoid (gray) in the upper panel. Arrows represent one-step ahead predictions. The lower panels present posterior probability distributions for the parameters, whose modes are similar to the maximum likelihood parameter estimates. Blue vertical lines represent “true”, simulated parameter values, while gray lines and black curves represent respectively histograms and kernel density plots of estimated parameter distributions. Simulated parameter values: $r_N = 2, b = 0.2, k = 0.5, r_P = 0, \sigma_N = \sigma_P = 1$.

R/JAGS code

```
### Code for analyzing noisy Host-Parasitoid system time series data
### F. Barraquand --- Modified May-Hassell model, 18/05/2015
### For use in "Moving forward in circles" Ecology Letters review paper
### Coding style inspired by and modified from Kéry, M., & Schaub, M. (2012).
### Bayesian population analysis using WinBUGS: a hierarchical perspective.
### Academic Press.
```

```

rm(list=ls())
graphics.off()
library("R2jags")      # Load R2jags package
##### Parameters #####
### Parameters for simulation of May-Hassell model
n.years<-50  # Number of years - 25 first, perhaps use 50 or 100
N1<-100 # Initial pop size host
P1<-30 # Initial pop size parasitoid
k<-0.5 # aggregation coefficient
b<-0.2 # attack rate
rmax_V<-2 # Max AVERAGE growth rate (thus not a true max)
rmax_P<-0 # Parasitoids emerging = 1 on average
sigma2.proc<-1 # Process sigma on the log-scale
#####

##### Simulation of data #####
set.seed(43)
y<-N<-P<-numeric(n.years)
N[1]<-N1
P[1]<-P1
rV<-rnorm(n.years-1,rmax_V,sqrt(sigma2.proc))
rP<-rnorm(n.years-1,rmax_P,sqrt(sigma2.proc))
for (t in 1:(n.years-1)){
  N[t+1]<-N[t]*exp(rV[t]) * ((1+b*P[t]/k)^(-k))
  P[t+1]<-N[t]*exp(rP[t]) * (1-(1+b*P[t]/k)^(-k))
}
## Plotting time series
plot(1:n.years,N,type="b")
lines(1:n.years,P,type="b")
# Bundle data
jags.data <- list(T=n.years,logN=log(N),logP=log(P))
##### end of data simulation #####

##### Model specification #####
# Formulating the model in BUGS and storing it
sink("ssm.hostpara1.txt")
cat("
  model {
    ##### Priors #####
    # Priors and constraints initial pop size
    logN[1] ~ dnorm(0,0.01) # Prior for hosts on the log scale
    logP[1] ~ dnorm(0,0.01) # Prior for parasitoids on the log scale
    # Priors for prey population dynamics
    r_V ~ dnorm(1,0.001) # below the truth, rather flat prior
    k ~ dunif(0.2,1) #restricted for now
    sigma_V ~ dunif(0.01,5) # rather vague
    sigma2_V<-pow(sigma_V, 2)
    tau_V<-pow(sigma_V,-2)
    #Priors predator population dynamics
    b ~ dgamma(0.01,0.1)
    r_P ~ dnorm(0,0.1)
    sigma_P ~ dunif(0.01,2) # rather vague
    sigma2_P<-pow(sigma_P, 2)
    tau_P<-pow(sigma_P,-2)
    ##### end of priors #####

```

```

##### Likelihood formulation #####
# state process
for (t in 1:(T-1))
{
  logN[t+1] ~ dnorm(logNupdate[t],tau_V)
  logNupdate[t] <- logN[t] + r_V - k*log(F[t])
  F[t]<-(P[t]/k)*b+1
  N[t]<-exp(logN[t])
  logP[t+1]~ dnorm(logPupdate[t],tau_P)
  logPupdate[t] <- logN[t] + r_P + log(1 - (F[t]^(-k)) )
  #don*t forget it is logN here
  P[t]<-exp(logP[t])
}
##### end of likelihood #####
}
",fill=TRUE)
sink()
##### end of model specification #####

##### Model fitting routines #####
# Initial values
inits <- function () {
  list(sigma_V=runif(1,0.1,2), sigma_P=runif(1,0.1,2),
        r_V=runif(1,0.1,2),r_P=runif(1,-1,1), k=runif(1,0.2,1), b=runif(1,0,2))}

# Parameters monitored
parameters<-c("r_V","k","r_P","sigma2_V","sigma2_P","b","logNupdate","logPupdate","F")

# MCMC settings
nc <- 3 #number of chains
nb <- 14000 # "burn in"
ni<-34000
nt <- 10 # "thinning"

# run model
out <- jags(jags.data, inits, parameters, "ssm.hostpara1.txt", n.chains=nc, n.thin=nt,
n.iter=ni, n.burnin=nb, working.directory = getwd())
print(out, dig = 2)
#store predictions
logNupdate=out$BUGSoutput$mean$logNupdate
logPupdate=out$BUGSoutput$mean$logPupdate
##### end of model fitting #####

##### Plotting time series and parameter estimates #####
pdf(file = "HostPara_wParaNoise_density.pdf",width = 9,height =9)
mat = matrix(c(1,1,1,2,3,4,5,6,7),3,3,byrow=TRUE)
layout(mat, widths=rep(1, ncol(mat)), heights=rep(1, ncol(mat)))
par(mar=c(5,5,2,4)+.1,cex.lab=2,lwd=2)
plot(1:(n.years-1),log(N[1:(n.years-1)]),type="o",pch = 16,bg="black",xlab="Year",
ylab="log(Abundance)")
#lines(1:(n.years-1),logNupdate,type="p")
arrows(1:(n.years-1),log(N[1:(n.years-1)]),2:(n.years),logNupdate,length = 0.05)
par(new=TRUE)
plot(1:(n.years-1),log(P[1:(n.years-1)]),type="o",col="grey",
pch = 16,bg = "grey",xaxt="n",yaxt="n",xlab="",ylab="")
#lines(1:(n.years-1),exp(logXupdate),type="p",col="grey")

```

```

arrows(1:(n.years-1),log(P[1:(n.years-1)]),2:(n.years),logPupdate,col="grey",length = 0.05)
#axis(4)
#mtext("log(P)",side=4,line=3)
legend("topleft",col=c("black","grey"),lty=1,legend=c("log(N)","log(P)"))

#With histogram added
hist(out$BUGSoutput$sims.list$r_V,breaks=50,xlab=expression(r[N]),probability=TRUE,
main=NULL,ylab="Posterior probability",cex.lab=2,
col="gray",border="white")
d<-density(out$BUGSoutput$sims.list$r_V)
lines(d)
#par(new=TRUE)
abline(v=rmax_V,col="blue",lwd=2)

x=out$BUGSoutput$sims.list$b[out$BUGSoutput$sims.list$b<5] ## to see well values
hist(x,breaks=50,xlab="b",probability=TRUE,main=NULL,ylab=NULL,xlim=c(0,5),
col="gray", border="white")
lines(density(out$BUGSoutput$sims.list$b))
abline(v=b,col="blue",lwd=2)

hist(out$BUGSoutput$sims.list$k,breaks=50,xlab="k",probability=TRUE,
main=NULL,ylab=NULL, col="gray", border="white")
lines(density(out$BUGSoutput$sims.list$k))
abline(v=k,col="blue",lwd=2)

hist(out$BUGSoutput$sims.list$r_P,breaks=50,xlab=expression(r[P]),probability=TRUE,
main=NULL,ylab="Posterior probability",cex.lab=2,col="gray", border="white")
lines(density(out$BUGSoutput$sims.list$r_P))
abline(v=rmax_P,col="blue",lwd=2)

hist(out$BUGSoutput$sims.list$sigma2_V,breaks=50,xlab=expression(sigma[N]^2),
probability=TRUE,main=NULL,ylab=NULL,cex.lab=1.5, col="gray", border="white")
lines(density(out$BUGSoutput$sims.list$sigma2_V))
abline(v=sigma2.proc,col="blue", lwd=2)

hist(out$BUGSoutput$sims.list$sigma2_P,breaks=50,xlab=expression(sigma[P]^2),
probability=TRUE, main=NULL,ylab=NULL,cex.lab=1.5, col="gray", border="white")
lines(density(out$BUGSoutput$sims.list$sigma2_P))
abline(v=sigma2.proc,col="blue", lwd=2)
dev.off()
##### End of plotting #####

```

References

- Hampton, S.E., Holmes, E.E., Scheef, L.P., Scheuerell, M.D., Katz, S.L., Pendleton, D.E. & Ward, E.J. (2013). Quantifying effects of abiotic and biotic drivers on community dynamics with multivariate autoregressive (MAR) models. *Ecology*, 94, 2663–2669.
- Hooten, M. & Hobbs, N. (2015). A guide to bayesian model selection for ecologists. *Ecological Monographs*, 85, 3–28.
- Ives, A.R., Einarsson, Á., Jansen, V.A. & Gardarsson, A. (2008). High-amplitude fluctuations and alternative dynamical states of midges in Lake Myvatn. *Nature*, 452, 84–87.
- Ives, A.R. & Jansen, V.A. (1998). Complex dynamics in stochastic tritrophic models. *Ecology*, 79, 1039–1052.
- Kéry, M. & Schaub, M. (2012). *Bayesian population analysis using WinBUGS: a hierarchical perspective*. Academic Press, Cambridge, MA.
- Lunn, D.J., Thomas, A., Best, N. & Spiegelhalter, D. (2000). WinBUGS - a Bayesian modelling framework: concepts, structure, and extensibility. *Statistics and computing*, 10, 325–337.
- Moran, P. (1953). The statistical analysis of the canadian lynx cycle. *Australian Journal of Zoology*, 1, 291–298.
- New, L., Matthiopoulos, J., Redpath, S. & Buckland, S. (2009). Fitting models of multiple hypotheses to partial population data: investigating the causes of cycles in red grouse. *American Naturalist*, 174, 399–412.
- Plummer, M. (2003). Jags: A program for analysis of bayesian graphical models using gibbs sampling. In: *Proceedings of the 3rd international workshop on distributed statistical computing*. Vienna, vol. 124, p. 125.
- Rosenzweig, M. (1971). Paradox of enrichment: destabilization of exploitation ecosystems in ecological time. *Science*, 171, 385–387.