

1     MicroxanoX: an R package for simulating an aquatic  
2         *MICRO*bial ecosystem that can occupy *OXic* or  
3         *ANOXic* 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**

*MicroxanoX* is an R package to simulate a three functional group ecosystem (cyanobacteria, phototrophic sulfur bacteria, and sulfate-reducing bacteria) with four chemical substrates (phosphorus, oxygen, reduced sulfur, and oxidized sulfur) using a set of ordinary differential equations. Simulations can be run individually or over a parameter range, ~~to find stable states~~. The model can be implemented with different numbers of species per functional group. The package is constructed in such a way that the results contain the input parameter used, so that a saved results can be loaded again and the simulation be repeated. Furthermore, the package framework and code should serve as a useful starting point for making simulation models of other types of ~~ecosystem~~ecosystems.

6     *Keywords:* reproducibility, regime shift, ~~stable~~final state, ordinary differential  
7     equations

---

---

\*Corresponding author

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

<sup>1</sup>Corresponding Author

<sup>2</sup>Equal contribution

8 **1. Required Metadata**

9 **1.1. Current code version**

10 Ancillary data table required for subversion of the codebase.

Nr.	Code metadata description	Please fill in this column
C1	Current code version	v0.9- <del>0</del> -1
C2	Permanent <del>link</del> -links 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	<a href="https://zenodo.org/record/6624125/files/10.5281/zenodo.6624125.7148667">10.5281/zenodo.6624125.7148667</a>
C4	Legal Code License	MIT
C5	Code versioning system used	<del>git</del> -git
C6	Software code languages, tools, and services used	R
C7	Compilation requirements, operating environments	R ( $\geq 4.1.0$ )
C8	If available Link to developer documentation/manual	magrittr tibble ggplot2 patchwork grDevices stats mgcv deSolve dplyr tidyr stringr multidplyr <a href="#">User Guide</a>
C9	Support email for questions	<del>Rainer.Krug@uzh.ch;</del> <del>Owen.Petchey@ieu.uzh</del> <a href="mailto:Rainer.Krug@uzh.ch">Rainer.Krug@uzh.ch</a> ; <a href="mailto:Owen.Petchey@ieu.uzh">Owen.Petchey@ieu.uzh</a>

Nr.	Code metadata description	Please fill in this column
-----	------------------------------	----------------------------

## 11 2. Motivation and significance

12 Mathematical models play a key role in the development of ~~understanding~~  
13 ~~about how ecosystems work and how they respond to environmental ecosystem~~  
14 ~~models and in increasing the understanding of how and why ecosystems change~~  
15 ~~when the environment~~ changes [? ? ? ]. They are also important for developing  
16 hypotheses to test in empirical studies. One area of ecology in which models  
17 have ~~played a influential role is how ecosystems respond~~ been influential is the  
18 understanding of ecosystem responses to gradual change in an environmental  
19 driver [? ]. An environmental driver is ~~a~~-an environmental condition that  
20 affects an ecosystem, but ~~is assumed to not be~~ which is not itself affected by the  
21 ecosystem, ~~such as the rate of nutrient input into a lake~~. An example driver  
22 would be the amount of nutrients entering a lake from run-off from surrounding  
23 agricultural land.

24 It is conceivable that an ecosystem state, such as the total biomass of a  
25 particular type of bacteria, may remain unchanged when an environmental driver  
26 changes. It is also possible that the ecosystem state changes gradually. It is  
27 also possible that the ecosystem state changes abruptly to a new state that is  
28 difficult to recover from [? ]. This possibility for abrupt, perhaps catastrophic  
29 changes ~~that~~, which are difficult to reverse, causes considerable concern [? ? ? ].

30 An example where a gradual change of an environmental variable causes  
31 an abrupt change ~~of~~-in the system is the switch from an aerobic (oxygen is  
32 available for metabolism) to an anaerobic (oxygen generally unavailable) state in  
33 a microbial ecosystem. This system has been investigated by ? ] in a simulation  
34 study of a mathematical model. Three types of microbes occur in the model:  
35 cyanobacteria (CB) dominating the oxic state, and two types of sulfur bacteria  
36 that dominate the anoxic state (sulfate reducing bacteria (SB) and phototrophic  
37 sulfur bacteria (PB)). The model shows that gradual change in the rate at which  
38 oxygen ~~could~~-can diffuse into the ecosystem (termed ~~the~~-oxygen diffusivity)  
39 could cause catastrophic changes in the ecosystem state that would be difficult  
40 to reverse.

41 One feature of the study by ? ] was limited biodiversity. Specifically, there  
42 was no biodiversity within each of the three types (i.e., functional groups) of  
43 bacteria. This leaves ~~open~~-the question of if and how biodiversity within these  
44 ~~types (i.e. functional groups ) of bacteria affects~~ functional groups might affect

45 the ecosystem response to environmental change. This limitation is not specific  
46 to the study of [? ]. There are few if any studies of the effects of biodiversity on  
47 abrupt transitions between ecosystem states.

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

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

### 68 3. 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 bio-  
73 geochemical processes and is based on [? ] (See [? ] for a detailed discussion  
74 of the model). At the core of the simulations is a set of ordinary differential  
75 equations (specified in the function `bushplus_dynamic_model()`, though this  
76 function need not be directly called). There are functions for running individual  
77 simulations and ~~for running~~ a set of simulations across, for example, a range of  
78 environmental conditions.

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

85 The package functions and code have a modular structure so that new  
86 functionality can be easily added. E.g., temporally defined events of any type  
87 could be specified. Further, all parameter values required to run a simulation  
88 are stored in one object. Lastly, the general structure of the code should make it

straightforward to adapt the model to other similar systems (described in more detail in the Impact section).

### 3.1. Software architecture

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

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

### 3.2. Running one simulation

A typical simulation would look as shown in Figure [?].

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

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

### 3.3. Examining ecosystem responses to environmental change

~~The general approach used to find the stable state of the system with a specific parameter set~~ One approach to finding how the ecosystem responds to environmental driver change is to examine the relationship between the environmental driver value and the numerical value of a system's state. The first component of this approach is to run the simulation for a long time and record the final state (i.e., the state of the system at the end of a simulation).

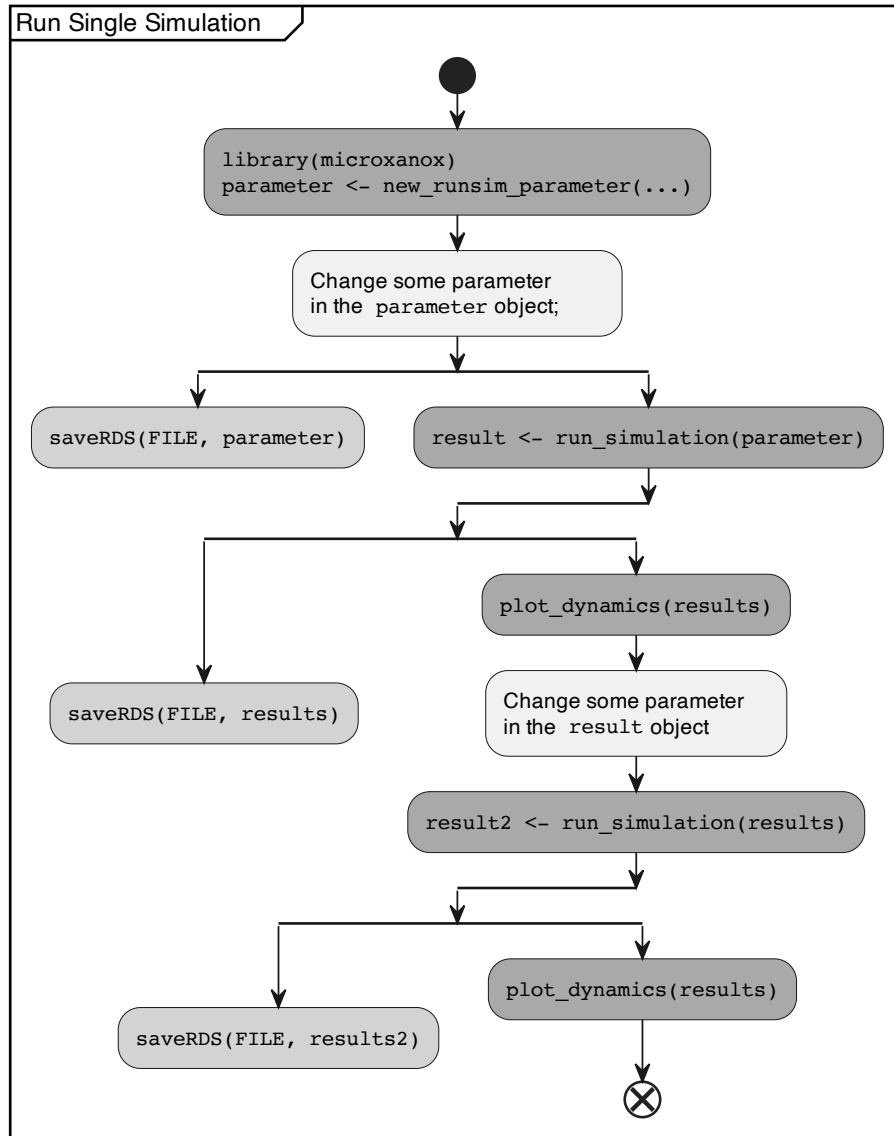


Figure 1: Typical flow of a simulation. Dark Grey boxes: commands necessary for simulation; Light Grey: Saving of parameter and results; Lightest Grey: Different non specified commands.

131 When one does this across a range of environmental conditions, one can discover  
132 how the ~~steady-final~~ state of the system ~~responds to changes with~~ the environ-  
133 mental conditions. The package contains functionality for finding ~~steady-final~~  
134 states that correspond to values of one environmental driver, namely the value of  
135 oxygen diffusivity. In terms of non-linear systems analysis, this would be termed  
136 a parameteric sensitivity analysis being conducted by running an open loop of  
137 the dynamic system under a set of initial conditions and a subset of parameter  
138 values (here oxygen diffusivity).

139 ~~Two methods for finding~~ When one wishes to be able to make conclusions  
140 about how the *steady state* (or an equilibrium point) of the system is affected  
141 by the environmental driver, it is very important to note that the *final state*  
142 (provided by the simulation) is not guaranteed to be a *steady state*. The  
143 software does not provide the user with a steady state. In order to somewhat  
144 safely assume that the final state is a steady state, the user must ensure that  
145 the simulation is run for sufficiently long time for any transient dynamics to  
146 disappeared, and must also check the type of long-term dynamics occurring. In  
147 the results presented here, and in the paper ? ], this was performed by visual  
148 inspection, and by checking the sensitivity of conclusions to the length of the  
149 simulation. Furthermore, the package does not include methods for a formal  
150 analysis of the stability of the system and users should take care to assess if  
151 steady states are ~~implemented. The first runs a separate unique.~~

152 Two numerical strategies for finding final states and their sensitivity to  
153 parameters are implemented. Two strategies are implemented in order to allow  
154 users to compare of their results. The first method runs a independent simula-  
155 tion for each combination of starting-initial conditions and oxygen diffusivity  
156 (we term this the *Replication method*). This is the method used in the ? ] study :  
157 ~~The second and was used to obtain the results in figures 3 and 4 of that article.~~

158  
159 The second method runs two simulations, one with step-wise and slowly  
160 temporally *increasing* oxygen diffusivity, and the other with step-wise and  
161 slowly *decreasing* oxygen diffusivity. (Put another way, to explore the sensitivity  
162 of final states of the dynamic model under oxygen diffusivity variation, subset  
163 values of this parameter are numerically evaluated. Two value vectors of oxygen  
164 diffusivity are presented step-wise, gradually increasing and decreasing.) During  
165 this temporal environmental change, the state of the system is recorded just  
166 before change to a new oxygen diffusivity (we term this the *Temporal method*).  
167 ~~We implemented two methods since there is no definitive best method, and in~~  
168 ~~order to check if results were sensitive to choice of method.~~

169 An potentially important difference between the two methods is in the  
170 system state when a new value of oxygen diffusivity is set. In the replication  
171 method, the system state when a new value of oxygen diffusivity is set is always  
172 the same. Whereas in the temporal method, the system state when a new value  
173 of oxygen diffusivity is set is the final state of the system for the previously  
174 set value of oxygen diffusivity. Since some modellers prefer one approach and  
175 others another, we decided to implement both.

176 The replication method is implemented in the function `run_replication_ssfind()`

177 which takes a parameter object as returned by the function `new_replication_ssfind_parameter()`  
 178 and the number of cores for multithreading the simulation. As the multithread-  
 179 ing uses the package function `mclapply()` from the package `parallel` [? ],  
 180 the multithreading only works on Linux and Mac. It is planned to move to  
 181 `parLapply()` [? ] in a future release.

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

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

### 187 3.4. [Analysing and visualising results](#)

188 From the results returned, summary measures about how the ecosystem  
 189 ~~stable states respond to final states changes with~~ environmental change can be  
 190 extracted. The function `get_stability_measures()` returns quantities such as  
 191 the amount of environmental change required for the system to abruptly change  
 192 to a different state.

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

## 197 4. [Impact](#)

198 The open source implementation and extension of the model used in [? ]  
 199 provides the means of reproducing the results published while at the same  
 200 time provides the means of ~~doing making~~ unique, innovative, and important  
 201 investigations of how ecosystems ~~respond to environmental change change as~~  
 202 ~~the environmental changes~~, and how biodiversity may modulate this ~~response~~.

203 The design of the package code and functionality is with reproducibility in  
 204 mind (~~for other example of reproducibility directed software, see e.g. [? ], [? ]~~  
 205 ~~or [? ]~~); the combination of all parameters being in a single parameter object ~~as~~  
 206 ~~well as and~~ the return of the simulation as a result object which inherits from  
 207 the parameter object, provides a relatively ~~easy to use~~ ~~easy to use~~ framework to  
 208 implement reproducible experiments.

209 Here we evidence the impact of the *microxanox* package by describing three  
 210 ~~use cases use cases~~ and then by describing how the package can be a starting  
 211 point for models of other ecosystems. The first two use cases are described in  
 212 detail (including the code for reproducing them) in the ~~User Guide~~ [User Guide](#)  
 213 and the *Partial Reproduction* vignettes. The third is taken from ~~[? ]~~ and [? ].

### 214 4.1. [Use case 1: Regime shifts during temporal environmental change](#)

215 The study of [? ] includes simulations of the effect of oxygen diffusivity (an  
 216 environmental driver, in the sense that it affects the ecosystem but is not affected  
 217 by it) on the ecosystem state (oxic or anoxic). The ~~microxanox~~ [microxonox](#)



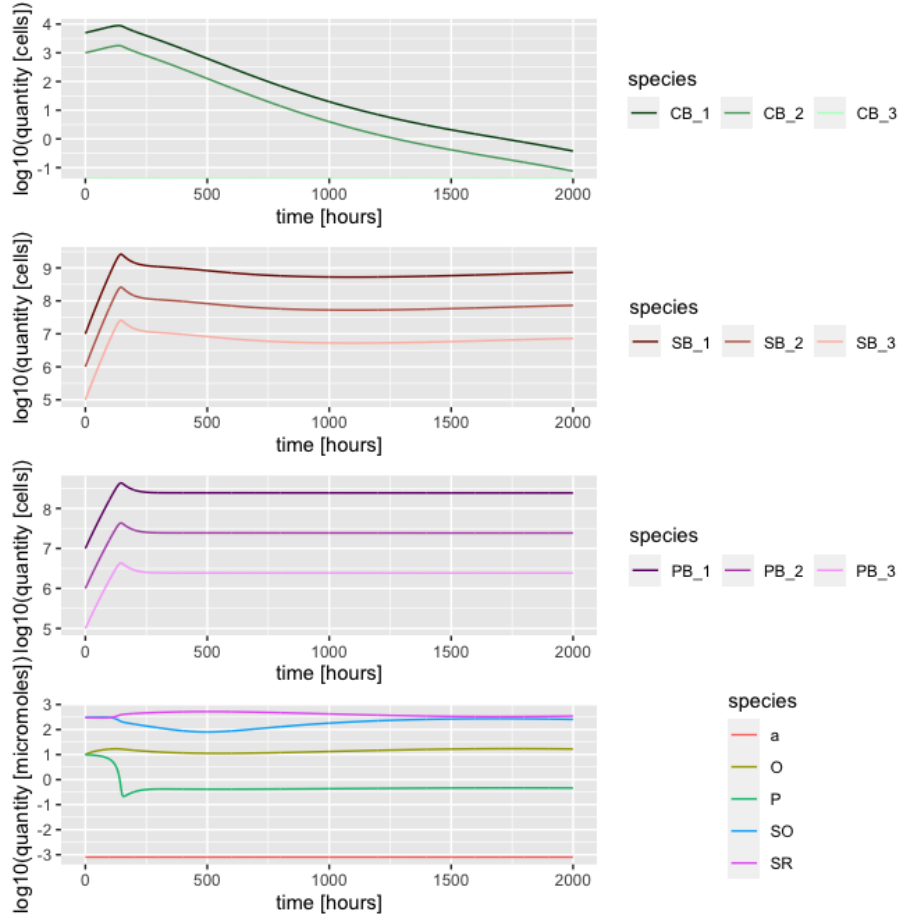


Figure 2: Results of a simulation run shown using the function `plot_dynamics()`. In this case, there were three strains per functional group, though strains within functional groups had identical properties in this example. CB\_1 = cyanobacteria strain 1; SB\_1 = sulfur reducing bacteria stain 1; PB\_1 = phototrophic sulfur bacteria strain 1.

218 package contains functionality to make a specific temporal pattern of change  
219 in the oxygen diffusivity. As well as allowing individual simulations during  
220 which oxygen diffusivity varies, this functionality forms the basis of the temporal  
221 method for finding ~~stable-final~~ states.

222 An example of this functionality is given in the *Partial Reproduction* vignette,  
223 which we briefly ~~describe and~~ show here (Figure 3). The example is composed of  
224 a single simulation, at the beginning of which the system is in the oxic state with  
225 high abundance of cyanobacteria. Oxygen diffusivity is then slowly decreased  
226 and eventually, around hour 30'000 the system switches to the anoxic state, with  
227 high abundance of both sulfur bacteria types. The oxygen diffusivity is then  
228 increased and at around hour 38'000 the system abruptly switches back to the  
229 oxic state.

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

#### 236 4.2. Use case 2: The extent of hysteresis depends on community composition

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

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

251 The results show that the amount of hysteresis depends greatly on the  
252 combinations of organisms present (Figure 4). For example, with only the CB  
253 (cyanobacteria) present, there was no hysteresis. In contrast, the presence of  
254 both CB and SB (sulfate reducing bacteria) led to a large amount of hysteresis.  
255 (These results are also given in the *Partial Reproduction* vignette.)

#### 256 4.3. Use case 3: Effects of functional diversity on regime shifts

257 As discussed in the Introduction section, the package was motivated by the  
258 question of how biodiversity influences ecosystem responses to environmental  
259 change. Extensive results concerning this question are given in a separate  
260 publication ~~?-?~~. Here we describe one of the results, which is that having

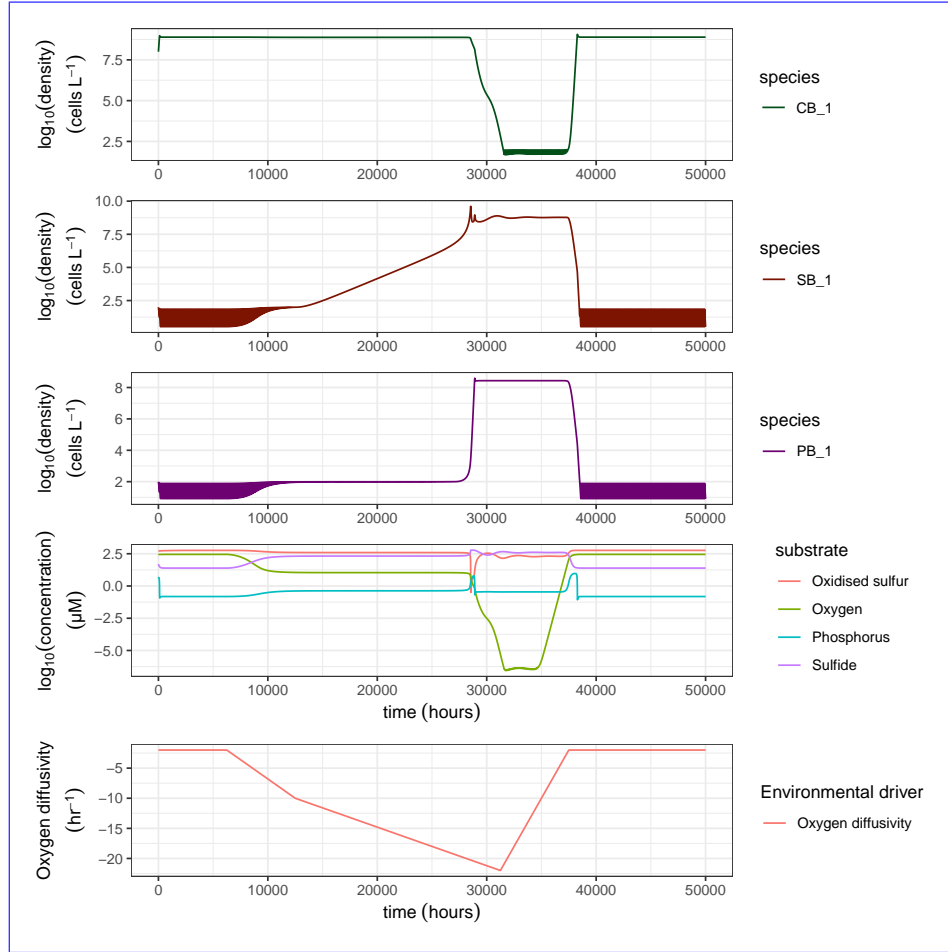


Figure 3: The temporal dynamics of the ecosystem model when an environmental condition (here parameter  $a$ , the oxygen diffusivity) changes. Plot of the ~~stable~~-final states of the simulation runs under different oxygen diffusivity. In this simulation there is only one strain in each functional 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.

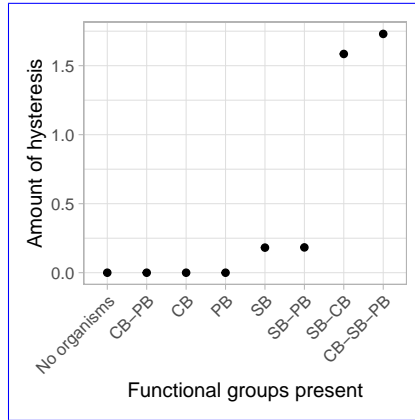


Figure 4: The amount of hysteresis depends on the combination of types of organisms present. The model is entirely deterministic, hence there are no error bars.

biodiversity in a functional group can allow state changes to occur that otherwise would not have. ~~I.e.~~ 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 abundance 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.

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

Researchers wanting to model a new ecosystem do not, therefore, have to start from scratch. This will relieve researchers from needing to make software

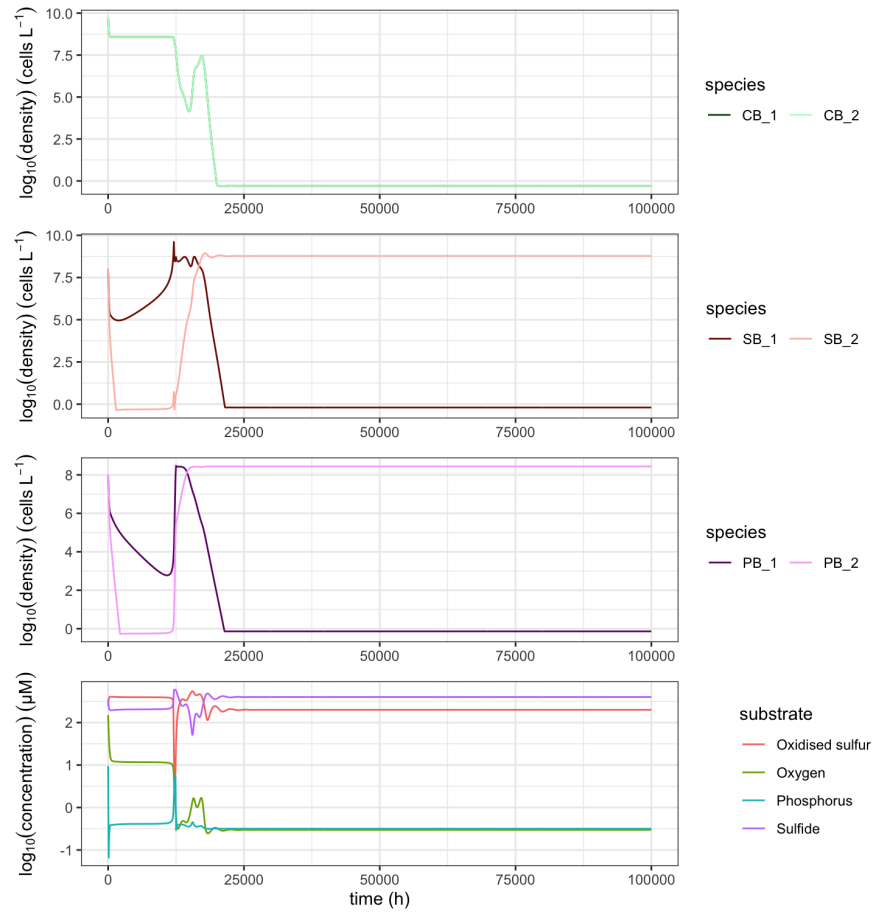


Figure 5: The dynamics of the ecosystem when there are two species in each functional group, and some variation (diversity) in species parameters.

288 design decisions, and rather focus on appropriately representing their ecosystem  
289 and finding the results that interest them. Nevertheless, adaptation of the code  
290 in the package will require a person / persons that can take a conceptual model of  
291 an ecosystem and then represent that in terms of parameters and rate equations,  
292 and ~~that is~~ a person or persons relatively proficient in R programming.

## 293 5. Conclusions

294 The *microxanoz* R package allows the simulation, visualisation, and analysis  
295 of a model of a microbial ecosystem while allowing variation in the amount  
296 of diversity ~~containing~~ present in each of the functional groups of organisms  
297 present. It has been used for the research described in another paper that  
298 ~~describes~~ provides one of the first investigations of the effects of diversity on  
299 ecosystem resilience ~~?~~ [? ]. In that paper, we show that diversity can have large  
300 and important effects ~~of~~ on ecosystem responses, highlighting the need for models  
301 such as ours, with which one can easily manipulate the amount of biodiversity.  
302 The *microxanoz* package has also been used to reproduce the results of the paper  
303 that inspired the package development [? ].

304 The package greatly lowers the amount of work required in further investi-  
305 gations of the specific ecosystem modelled. There has, for example, been quite  
306 limited investigation of how biodiversity influences the short-term responses of  
307 the modelled ecosystem to environmental change. Likewise, the package could be  
308 used to power an investigation of the effects of biodiversity on the usefulness of  
309 early warning signals of abrupt ecosystem change [? ]. In addition this package  
310 could be used as a template for ~~the implementation for~~ developing models of  
311 other types of ecosystems and organism. By doing so, other models can profit  
312 from the overall framework used, and the reproducibility aspects as well as the  
313 flexibility implemented.

## 314 6. Conflict of Interest

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

## 318 7. Acknowledgements

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

## 322 References

- 323 [ ] Amrei Binzer, Christian Guill, Björn C. Rall, and Ulrich Brose. Interactive  
324 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.
- Timothy Bush, Muhe Diao, Rosalind J. Allen, Ruben Sinnige, Gerard Muiyzer, and Jef Huisman. Oxic-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.
- 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.
- 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.
- Adrian A. Correndo, Austin Pearce, Carl H. Bolster, John T. Spargo, Deanna Osmond, and Ignacio A. Ciampitti. The soiltestcorr R package: An accessible framework for reproducible correlation analysis of crop yield and soil test data. *SoftwareX*, 21:101275, February 2023. ISSN 23527110. doi: 10.1016/j.softx.2022.101275.
- Nathan Fox, Tom August, Francesca Mancini, Katherine E. Parks, Felix Eigenbrod, James M. Bullock, Louis Sutter, and Laura J. Graham. “photosearcher” package in R: An accessible and reproducible method for harvesting large datasets from Flickr. *SoftwareX*, 12:100624, July 2020. ISSN 23527110. doi: 10.1016/j.softx.2020.100624.
- Rainer M Krug and Owen L Petchey. Microxanox: Model to simulate three functional group system. June 2022. doi: 10.5281/ZENODO.6624125.
- Romana Limberger, Uriah Daugaard, Anubhav Gupta, Rainer M. Krug, Kimberley D. Lemmen, Sofia J. Van Moorsel, Marcel Suleiman, Debra Zuppinger-Dingley, and Owen L. Petchey. Functional diversity can facilitate the collapse of an undesirable ecosystem state. *Ecology Letters*, 26(6): 883–895, June 2023. ISSN 1461-023X, 1461-0248. doi: 10.1111/ele.14217.
- 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.
- Owen L Petchey, Uriah Daugaard, Anubhav Gupta, Rainer M Krug, Kimberley D. Lemmen, van Moorsel, Sofia J., Marcel Suleiman, Zuppinger-Dingley, Debra, and Romana Limberger. Code package for: "Functional diversity

- 365 can facilitate the collapse of an undesirable ecosystem state". Zenodo, June  
366 2022.
- 367 [ ] R Core Team. *R: A Language and Environment for Statistical Computing*.  
368 R Foundation for Statistical Computing, Vienna, Austria, 2022.
- 369 [ ] Marten Scheffer, Steve Carpenter, Jonathan A. Foley, Carl Folke, and Brian  
370 Walker. Catastrophic shifts in ecosystems. *Nature*, 413(6856):591–596,  
371 October 2001. ISSN 0028-0836, 1476-4687. doi: 10.1038/35098000.
- 372 [ ] Marten Scheffer, Jordi Bascompte, William A. Brock, Victor Brovkin,  
373 Stephen R. Carpenter, Vasilis Dakos, Hermann Held, Egbert H. van Nes,  
374 Max Rietkerk, and George Sugihara. Early-warning signals for critical  
375 transitions. *Nature*, 461(7260):53–59, September 2009. ISSN 0028-0836,  
376 1476-4687. doi: 10.1038/nature08227.
- 377 [ ] Karline Soetaert, Thomas Petzoldt, and R. Woodrow Setzer. Solving  
378 differential equations in R: Package deSolve. *Journal of Statistical Software*,  
379 33(9):1–25, 2010. doi: 10.18637/jss.v033.i09.
- 380 [ ] Josef Spillner, Panagiotis Gkikopoulos, Pamela Delgado, and Christine  
381 Choirat. Towards reproducible software studies with MAO and Renku.  
382 *SoftwareX*, 17:100947, January 2022. ISSN 23527110. doi: 10.1016/j.softx.  
383 2021.100947.
- 384 [ ] John Vandermeer and Ivette Perfecto. Hysteresis and critical transitions in  
385 a coffee agroecosystem. *Proceedings of the National Academy of Sciences*,  
386 116(30):15074–15079, July 2019. ISSN 0027-8424, 1091-6490. doi: 10.1073/  
387 pnas.1902773116.
- 388 [ ] David A. Vasseur and Kevin S. McCann. A Mechanistic Approach for  
389 Modeling Temperature-Dependent Consumer-Resource Dynamics. *The*  
390 *American Naturalist*, 166(2):184–198, August 2005. ISSN 0003-0147, 1537-  
391 5323. doi: 10.1086/431285.