Experiment 3

Owen Petchey

6/25/2021

Setup

R

```r
rm(list = ls())

knitr::opts_knit$set(progress = TRUE, verbose = FALSE, cache = TRUE)

microxanox_release <- "0.2.1"

#tmplib <- tempfile()
#dir.create(tmplib)


### From '?remotes::install_github`:
# auth_token
#   To install from a private repo, generate a personal access token (PAT) in
#   "https://github.com/settings/tokens" and supply to this argument. This is
#   safer than using a password because you can easily delete a PAT without
#   affecting any others. Defaults to the GITHUB_PAT environment variable.

# remotes::install_github(
#   "UZH-PEG/microxanox",
#   ref = microxanox_release,
#   # auth_token = "ENTER YOUR TOKEN or PROVED AS ENVIRONMENT VARIABLE",
#   build_vignettes = FALSE,
#   force = TRUE,
#   upgrade = FALSE
# )

library(microxanox)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.4     v purrr   0.3.4
## v tibble  3.1.2     v dplyr   1.0.7
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(patchwork)
library(here)
```

```
## here() starts at /Users/owenpetchey/Desktop/microxanox/diversity_envresp1
```

```
source(here("experiments/r functions/various_useful_functions.r"))
zero <- 0 ## don't change
unity <- 1 ## don't change!!!
#options(mc.cores = 8)
```

Version of `microxanox` package used: 0.2

General simulation conditions

```
default_dynamic_model <- bushplus_dynamic_model
default_event_definition <- event_definition_1
default_event_interval <- 100
default_noise_sigma <- 0
default_minimum_abundances <- rep(1, 3)
names(default_minimum_abundances) <- c("CB", "PB", "SB")
default_sim_duration <- 80000
default_sim_sample_interval <- 100
initial_pars_from <- "bush_ssfig3"
## note that next line (log10a_series is over-ridden with getting stable states)
#default_log10a_series <- c(-2, -2, -2, -2, -10, -10, -10, -10, -10)
```

Define diversity

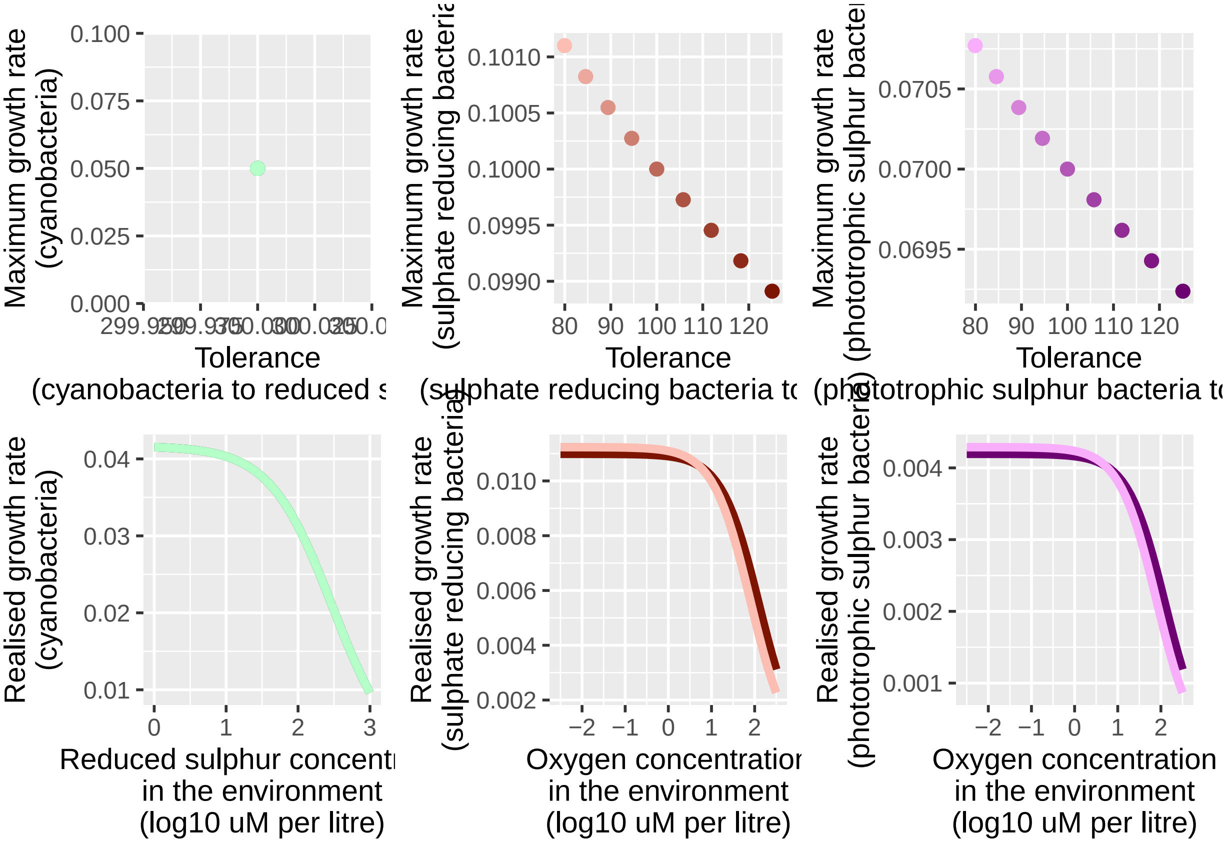
```
num_CB_strains <- 9
num_SB_strains <- 9
num_PB_strains <- 9
CB_gmax_div <- 0.015789474 * 0
CB_h_div <- -0.08 * 0
SB_gmax_div <- 0.015789474
SB_h_div <- -0.323
PB_gmax_div <- 0.015789474
PB_h_div <- -0.323

num_div_treatment_levels <- 20
```

Create diversity

```r
var_expt <- create_diversity()
```

Display diversity

```r
display_diversity()
```



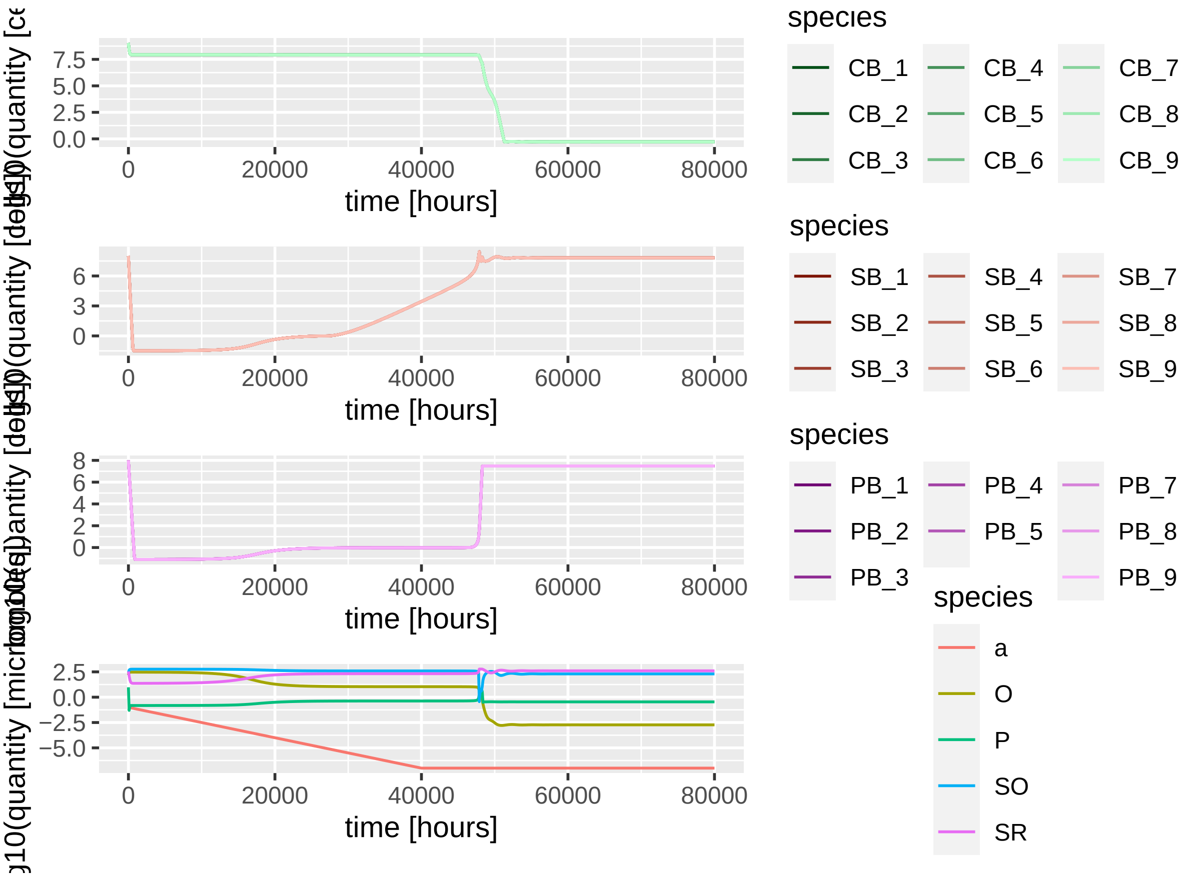
Temporal switching

Oxic to anoxic

No diversity

```r
default_sim_duration <- 80000
```

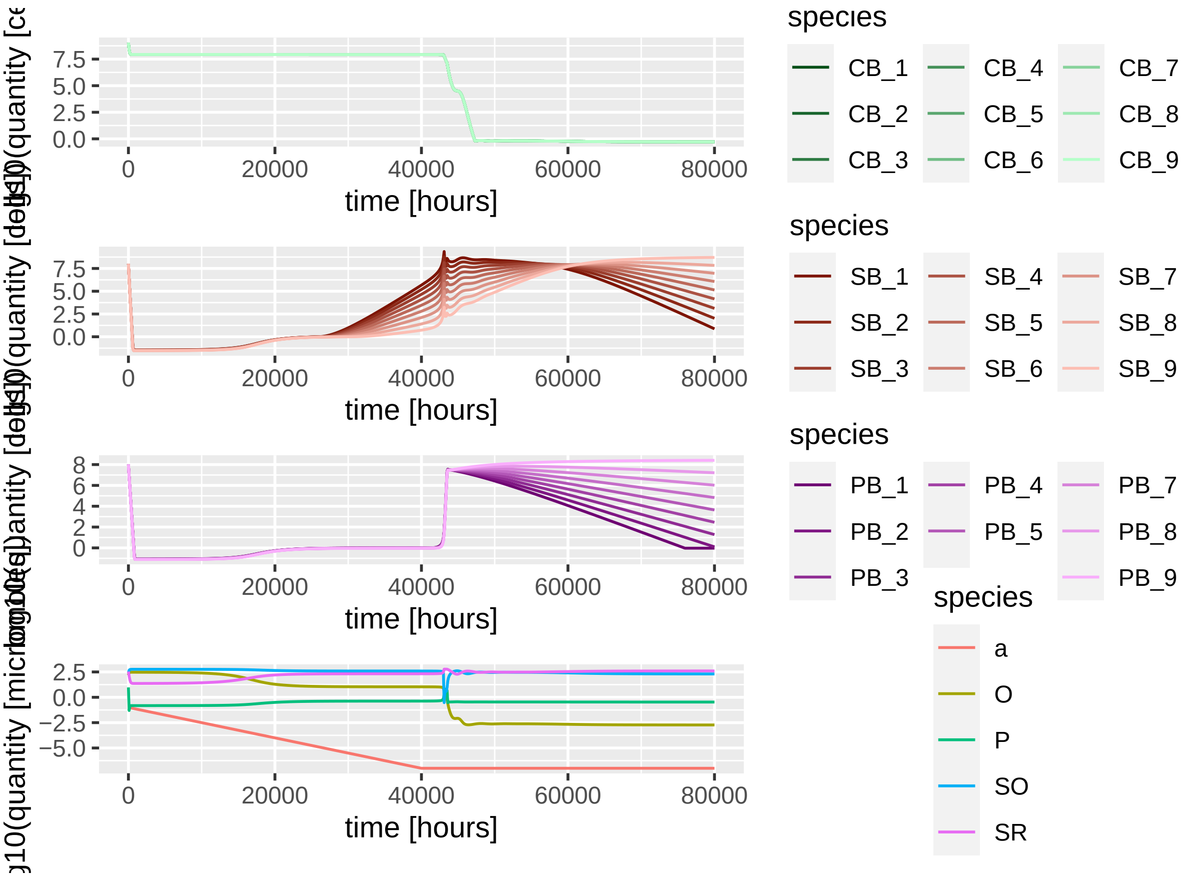
```r
sim_number <- 1
default_log10a_series <- c(-1, -7, -7)
initial_state <- new_initial_state(num_CB_strains,
                                   num_PB_strains,
                                   num_SB_strains,
                                    values = "bush_ssfig3")
initial_state[grep("CB_", names(initial_state))] <- 10^10/num_CB_strains
sim_res_novar <- run_simulation(parameter_values = var_expt$pars[[sim_number]],
                                initial_state = initial_state)
plot_dynamics(sim_res_novar)
```



```
#ggsave(here("simulations/expt2/figures/switching_novar.pdf"), width = 10)
```

Maximum diversity

```
sim_number <- num_div_treatment_levels
sim_res_highvar <- run_simulation(parameter_values = var_expt$pars[[sim_number]],
                                initial_state = initial_state)
plot_dynamics(sim_res_highvar)
```



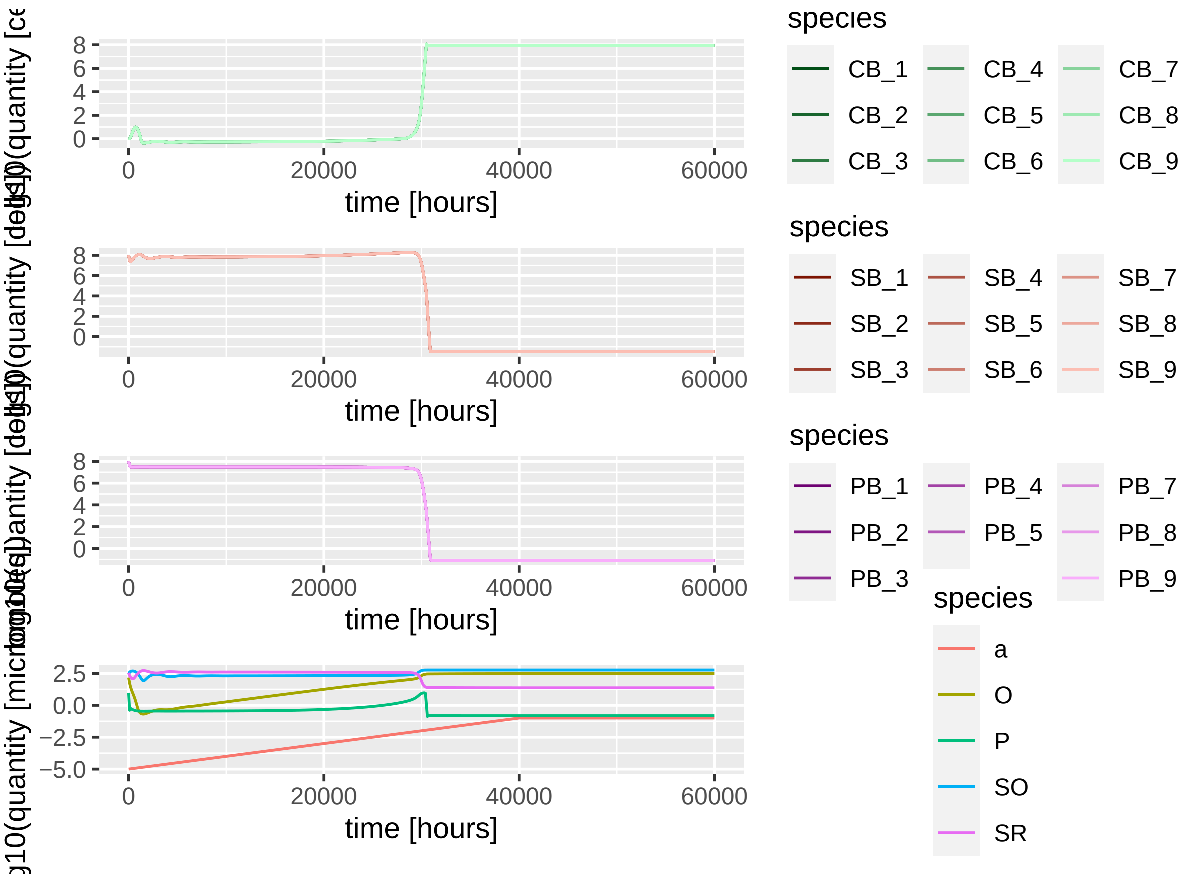
```
#ggsave(here("simulationsexpt2/figures/switching_highvar.pdf"), width = 10)
```

Anoxic to oxic

No diversity

```
default_sim_duration <- 60000
```

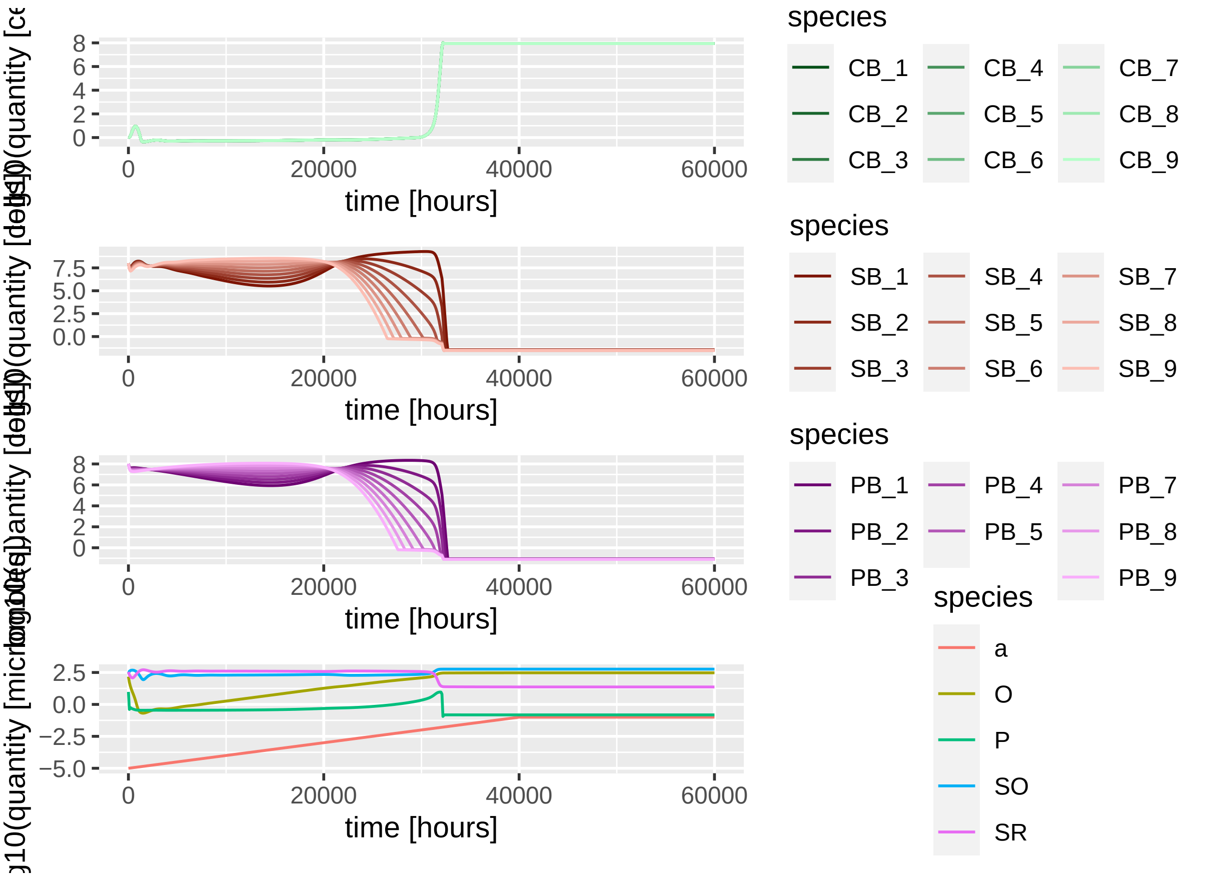
```
sim_number <- 1
default_log10a_series <- c(-5, -3, -1, -1)
initial_state <- new_initial_state(num_CB_strains,
                                   num_PB_strains,
                                   num_SB_strains,
                                   values = "bush_ssfig3")
initial_state[grep("CB_", names(initial_state))] <- 10/num_CB_strains
sim_res_novar <- run_simulation(parameter_values = var_expt$pars[[sim_number]],
                         initial_state = initial_state)
plot_dynamics(sim_res_novar)
```



```
#ggsave(here("simulations/expt2/figures/switching_novar.pdf"), width = 10)
```

Maximum diversity

```
sim_number <- num_div_treatment_levels
sim_res_highvar <- run_simulation(parameter_values = var_expt$pars[[sim_number]],
                                  initial_state = initial_state)
plot_dynamics(sim_res_highvar)
```



```
#ggsave(here("simulationsexpt2/figures/switching_highvar.pdf"), width = 10)
```

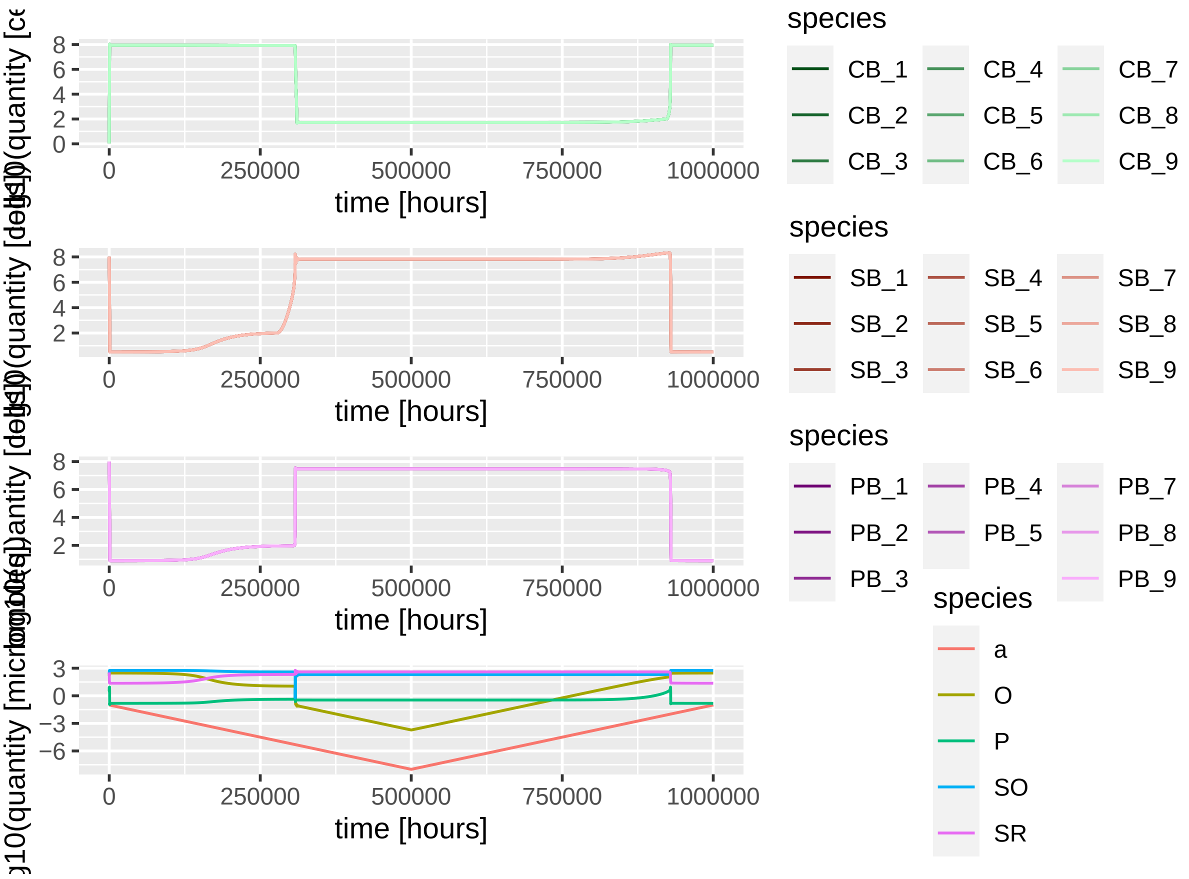
Anoxic to oxic to anoxic

No diversity

```
default_sim_duration <- 1000000
```

```
default_minimum_abundances <- rep(100, 3)
names(default_minimum_abundances) <- c("CB", "PB", "SB")
```

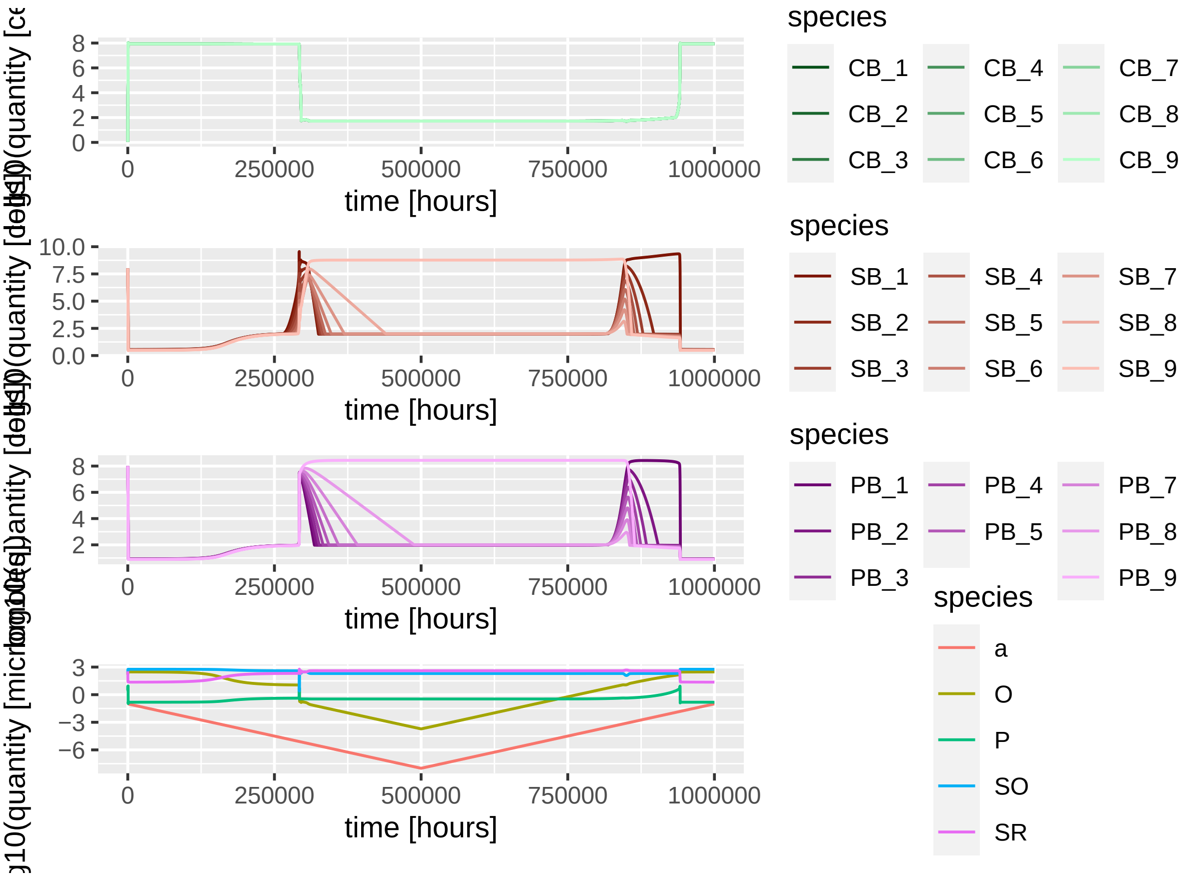
```
sim_number1 <- 1
default_log10a_series <- c(-1, -8, -1)
initial_state <- new_initial_state(num_CB_strains,
                                   num_PB_strains,
                                   num_SB_strains,
                                   values = "bush_ssfig3")
initial_state[grep("CB_", names(initial_state))] <- 10/num_CB_strains
sim_res_novar <- run_simulation(parameter_values = var_expt$pars[[sim_number1]],
                          initial_state = initial_state)
plot_dynamics(sim_res_novar)
```



```
#ggsave(here("simulations/expt2/figures/switching_novar.pdf"), width = 10)
```

Maximum diversity

```
sim_number2 <- num_div_treatment_levels
sim_res_highvar <- run_simulation(parameter_values = var_expt$pars[[sim_number2]],
                                  initial_state = initial_state)
plot_dynamics(sim_res_highvar)
```



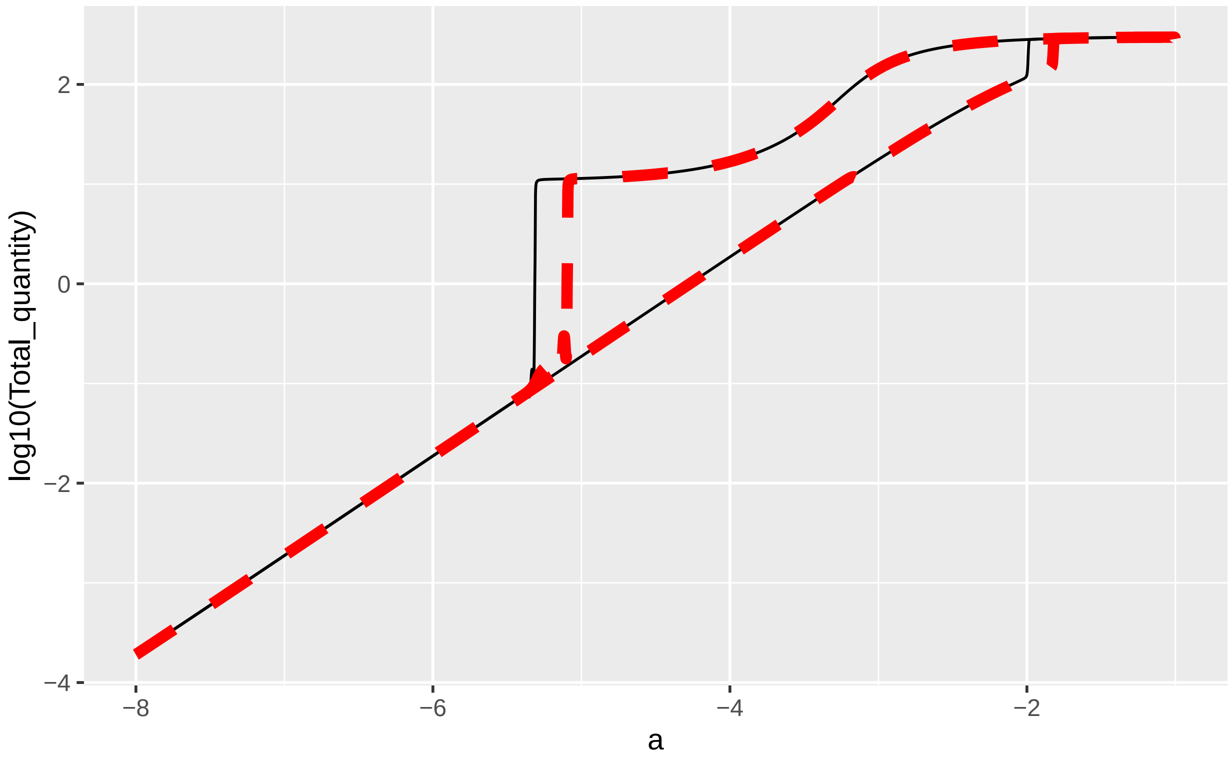
```
#ggsave(here("simulationsexpt2/figures/switching_highvar.pdf"), width = 10)
```

Visualise

```
visualise_temporal_env_eco()
```

```
## `summarise()` has grouped output by 'functional_group', 'a', 'var_type'. You can override using the
## `summarise()` has grouped output by 'functional_group', 'a', 'var_type'. You can override using the
```

Solid line is with no intraspecific diversity.
Dashed line is with intraspecific diversity



Stable state finding

Setup

```
options(mc.cores = 4)
```

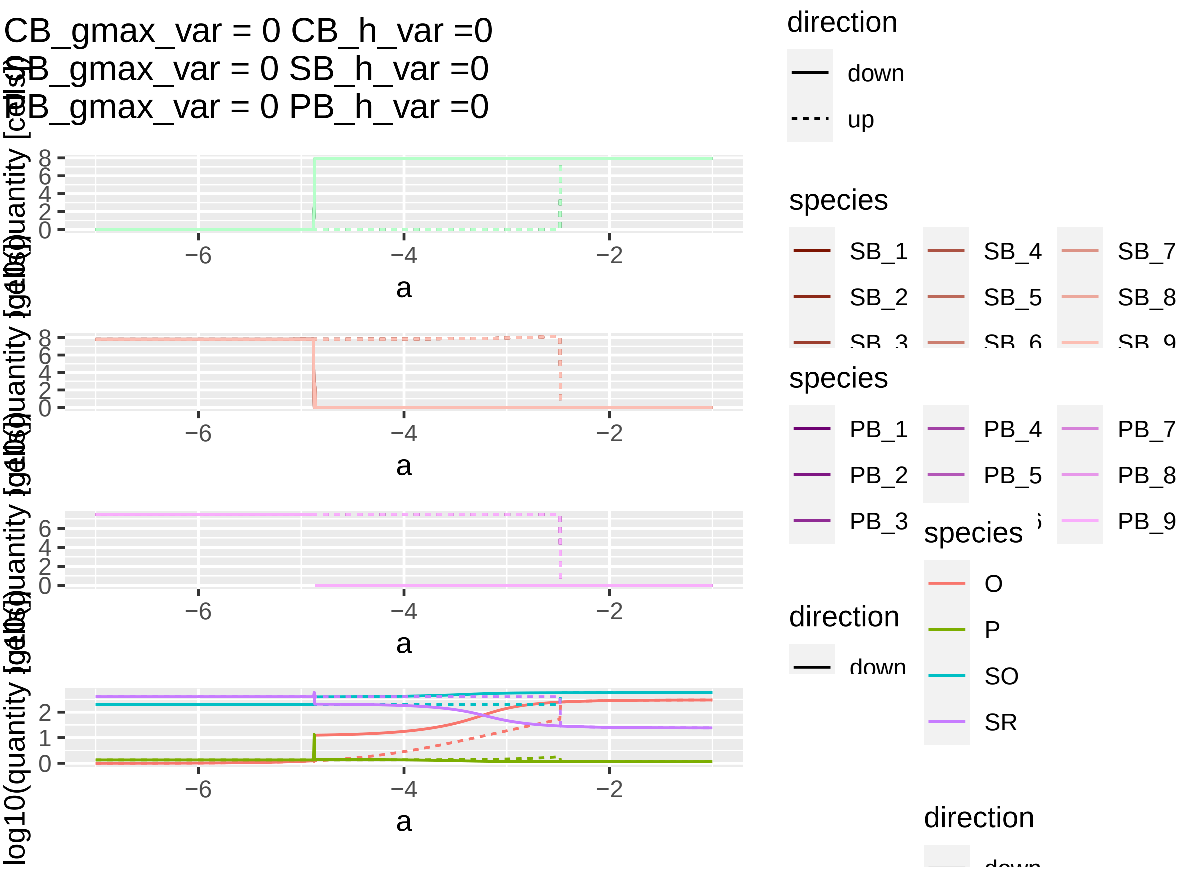
```
ssfind_minimum_abundances <- rep(0, 3)
names(ssfind_minimum_abundances) <- c("CB", "PB", "SB")
ssfind_simulation_duration <- default_sim_duration
ssfind_simulation_sampling_interval <- ssfind_simulation_duration
ssfind_event_interval <- ssfind_simulation_duration
grid_num_a <- 1000 ## number of a_0 values
a_Os <- 10^seq(-7, -1, length=grid_num_a) ## sequence of a_0 values
grid_num_N <- 2 ## number of N values
initial_CBs <- 10^seq(0, 10, length=grid_num_N) ## sequence of N values
initial_PBs <- 1e8 ## not varied
initial_SBs <- 1e8 ## not varied
# next line creates all possible combinations
ss_expt <- expand.grid(N_CB = initial_CBs,
                       N_PB = initial_PBs,
                       N_SB = initial_SBs,
                       a_0 = a_Os)
```

```r
#var_expt <- run_ss_var_experiment()
#saveRDS(var_expt, here("experiments/experiment 3/data/ss_data.RDS"))
```

```r
var_expt <- readRDS(here("experiments/experiment 3/data/ss_data.RDS"))

result_index1 <- 1
p1  <- plot_ss_result1(var_expt,
                result_index = result_index1,
                filename_prefix = NULL,
                save_image_file = FALSE)
```

```
## Warning in mask$eval_all_mutate(quo): NaNs produced
```

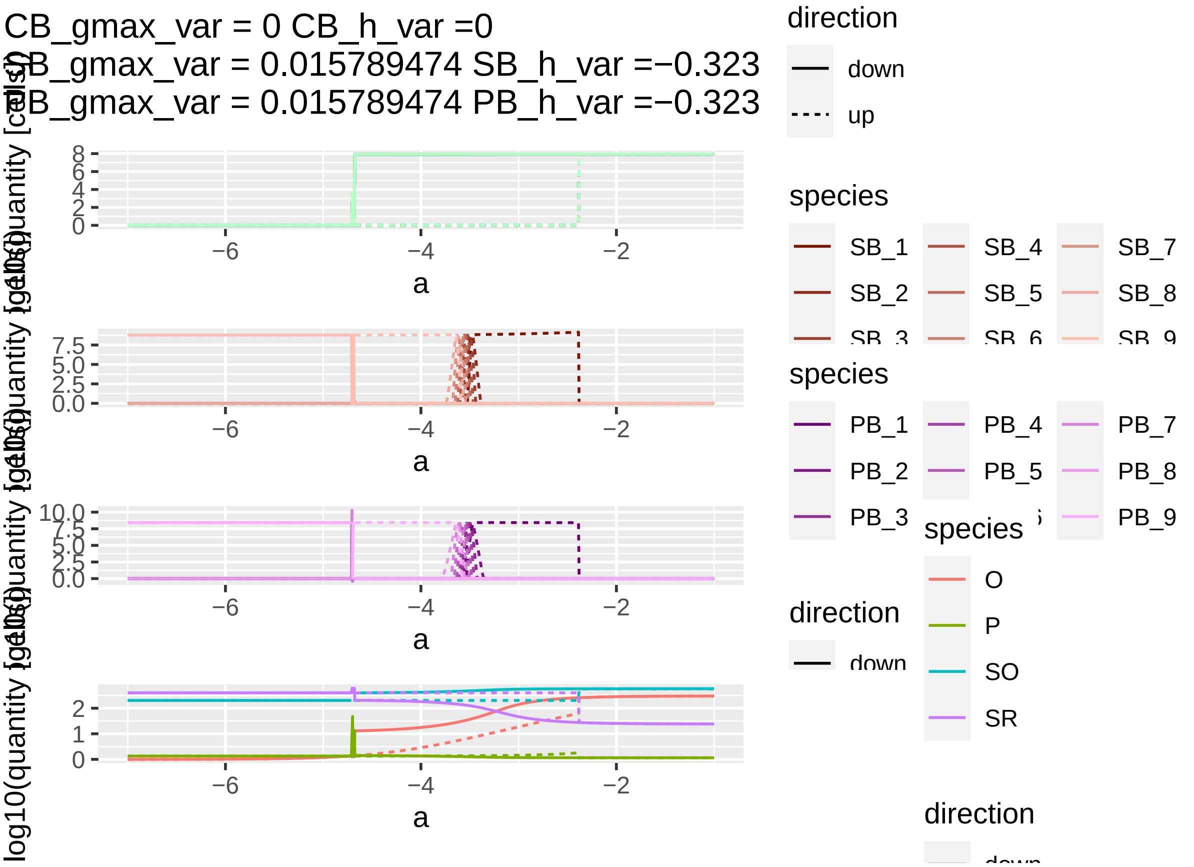
```r
p1
```



```r
result_index2 <- num_div_treatment_levels
p2  <- plot_ss_result1(var_expt,
                result_index = result_index2,
                filename_prefix = NULL,
                save_image_file = FALSE)
```

```
## Warning in mask$eval_all_mutate(quo): NaNs produced
```

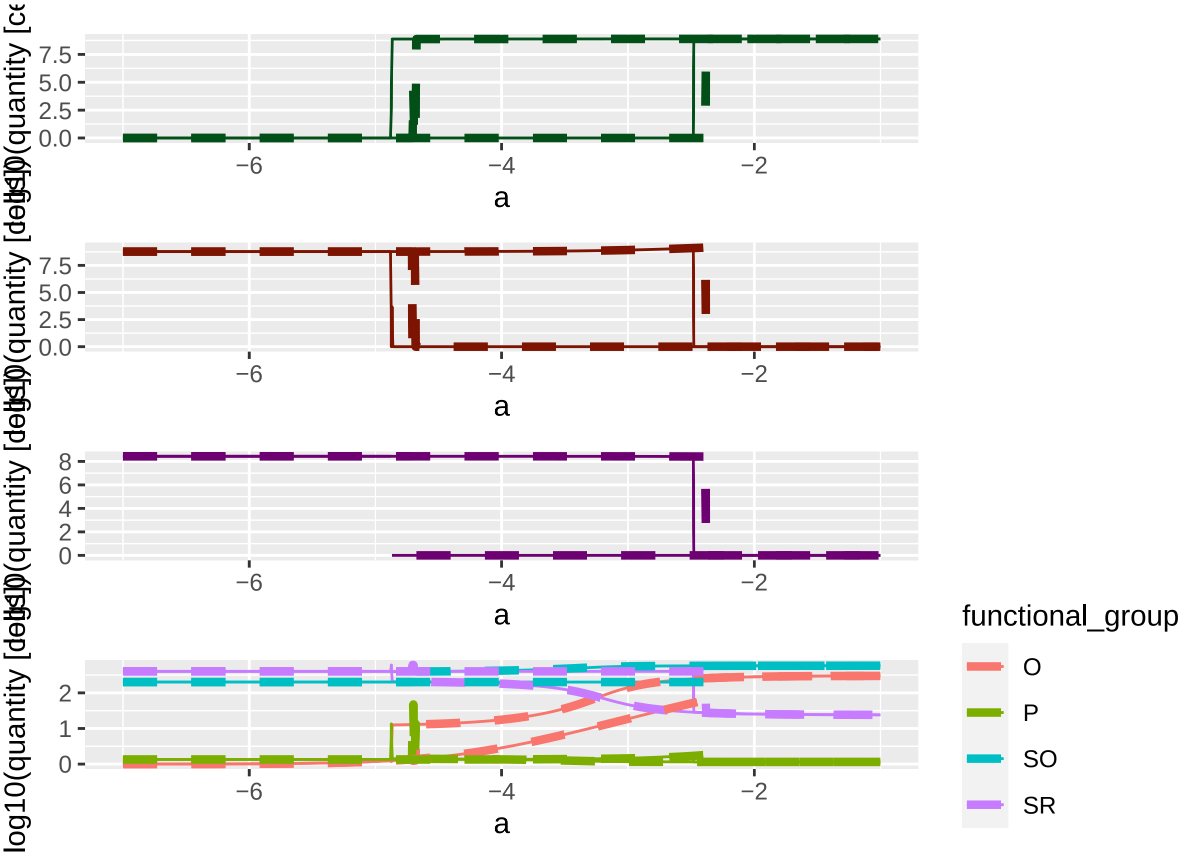
```
p2
```



```
p_overlay <- plot_ss_result2(var_expt[result_index1,]$ss_res[[1]],
                             var_expt[result_index2,]$ss_res[[1]],
                             xlims = c(-7, -1))
```

```
## ‘summarise()‘ has grouped output by ’a’, ’direction’, ’var_type’. You can override using the ‘.groups

## Warning in mask$eval_all_mutate(quo): NaNs produced

## Warning in mask$eval_all_mutate(quo): NaNs produced

## ‘summarise()‘ has grouped output by ’a’, ’direction’, ’var_type’. You can override using the ‘.groups

## Warning in mask$eval_all_mutate(quo): NaNs produced

## Warning in mask$eval_all_mutate(quo): NaNs produced

## Warning in mask$eval_all_mutate(quo): NaNs produced

## Warning in mask$eval_all_mutate(quo): NaNs produced

## Warning in mask$eval_all_mutate(quo): NaNs produced

## Warning in mask$eval_all_mutate(quo): NaNs produced
```

```
p_overlay
```



```
#ss_result1 <- var_expt[result_index1,]$ss_res[[1]]
#ss_result2 <- var_expt[result_index2,]$ss_res[[1]]
#xlims = c(-7, -1)
```