

# model description

Justin Pomeranz

1/26/2021

## Introduction

This model extends that of Zhou et al. 2013 which incorporates continuous predation and seasonal reproduction (typical life history of many aquatic organisms) of a 2-species predator-prey model.

We extend this model to include intraguild predation, in which a basal species B is consumed by two predators C, and P, where P (also called intraguild predator) can also consume C (also called intraguild prey).

## Model overview

Assume there are habitat patches,  $x$ , located across a landscape. Within each patch, three species may exist. A basal species B, a consumer species C, and a predator species P. In the absence of predation, the basal species population abundance is assumed to grow exponentially to to some population level,  $K$ , (logistic growth) based on some characteristic of the habitat patch (i.e. environmental condition, patch size).

The consumer species (also know as the intraguild prey) receives all of its energy from the basal species. The consumer species population varies in direct proportion to the density of B, following a type II functional response (Holling's disk equation). The consumer species is also predated upon by the predator species, P.

The predator species consumes both B and C, and responds to each following a type II functional response.

## Dispersal

The model starts with some initial nonzero population abundances for all three species in each patch. First, dispersal from all patches is accounted for. The number of emmigrants,  $E$ , for each species from each patch is estimated, Then, the number of emmigrants from patch  $i$  is divided into immigrants,  $I$ , for each patch  $j$ . The likelihood of an individual dispersal from patch  $i$  to  $j$  declines exponentially with increasing distance from patch  $i$  to  $j$ .

The probability that an individual disperses can be set with `p_dispersal`. A single value can be used for all trophic levels, or can be set for each individually.

---

## NOTE

The above formulation uses a single value of dispersal probability `p_dispersal`. Might be worth looking at a distribution of probabilities, such as  $p_{dispersal} \sim \text{Beta}(\alpha, \beta)$  to add stochasticity to this function.

---

## Predation

After dispersal, continuous predation occurs. The number of prey  $i$  eaten by consumer  $j$ ,  $W_{ij}$ , in a season is calculated as:

$$W_{ij} = \frac{\alpha_{ij} N_i N_j}{\beta_{ij} N_j + N_i}$$

where  $N_i$  is the population abundance of resource  $i$ ,  $N_j$  is the population abundance of consumer  $j$ , and  $\alpha_{ij}$  and  $\beta_{ij}$  are coefficients describing the functional response as the ratio of resources to consumers increases.  $\beta_{ij}$  describes how fast the predation curve reaches the asymptote of the type II functional response, and  $\frac{\alpha}{\beta}$  determines the location of the asymptote.

The total number of Basal species individuals eaten is the sum of the individuals eaten by C and P:

$$W_B = W_{BC} + W_{BP}$$

The predator species P, searches for and consumes both species in a patch. Its total effort is controlled by an  $\alpha_p$  and a  $\beta_p$  parameter. The preference parameter  $\delta$ , controls how much time it spends searching for the B resource, and  $1 - \delta$  is the effort spent searching for the C resource.

$$W_{BP} = \delta \left( \frac{\alpha_p B P}{\beta_p P + B} \right)$$

$$W_{CP} = (1 - \delta) \left( \frac{\alpha_p C P}{\beta_p P + C} \right)$$

## Survival

After accounting for predation, the remaining individuals are reduced by the base survival probability:

$$B' = S_0(B - W_B)$$

$$C' = S_0(C - W_{CP})$$

$$P' = S_0 P$$

## ##Reproduction

Finally, those individuals remaining ( $X'_i$ ) reproduce according to the following:

$$B_{t+1} = \frac{r_{max}}{1 + b B'} B'$$

$$C_{t+1} = e_{bc} W_{BC} C'$$

$$P_{t+1} = (e_{bp} W_{BP} + e_{cp} W_{CP}) P'$$

where  $r_{max}$  is the maximum per-capita reproduction rate of the basal species and  $b = \frac{r_{max}-1}{K}$  determines the asymptotic relationship of population growth, where  $K$  is carrying capacity. In the absence of predation, population growth of the basal species follows a logistic growth relationship. Note the use of  $X_i$  in determining the predation effects (terms inside of the parentheses) and  $X'_i$  to estimate reproduction in the next generation. This can be viewed as a time lag, which incorporates continuous predation through the season, and synchronous reproduction only for the surviving individuals.

For the consumer species (C and P)  $e_{ij}$  is the efficiency of converting resource  $i$  into new consumer  $j$ . the  $e_{ij}$  parameter can be viewed as the relative “value” of a resource to a consumer. For example, we may assume that  $e_{cp} \geq e_{bp}$  if the resource C is of higher quality than resource B. This may translate to a higher search

effort for resource C. Whereas if the quality of both resources is similar,  $e_{cp} \approx e_{bp}$ , than the predator P may equally split its effort.

The product of  $e_{ij}\alpha_{ij}$  is the maximum reproductive rate of consumer  $j$  on resource  $i$ . The carrying capacity of consumer  $j$  on resource  $i$  is  $\frac{N_i(e_{ij}\alpha_{ij}-1)}{\beta_{ij}}$  when  $N_i$  is constant

---

## NOTE

An alternative formulation discards the  $\delta$  parameter and uses individual parameters for  $e_{ij}$ ,  $\alpha_{ij}$  and  $\beta_{ij}$  for all pairwise combinations (i.e.  $3 * (e + \alpha + \beta) = 9$  parameters). This was my original formulation, and is retained in `igp_sim2()` function.

The `igp_sim()` function, demonstrated here, uses the  $\delta$  preference parameter.

---

## Time $t + 1$

To ensure that the number of individuals in time  $t + 1$  is an integer and to include stochasticity in our model, the population abundance for  $t + 1$  is drawn from a Poisson distribution:

$$X_i \sim \text{Poisson}(\lambda)$$

where  $\lambda$  is a vector of abundances at  $t+1$ .

$$\vec{\lambda} = B_{t+1}, C_{t+1}, P_{t+1}$$

## Disturbance

We also include a disturbance probability `p_disturb`, which determines the probability of a disturbance occurring in patch  $X$  at time  $t$ . If a disturbance occurs, the abundance of each species is reduced by the value set with `mag_disturb` e.g., `mag_disturb = 0.25` a disturbance in a patch reduces the abundance of all species by 25%.

Disturbance correlation. The correlation value can be set such that if one patch undergoes a disturbance as described above, then patches located nearby (euclidean distance, same branch, etc.) are also more likely to undergo that disturbance. This can be viewed as a flood disturbance through a river branch, or a large storm even across a 2-dimensional landscape.

*NOTE:* correlation is not yet implemented.

We also include a disturbance probability `p_disturbance` which reduces the abundance of each species present in patch  $x$  at time  $i$  by some constant value (e.g., if disturbance occurs, populations are reduced by 50%).

Disturbance correlation. The correlation value can be set such that if one patch undergoes a disturbance as described above, then patches located nearby (euclidean distance, same branch, etc.) are also more likely to undergo that disturbance. This can be viewed as a flood disturbance through a river branch, or a large storm even across a 2-dimensional landscape.

## Resource extinction

An explicit assumption of our model is that the presence of a consumer is dependent on the presence of at least one of its' resources. Therefore, if the population abundance of B is equal to 0 at the beginning of a timestep, we force the abundances of C and P to also equal zero. Note that C and P can become locally extinct and the simulation will continue as normal, and any secondary extinctions are the direct result of trophic dynamics.

## FCL: Food Chain Length

There are 5 possible outcomes, and their corresponding food chain lengths

- all species are extinct FCL = 0
- B only FCL = 1
- B present FCL = 1
- B + C FCL = 2
- B + P FCL = 2.5
- B + C + P FCL = 3

## Environment

The carrying capacity,  $K$ , is a function which is determined by ecosystem area (i.e., number of patches upstream) or some environmental condition. If determined by environmental condition, patches can either be randomly assigned a value, or can be spatially auto-correlated, such that nearby patches are more likely to have similar environmental values.

*NOTE:* this is a work in progress.

## Simulation function `igp_sim()`

The main function is `igp_sim()`, which simulates trophic dynamics between three species in patches. By default, the species are classified as basal B, consumer C, and predator P.

A distance matrix describing how patches are oriented spatially can be included with the `dist_mat` argument, or otherwise patches are assumed to be randomly distributed in a 10x10 square.

Carrying capacity of the basal species can be determined by the `k` argument. If one value is supplied, it is assumed that carrying capacity is the same for all patches. You can set `k` for each patch individually with a vector of length = `n_patch` in the `k` argument. In the **future**, a function will be implemented to calculate carrying capacity based on number of patches upstream, or randomly across all patches (possibly other implementations). `r_max` controls the maximum reproduction rate for the basal species, B.

The initial population abundances can be set with the `n_0` argument. The default is to start B, C and P at  $(0.8*k, 0.5*k, 0.25*k)$  respectively. If one value is supplied, all three species start at the same value, or a vector of length 3 can be used to set the starting abundances for each individually.

The probability that an individual will disperse from a patch is set with the `p_dispersal` argument, and can take values from 0 to 1. If one value is supplied it is applied to all three species, otherwise a vector of length 3 can be used to set probabilities for species B, C and P (in that order). The `theta` argument determines how quickly dispersal ability declines with increasing distance (higher values of `theta` means individuals will disperse shorter distances). Currently, only one value of `theta` is applied to all three species.

Trophic dynamics are determined with the arguments `eij`, `alphaij` and `betaij`, where *ij* describes the relationship between resource *i* and consumer *j*. `eij` refers to the conversion efficiency, and `alphaij` and

`betaij` are parameters controlling the type II functional response. `P_pref` controls the preference parameter  $\delta$  of resource B over C for predator P. When  $\delta < 0.5$ , the predator prefers the C resource, and When  $\delta > 0.5$  the predator prefers resource B, and When  $\delta = 0.5$  the predator does not prefer either resource.

The `s0` argument controls the base survival probability. Currently it sets the same survival probability for all species.

The `p_disturb` argument sets the probability that any given patch experiences a disturbance in a given time step. This value should be set very low (i.e.,  $1e-4$ ) due to the high number of rolls in a given simulation `n_patch * t`. The `mag_disturb` argument controls the strength of disturbance, and reduces the population of all species by that value. i.e. `mag_disturb = 0.25` reduces the population by 25%. `mag_disturb = 1` removes 100% of the population e.g. local extinction.

`t` controls the number of timesteps to run the simulation for.

## default parameter values

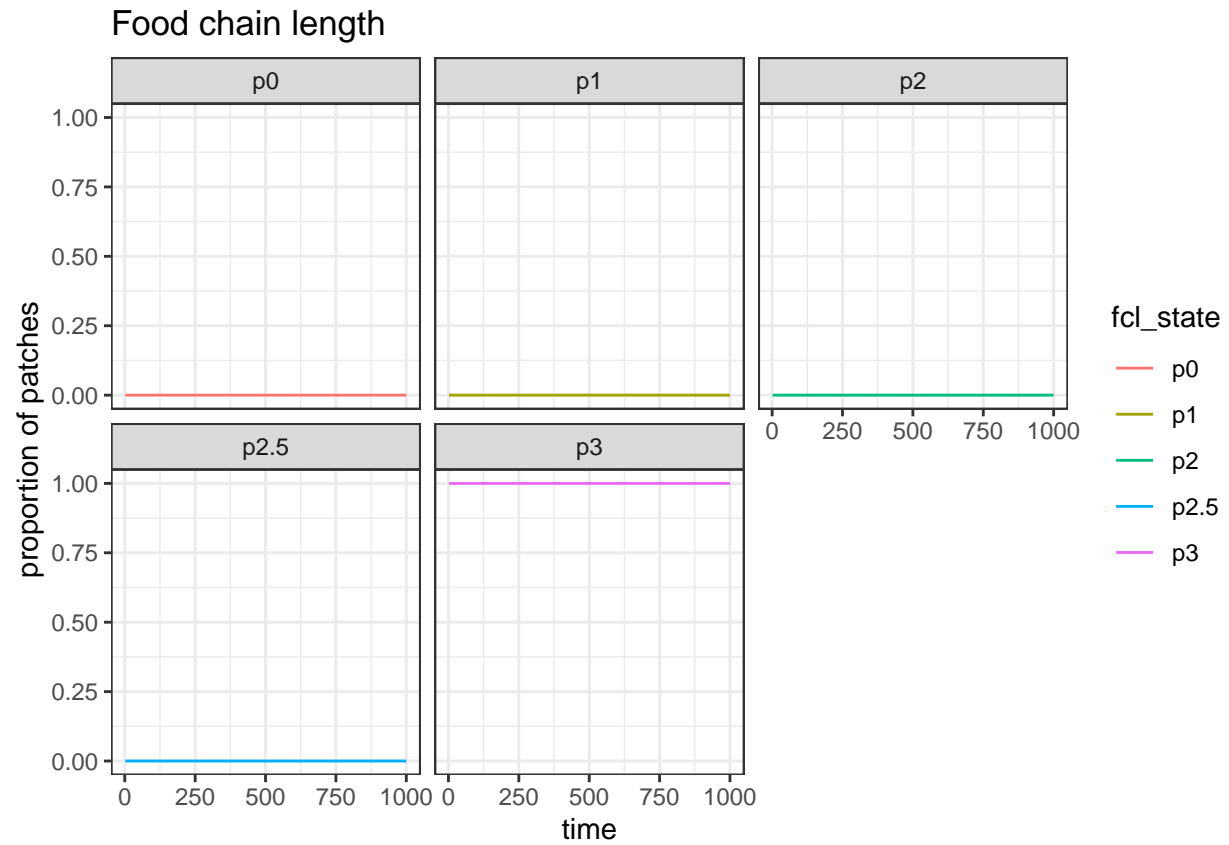
Carrying capacity: `k = 500`, Probability of dispersal: `p_dispersal = 0.1`, Strength of distance-dependence on dispersal: `theta = 1`, Basal species B max reproduction rate: `r_max = 2.5`, All resources are of equal quality: `ebc = ebp = ecp = 2`, Consumer species C and P have same equal search efforts: `alphabc = alphap = 4`, and `betabc = betap = 20`, Predator P prefers resource C: `P_pref = 0.25` (values  $> 0.5$  predator prefers B). The base survival rate is: `s0 = 0.75`, Probability of a disturbance happening a given patch per time step: `p_disturb = 1e-4`, Disturbances reduce abundance by 50%: `mag_disturb = 0.5`, Time steps: `t = 1000`,

## visualization

The `plot_{descriptor}` arguments control which plots will be displayed.

## Proportion of patches with food chain length $S$ at each time step:

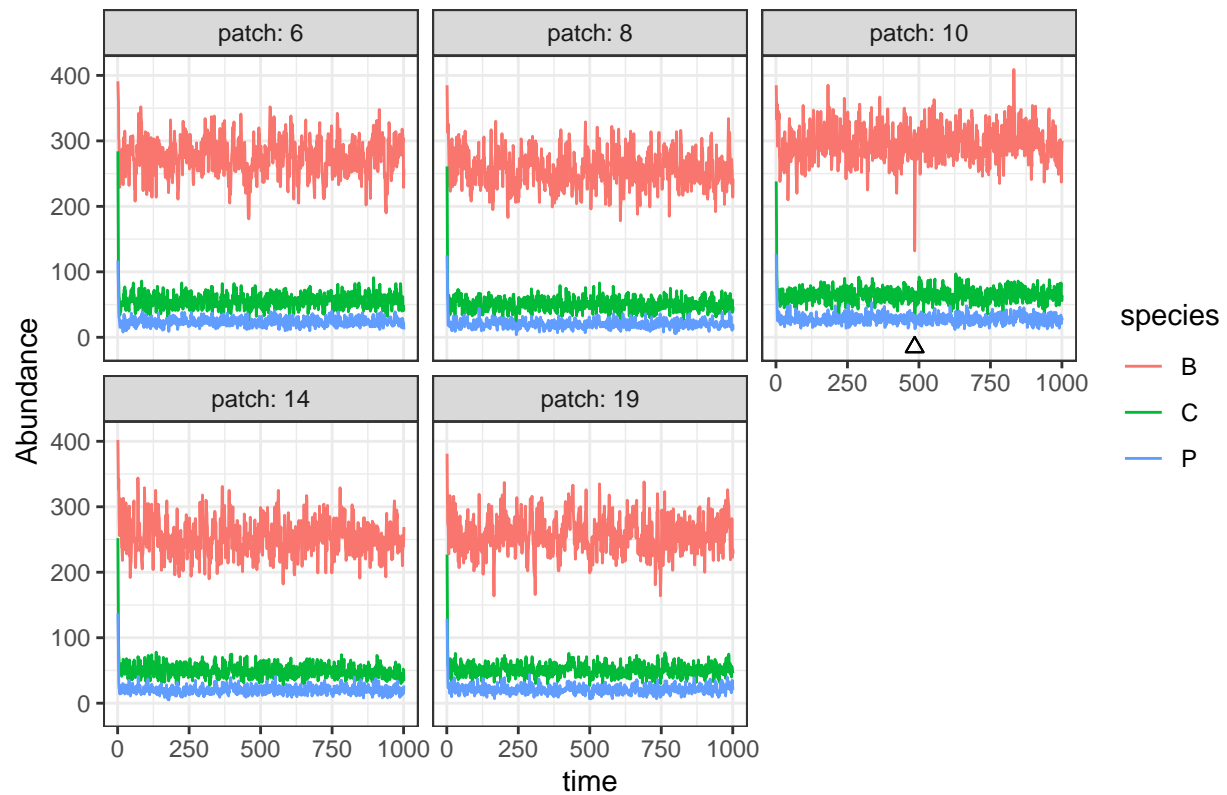
```
set.seed(1222) # for reproducibility
d <- igp_sim(plot_disturbance = FALSE,
             plot_fcl = TRUE,
             plot_patch_dynamics = FALSE)
```



### Dynamics of 5 random patches

```
set.seed(1222)
d <- igp_sim(plot_disturbance = FALSE,
             plot_fcl = FALSE,
             plot_patch_dynamics = TRUE)
```

## Dynamics for 5 random patches

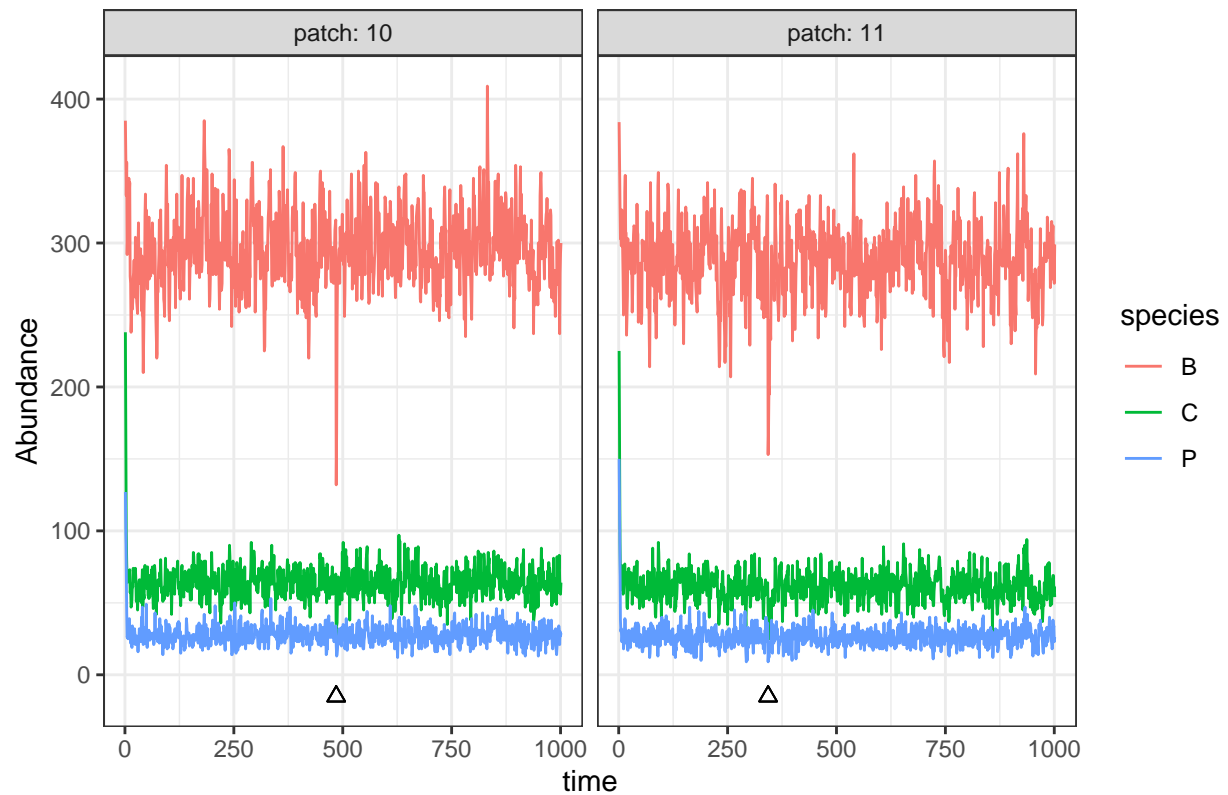


## dynamics of all patches which experienced a disturbance:

(if no disturbances occur, currently throws an error. Maybe remove this from main function and add it as a separate function to run on the output of `igp_sim()`?)

```
set.seed(1222) # to ensure that at least one disturbance occurs
d <- igp_sim(plot_disturbance = TRUE,
             plot_fcl = FALSE,
             plot_patch_dynamics = FALSE)
```

## All patches experiencing disturbances



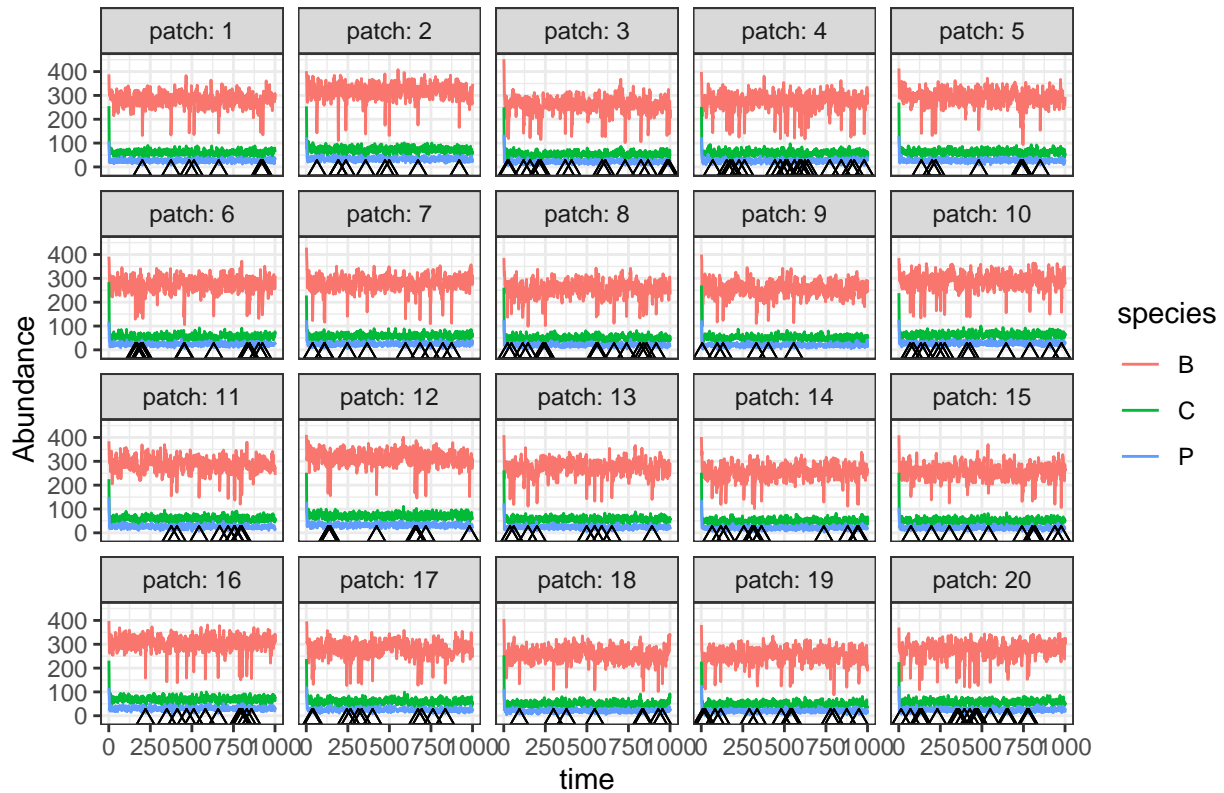
When a disturbance occurs at time step =  $t$ , it is displayed with a triangle below the x axis.

Increasing the frequency of disturbance makes this clear:

```
set.seed(1222)
d <- igp_sim(p_disturb = 0.01,
             plot_disturbance = TRUE,
             plot_fcl = FALSE,
             plot_patch_dynamics = FALSE)
```



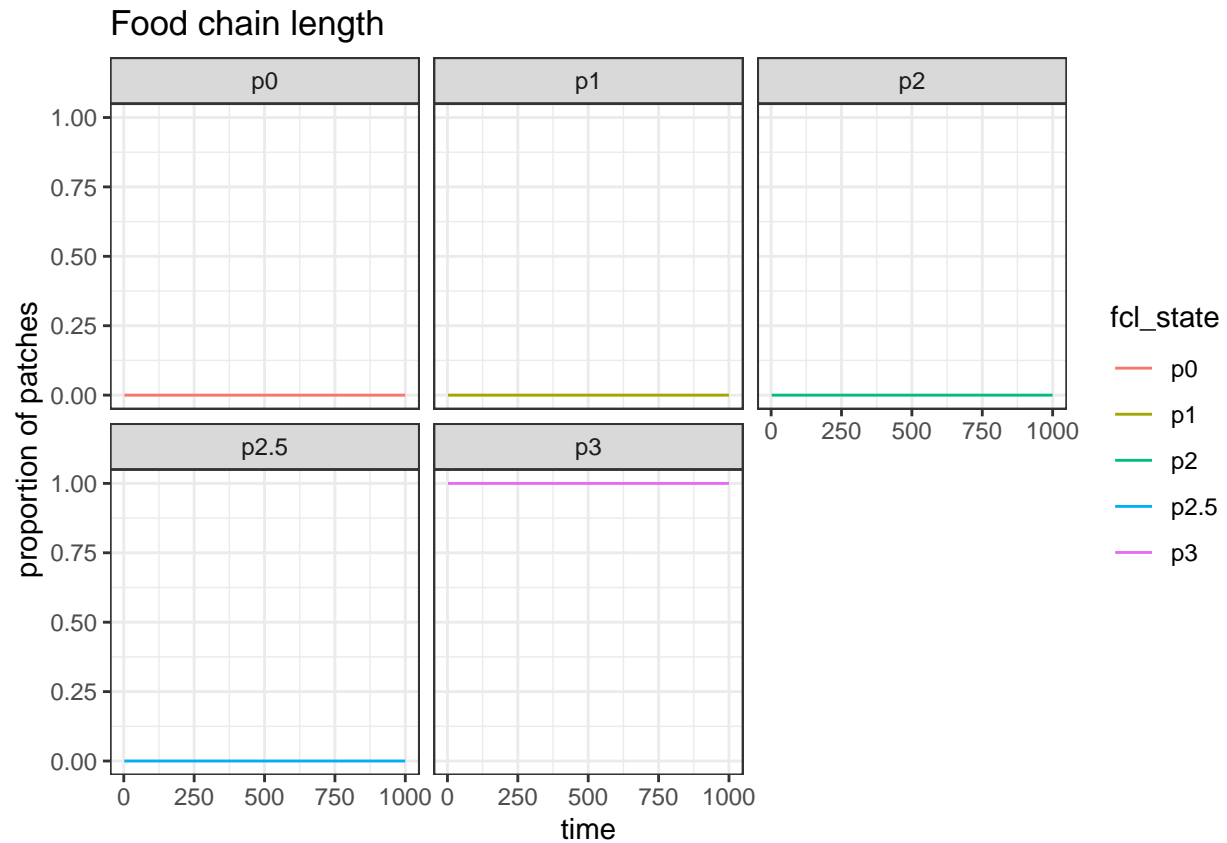
## All patches experiencing disturbances



## Customize settings

The arguments controlling the trophic dynamics are currently set to maximize stability and coexistence of all three species (parameter values based on the results of Zhou et al. 2013). This can be seen if we turn off dispersal and reduce disturbance probability

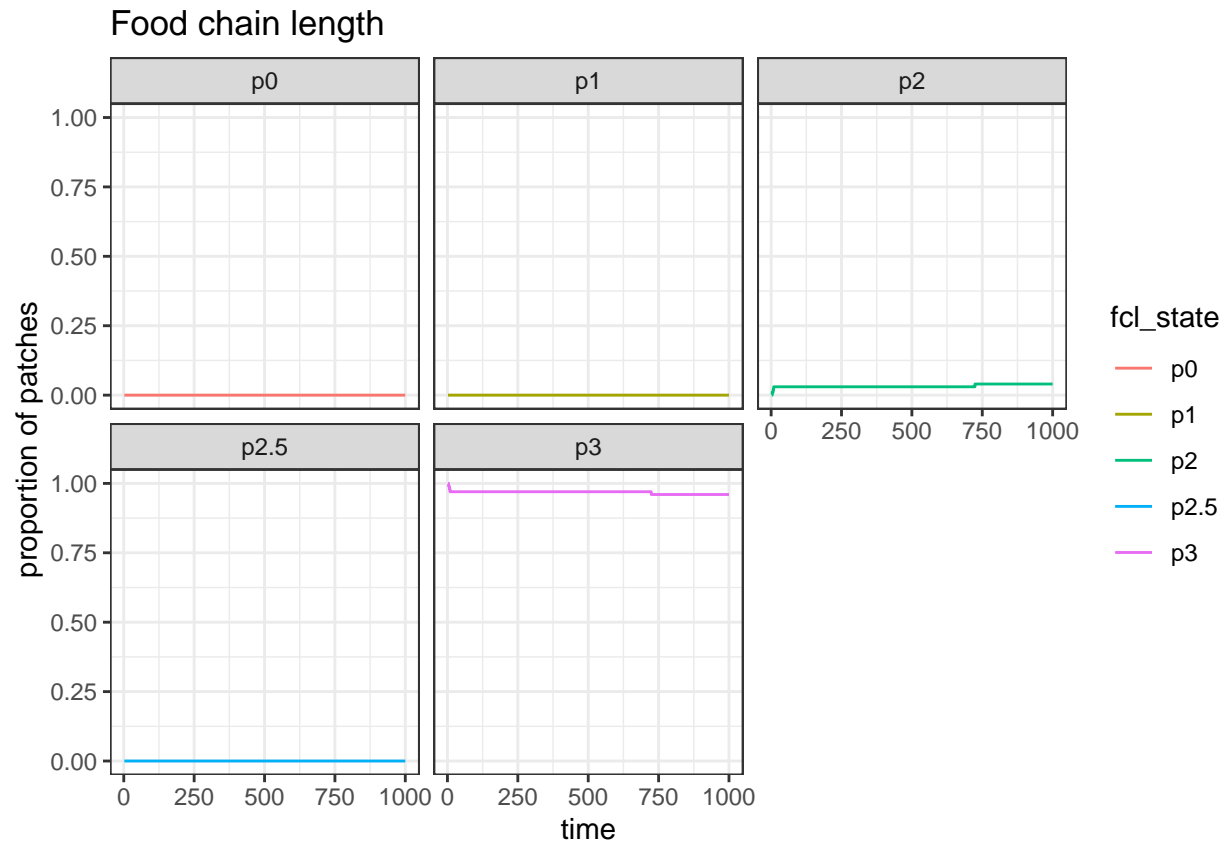
```
set.seed(1222) # for reproducibility
d <- igp_sim(n_patch = 100,
             p_dispersal = 0,
             mag_disturb = 0, # disturbance does not reduce population
             plot_disturbance = FALSE,
             plot_patch_dynamics = FALSE)
```



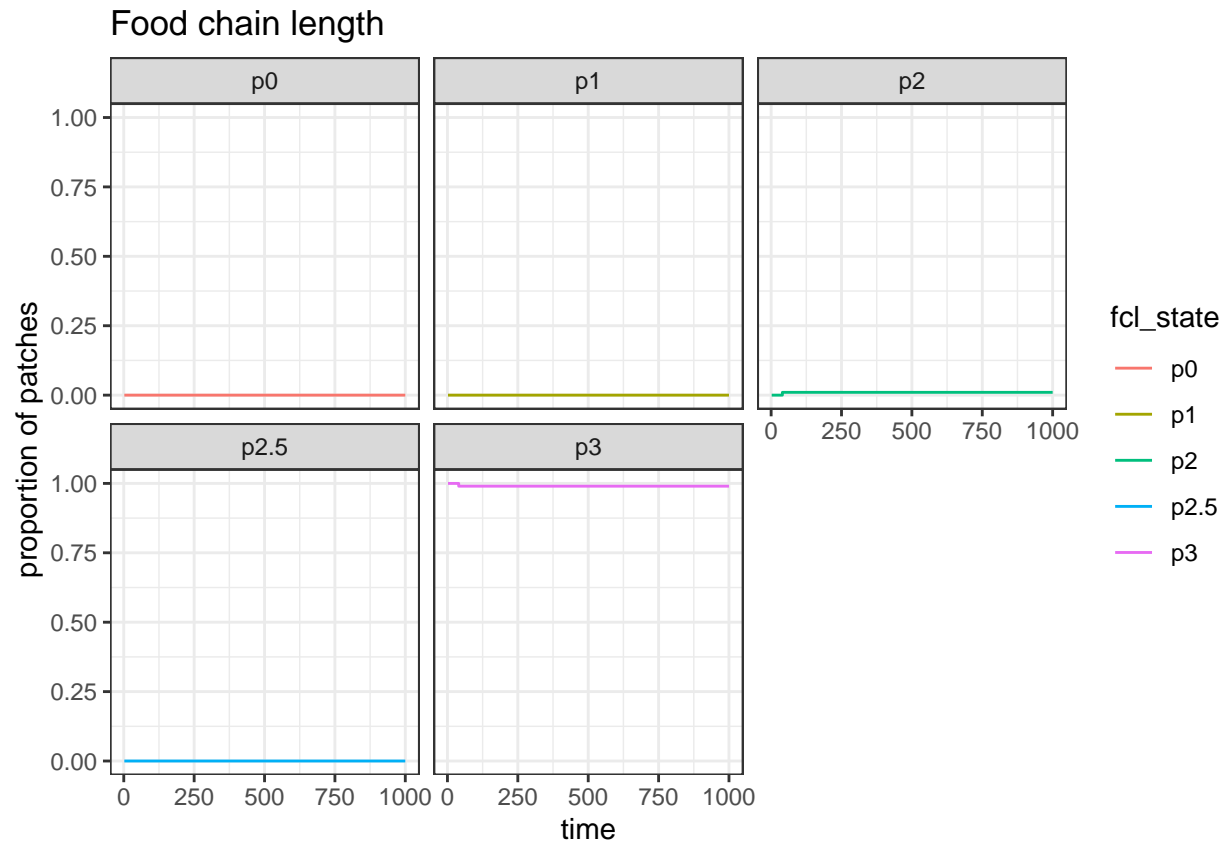
As we can see, the proportion of patches with FCL\_state = 3 (all species present) is 100%.

This is relatively insensitive to the starting abundances. (Starting abundances above = 400, 250, and 125, respectively.)

```
set.seed(1222)
d <- igp_sim(n_0 = 100, # start all at 100
            n_patch = 100,
            p_dispersal = 0,
            mag_disturb = 0, # disturbance does not reduce population
            plot_disturbance = FALSE,
            plot_patch_dynamics = FALSE)
```

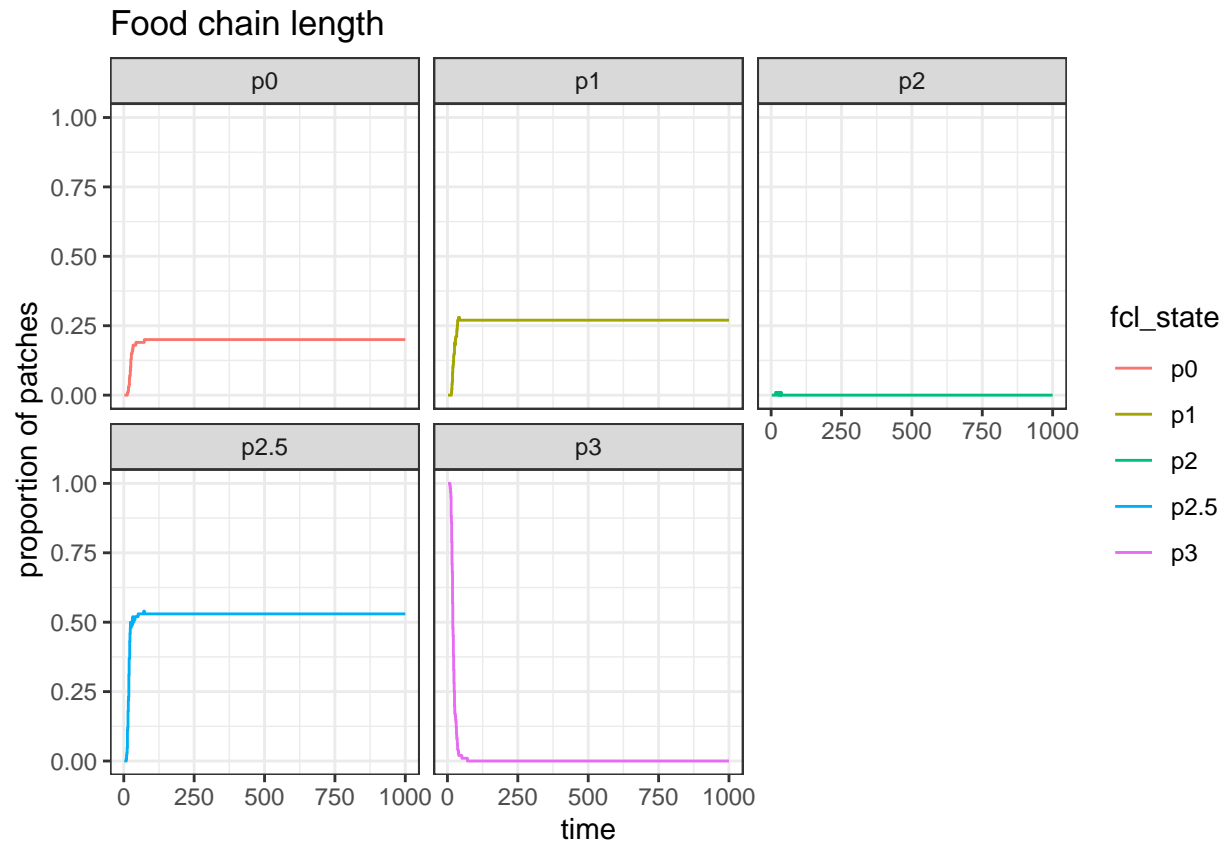


```
set.seed(1222)
d <- igp_sim(n_0 = c(100, 200, 50),
             n_patch = 100,
             p_dispersal = 0,
             mag_disturb = 0, # disturbance does not reduce population
             plot_disturbance = FALSE,
             plot_patch_dynamics = FALSE)
```



Keeping dispersal off and disturbance off, we can modify the strength of the trophic dynamics. A key point that increases stability, is ensuring that C out competes P for their shared resource (weak IGP). We can increase probability of extinction if we make P a stronger competitor

```
set.seed(1222) # for reproducibility
d <- igp_sim(n_patch = 100,
  p_dispersal = 0,
  mag_disturb = 0, # disturbance does not reduce population
  alphap = 10, # increase search effort
  P_pref = 0.75, # P mainly focuses on B
  plot_disturbance = FALSE,
  plot_patch_dynamics = FALSE)
```



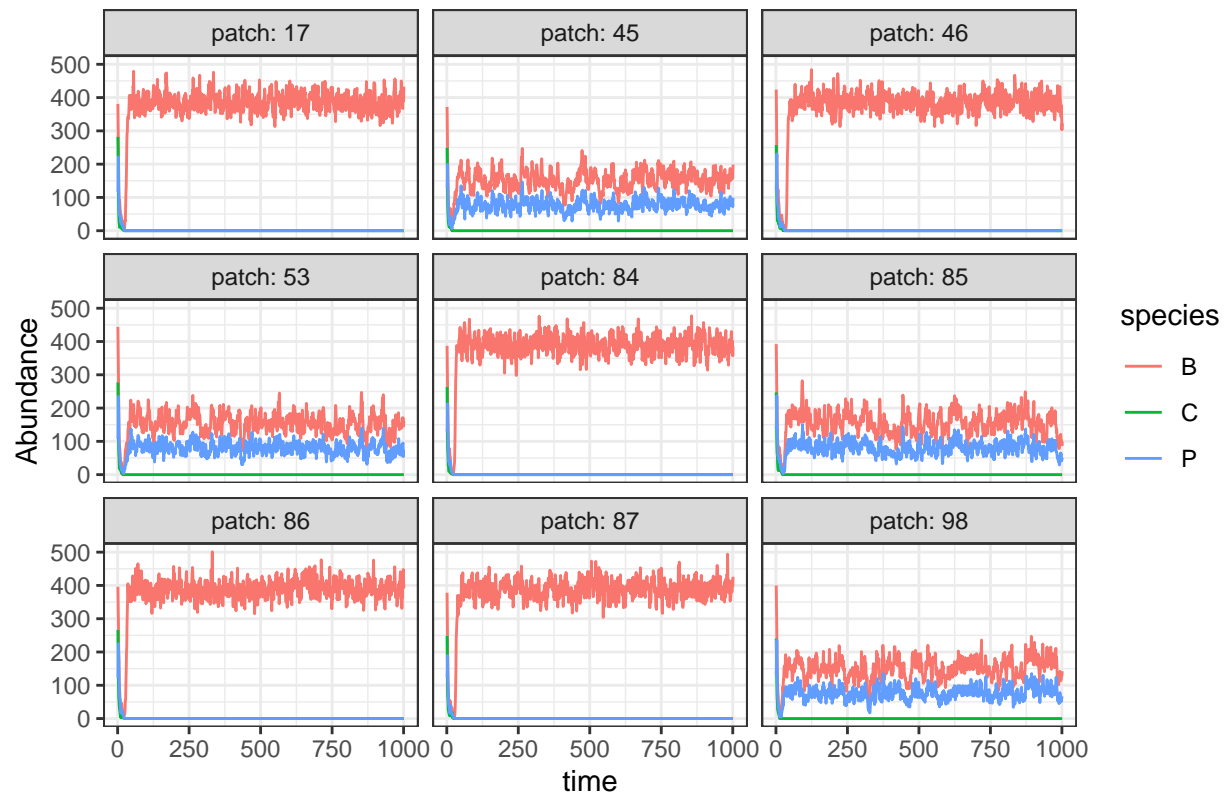
FCL\_state = 2 → has species B and C [no patches] FCL\_state = 2.5 → has species B and P, and C is removed.

Based on this, we can see that Species C is extinct in all patches, ~50% of patches have B+P, and ~25% have no or only B species present.

Plot of patch dynamics for 9 random patches:

```
set.seed(1222)
d$sp_dynamics %>%
  filter(patch %in% sample(1:100, 9)) %>%
  ggplot(
    aes(y = value,
        x = time,
        color = species)) +
  geom_line() +
  facet_wrap(~patch, labeller = label_both) +
  labs(y = "Abundance",
       title = "Dynamics for 9 random patches") +
  theme_bw() +
  NULL
```

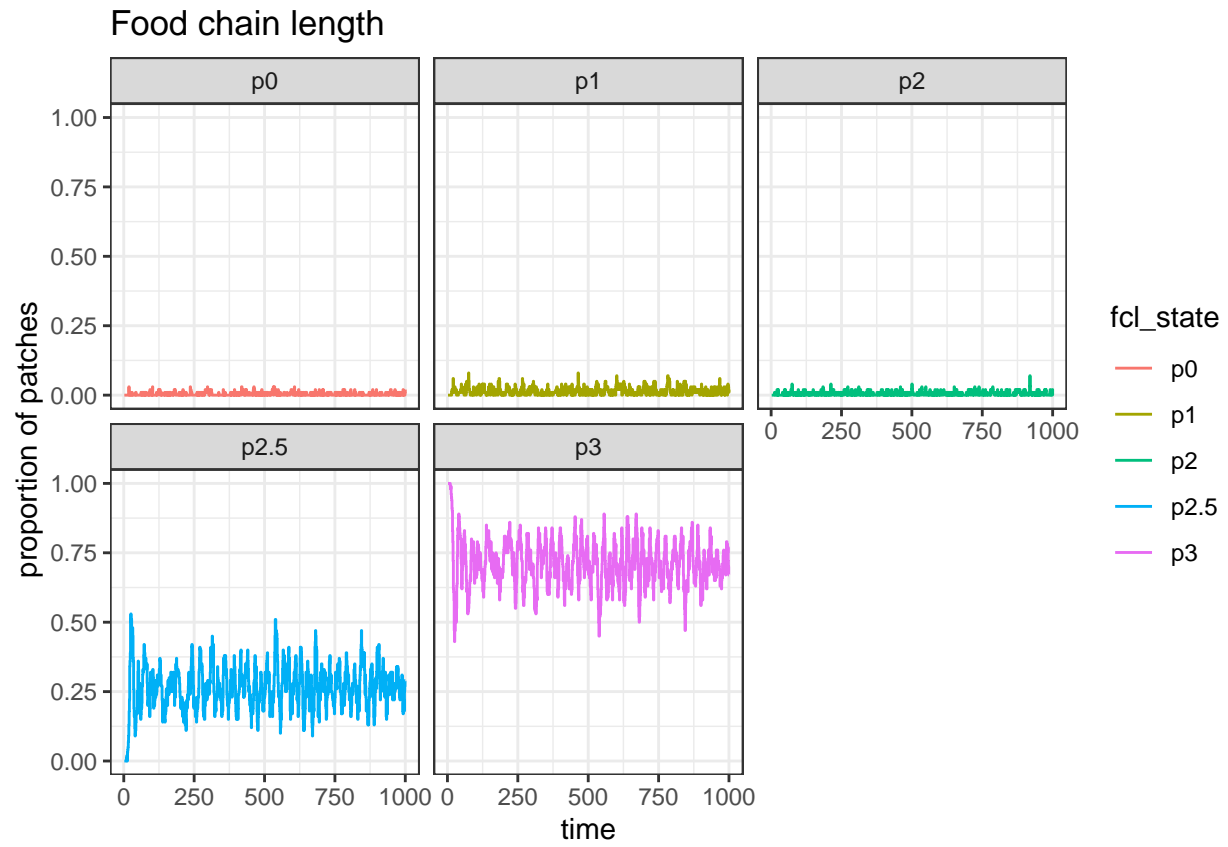
## Dynamics for 9 random patches



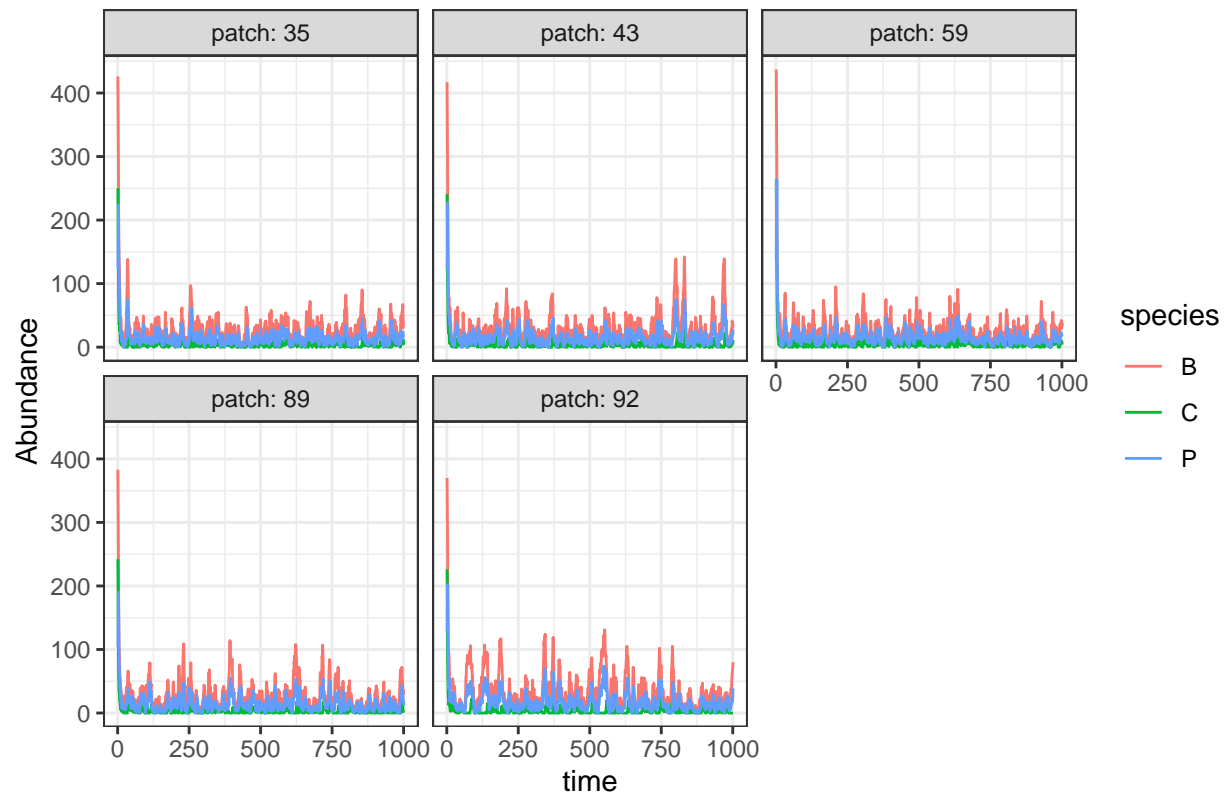
Here we can see that the green species (C) is extinct, and the blue species (P) also sometimes goes extinct.

## Turning on dispersal

```
set.seed(1222)
d <- igp_sim(n_patch = 100,
  p_dispersal = 0.1, # 10% individuals leave per time
  mag_disturb = 1, # disturbance effect
  alphap = 10, # increase search effort
  P_pref = 0.75, # P mainly focuses on B
  plot_disturbance = FALSE,
  plot_patch_dynamics = TRUE,
  plot_fcl = TRUE)
```



## Dynamics for 5 random patches



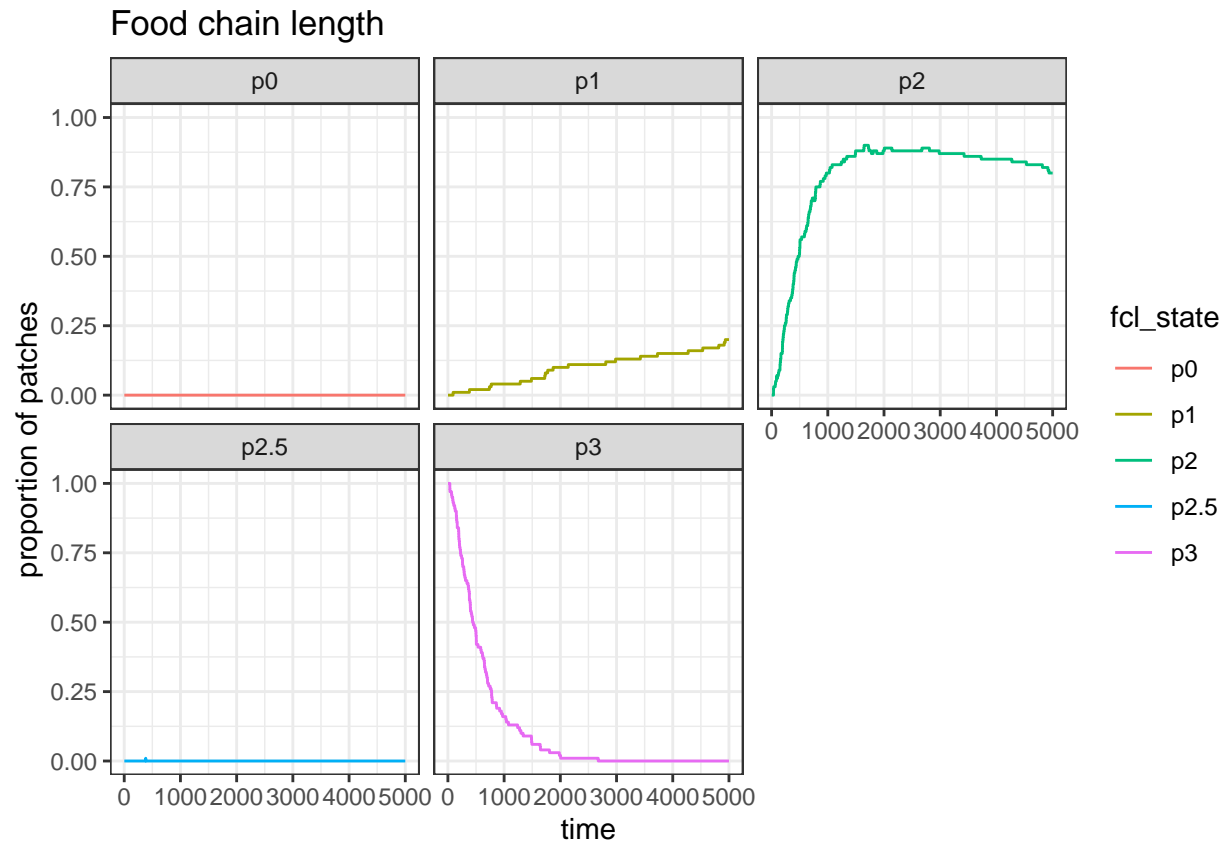
Dispersal is able to “rescue” most of the patches with C going locally extinct and then recolonizing. This results in patches oscillating between  $FCL=2.5$  (B+P) and  $FCL=3$  (B+C+P)

## Disturbance

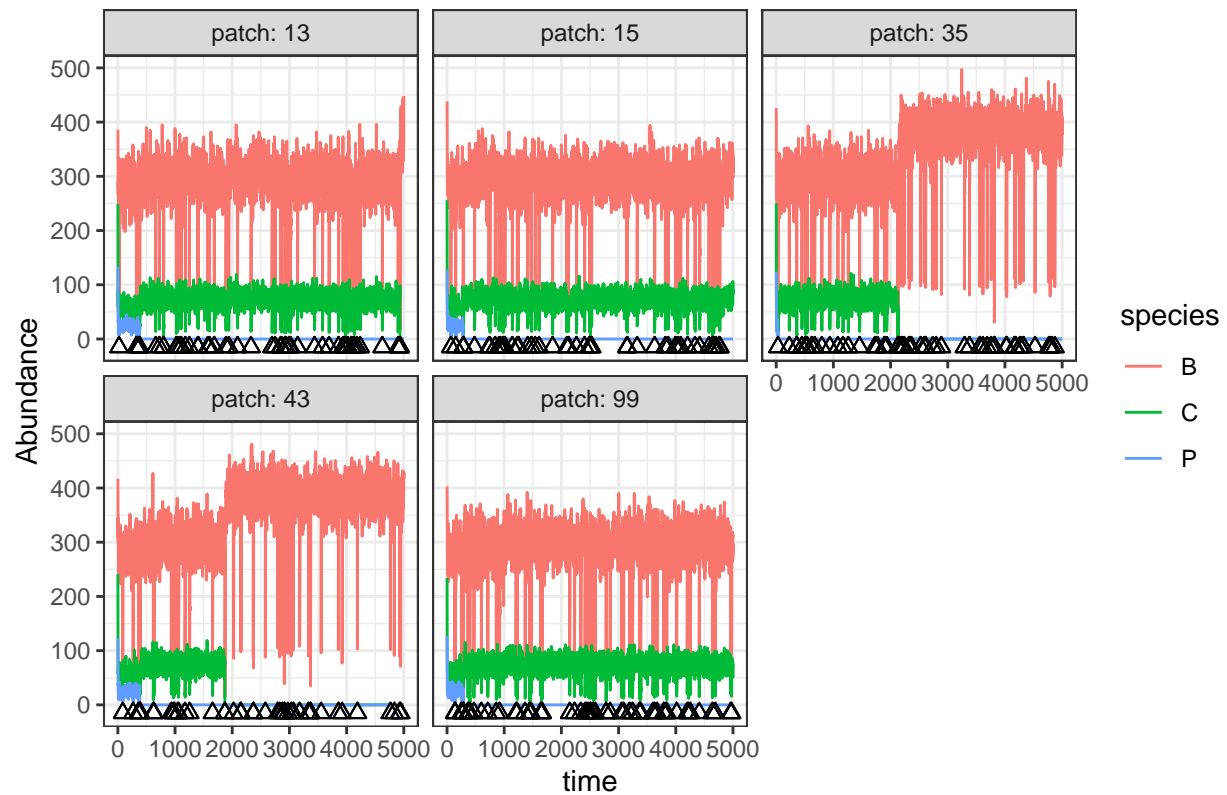
Increasing the frequency and magnitude of disturbance with no dispersal causes shorter food chains:

```
set.seed(1222)
d <- igp_sim(n_patch = 100,
  p_dispersal = 0, # no dispersal
  p_disturb = 1e-2, # increasing disturbance frequency
  mag_disturb = 0.75, # reduce population by 75%
  plot_disturbance = FALSE,
  plot_patch_dynamics = TRUE,
  plot_fcl = TRUE,
  t = 5000) # increasing timesteps to see pattern
```



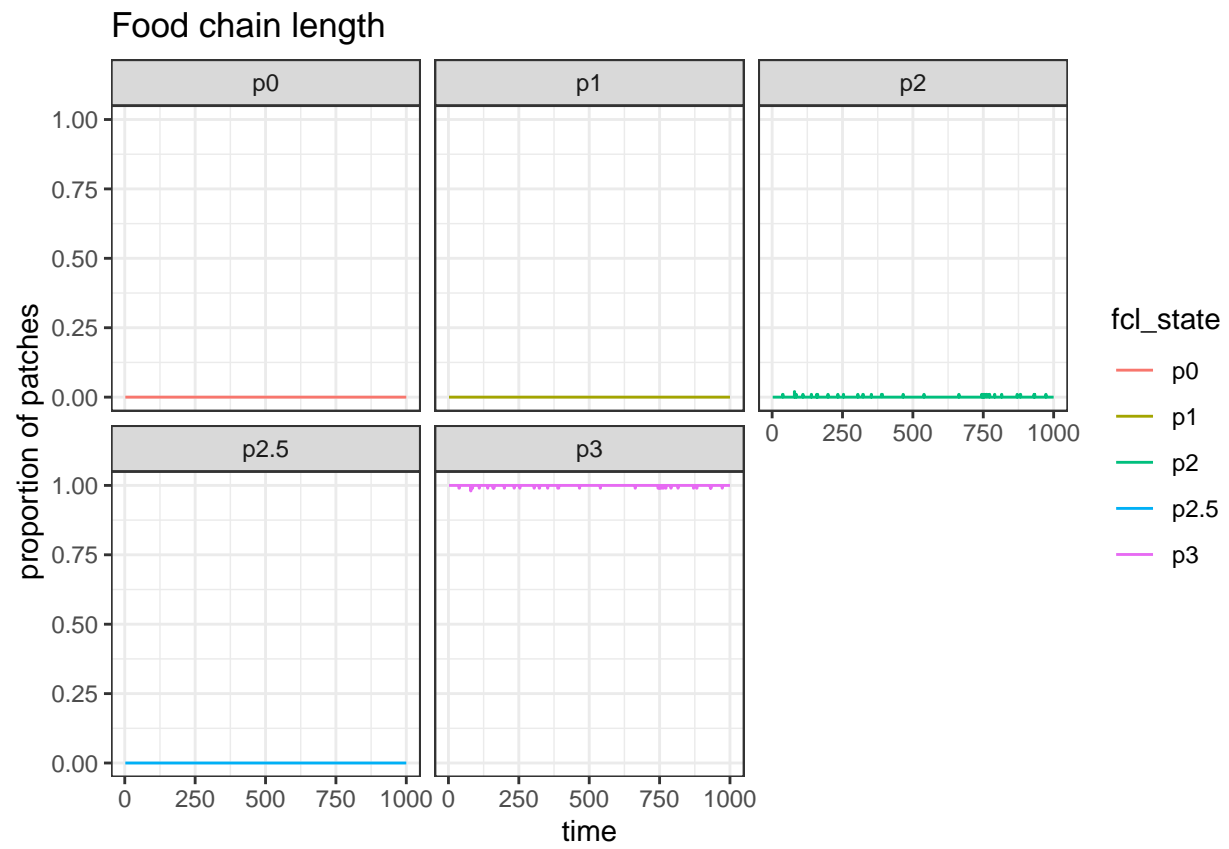


## Dynamics for 5 random patches

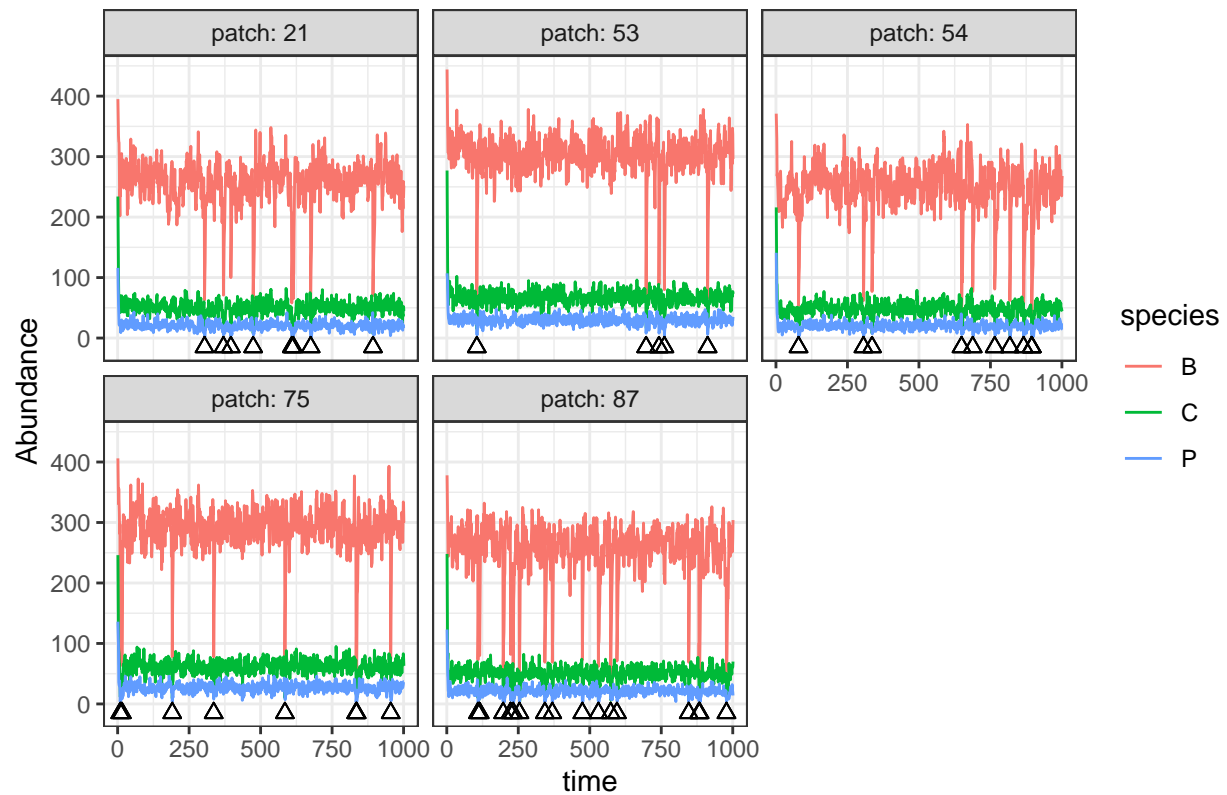


As time progresses, higher trophic levels are continually lost in the absence of dispersal.

```
set.seed(1222)
d <- igp_sim(n_patch = 100,
             p_dispersal = 0.1, # Dispersal back on
             p_disturb = 1e-2, # increasing frequency
             mag_disturb = 0.75, # reduce population by 75%
             plot_disturbance = FALSE,
             plot_patch_dynamics = TRUE,
             plot_fcl = TRUE)
```



## Dynamics for 5 random patches

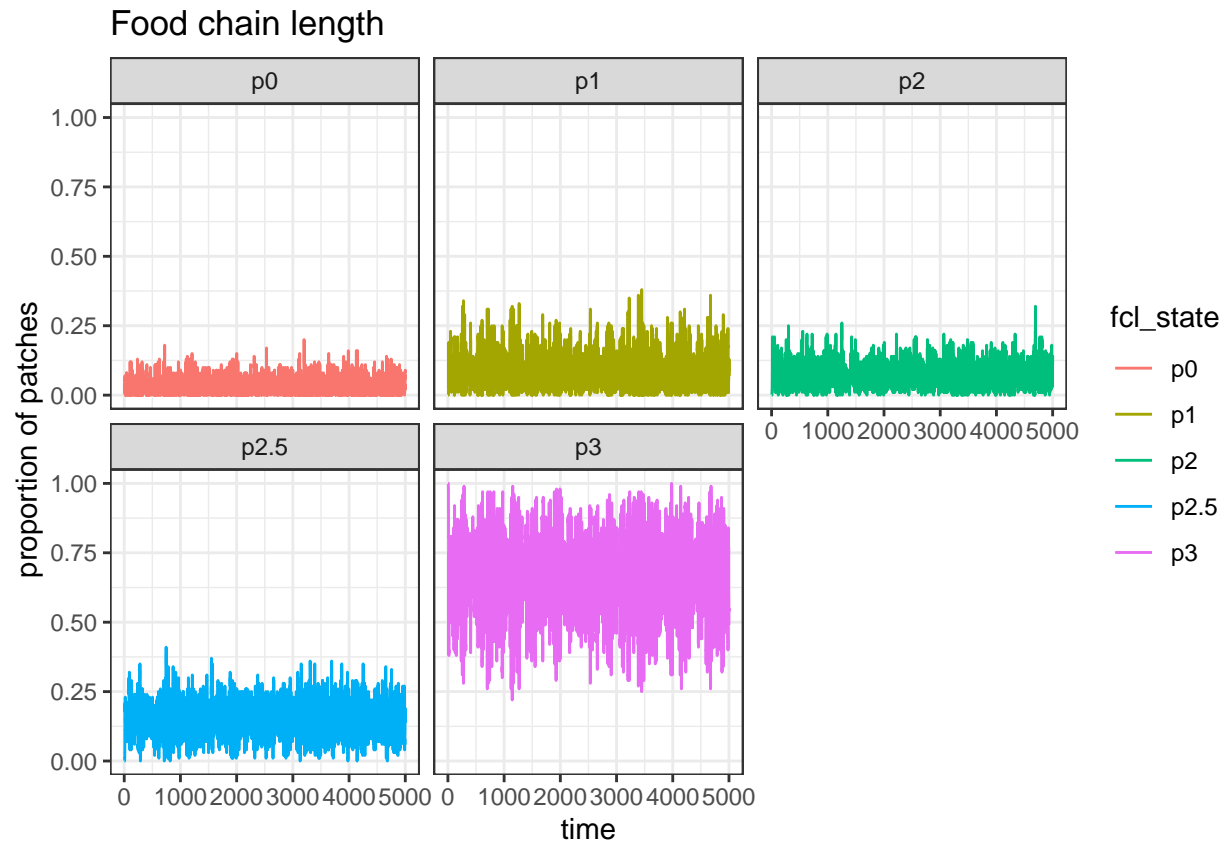


With parameters supporting coexistence and dispersal on, we can see that disturbance generally does not eradicate species (but see how abundance of red generally drops significantly in response to disturbance event (triangle)).

## Trophic dynamics, Disturbance, dispersal

Increase alpha (higher reproduction) and decrease beta (higher K) creates chaotic dynamics

```
set.seed(1222)
d <- igp_sim(n_patch = 100,
             p_dispersal = 0.1, # dispersal is on
             mag_disturb = 0.75, # reduce population by 75%
             alphabc = 6,
             alphap = 8,
             betabc = 15,
             betap = 15,
             P_pref = 0.5,
             plot_disturbance = FALSE,
             plot_patch_dynamics = TRUE,
             plot_fcl = TRUE,
             t = 5000)
```



## Dynamics for 5 random patches

