

experiment 1

Owen Petchey

6/25/2021

This experiment supercedes all previous ones. It is a factorial manipulation of diversity of the three groups. It takes about 50 hours to run while using 12 cores.

Setup

R

```
rm(list = ls())

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

microxanox_release <- "0.2"

#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(
#   "opetchey/microxanox",
#   ref = microxanox_release,
#   # auth_token = "ENTER YOUR TOKEN or PROVED AS ENVIRONMENT VARIABLE",
#   build_vignettes = FALSE,
#   force = TRUE,
#   upgrade = FALSE,
#   lib = tmplib
# )

#library(microxanox, lib.loc = tmplib)

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/microxano/ 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)
eval_dynamics_flag <- FALSE
```

Version of microxano 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
CB_h_div <- -0.08
SB_gmax_div <- 0.015789474
SB_h_div <- -0.323
PB_gmax_div <- 0.015789474
```

```
PB_h_div <- -0.323

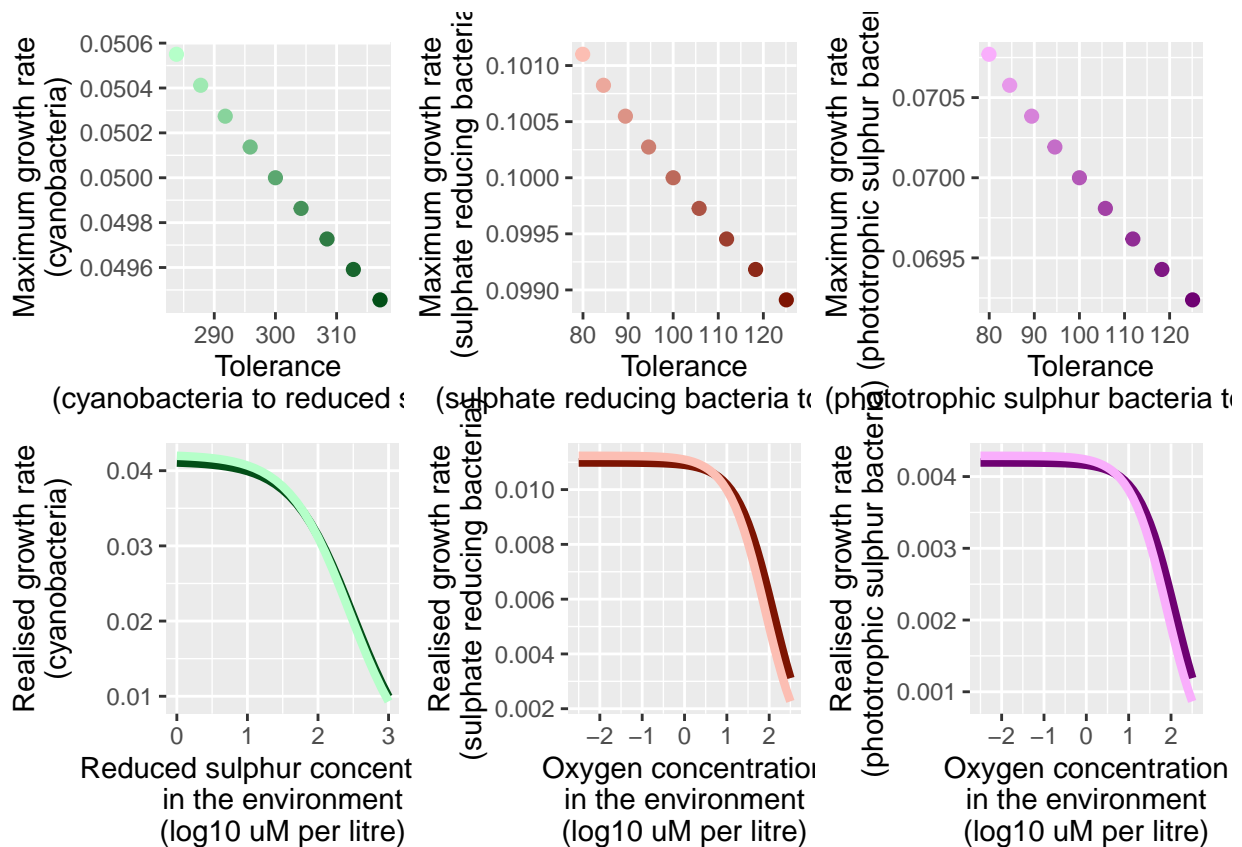
num_div_treatment_levels <- 20
```

Create diversity

```
var_expt <- create_diversity_factorial()
```

Display diversity

```
display_diversity(400)
```



Temporal switching

```
var_expt_levels <- var_expt[,1:6]

no_diversity <- which(rowSums(abs(var_expt_levels))==0)
max_diversity_all <- which(max(rowSums(abs(var_expt_levels))) ==
```

```

      rowSums(abs(var_expt_levels)))
max_only_CB_diversity <- which(max(rowSums(abs(var_expt_levels[,1:2]))) ==
      rowSums(abs(var_expt_levels[,1:2])) &
      rowSums(abs(var_expt_levels[,3:6]))==0)
#var_expt_levels[381,]

max_only_SBPB_diversity <- which(max(rowSums(abs(var_expt_levels[,3:6]))) ==
      rowSums(abs(var_expt_levels[,3:6])) &
      rowSums(abs(var_expt_levels[,1:2]))==0)
#var_expt_levels[20,]

```

Oxic to anoxic

No diversity

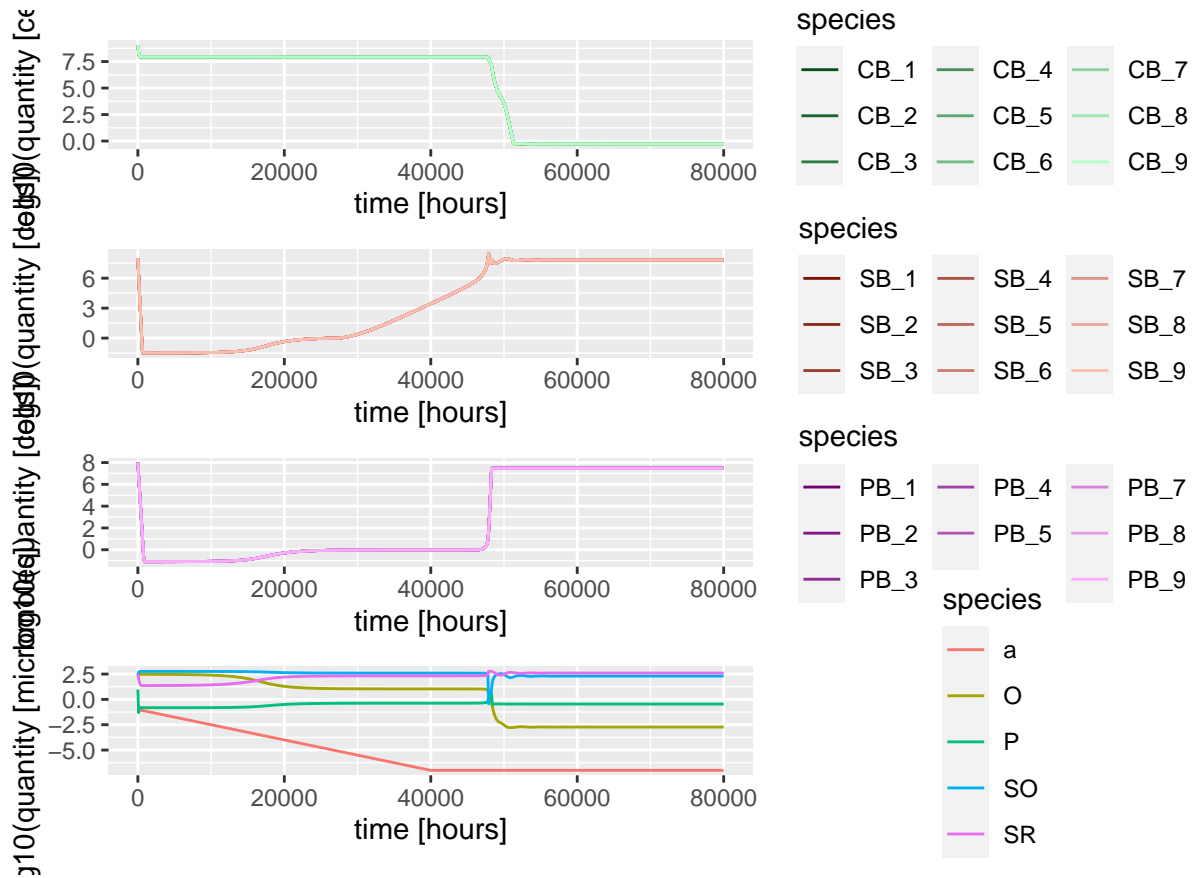
```

default_sim_duration <- 80000

default_log10a_series <- c(-1, -7, -7)
initial_state <- new_initial_state(num_CB_strains,
      num_PB_strains,
      num_SB_strains,
      values = "bush_ssf3")
initial_state[grepl("CB_", names(initial_state))] <- 10^10/num_CB_strains
sim_res_novar1 <- run_simulation(parameter_values = var_expt$pars[[no_diversity]],
      initial_state = initial_state)
saveRDS(sim_res_novar1, here("experiments/experiment 1/data/sim_res_novar1.RDS"))

sim_res_novar1 <- readRDS(here("experiments/experiment 1/data/sim_res_novar1.RDS"))
plot_dynamics(sim_res_novar1)

```

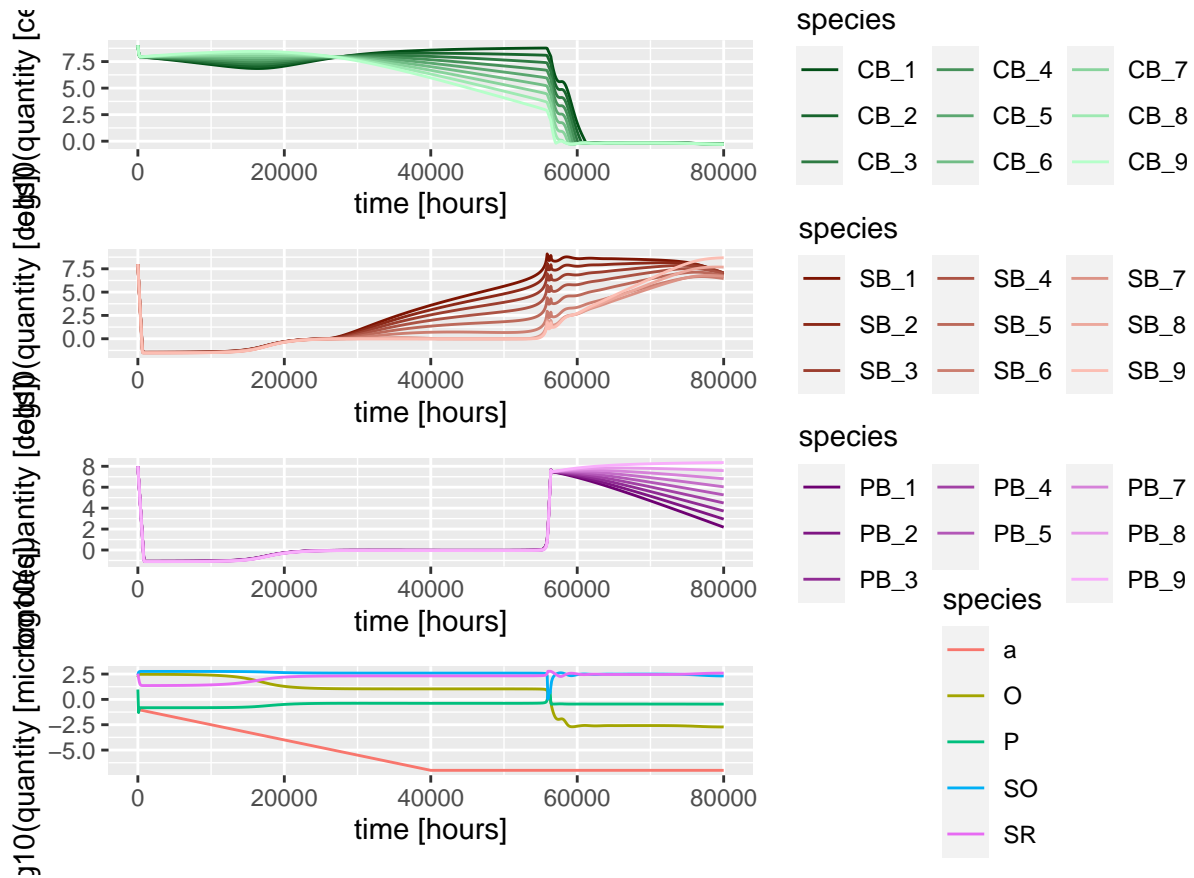


```
#ggsave(here("simulations/expt2/figures/switching_novar.pdf"), width = 10)
```

Maximum diversity

```
sim_number <- num_div_treatment_levels
sim_res_highvar1 <- run_simulation(parameter_values = var_expt$pars[[max_diversty_all]],
                                  initial_state = initial_state)
saveRDS(sim_res_highvar1, here("experiments/experiment 1/data/sim_res_highvar1.RDS"))
```

```
sim_res_highvar1 <- readRDS(here("experiments/experiment 1/data/sim_res_highvar1.RDS"))
plot_dynamics(sim_res_highvar1)
```



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

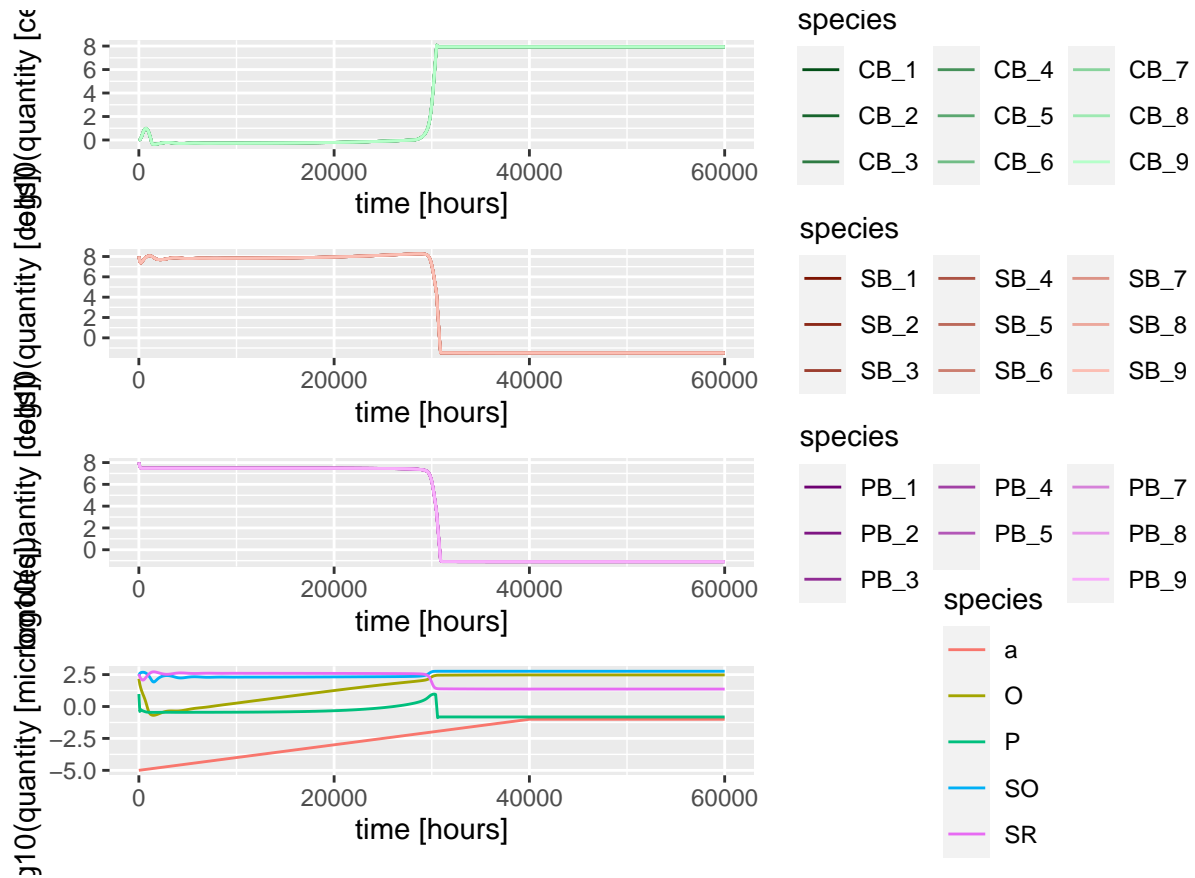
Anoxic to oxic

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_novar2 <- run_simulation(parameter_values = var_expt$pars[[no_diversity]],
                                initial_state = initial_state)
saveRDS(sim_res_novar2, here("experiments/experiment 1/data/sim_res_novar2.RDS"))
```

```
sim_res_novar2 <- readRDS(here("experiments/experiment 1/data/sim_res_novar2.RDS"))
plot_dynamics(sim_res_novar2)
```

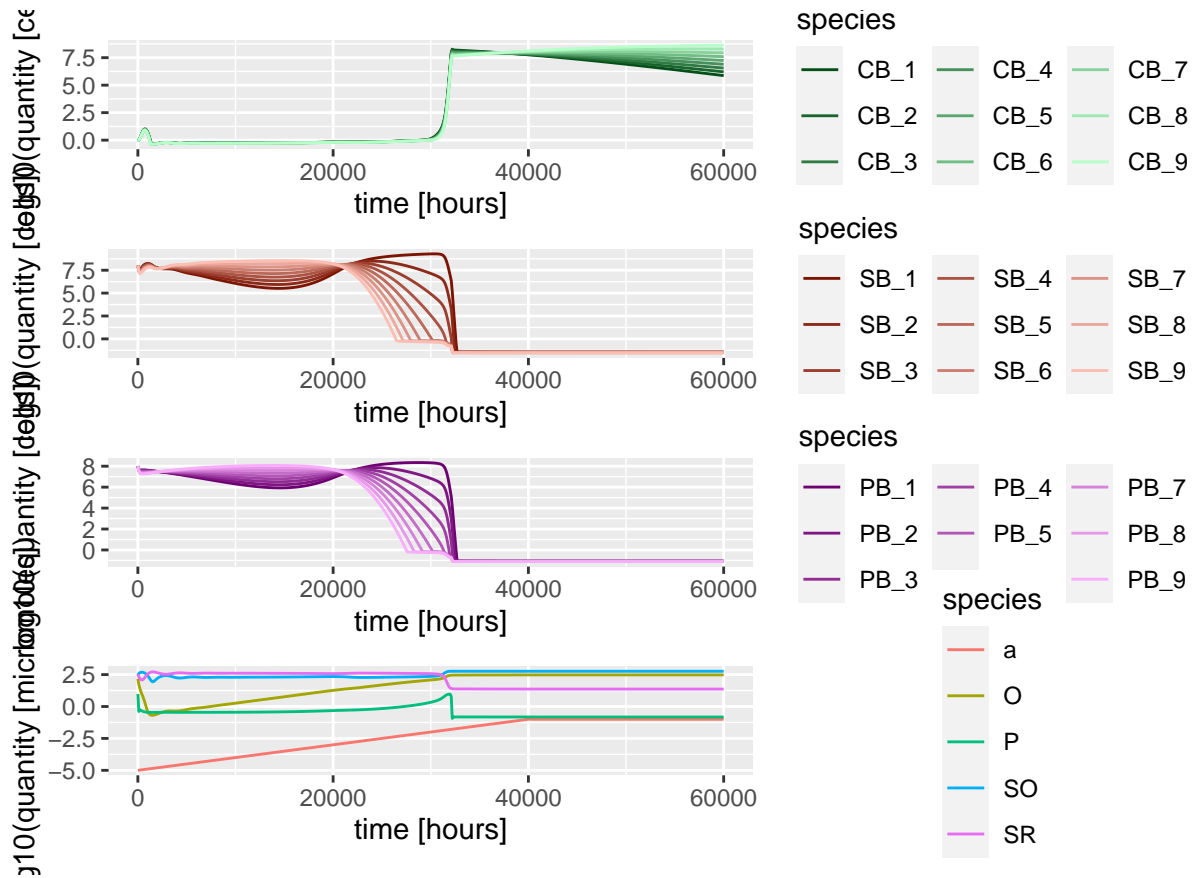


```
#ggsave(here("simulations/expt2/figures/switching_novar.pdf"), width = 10)
```

Maximum diversity

```
sim_number <- num_div_treatment_levels
sim_res_highvar2 <- run_simulation(parameter_values = var_expt$pars[[max_diversty_all]],
                                  initial_state = initial_state)
saveRDS(sim_res_highvar2, here("experiments/experiment 1/data/sim_res_highvar2.RDS"))
```

```
sim_res_highvar2 <- readRDS(here("experiments/experiment 1/data/sim_res_highvar2.RDS"))
plot_dynamics(sim_res_highvar2)
```



```
#ggsave(here("simulationsext2/figures/switching_highvar.pdf"), width = 10)
```

Anoxic to oxix to anoxic

No diversity

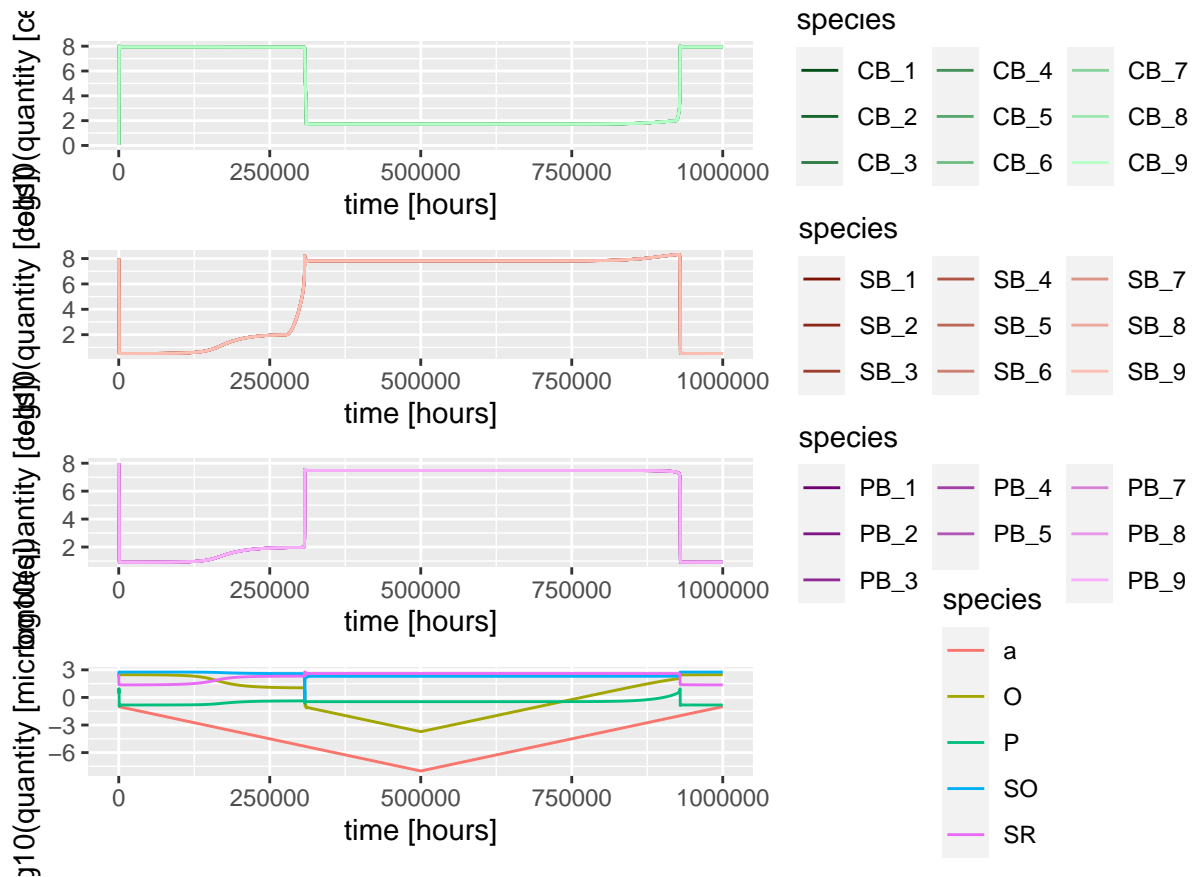
```
default_sim_duration <- 1000000
```

```
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_novar3 <- run_simulation(parameter_values = var_expt$pars[[no_diversity]],
                                initial_state = initial_state)
saveRDS(sim_res_novar3, here("experiments/experiment 1/data/sim_res_novar3.RDS"))
```



```
sim_res_novar3 <- readRDS(here("experiments/experiment 1/data/sim_res_novar3.RDS"))
plot_dynamics(sim_res_novar3)
```

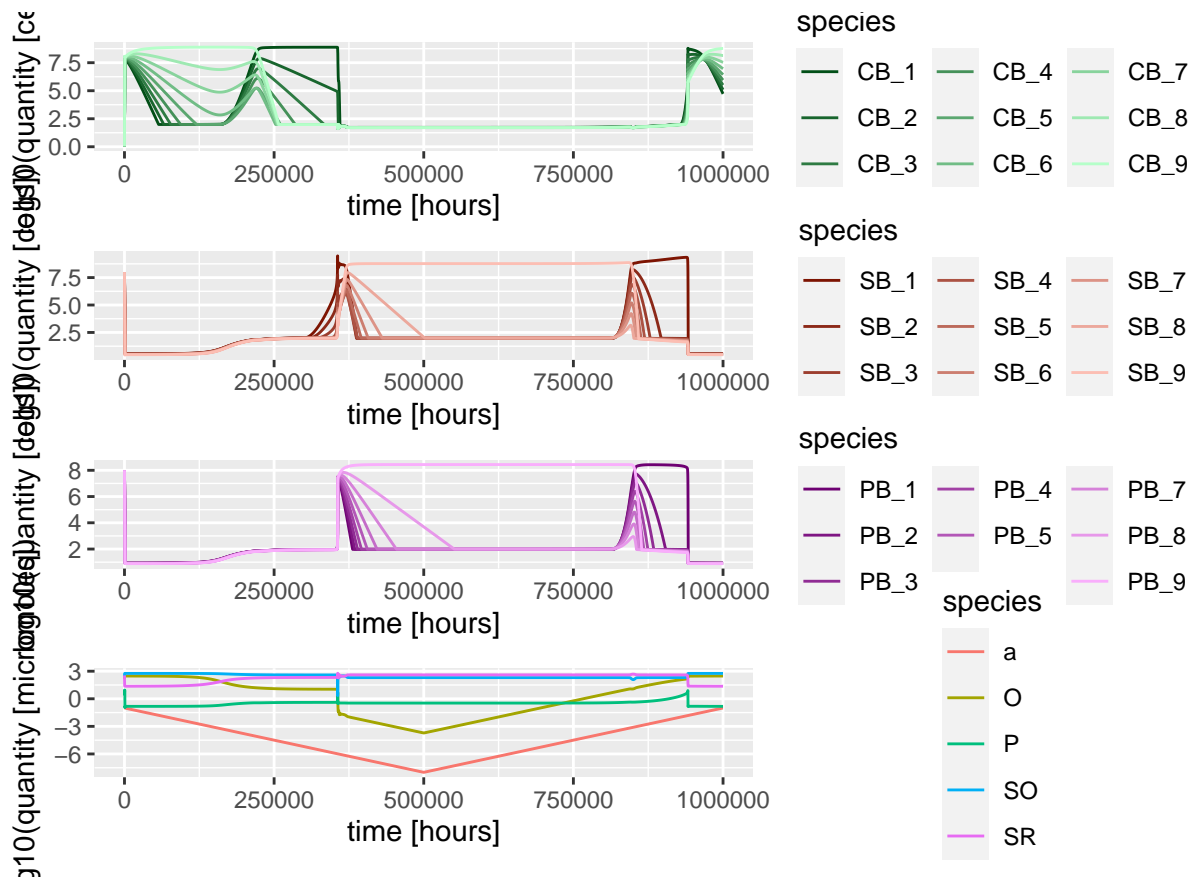


```
#ggsave(here("simulations/expt2/figures/switching_novar.pdf"), width = 10)
```

Maximum diversity

```
sim_number2 <- num_div_treatment_levels
sim_res_highvar3 <- run_simulation(parameter_values = var_expt$pars[[max_diversty_all]],
                                   initial_state = initial_state)
saveRDS(sim_res_highvar3, here("experiments/experiment 1/data/sim_res_highvar3.RDS"))
```

```
sim_res_highvar3 <- readRDS(here("experiments/experiment 1/data/sim_res_highvar3.RDS"))
plot_dynamics(sim_res_highvar3)
```



```
#ggsave(here("simulationsext2/figures/switching_highvar.pdf"), width = 10)
```

Visualise

```
visualise_temporal_env_eco()
```

Stable state finding

Setup

```
options(mc.cores = 12)
```

```
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 #usually 1000 ## number of a_0 values
a_0s <- 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_0s)

```

Run stable state finding

*Careful, this simulation takes about 600 hours on a single core

```

#var_expt <- run_ss_var_experiment()
#saveRDS(var_expt, here("experiments/experiment 1/data/ss_data.RDS"))

```

Process the stable state data

```

## find various combinations of diversity
var_expt <- readRDS(here("experiments/experiment 1/data/ss_data.RDS"))

var_expt_levels <- var_expt[,1:6]

no_diversity <- which(rowSums(abs(var_expt_levels))==0)
max_diversity_all <- which(max(rowSums(abs(var_expt_levels))) ==
                          rowSums(abs(var_expt_levels)))
max_only_CB_diversity <- which(max(rowSums(abs(var_expt_levels[,1:2]))) ==
                              rowSums(abs(var_expt_levels[,1:2])) &
                              rowSums(abs(var_expt_levels[,3:6]))==0)

#var_expt_levels[381,]

max_only_SBPB_diversity <- which(max(rowSums(abs(var_expt_levels[,3:6]))) ==
                                rowSums(abs(var_expt_levels[,3:6])) &
                                rowSums(abs(var_expt_levels[,1:2]))==0)

#var_expt_levels[20,]

```

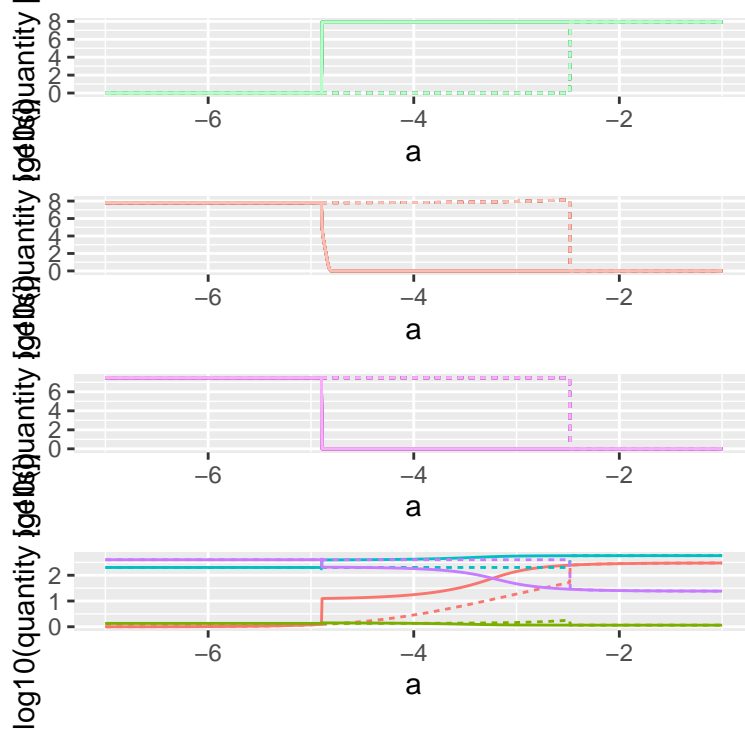
various graphs

```

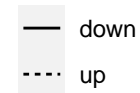
p1 <- plot_ss_result1(var_expt,
                      result_index = no_diversity,
                      filename_prefix = NULL,
                      save_image_file = FALSE)
p1

```

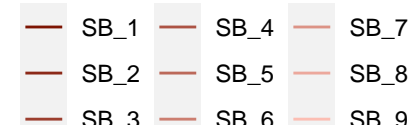
CB_gmax_var = 0 CB_h_var = 0
 SB_gmax_var = 0 SB_h_var = 0
 PB_gmax_var = 0 PB_h_var = 0



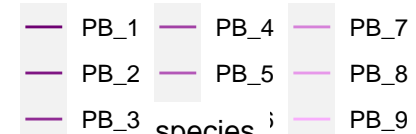
direction



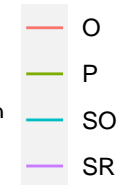
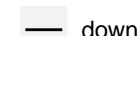
species



species



direction

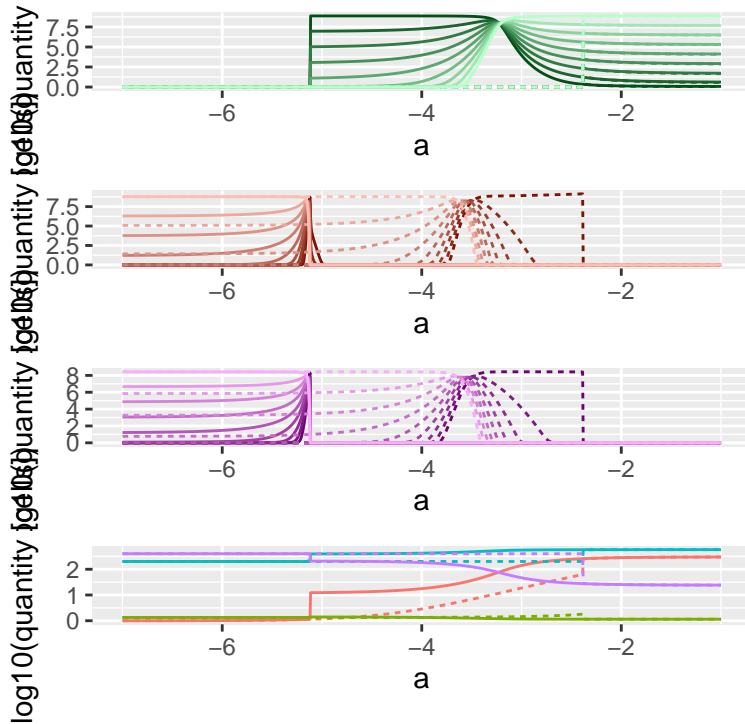


direction



```
p2 <- plot_ss_result1(var_expt,
  result_index = max_diversty_all,
  filename_prefix = NULL,
  save_image_file = FALSE)
p2
```

CB_gmax_var = 0.015789474 CB_h_var = -0.08
 SB_gmax_var = 0.015789474 SB_h_var = -0.323
 PB_gmax_var = 0.015789474 PB_h_var = -0.323



direction

— down
 up

species

SB_1	SB_4	SB_7
SB_2	SB_5	SB_8
SR_3	SR_6	SR_9

species

PB_1	PB_4	PB_7
PB_2	PB_5	PB_8
PB_3		PB_9

direction

— down

species
 O
 P
 SO
 SR

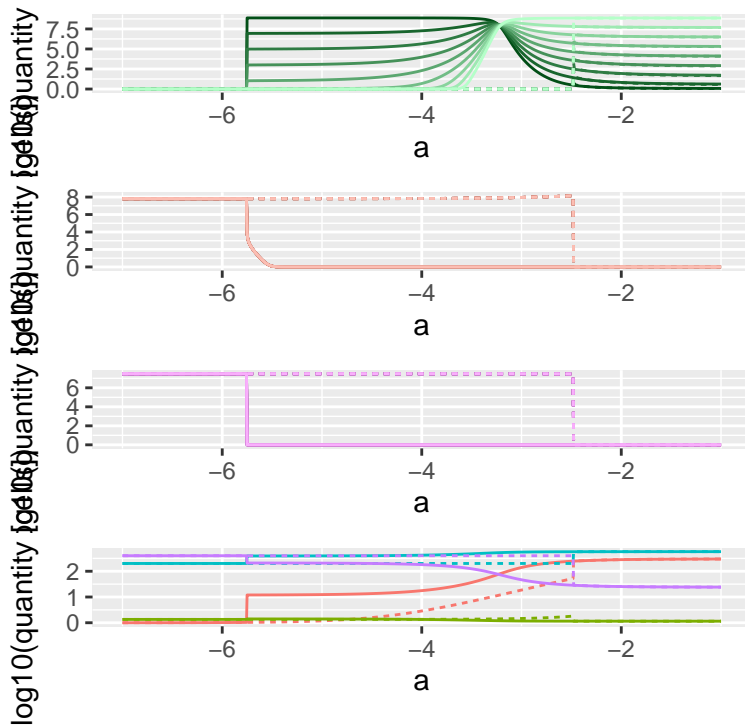
direction

— down

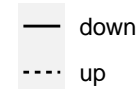
```
p3 <- plot_ss_result1(var_expt,
  result_index = max_only_CB_diversity,
  filename_prefix = NULL,
  save_image_file = FALSE)
```

p3

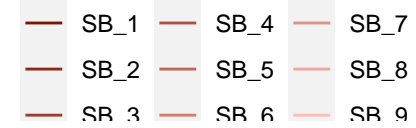
CB_gmax_var = 0.015789474 CB_h_var = -0.08
 SB_gmax_var = 0 SB_h_var = 0
 PB_gmax_var = 0 PB_h_var = 0



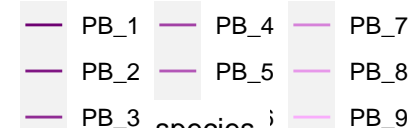
direction



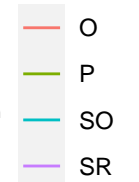
species



species



direction



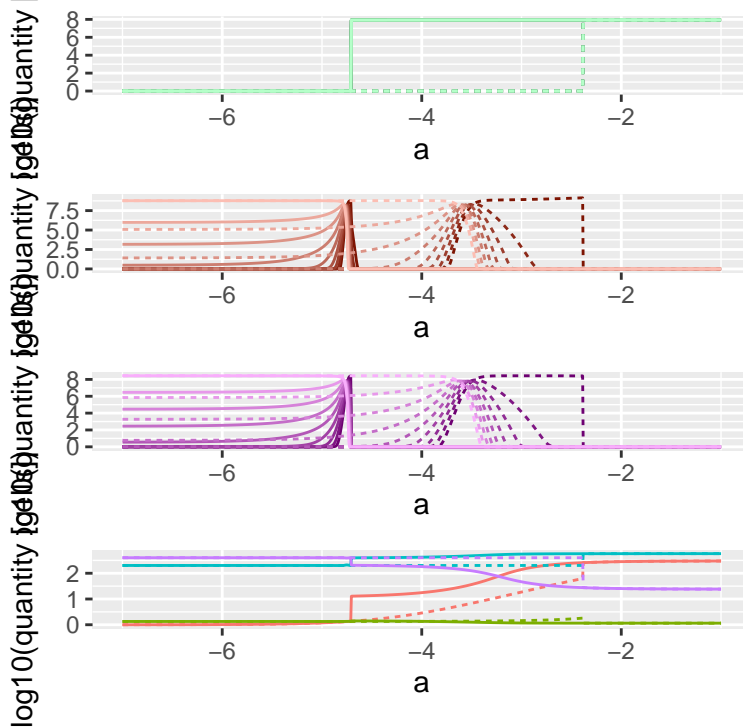
direction



```
p4 <- plot_ss_result1(var_expt,
  result_index = max_only_SBPB_diversity,
  filename_prefix = NULL,
  save_image_file = FALSE)
```

p4

CB_gmax_var = 0 CB_h_var = 0
 SB_gmax_var = 0.015789474 SB_h_var = -0.323
 PB_gmax_var = 0.015789474 PB_h_var = -0.323



direction

— down
 up

species

— SB_1	— SB_4	— SB_7
— SB_2	— SB_5	— SB_8
— SB_3	— SB_6	— SB_9

species

— PB_1	— PB_4	— PB_7
— PB_2	— PB_5	— PB_8
— PB_3	— PB_6	— PB_9

direction

— down

species
 — O
 — P
 — SO
 — SR

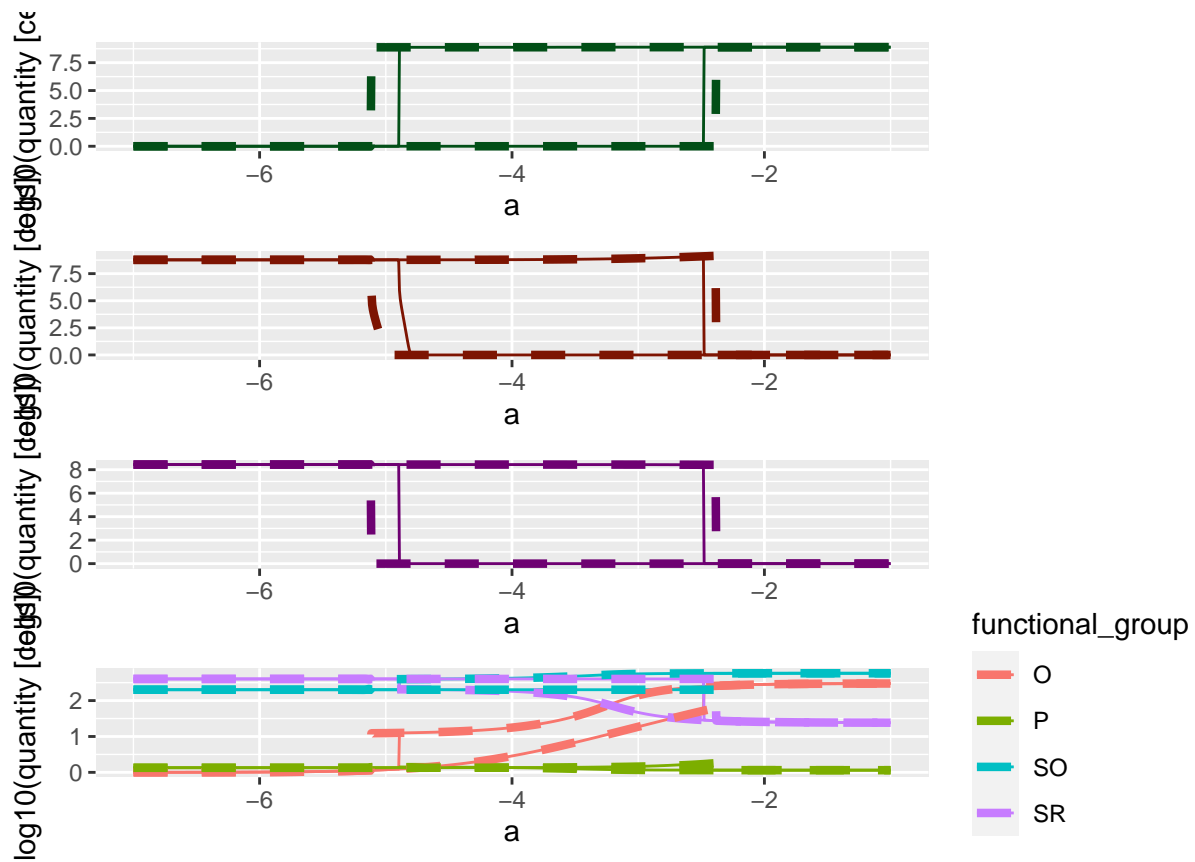
direction

— down

```
p_overlay1 <- plot_ss_result2(var_expt[no_diversity,]$ss_res[[1]],
                              var_expt[max_diversity_all,]$ss_res[[1]],
                              xlims = c(-7, -1))
```

'summarise()' has grouped output by 'a', 'direction', 'var_type'. You can override using the '.group' argument.
 ## 'summarise()' has grouped output by 'a', 'direction', 'var_type'. You can override using the '.group' argument.

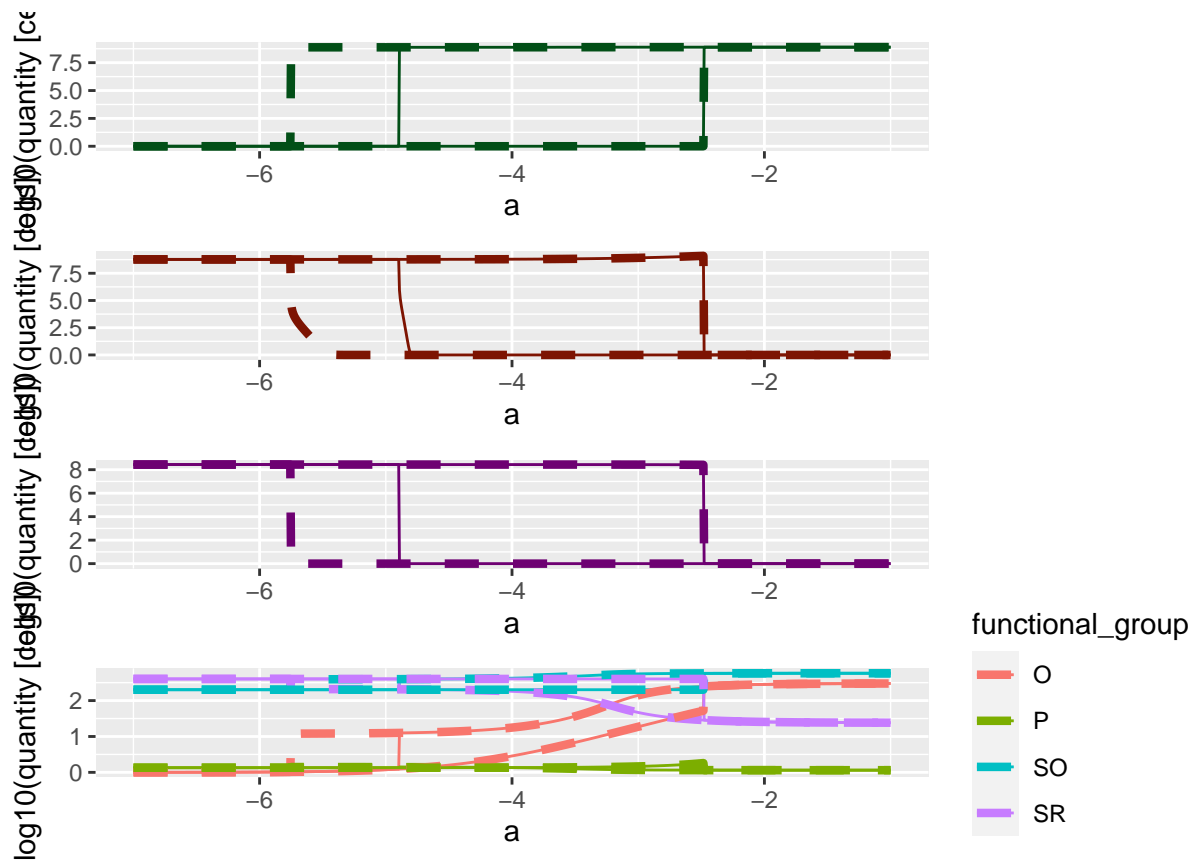
```
p_overlay1
```



```
p_overlay2 <- plot_ss_result2(var_expt[no_diversity,]$ss_res[[1]],
                             var_expt[max_only_CB_diversity,]$ss_res[[1]],
                             xlims = c(-7, -1))
```

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

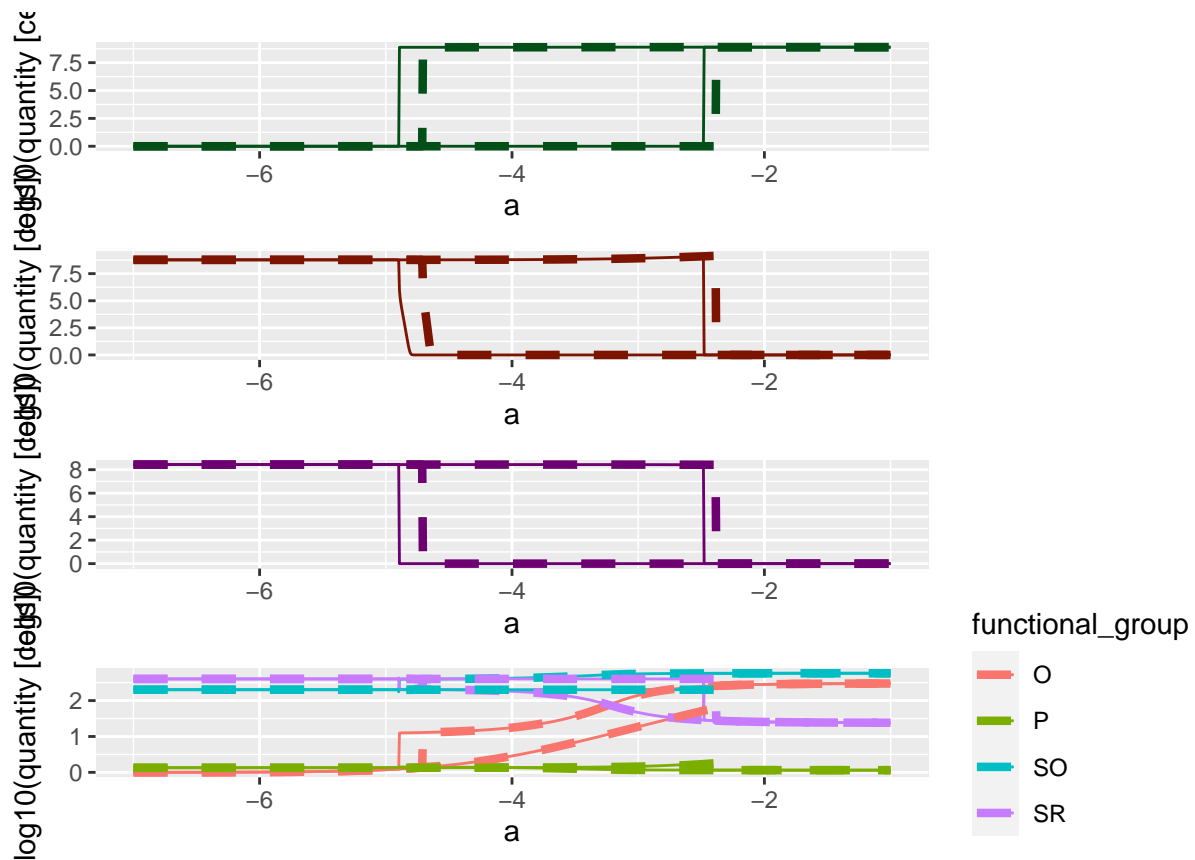
```
p_overlay2
```

```
p_overlay3 <- plot_ss_result2(var_expt[no_diversity,]$ss_res[[1]],
                             var_expt[max_only_SBPB_diversity,]$ss_res[[1]],
                             xlims = c(-7, -1))
```

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

```
p_overlay3
```



Calculate stability measures

```
stab_data <- var_expt %>%
  group_by(CB_var_gmax_s, CB_var_h_s,
           SB_var_gmax_s, SB_var_h_s,
           PB_var_gmax_s, PB_var_h_s) %>%
  do(stability_measures = get_stability_measures(.$ss_res[[1]]))
stab_data <- unnest(stab_data, cols = c(stability_measures))
#saveRDS(stab_data, here("experiments/experiment 1/data/stab_data.RDS"))
```

Process and visualise stability measures

```
stab_data <- readRDS(here("experiments/experiment 1/data/stab_data.RDS"))

CB_vars <- unique(stab_data$CB_var_gmax_s)
SB_vars <- unique(stab_data$CB_var_gmax_s)

CB_stab_data <- stab_data %>%
  filter(SB_var_gmax_s == 0) %>%
  mutate(var_treat = "CB",
         var_gmax = CB_var_gmax_s)
```

```

SBPB_stab_data <- stab_data %>%
  filter(CB_var_gmax_s == 0) %>%
  mutate(var_treat = "SB-PB",
         var_gmax = SB_var_gmax_s)

for_join <- tibble(CB_var_gmax_s = CB_vars,
                  SB_var_gmax_s = SB_vars)
CBSBPB_stab_data <- stab_data %>%
  right_join(for_join) %>%
  mutate(var_treat = "CB-SB-PB",
         var_gmax = CB_var_gmax_s)

## Joining, by = c("CB_var_gmax_s", "SB_var_gmax_s")

all_stab_results <- CB_stab_data %>%
  bind_rows(SBPB_stab_data) %>%
  # bind_rows(results3) %>%
  # bind_rows(results4) %>%
  bind_rows(CBSBPB_stab_data)

all_stab_results <- all_stab_results %>%
  mutate(var_treat = forcats::fct_relevel(var_treat, levels = c("CB", "SB-PB", "CB-SB-PB")))

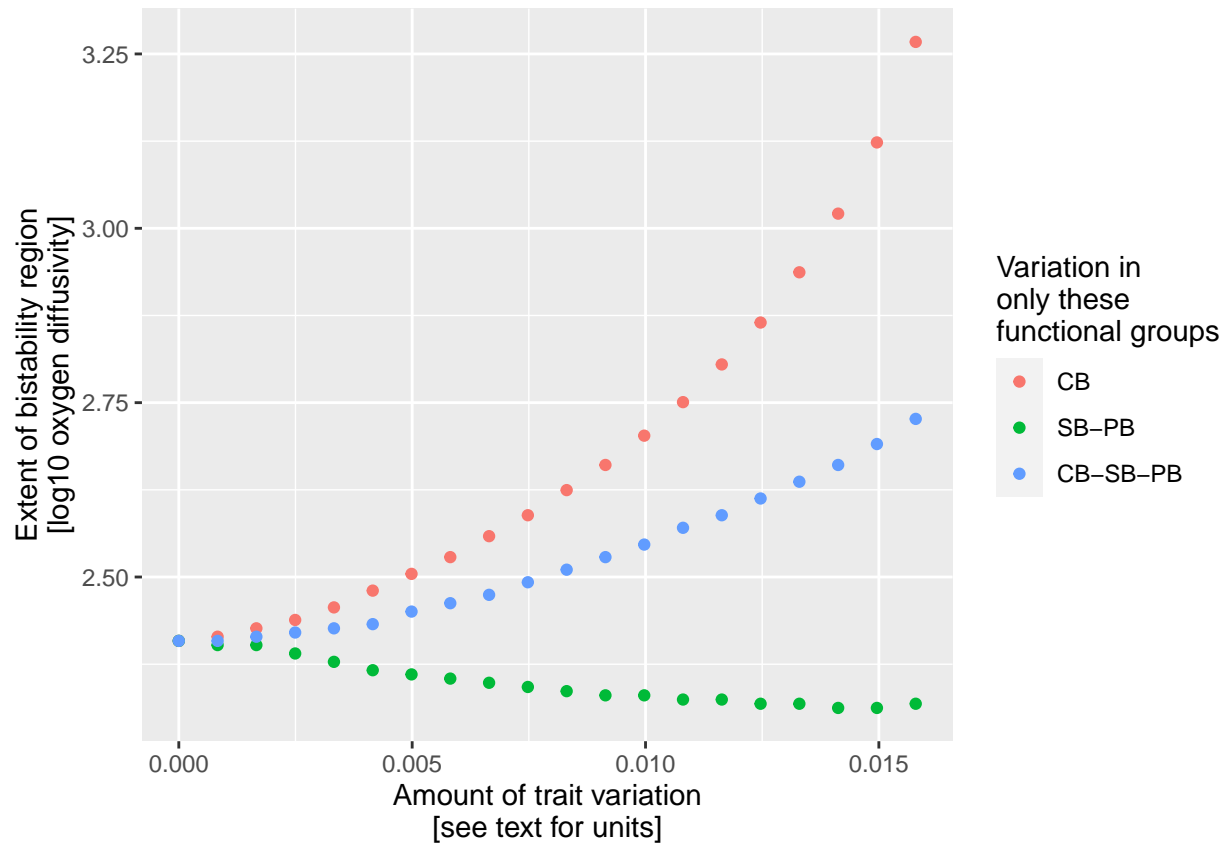
## Warning: Outer names are only allowed for unnamed scalar atomic inputs

#saveRDS(all_stab_results, here("experiments/experiment summary/all_stab.RDS"))

#all_stab_results <- readRDS(here("experiments/experiment summary/all_stab.RDS"))

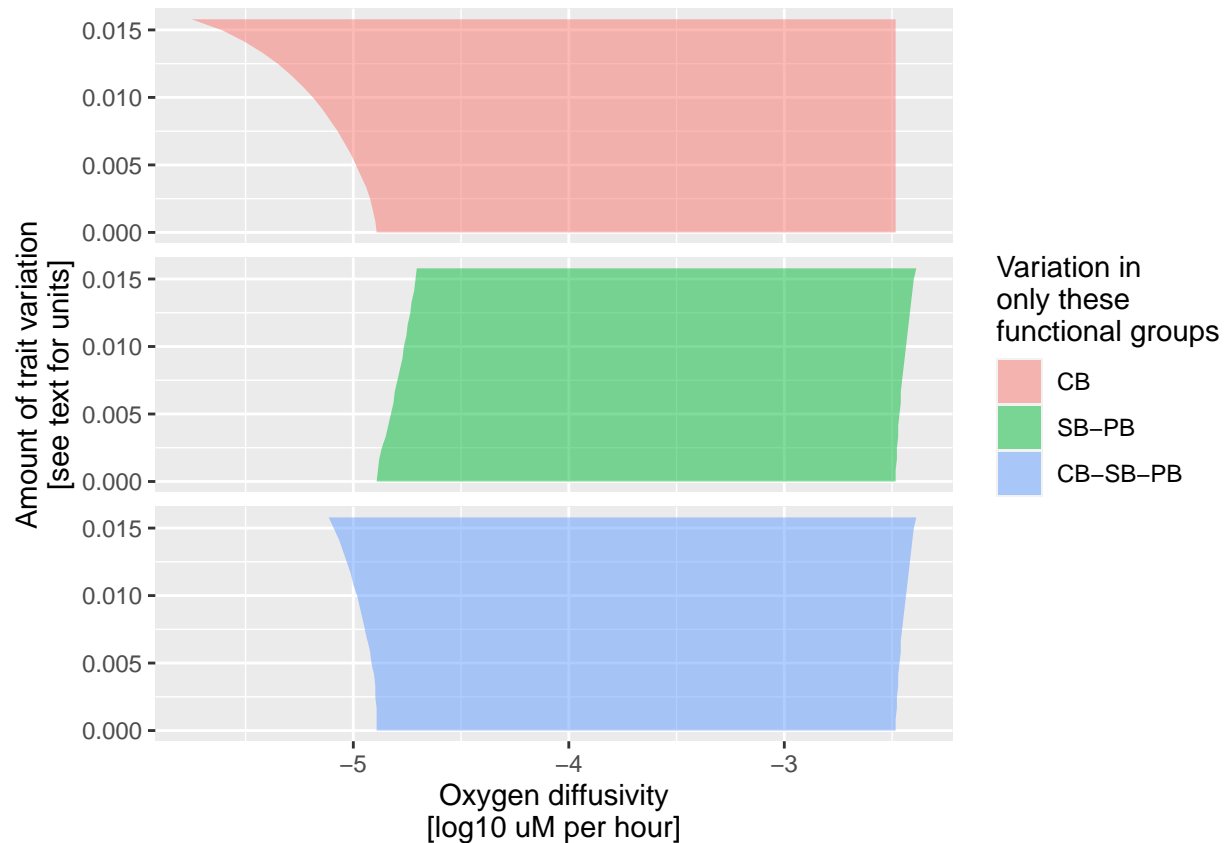
all_stab_results %>%
  filter(Species == "0") %>%
  ggplot(aes(x = var_gmax, y = hyst_range, col=var_treat)) +
  geom_point() +
  xlab("Amount of trait variation\n[see text for units]") +
  ylab("Extent of bistability region\n[log10 oxygen diffusivity]") +
  labs(col = "Variation in\nonly these\nfunctional groups")

```



```
##ggsave("manuscript/figures/extent_of_bistab1.pdf", height = 4)

all_stab_results %>%
  #filter(var_treat == "CB") %>%
  filter(Species == "0") %>%
  ggplot(aes(x = var_gmax,
             ymin = hyst_min,
             ymax = hyst_max,
             fill=var_treat)) +
  geom_ribbon(alpha = 0.5) +
  facet_wrap( ~ var_treat, nrow = 3) +
  xlab("Amount of trait variation\n[see text for units]") +
  ylab("Oxygen diffusivity\n[log10 uM per hour]") +
  labs(fill = "Variation in\nonly these\nfunctional groups") +
  coord_flip() +
  theme(
    strip.background = element_blank(),
    strip.text.x = element_blank()
  )
)
```



```
##ggsave("manuscript/figures/extent_of_bistab2.pdf", height = 4)
```

Zoom in on SS

```
a_0s <- 10^seq(-2.479, -2.4785, length=grid_num_a) ## sequence of a_0 values
initial_CBs <- 1#10^seq(0, 0, 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_0s)
var_expt_master <- var_expt
var_expt <- var_expt_master[381,]
```

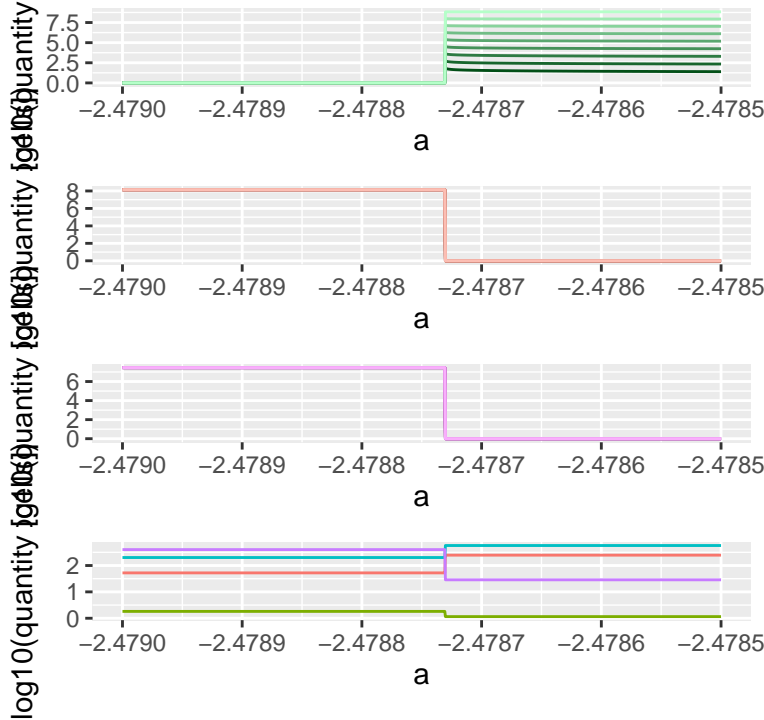
```
#var_expt <- run_ss_var_experiment()
#saveRDS(var_expt, here("experiments/experiment 1/data/ss_data_zoom.RDS"))
```

```
zoom <- readRDS(here("experiments/experiment 1/data/ss_data_zoom.RDS"))
p1 <- plot_ss_result1(zoom,
```

p1

```
result_index = 1,
filename_prefix = NULL,
save_image_file = FALSE)
```

CB_gmax_var = 0.015789474 CB_h_var = -0.08
 SB_gmax_var = 0 SB_h_var = 0
 PB_gmax_var = 0 PB_h_var = 0



direction

— up

species
species

— SB_1	— SB_4	— SB_7
— SB_2	— SB_5	— SB_8
— SB_3	— SB_6	— SB_9

direction

— up

species

— PB_1	— up	— PB_7
— PB_2	— PB_3	— PB_8

species

— O
— P
— SO

Negative abundance investigation

I (Owen) found that the sampling interval had an effect on the stability of the simulation. If the sampling interval was long, then in some rare cases (see below) the odesolver failed, with negative abundances occurring. I think this is due to abundances becoming very small, and then the computer having trouble with precision. I guess that when a sample is taken, the abundance is somehow altered if it is very low, probably by some rounding.

```
var_expt$pars[[1]]
dd <- var_expt$ss_res[[1]]
dd1 <- filter(dd, PB_1 < (-0.0001))
dd1$a_0

ss_expt_master <- ss_expt
ss_expt <- ss_expt_master[abs(ss_expt_master$a_0 - 1.336984e-05) < 1e-10,]

var_expt_master <- var_expt
#var_expt <- var_expt[1,]
```

```

var_expt_test <- run_ss_var_experiment()
res <- var_expt_test$ss_res[[1]]

test1 <- ss_by_a_N(ss_expt, var_expt$pars[[1]])
x <- ss_expt[2,]
param <- var_expt$pars[[1]]
get_final_states_a_N(x, param)
ssfind_parameters <- param
ssfind_simulation_sampling_interval <- 1000
## now run inside the function "get_final_states_a_N"
simres1 <- simres
ssfind_simulation_sampling_interval <- 5000
## now run inside the function "get_final_states_a_N"
simres2 <- simres # this fails

## now run inside the function "get_final_states_a_N"
plot_dynamics(simres2)

ggplot() +
  geom_line(data = simres1$result,
            mapping = aes(x = time, y = log10(PB_1))) +
  geom_point(data = simres2$result,
             mapping = aes(x = time, y = log10(PB_1))) +
  xlim(c(0, 250000))

ccc <- simres2$result

simres2$result$PB_1
simres2$result$time

log10_a <- log10(ss_expt$a_0[1]) ## very slowly goes anoxic
#log10_a <- log10(a_0s[354]) ## very slowly goes anoxic
#log10_a <- log10(a_0s[356]) ## very very very slowly goes anoxic
#log10_a <- log10(a_0s[357]) ## does not go anoxic

default_dynamic_model <- bushplus_dynamic_model
default_event_definition <- event_definition_1
default_event_interval <- ssfind_simulation_duration
default_noise_sigma <- 0
default_minimum_abundances <- ssfind_minimum_abundances
default_sim_duration <- ssfind_simulation_duration
default_sim_sample_interval <- ssfind_simulation_duration
#initial_pars_from <- "bush_ssfig3"

default_log10a_series <- c(log10_a, log10_a)
initial_state <- new_initial_state(num_CB_strains,
                                   num_PB_strains,
                                   num_SB_strains,
                                   values = "bush_ssfig3")

```

```

initial_state[grepl("CB_", names(initial_state))] <- 10^10/num_CB_strains
sim_res_novar <- run_simulation(parameter_values = var_expt$pars[[1]],
                               initial_state = initial_state)
plot_dynamics(sim_res_novar)

simulation_result <- sim_res_novar
every_n <- 1

chk <- sim_res_novar$result

sim_res_novar$result %>%
  ggplot() +
  geom_line(mapping = aes(x = time, y = PB_1))

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

```