

1 Microxanox: an R package for simulating an aquatic
2 *MICRobial* ecosystem that can occupy *OXic* or
3 *ANOXic* states.

4 Rainer M Krug^{a,1}, Owen L. Petchey^a

^a*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~~steady state, final state,
7 ordinary differential equations

*Corresponding author

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

¹Corresponding Author

²Equal 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- 0 -1
C2	Permanent link -links to code/repository used for this code version	https://github.com/UZH- PEG/microxanox 10.5281/zenodo-6624125.7148667
C3	Code Ocean compute capsule	
C4	Legal Code License	MIT
C5	Code versioning system used	git -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 User Guide
C9	Support email for questions	Rainer.Krug@uzh.ch; Owen.Petchey@ieu.uzh Rainer.Krug@uzh.ch ; Owen.Petchey@ieu.uzh

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 steady state (if it exists). 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

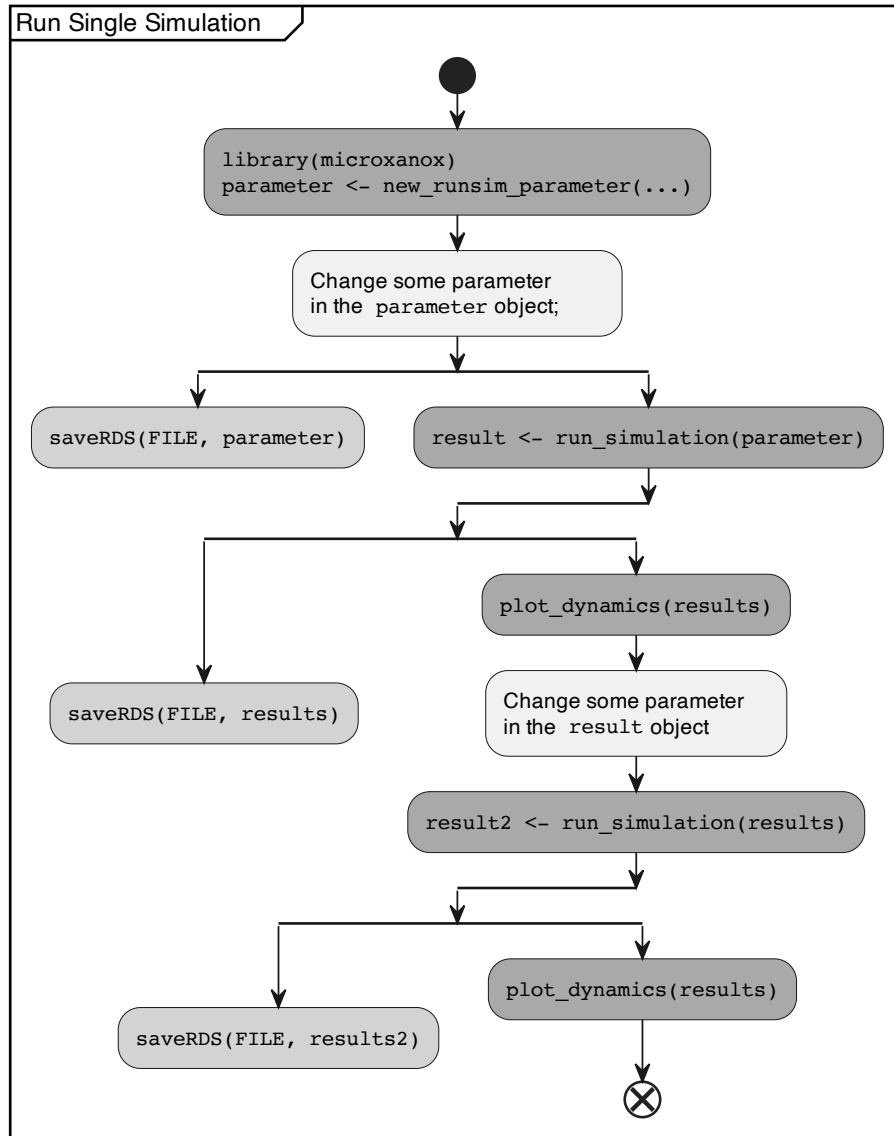


Figure 1: Typical flow of a simulation. Dark Grey boxes: commands necessary for simulation; Light Grey: Saving of parameter and results; Lightest Grey: Different non specified commands.

131 [simulation](#)). When one does this across a range of environmental conditions, one
132 discovers how the ~~steady-final~~ state of the system ~~responds to changes with~~ the
133 environmental conditions. The package contains functionality for finding ~~steady~~
134 ~~final~~ states that correspond to values of one environmental driver, namely the
135 value of oxygen diffusivity.

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

147 Two numerical strategies for finding final states and their sensitivity to
148 parameters are implemented. Two strategies are implemented in order to allow
149 comparison of their results. The first method runs a independent simulation for
150 each combination of ~~starting-initial~~ conditions and oxygen diffusivity (we term
151 this the *Replication method*). This is the method used in the [?] study. ~~The~~
152 ~~second and was used to obtain the results in figures 3 and 4 of that article.~~

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

163 The replication method is implemented in the function `run_replication_ssfind()`
164 which takes a parameter object as returned by the function `new_replication_ssfind_parameter()`
165 and the number of cores for multithreading the simulation. As the multithread-
166 ing uses the package function `mclapply()` from the package `parallel` [?],
167 the multithreading only works on Linux and Mac. It is planned to move to
168 `parLapply()` [?] in a future release.

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

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

174 3.4. Analysing and visualising results

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

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

184 4. Impact

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

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

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

201 4.1. Use case 1: Regime shifts during temporal environmental change

202 The study of ?] includes simulations of the effect of oxygen diffusivity (an
203 environmental driver, in the sense that it affects the ecosystem but is not affected
204 by it) on the ecosystem state (oxic or anoxic). The ~~microxanox~~ microxonox
205 package contains functionality to make a specific temporal pattern of change
206 in the oxygen diffusivity. As well as allowing individual simulations during
207 which oxygen diffusivity varies, this functionality forms the basis of the temporal
208 method for finding ~~stable final~~ states.

209 An example of this functionality is given in the *Partial Reproduction* vignette,
210 which we briefly describe and show here (Figure 3). The example is composed of
211 a single simulation, at the beginning of which the system is in the oxic state with
212 high abundance of cyanobacteria. Oxygen diffusivity is then slowly decreased
213 and eventually, around hour 30'000 the system switches to the anoxic state, with
214 high abundance of both sulfur bacteria types. The oxygen diffusivity is then

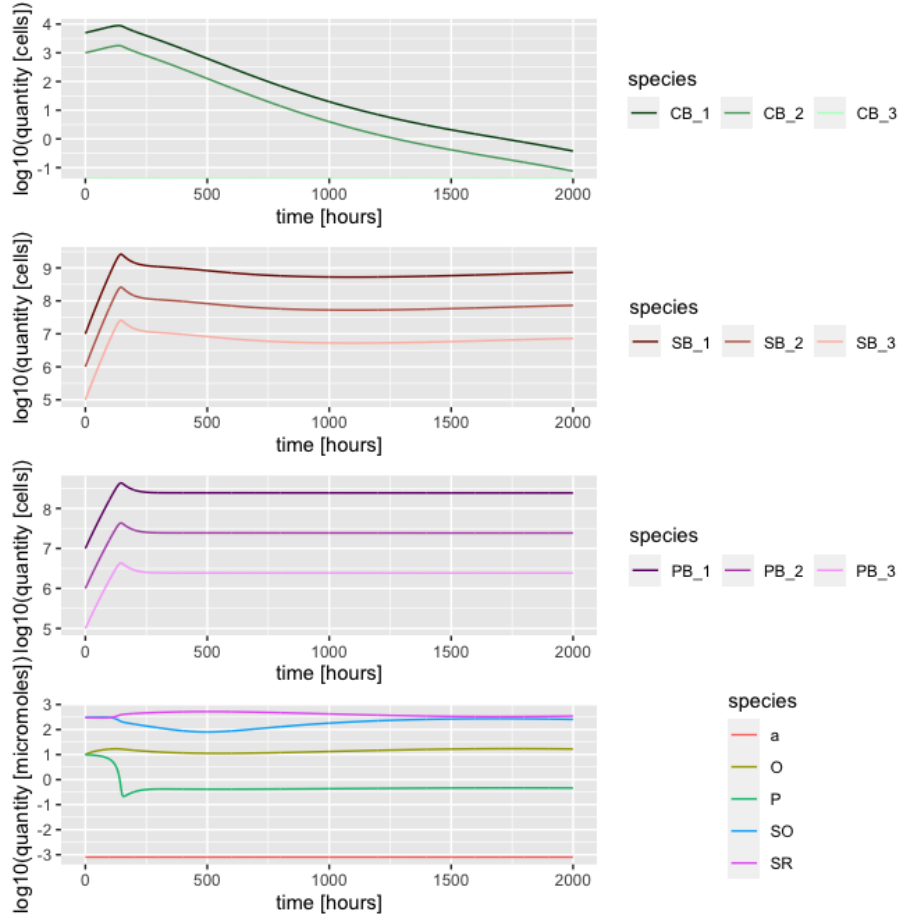


Figure 2: Results of a simulation run shown using the function `plot_dynamics()`. In this case, there were three strains per functional group, though strains within functional groups had identical properties in this example. CB_1 = cyanobacteria strain 1; SB_1 = sulfur reducing bacteria stain 1; PB_1 = phototrophic sulfur bacteria strain 1.

215 increased and at around hour 38'000 the system abruptly switches back to the
216 oxic state.

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

223 4.2. Use case 2: The extent of hysteresis depends on community composition

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

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

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

243 4.3. Use case 3: Effects of functional diversity on regime shifts

244 As discussed in the Introduction section, the package was motivated by the
245 question of how biodiversity influences ecosystem responses to environmental
246 change. Extensive results concerning this question are given in a separate
247 publication ~~?~~?. Here we describe one of the results, which is that having
248 biodiversity in a functional group can allow state changes to occur that otherwise
249 would not have. ~~I.e.,~~ i.e. biodiversity can qualitatively change the state of the
250 ecosystem.

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

257 Figure 5 shows a simulation with two species (strains) in each of the three
258 functional groups. The ecosystem starts in the oxic state, though with relatively

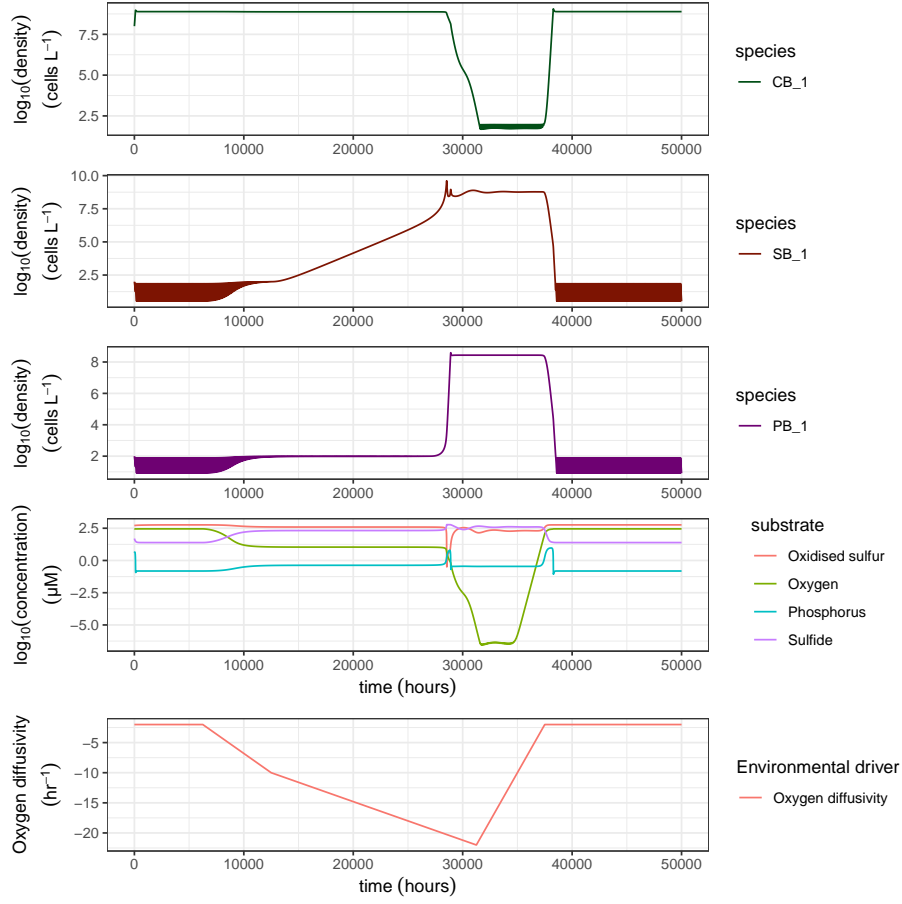


Figure 3: The temporal dynamics of the ecosystem model when an environmental condition (here parameter a , the oxygen diffusivity) changes. Plot of the ~~stable~~final states of the simulation runs under different oxygen diffusivity. In this simulation there is only one strain in each functional group. CB extunderscore1 = cyanobacteria strain 1; SB ~~1~~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.

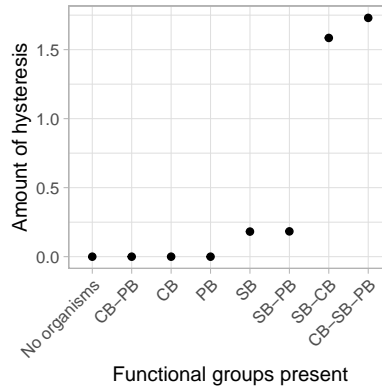


Figure 4: The amount of hysteresis depends on the combination of types of organisms present. The model is entirely deterministic, hence there are no error bars.

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

5. Conclusions

The *microxanox* R package allows the simulation, visualisation, and analysis of a model of a microbial ecosystem while allowing variation in the amount of diversity ~~containing~~ present in each of the functional groups of organisms present. It has been used for the research described in another paper that

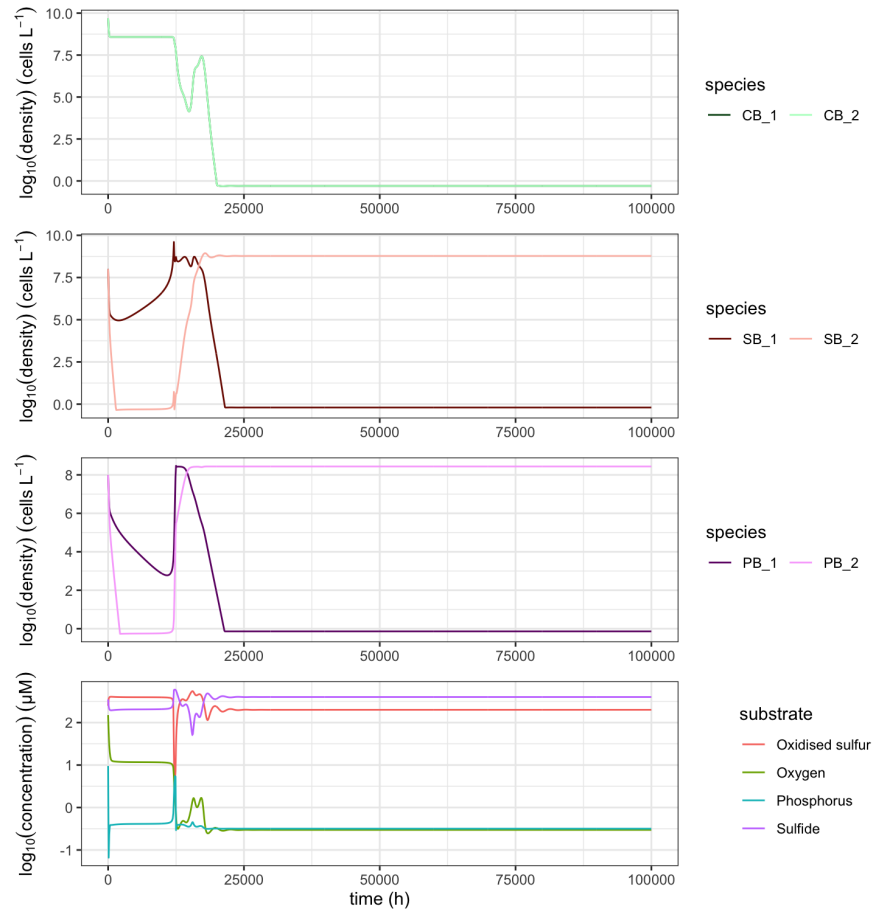


Figure 5: The dynamics of the ecosystem when there are two species in each functional group, and some variation (diversity) in species parameters.

describes provides one of the first investigations of the effects of diversity on ecosystem resilience [?]. In that paper, we show that diversity can have large and important effects of on ecosystem responses, highlighting the need for models such as ours, with which one can easily manipulate the amount of biodiversity. The *microxanox* package has also been used to reproduce the results of the paper that inspired the package development [?].

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

6. Conflict of Interest

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

7. Acknowledgements

This project was part of SNF Project 310030_188431. The project was also supported by the University of Zurich Research Priority Programme in Global 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. 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.
- [3] P. Catalina Chaparro-Pedraza. Fast environmental change and 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.

324 [] Scott L. Collins, Jesse B. Nippert, John M. Blair, John M. Briggs, Pamela
 325 Blackmore, and Zak Ratajczak. Fire frequency, state change and hysteresis
 326 in tallgrass prairie. *Ecology Letters*, 24(4):636–647, April 2021. ISSN
 327 1461-023X, 1461-0248. doi: 10.1111/ele.13676.

328 [] Adrian A. Correndo, Austin Pearce, Carl H. Bolster, John T. Spargo,
 329 Deanna Osmond, and Ignacio A. Ciampitti. The soiltestcorr R package:
 330 An accessible framework for reproducible correlation analysis of crop yield
 331 and soil test data. *SoftwareX*, 21:101275, February 2023. ISSN 23527110.
 332 doi: 10.1016/j.softx.2022.101275.

333 [] Nathan Fox, Tom August, Francesca Mancini, Katherine E. Parks, Fe-
 334elix Eigenbrod, James M. Bullock, Louis Sutter, and Laura J. Graham.
 335 “photosearcher” package in R: An accessible and reproducible method for
 336 harvesting large datasets from Flickr. *SoftwareX*, 12:100624, July 2020.
 337 ISSN 23527110. doi: 10.1016/j.softx.2020.100624.

338 [] Rainer M Krug and Owen L Petchey. Microxanox: Model to simulate three
 339 functional group system. June 2022. doi: 10.5281/ZENODO.6624125.

340 [] Romana Limberger, Uriah Daugaard, Anubhav Gupta, Rainer M. Krug,
 341 Kimberley D. Lemmen, Sofia J. Van Moorsel, Marcel Suleiman, Debra
 342 Zuppinger-Dingley, and Owen L. Petchey. Functional diversity can facilitate
 343 the collapse of an undesirable ecosystem state. *Ecology Letters*, 26(6):
 344 883–895, June 2023. ISSN 1461-023X, 1461-0248. doi: 10.1111/ele.14217.

345 [] Amanda C. Northrop, Vanessa Avalone, Aaron M. Ellison, Bryan A. Ballif,
 346 and Nicholas J. Gotelli. Clockwise and counterclockwise hysteresis charac-
 347 terize state changes in the same aquatic ecosystem. *Ecology Letters*, 24(1):
 348 94–101, January 2021. ISSN 1461-023X, 1461-0248. doi: 10.1111/ele.13625.

349 [] Owen L Petchey, Uriah Daugaard, Anubhav Gupta, Rainer M Krug, Kimber-
 350 ley D. Lemmen, van Moorsel, Sofia J., Marcel Suleiman, Zuppinger-Dingley,
 351 Debra, and Romana Limberger. Code package for: "Functional diversity
 352 can facilitate the collapse of an undesirable ecosystem state". Zenodo, June
 353 2022.

354 [] R Core Team. *R: A Language and Environment for Statistical Computing*.
 355 R Foundation for Statistical Computing, Vienna, Austria, 2022.

356 [] Marten Scheffer, Steve Carpenter, Jonathan A. Foley, Carl Folke, and Brian
 357 Walker. Catastrophic shifts in ecosystems. *Nature*, 413(6856):591–596,
 358 October 2001. ISSN 0028-0836, 1476-4687. doi: 10.1038/35098000.

359 [] Marten Scheffer, Jordi Bascompte, William A. Brock, Victor Brovkin,
 360 Stephen R. Carpenter, Vasilis Dakos, Hermann Held, Egbert H. van Nes,
 361 Max Rietkerk, and George Sugihara. Early-warning signals for critical
 362 transitions. *Nature*, 461(7260):53–59, September 2009. ISSN 0028-0836,
 363 1476-4687. doi: 10.1038/nature08227.

- 364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788