

1     MicroxanoX: an R package for simulating an aquatic  
2         *MICRO*bial ecosystem that can occupy *OX*ic or  
3             *ANOX*ic states.

4                     Rainer M Krug<sup>a,1</sup>, Owen L. Petchey<sup>a</sup>

<sup>a</sup>*Department of Evolutionary Biology and Environmental Studies Winterthurerstrasse 190  
8057 Zurich*

---

5     **Abstract**

Ca. 100 words.

6     *Keywords:* reproducibility; regime shift;, reproducibility; regime shift;

---

7     **1. TODO**

- 8         • **update links**  
9         ☒ **~~update cross references~~**  
10        • **check keywords**  
11        ☒ **~~add more references~~**  
12        • **expand on Impact and Conclusion section**

---

\*Corresponding author

*Email addresses:* `Rainer.Krug@uzh.ch`; `Rainer@krugs.de` (Rainer M Krug),  
`Owen.Petchey@ieu.uzh.ch` (Owen L. Petchey)

<sup>1</sup>Corresponding Author

<sup>2</sup>Equal contribution

## 13 2. Required Metadata

### 14 2.1. Current code version

15 Ancillary data table required for subversion of the codebase.

Nr.	Code metadata description	Please fill in this column
C1	Current code version	v0.9.0
C2	Permanent link to code/repository used for this code version	<a href="https://github.com/UZH-PEG/microxanox">https://github.com/UZH-PEG/microxanox</a>
C3	Code Ocean compute capsule	
C4	Legal Code License	CC BY 4.0
C5	Code versioning system used	git
C6	Software code languages, tools, and services used	<a href="#">R</a>
C7	Compilation requirements, operating environments	<a href="#">R (&gt;= 4.1.0)</a> magrittr tibble ggplot2 patchwork grDevices stats mgcv deSolve dplyr tidyr stringr multidplyr
C8	If available Link to developer documentation/manual	<b>TO BE ADDED</b>
C9	Support email for questions	Rainer.Krug@uzh.ch; Rainer@krugs.de

## 16 3. Motivation and significance

17 Mathematical models play a key role in the development of understanding  
 18 about how ecosystems work and how they respond to environmental changes  
 19 [1, 3, 13]. They are also important for developing hypotheses that can then be  
 20 tested in empirical studies. One area of ecology in which simple models have

played a influential role is the area of how ecosystems respond to gradual change in an environmental driver [9]. An environmental driver is a environmental condition that affects an ecosystem, but is assumed to not be affected by the ecosystem, such as the rate of nutrient input into a lake.

It is conceivable that an ecosystem state, such as the total biomass of a particular type of bacteria, may remain unchanged when an environmental driver changes. It is also possible that the ecosystem state changes gradually. It is also possible that the ecosystem state changes abruptly to a new state that is difficult to recover from [9]. This possibility for abrupt, perhaps catastrophic changes that are difficult to reverse causes considerable concern [4, 6, 12].

An example where a gradual change of an environmental variable causes an abrupt change of the system is the switch from an aerobic (oxygen is available for metabolism) to anaerobic (oxygen generally unavailable) state in a microbial ecosystem. Three types of microbes occur in the model: cyanobacteria (CB) dominating the oxic state, and two types of sulfur bacteria that dominate the anoxic state (sulfate reducing bacteria (SB) and phototrophic sulfur bacteria (PB)). This system has been investigated by Bush et al. [2] in a simulation study of a mathematical model. They showed that gradual change in the rate at which oxygen could diffuse into the ecosystem (termed the oxygen diffusivity) could cause catastrophic changes in the ecosystem state that would be difficult to reverse.

One feature of the study by Bush et al. [2] was limited biodiversity. Specifically, there was no biodiversity within each of the three types of bacteria. This leaves open the question of if and how biodiversity within these types (i.e. functional groups) of bacteria, in their model, affects the ecosystem response to environmental change. This limitation is not specific to the study of Bush et al. [2], there are few if any studies of the effects of biodiversity on abrupt transitions between ecosystem states.

We decided to fill this research gap by making a simulation study of how within functional group biodiversity affects ecosystem responses to environmental change Limberger [5], and to base our work on the work and model of Bush et al. [2]. It was with this goal in mind that we developed the *microxanox* package. The first stage of development was to write code from scratch (as there was no available code to start from) and to confirm that this new implementation would reproduce the previously published results. The resulting reproduction is available as one of the package vignettes: [vignette Partial reproduction of Bush et al.](#)

The second stage was to add functionality that would be necessary to answer our research question. Most importantly, we made it possible to have multiple species of bacteria within each of the three functional groups, for the multiple species to differ in their characteristics, and to vary the number of species and amount of variability among them. We also added functionality that allowed: temporally varying environmental conditions, addition of random noise to state variables, and immigration. In addition to the model itself, the package includes some functions to analyse the results as well as visualize the results to provide a starting point for customized visualizations based on own requirements. The

67 basic and additional functionality is described in the package [User Guide](#).

## 68 4. Software description

69 The *microxanox* package is for simulating a three functional group system  
70 (*CB*: cyanobacteria, *PB*: phototrophic sulfur bacteria, and *SB*: sulfate-reducing  
71 bacteria) with four chemical substrates (*P*: phosphorus, *O*: oxygen, *SR*: reduced  
72 sulfur, *SO*: oxidized sulfur). It includes feedback between organisms and biogeo-  
73 chemical processes and is based on Bush et al. [2] (See Bush et al. [2] for a detailed  
74 discussion of the model). At the core of the simulations is a set of ordinary differ-  
75 ential equations (specified in the function `bushplus_dynamic_model()`, though  
76 this function need not be directly called). There are functions for running  
77 individual simulations and for running a set of simulations across, for example,  
78 a range of environmental conditions.

79 To make the simulation run with multiple species, we expressed different  
80 species characteristics in the elements of vectors and matrices. We also coded  
81 the ordinary differential equations to include the vectors and matrices, and to  
82 use matrix mathematics. In this way, we made it possible to run simulations  
83 with different numbers of species without having to change the underlying code.

84 The package functions and code have modular structure, so that new func-  
85 tionality can be easily added. E.g. temporally defined events of any type could  
86 be specified. Further, all parameter values required to run a simulation are  
87 stored in one object. Lastly, the general structure of the code should make it  
88 straightforward to adapt the model to other similar systems (described in more  
89 detail in the Impact section).

### 90 4.1. Software architecture

91 The framework used when writing this package aimed to maximise simplicity  
92 for the user, and to make it easy to reproduce results (see the supplement to  
93 Limberger [5] for an example of how this is used). As such, all the parameters  
94 needed to run a simulation or find a stable state (i.e. the final state of the  
95 ecosystem) are contained in a single object (which can easily be created using  
96 included functions). This parameter object is given to a function that runs the  
97 simulations and returns the results. The returned results object is identical  
98 to the parameter object but with an additional slot named `results`, which  
99 contains the results of the run. Thus the returned results object contains the  
100 simulation conditions (parameters) as well as the results, and can be used to  
101 run the simulation again. This promotes reproducibility and makes incremental  
102 changes of individual parameters and re-running the simulations straightforward.

103 In the following sections we describe how to use the package to run one  
104 simulation and to find steady states across an environmental gradient.

### 105 4.2. Running one simulation

106 A typical simulation would look as shown in 1.



Figure 1: Typical flow of a simulation. Dark Grey boxes: commands necessary for simulation; Light Grey: Saving of parameter and results; Lightest Grey: Different non specified commands.

107 A simulation is run using the `run_simulation()` function. In this function,  
 108 the ODEs are solved using the function `ode()` in the package *deSolve* package [11].  
 109 The `run_simulation()` function needs only one argument - an object as created  
 110 by the function `new_runsim_parameter()`. The parameter object returned by  
 111 `new_runsim_parameter()` contains among other things the `strain_parameter`  
 112 object, which can be created by the function `new_strain_parameter()`. For  
 113 a detailed description of the parameter objects, their meaning and how they  
 114 are created and have values set and changed please see the *User Guide* which  
 115 accompanies the package or is available at [User Guide](#)

116 After the parameter object has been defined, it can be used in the `run_simulation()`  
 117 function. The function returns an object which is identical to the parameter  
 118 object, except of an additional slot containing the results. This design produces  
 119 a fully reproducible object as it can be used as a parameter object to be fed  
 120 back into the `run_simulation()` function to run the simulation again from the  
 121 parameter used to generate the results before.

#### 122 4.3. Finding Stable States

123 The general approach used to find the stable state of the system with a  
 124 specific parameter set is to run the simulation for a long time and record the  
 125 final state. When one does this across a range of environmental conditions, one  
 126 discovers how the steady state of the system responds to the environmental  
 127 conditions. The package contains functionality for finding steady states that  
 128 correspond to values of one environmental driver, namely the value of oxygen  
 129 diffusivity.

130 Two methods for finding steady states are implemented. The first runs a  
 131 separate simulation for each combination of starting conditions and oxygen  
 132 diffusivity (we term this the *Replication method*). This is the method used in  
 133 the Bush et al. [2] study. The second runs two simulations, with step-wise and  
 134 slowly temporally increasing oxygen diffusivity, and the other with step-wise  
 135 and slowly decreasing or decreasing oxygen diffusivity. During this temporal  
 136 environmental change, the state of the system was recorded just before change to  
 137 a new oxygen diffusivity (we term this the *Temporal method*). We implemented  
 138 two methods since there is no definitive best method, and in order to check if  
 139 results were sensitive to choice of method.

140 The replication method is implemented in the function `run_replication_ssfind()`  
 141 which takes a parameter object as returned by the function `new_replication_ssfind_parameter()`  
 142 and the number of cores for multithreading the simulation. As the multithreading  
 143 uses the package function `mclapply()` from the package `parallel` [7], the multithreading  
 144 only works on Linux and Mac. It is planned to move to `parLapply()`  
 145 [7] in a future release.

146 This temporal method implemented in the function `run_temporal_ssfind()`,  
 147 which takes a parameter object as created by the function `new_temporal_ssfind_parameter()`.  
 148 It is planned for a later release, to run these two simulations in parallel.

149 For a more detailed walk-through of these two approaches and explanation  
 150 please see the [User Guide](#).

#### 151 4.4. Analysing and visualising results

152 From the results returned, summary measures about how the ecosystem  
153 stable states respond to environmental change can be extracted. The function  
154 `get_stability_measures()` returns quantities such as the amount of environ-  
155 mental change required for the system to abruptly change to a different state.

156 The function `plot_dynamics()` plots a single simulation run, as returned  
157 from the `run_simulation()` function. This function is only provided as a  
158 convenience function to provide a way to easily see the results of a simulation  
159 run. An example plot resulting from this function is shown in 2.

### 160 5. Impact

161 The open source implementation and extension of the model used in Bush  
162 et al. [2] provides the means of reproducing the results published while at  
163 the same time provides the means of doing unique, innovative, and important  
164 investigations of how ecosystems respond to environmental change, and how  
165 biodiversity may modulate this response.

166 The design of the package code and functionality is with reproducibility in  
167 mind: the combination of all parameters being in a single parameter object  
168 as well as the return of the simulation as a result object which inherits from  
169 the parameter object provides a relatively easy to use framework to implement  
170 reproducible experiments.

171 Here we evidence the impact of the *microxanox* package by describing three  
172 use cases and then by describing how the package can be a starting point for  
173 models of other ecosystems. The first two use cases are described in detail  
174 (including the code for reproducing them) in the *User Guide* and the *Partial*  
175 *Reproduction* vignettes. The third is taken from REF NEEDED [8].

#### 176 5.1. Use case 1: Regime shifts during temporal environmental change

177 The study of Bush et al. [2] includes simulations of the effect of oxygen  
178 diffusivity (an environmental driver, in the sense that it affects the ecosystem but  
179 is not affected by it) on the ecosystem state (oxic or anoxic). The *microxanox*  
180 package contains functionality to make a specific temporal pattern of change  
181 in the oxygen diffusivity. As well as allowing individual simulations during  
182 which oxygen diffusivity varies, this functionality forms the basis of the temporal  
183 method for finding stable states.

184 An example of this functionality is given in the *Partial Reproduction* vignette,  
185 which we briefly show here (Figure 3. The example is composed of a single  
186 simulation, at the beginning of which the system is in the oxic state with high  
187 abundance of cyanobacteria. Oxygen diffusivity is then slowly decreased and  
188 eventually, around hour 30'000 the system switches to the anoxic state, with  
189 high abundance of both sulfur bacteria types. The oxygen diffusivity is then  
190 increased and at around hour 38'000 the system abruptly switches back to the  
191 oxic state.

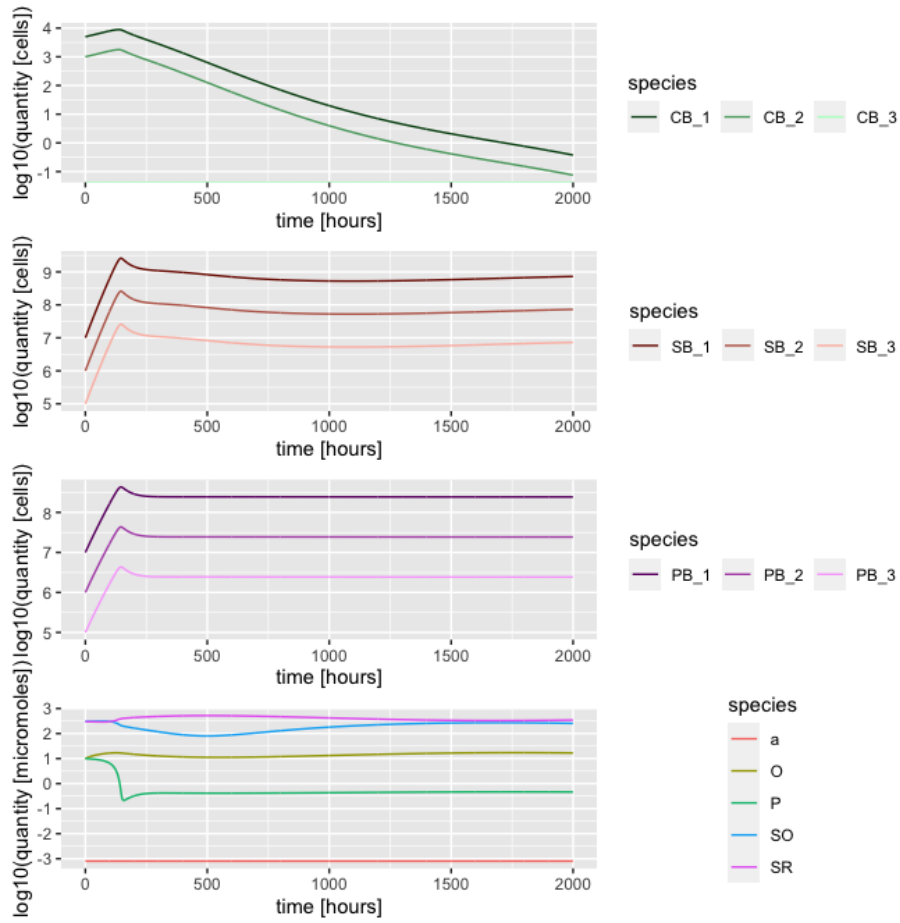


Figure 2: Results of a simulation run shown using the function `plot_dynamics()`. In this case, there were three strains per functional group, though strains within functional groups had identical properties in this example. CB extunderscore1 = cyanobacteria strain 1; SB extunderscore1 = sulfur reducing bacteria stain 1; PB = phototrophic sulfur bacteria strain 1. Here we show the figure produce by `plot_dynamics()`, which produces figures not with all features ideal for publication.



Also visible in the results are thick lines showing abundances of bacteria when abundances are low. This is due to the implementation of a function that at regular intervals, increases the abundance to a preset level. This prevents abundances reducing to very small numbers. The function that implements this increase abundance can also be made to add a certain abundance to each strain at regular intervals, thus simulating immigration in to the system.

```
#>
#> Attaching package: 'dplyr'
#> The following objects are masked from 'package:stats':
#>
#> filter, lag
#> The following objects are masked from 'package:base':
#>
#> intersect, setdiff, setequal, union
#> Warning in knitr::include_graphics(fig): It is highly recommended
#> to use relative paths for images. You had absolute paths: "/"
#> Users/rainer/git/0.RPackages/microxanox/inst/manuscript/figures/
#> gen_uc1_partrep_temporal_state_switching.pdf"
```

## 5.2. Use case 2: The extent of hysteresis depends on community composition

The package contains a function to extract summary features of ecosystem responses to environmental change, such as the amount of hysteresis displayed by the ecosystem. Hysteresis is a key feature of ecosystem responses to environmental change, because it is related to how difficult it can be to reverse the effects of environmental change [9]. The amount of hysteresis is measured as the extent of the environmental condition (here oxygen diffusivity) for which there were two stable states. I.e. it is the extent of the environmental conditions for which historical conditions play an important role in determining the current system state (a definition of hysteresis).

Using the package to calculate the extent of hysteresis involves setting ecosystem and simulation parameters, including parameters for the finding of stable states across an environmental gradient, running the stable state finding function, and analysing the results with the function that calculates extent of hysteresis. The code for this is provided in the *User Guide* vignette.

The results show that the amount of hysteresis depends greatly on the combinations of organisms present (Figure ??figL:uc2)). For example, with only the CB (cyanobacteria) present, there was no hysteresis. In contrast, the presence of both CB and SB (sulfate reducing bacteria) led to a large amount of hysteresis. (These results are also given in the *Partial Reproduction* vignette.)

```
#> Warning in knitr::include_graphics(fig): It is highly recommended
#> to use relative paths for images. You had absolute paths: "/"
#> Users/rainer/git/0.RPackages/microxanox/inst/manuscript/figures/
#> gen_uc2_user_guide_hysteresis.pdf"
```



Figure 3: The temporal dynamics of the ecosystem model when an environmental condition (here parameter  $*a^*$ , the oxygen diffusivity) changes. Plot of the stable states of the simulation runs under different oxygen diffusivity. In this simulation there is only one strain in each functional group. CB extunderscore1 = cyanobacteria strain 1; SB extunderscore1 = sulfur reducing bacteria stain 1; PB = phototrophic sulfur bacteria strain 1. Here we show a figure adapted from the output of the `plot_dynamics()` function.

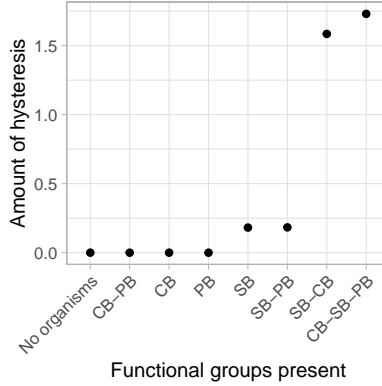


Figure 4: The amount of hysteresis depends on the combination of types of organisms present. The model is entirely deterministic, hence there are no error bars.

### 5.3. Use case 3: Effects of functional diversity on regime shifts

As discussed in the Introduction section, the package was motivated by the question of how biodiversity influences ecosystem responses to environmental change. Extensive results concerning this question are given in a separate publication Limberger [5]. Here we describe one of the results, which is that having biodiversity in a functional group can allow state changes to occur that otherwise would not have. I.e. biodiversity can qualitatively change the state of the ecosystem.

Biodiversity was added to the functional groups using the `new_strain_parameter()` function to create a parameter set with multiple species per functional group (albeit all with identical features) and then to add variability among the species by calling the `add_strain_var()` function. This function takes an already existing parameter set and adds the specified amount of variation. The new parameter object is then used as before.

Figure 5 shows a simulation with two species (strains) in each of the three functional groups. The ecosystem starts in the oxic state, though with relatively high abundance of each functional group. The strain of SB that is more tolerant to oxygen (SB\_1) initially decreases in abundance, but then increases, and the other (SB\_2) strain then becomes abundant and SB\_1 declines. Furthermore, the cyanobacteria crash in abundance, and the system switches to the anoxic state. In contrast, if there are two identical strains with tolerance half way between those in Figure 5 the ecosystem remains in the oxic state.

### 5.4. Adapting for other ecosystems and organisms

We anticipate that the package can be a useful starting point for investigating models of other types of ecosystem and how biodiversity in them affects responses to environmental change. The overall framework of the package, the purpose of each function, and the objects used for storing parameters and results could be retained. E.g., all such models would have parameters that differ among

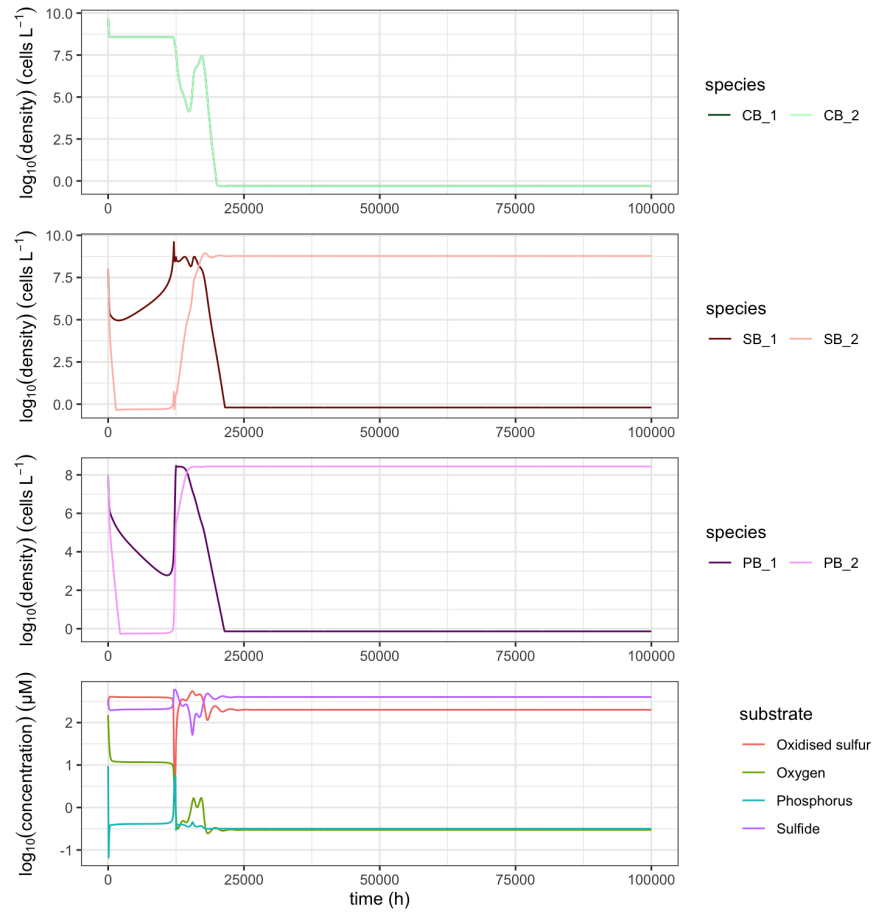


Figure 5: The dynamics of the ecosystem when there are two species in each functional group, and some variation (diversity) in species parameters.

262 species and need to be described in an object, and studies will often need to run  
263 simulations and sets of simulations across environmental conditions.

264 Researchers wanting to model a new ecosystem do not, therefore, have to  
265 start from scratch. This will relieve researchers from needing to making software  
266 design decisions, and rather focus on appropriately representing their ecosystem,  
267 and finding the results that interest them. Nevertheless, adaptation of the code  
268 in the package will require a person / persons that can take a conceptual model of  
269 an ecosystem and then represent that in terms of parameters and rate equations,  
270 and that is relatively proficient in R programming.

## 271 6. Conclusions

272 The *microxanoz* R package allows the simulation, visualisation, and analysis  
273 of a model of a microbial ecosystem while allowing variation in the amount of  
274 diversity containing in each of the functional groups of organisms present. It has  
275 been used for the research described in another paper that describes one of the  
276 first investigations of the effects of diversity on ecosystem resilience Limberger  
277 [5]. In that paper, we show that diversity can have large and important effects  
278 of ecosystem responses, highlighting the need for models such as ours, with  
279 which one can easily manipulate the amount of biodiversity. The *microxanoz*  
280 package has also been used to reproduce the results of the paper that inspired  
281 the package development Bush et al. [2].

282 The package greatly lowers the amount of work required in further investi-  
283 gations of the specific ecosystem modelled. There has, for example, been quite  
284 limited investigation of how biodiversity influences the short-term responses of  
285 the modelled ecosystem to environmental change. Likewise, the package could be  
286 used to power an investigation of the effects of biodiversity on the usefulness of  
287 early warning signals of abrupt ecosystem change [10]. In addition this package  
288 could be used as a template for the implementation for developing models of  
289 other types of ecosystems and organism. By doing so, other models can profit  
290 from the overall framework used, and the reproducibility aspects as well as the  
291 flexibility implemented.

## 292 7. Conflict of Interest

293 The authors declare no known conflicting or competing interests associated  
294 with this publication and there has been no significant financial support for this  
295 work that could have influenced its outcome.

## 296 8. Acknowledgements

297 This project was part of SNF Project 310030\_188431. The project was also  
298 supported by the University of Zurich Research Priority Programme in Global  
299 Change and Biodiversity.

## References

- [1] Amrei Binzer, Christian Guill, Björn C. Rall, and Ulrich Brose. Interactive effects of warming, eutrophication and size structure: Impacts on biodiversity and food-web structure. *Global Change Biology*, 22(1):220–227, January 2016. ISSN 13541013. doi: 10.1111/gcb.13086.
- [2] Timothy Bush, Muhe Diao, Rosalind J. Allen, Ruben Sinnige, Gerard Muyzer, and Jef Huisman. Oxidic-anoxic regime shifts mediated by feedbacks between biogeochemical processes and microbial community dynamics. *Nature Communications*, 8(1):789, October 2017. ISSN 2041-1723. doi: 10.1038/s41467-017-00912-x.
- [3] P. Catalina Chaparro-Pedraza. Fast environmental change and eco-evolutionary feedbacks can drive regime shifts in ecosystems before tipping points are crossed. *Proceedings of the Royal Society B: Biological Sciences*, 288(1955):20211192, July 2021. ISSN 0962-8452, 1471-2954. doi: 10.1098/rspb.2021.1192.
- [4] Scott L. Collins, Jesse B. Nippert, John M. Blair, John M. Briggs, Pamela Blackmore, and Zak Ratajczak. Fire frequency, state change and hysteresis in tallgrass prairie. *Ecology Letters*, 24(4):636–647, April 2021. ISSN 1461-023X, 1461-0248. doi: 10.1111/ele.13676.
- [5] Romana Limberger. #####TO BE UPDATED### - The role of functional diversity in ecosystem regime shifts. 2022.
- [6] Amanda C. Northrop, Vanessa Avalone, Aaron M. Ellison, Bryan A. Ballif, and Nicholas J. Gotelli. Clockwise and counterclockwise hysteresis characterize state changes in the same aquatic ecosystem. *Ecology Letters*, 24(1): 94–101, January 2021. ISSN 1461-023X, 1461-0248. doi: 10.1111/ele.13625.
- [7] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2022.
- [8] Author REF NEEDED. REFERNECE NEEDED. *Journal of missing references*, 2222.
- [9] Marten Scheffer, Steve Carpenter, Jonathan A. Foley, Carl Folke, and Brian Walker. Catastrophic shifts in ecosystems. *Nature*, 413(6856):591–596, October 2001. ISSN 0028-0836, 1476-4687. doi: 10.1038/35098000.
- [10] Marten Scheffer, Jordi Bascompte, William A. Brock, Victor Brovkin, Stephen R. Carpenter, Vasilis Dakos, Hermann Held, Egbert H. van Nes, Max Rietkerk, and George Sugihara. Early-warning signals for critical transitions. *Nature*, 461(7260):53–59, September 2009. ISSN 0028-0836, 1476-4687. doi: 10.1038/nature08227.

- 337 [11] Karline Soetaert, Thomas Petzoldt, and R. Woodrow Setzer. Solving  
338 differential equations in R: Package deSolve. *Journal of Statistical Software*,  
339 33(9):1–25, 2010. doi: 10.18637/jss.v033.i09.
- 340 [12] John Vandermeer and Ivette Perfecto. Hysteresis and critical transitions in  
341 a coffee agroecosystem. *Proceedings of the National Academy of Sciences*,  
342 116(30):15074–15079, July 2019. ISSN 0027-8424, 1091-6490. doi: 10.1073/  
343 pnas.1902773116.
- 344 [13] David A. Vasseur and Kevin S. McCann. A Mechanistic Approach for  
345 Modeling Temperature-Dependent Consumer-Resource Dynamics. *The*  
346 *American Naturalist*, 166(2):184–198, August 2005. ISSN 0003-0147, 1537-  
347 5323. doi: 10.1086/431285.