

Recreating the analyses with the `metafolio` R package

Sean C. Anderson^{1*}, Jonathan W. Moore^{1,2}, Michelle M. McClure³,
Nicholas K. Dulvy¹ Andrew B. Cooper²

¹Earth to Ocean Research Group, Department of Biological Sciences, Simon Fraser University, Burnaby BC, V5A 1S6, Canada

²School of Resource and Environmental Management, Simon Fraser University, Burnaby, BC, V5A 1S6, Canada

³Fishery Resource Analysis and Monitoring Division, Northwest Fisheries Science Center, National Marine Fisheries Service, National Oceanic and Atmospheric Administration, 2725 Montlake Blvd. E., Seattle, WA 98112, USA

*Corresponding author

This supplement accompanies the paper *Portfolio conservation of metapopulations under climate change* in *Ecological Applications*.

1 Loading the package

The `metafolio` R package is a tool to simulate metapopulations and apply financial portfolio optimization concepts to those metapopulations. The package is primarily intended for salmon simulations, but could be adapted for other taxonomic groups. In this document, we describe how to install and load the package, and run the analyses in our paper.

`metafolio` is available on CRAN: <http://cran.r-project.org/package=metafolio> and can be installed with:

```
install.packages("metafolio")
```

The analyses in this paper used `metafolio` version 0.1.0.

Alternatively, you can view the code and install the package from <http://github.com/seananderson/metafolio>. Installing the package from GitHub will require that you have a C++ compiler installed.

Load the package with:

```
library("metafolio")
```

The included vignette describes the package and illustrates some example simulations. You can view the vignette with:

```
vignette("metafolio")
```

You can view the help for the package with:

```
?metafolio  
help(package = "metafolio")
```

The figures from this paper can be re-created by downloading the source code from CRAN or GitHub and sourcing the file README.R in the `inst/examples` folder:

```
setwd("metafolio/inst/examples")  
source("README.R")
```

We have copied some examples from that code below. See the source code on GitHub for the full analysis. To save time compiling this document, we have run fewer iterations in these examples than in the paper.

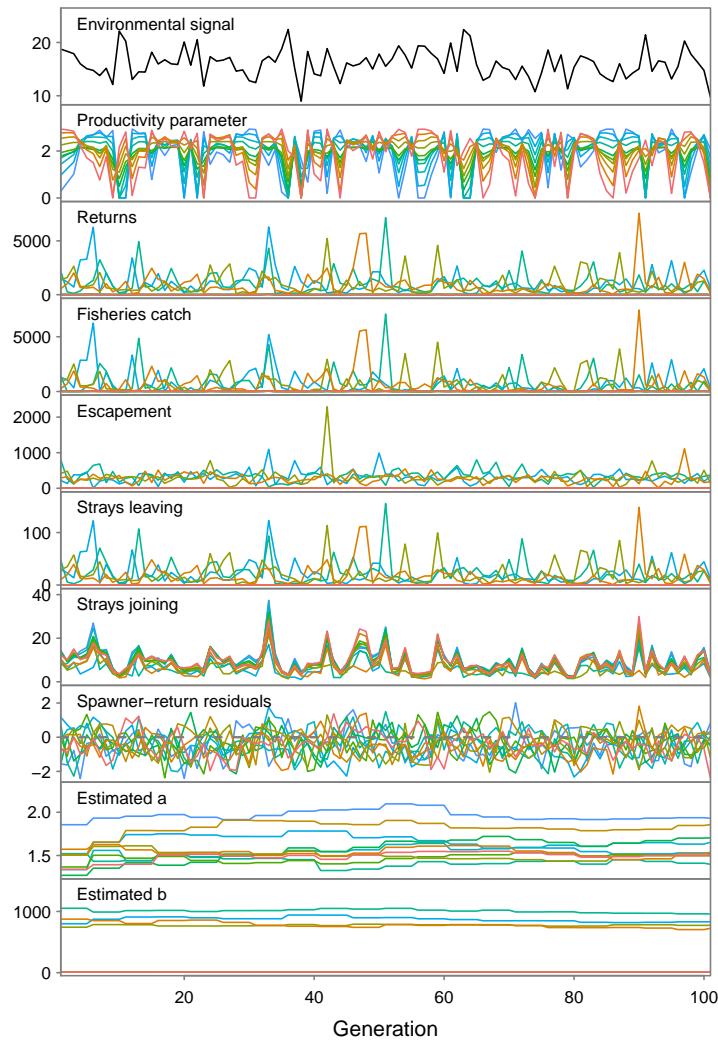
2 Prioritizing spatial response diversity

Setup the scenarios:

```
set.seed(1)  
w_plans <- list()  
w_plans[[1]] <- c(5, 1000, 5, 1000, 5, 5, 1000, 5, 1000, 5)  
w_plans[[2]] <- c(5, 5, 5, 1000, 1000, 1000, 1000, 5, 5, 5)  
w_plans[[3]] <- c(rep(1000, 4), rep(5, 6))  
w_plans[[4]] <- rev(w_plans[[3]])  
plans_name_sp <- c("Full response range", "Most stable only",  
  "Lower half", "Upper half")  
n_trials <- 200 # number of trials at each conservation plan  
num_pops <- c(10, 10, 10, 10)  
n_plans <- length(num_pops) # number of plans  
w <- list()  
for(i in 1:n_plans) { # loop over plans  
  w[[i]] <- list()  
  for(j in 1:n_trials) { # loop over trials  
    w[[i]][[j]] <- matrix(w_plans[[i]], nrow = 1)  
  }  
}
```

Here's one example of generating a time series plot:

```
set.seed(123)
arma_env_params <- list(mean_value = 16, ar = 0.1, sigma_env = 3, ma = 0)
eg_arma <- meta_sim(b = w[[1]][[1]], n_pop = 10, env_params = arma_env_params,
  env_type = "arma", assess_freq = 5)
plot_sim_ts(eg_arma, years_to_show = 100, burn = 30)
```



Now, we'll run the simulations:

```
x_arma_sp <- run_cons_plans(w, env_type = "arma", env_params =
  arma_env_params, show_progress = FALSE)
x_arma_sp$plans_port <- NULL # save space
```

```

linear_env_params <- list(min_value = 15, max_value = 19, sigma_env = 0.001,
  start_t = 30)
x_linear_sp <- run_cons_plans(w, env_type = "linear", env_params =
  linear_env_params, max_a = thermal_integration(10),
  show_progress = FALSE)
x_linear_sp$plans_port <- NULL # save space

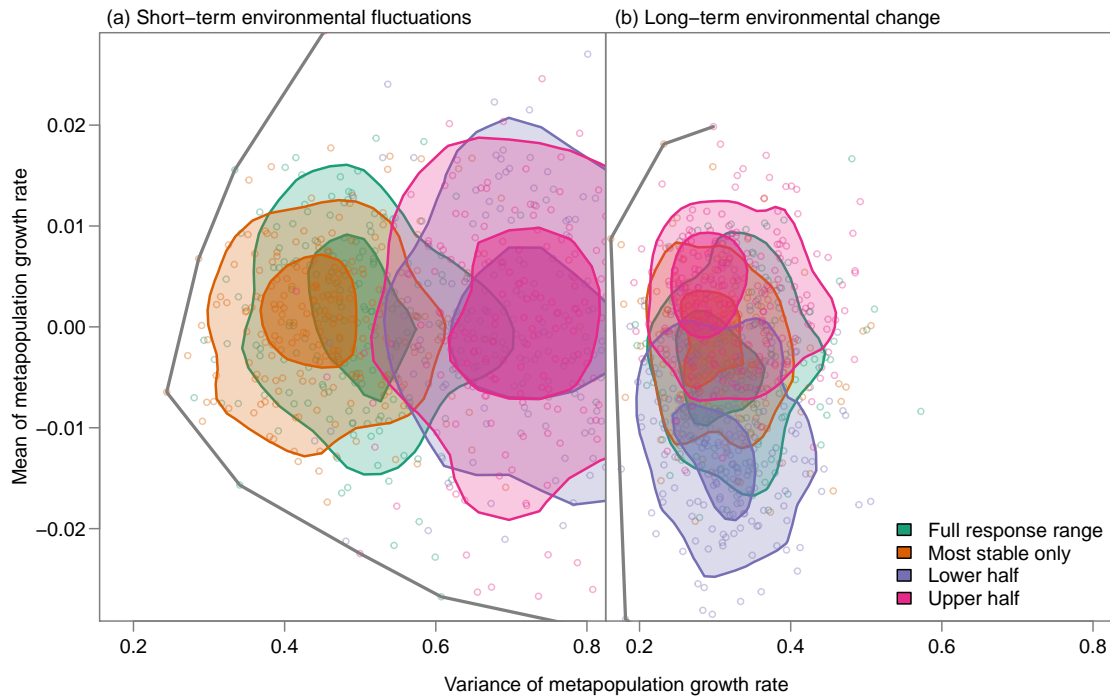
```

And plot the output:

```

cols <- RColorBrewer::brewer.pal(5, "Dark2")
xlim <- c(0.18, 0.80)
ylim <- c(-0.027, 0.027)
par(mfrow = c(1, 2))
par(las = 1, cex = 0.8, mar = c(0, 0, 0, 0), oma = c(4, 5.2, 1.5, .5),
  tck = -0.02, mgp = c(2, .6, 0))
plot_cons_plans(x_arma_sp$plans_mv, plans_name = plans_name_sp, cols = cols,
  add_all_efs = FALSE, xlim = xlim, ylim = ylim, add_legend = FALSE)
mtext("(a) Short-term environmental fluctuations", side = 3, line = 0.2, cex =
  0.8, adj = 0.05)
par(las = 0)
mtext("Mean of metapopulation growth rate", side = 2, line = 3, outer = FALSE,
  cex = 0.8)
par(las = 1)
plot_cons_plans(x_linear_sp$plans_mv, plans_name = plans_name_sp, cols = cols,
  add_all_efs = FALSE, xlim = xlim, ylim = ylim, y_axis = FALSE, add_legend =
  TRUE, legend_pos = "bottomright")
mtext("(b) Long-term environmental change", side = 3, line = 0.2, cex = 0.8,
  adj = 0.05)
mtext("Variance of metapopulation growth rate", side = 1, line = 2.25,
  outer = FALSE, cex = 0.8, adj = -1)

```



3 Numbers of populations conserved

Here, we'll show the effect of increasing the number of streams that conservation effort is distributed across. We'll conserve the populations without knowing how response diversity is distributed.

We'll set up the scenarios:

```
set.seed(1)
# In this version we start with a set amount of b and can split it up among many
# or invest it in a few
n_trials <- 200 # number of trials at each n conservation plan
num_pops <- c(2, 4, 8, 16) # n pops to conserve
b_conserve <- 2000 / num_pops
n_plans <- length(num_pops) # number of plans
w <- list()
for(i in 1:n_plans) { # loop over number conserved
  w[[i]] <- list()
  for(j in 1:n_trials) { # loop over trials
    w[[i]][[j]] <- matrix(rep(b_conserve[i], 16), nrow = 1)
    # conserve num_pops[i] populations; wipe out rest:
    w[[i]][[j]][-sample(1:16, num_pops[i])] <- 5
  }
}
plans_name_n <- paste(num_pops, "populations")
```

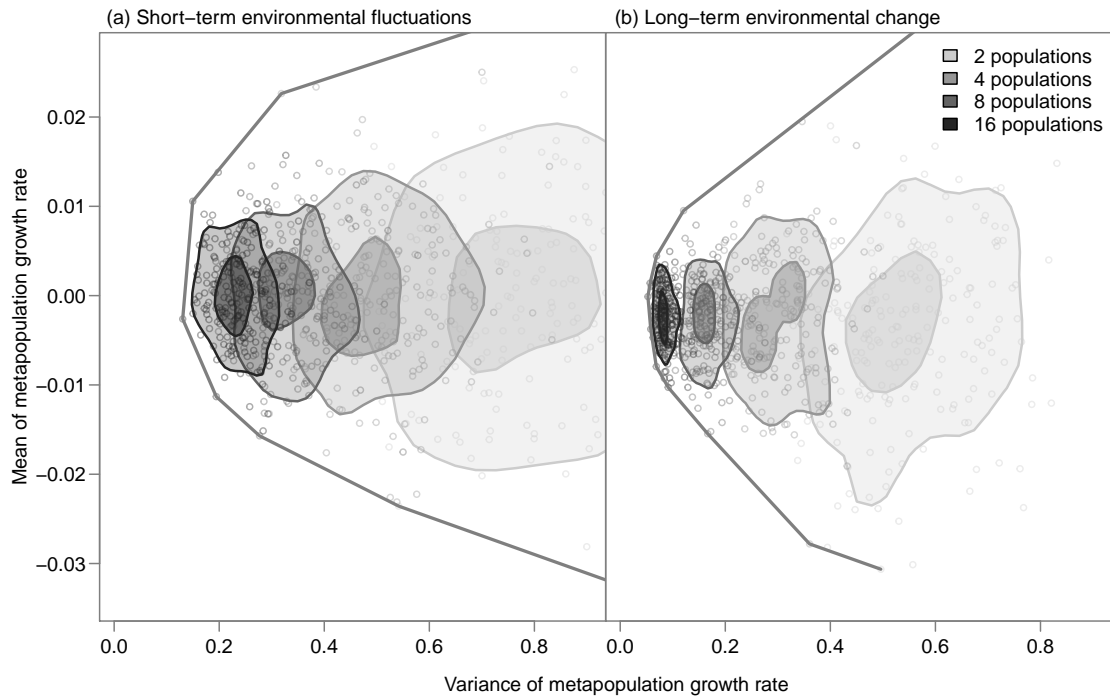
Now we'll run the simulations for the stationary climate variability and the non-stationary scenarios:

```
x_arma_n <- run_cons_plans(w, env_type = "arma", env_params =
  arma_env_params, max_a = thermal_integration(16),
  show_progress = FALSE)
x_arma_n$plans_port <- NULL # to save space
```

```
linear_env_params <- list(min_value = 15, max_value = 19, sigma_env = 0.001,
  start_t = 30)
x_linear_n <- run_cons_plans(w, env_type = "linear",
  env_params = linear_env_params, max_a = thermal_integration(16),
  show_progress = FALSE)
x_linear_n$plans_port <- NULL # save space
```

And plot the output:

```
cols <- RColorBrewer::brewer.pal(5, "Greys")[c(2:5)]
xlim <- c(0.008, 0.90)
ylim <- c(-0.034, 0.027)
par(mfrow = c(1, 2))
par(las = 1, cex = 0.8, mar = c(0, 0, 0, 0), oma = c(4, 5.2, 1.5, .5),
  tck = -0.02, mgp = c(2, .6, 0))
plot_cons_plans(x_arma_n$plans_mv, plans_name = plans_name_n, cols = cols,
  add_all_efs = FALSE, xlim = xlim, ylim = ylim, add_legend = FALSE)
mtext("(a) Short-term environmental fluctuations", side = 3, line = 0.2,
  cex = 0.8, adj = 0.05)
par(las = 0)
mtext("Mean of metapopulation growth rate", side = 2, line = 3,
  outer = FALSE, cex = 0.8)
par(las = 1)
plot_cons_plans(x_linear_n$plans_mv, plans_name = plans_name_n, cols = cols,
  add_all_efs = FALSE, xlim = xlim, ylim = ylim, y_axis = FALSE, add_legend = TRUE)
mtext("(b) Long-term environmental change", side = 3, line = 0.2, cex = 0.8,
  adj = 0.05)
par(xpd = NA)
mtext("Variance of metapopulation growth rate", side = 1, line = 2.25,
  outer = FALSE, cex = 0.8, adj = -1)
```



4 Reduction of habitat over time

Here we'll reduce the available habitat by a fixed quantity at every generation. The climate will follow a combination of short-term variability and long-term change.

Set up the scenarios:

```
set.seed(1)
n_trials <- 200 # number of trials at each n conservation plan
num_pops <- c(2, 4, 8, 12, 16) # n pops to conserve
b_conserve <- 2000 / num_pops
n_plans <- length(num_pops)
w <- list()
for(i in 1:n_plans) { # loop over number conserved
  w[[i]] <- list()
  for(j in 1:n_trials) { # loop over trials
    w[[i]][[j]] <- matrix(rep(b_conserve[i], 16), nrow = 1)
    # conserve num_pops[i] populations; wipe out rest:
    w[[i]][[j]][-sample(1:16, num_pops[i])] <- 5
  }
}
plans_name_n <- paste(num_pops, "populations")
cols <- RColorBrewer::brewer.pal(6, "Greys")[c(2:6)]
```

Run the simulations:

```
linear_arma_env_params <- list(min_value = 15, max_value = 19,  
  start_t = 30, mean_value = 16, ar = 0.1, sigma_env = 2, ma = 0)  
x_linear_arma_n <- run_cons_plans(w, env_type = "linear_arma", env_params =  
  linear_arma_env_params, max_a = thermal_integration(16), decrease_b = 0.85,  
  show_progress = FALSE)  
x_linear_arma_n$plans_port <- NULL # save space
```

Plot the output:

```
xlim <- c(0.08, 0.9)  
ylim <- c(-0.038, 0.028)  
par(las = 1, cex = 0.8, mar = c(0, 0, 0, 0), oma = c(4, 5.2, 1.8, .5),  
  tck = -0.02, mgp = c(2, .5, 0))  
plot_cons_plans(x_linear_arma_n$plans_mv, plans_name = plans_name_n, cols = cols,  
  add_all_efs = FALSE, xlim = xlim, ylim = ylim, add_legend = TRUE,  
  add_poly = TRUE, legend_pos = "bottomright")  
mtext("Reduction in stream flow", side = 3, line = .4,  
  cex = 0.8, adj = 0.05)  
mtext("Variance of metapopulation growth rate", side = 1, line = 2.25,  
  outer = FALSE, cex = 0.8)  
par(las = 0)  
mtext("Mean of metapopulation growth rate", side = 2, line = 3,  
  outer = FALSE, cex = 0.8)
```

