Supplement for: "Predicting the Unpredictable: Using replay experiments to disentangle how evolutionary outcomes are altered by adaptive momentum"

Austin J. Ferguson, Charles Ofria, Clifford Bohm

2024-06-06

Contents

1 Introduction **5**

2 Replaying disequilibrium populations **7**
2.1 Dependencies 7
2.2 Populations that did not cross 8
2.3 Populations that crossed once 15
2.4 Populations that crossed twice 21

3 Replaying equilibrium populations **25**
3.1 Dependencies 25
3.2 Populations that did not cross 26
3.3 Populations that crossed once 32
3.4 Populations that crossed twice 39

4 Replaying shuffled populations **41**
4.1 Dependencies 41
4.2 Populations that crossed once 42

Chapter 1

Introduction

Here we show all additional graphs that would not fit in the paper. Please use the navigation on the left to view the different sections.

For data, source code, and analyses, please refer to the GitHub repo: https://github.com/FergusonAJ/alife_2024_replaying_adaptive_momentum

Chapter 2

Replaying disequilibrium populations

The paper shows fine-grained replays for three replicates that were experiencing adaptive momentum (one that failed to cross, one that crossed one valley, and one that crossed two valleys).

Prior to these fine-grained replays, we replayed 24 populations at with less precision. We replayed every fourth generation with 1,000 replicates per replay (compared to 10,000 replicates for every generation in the fine-grained replays). Here we show all 24 coarse-grained replays.

The functions to generate the plots are not shown here. They can be viewed in `/experiments/2024_03_18_01__replays/analysis`

For each plot, the top subplot shows the potential to cross the first valley (orange line) and the second valley (yellow line). The background of the top subplot shows the expectation based on the position of the leading edge, as in the paper. The bottom subplot shows the state of the population at each generation of the original replicate.

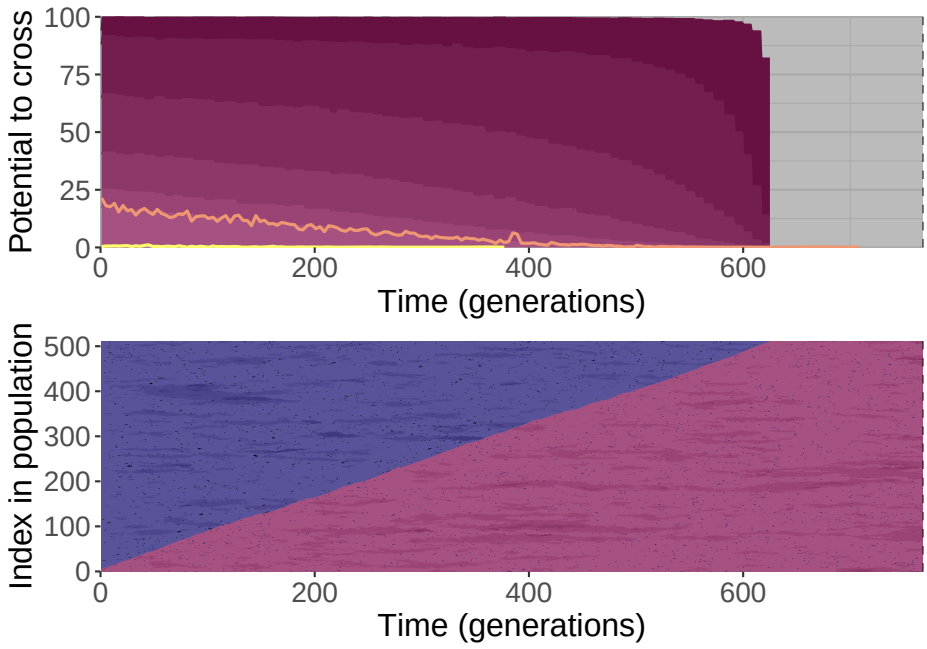
2.1 Dependencies

```
# External
library(ggplot2)
library(dplyr)
library(cowplot)

base_repo_dir = '../..'
exp_dir = paste0(base_repo_dir, '/experiments/2024_03_18_01__replays/')
```

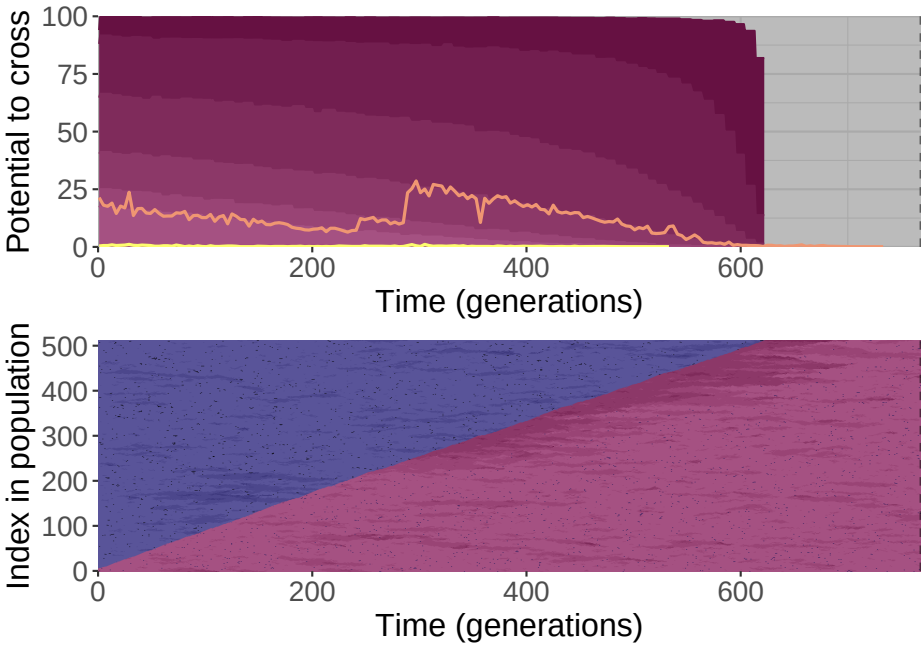
```
# Internal
source(paste0(base_repo_dir, '/global_shared_files/global_analysis_variables.R'))
```

2.2 Populations that did not cross

```
plot_replay_with_adjusted_benchmark('134')
```

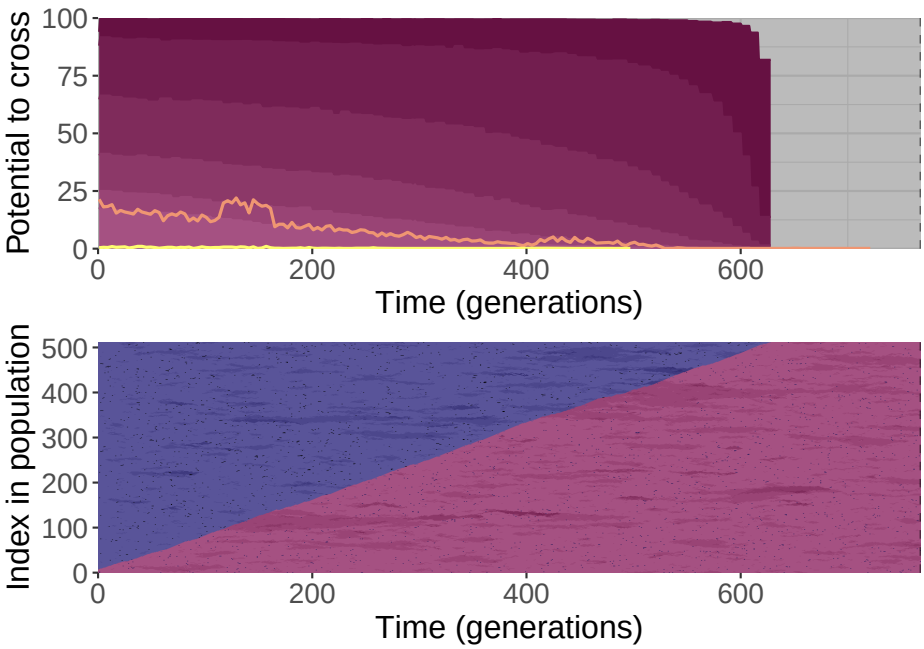



```
plot_replay_with_adjusted_benchmark('158')
```



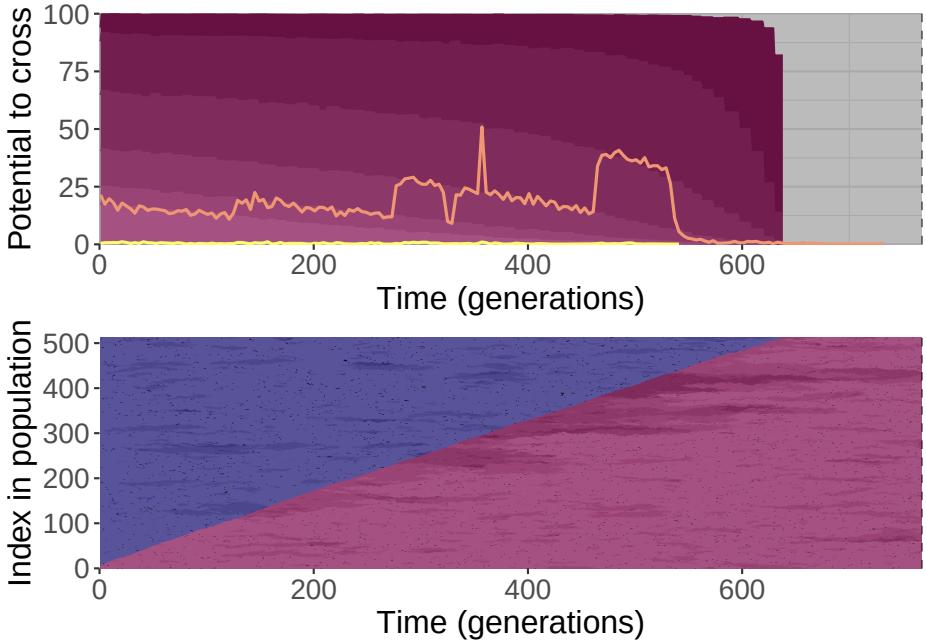
2.2.3 Seed 164

```
plot_replay_with_adjusted_benchmark('164')
```



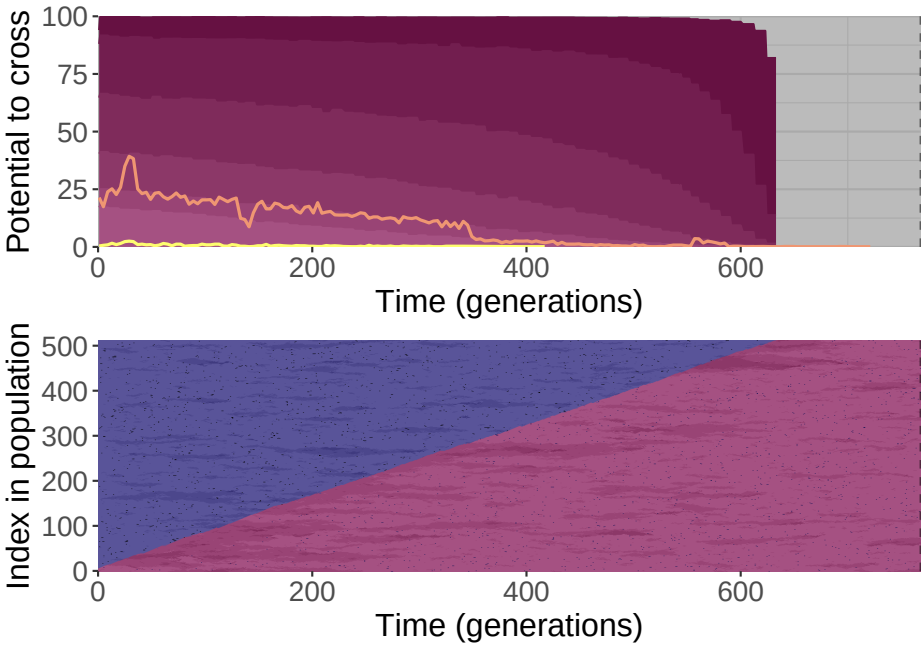
```
plot_replay_with_adjusted_benchmark('175')
```


2.2.4 Seed 175




```
plot_replay_with_adjusted_benchmark('252')
```

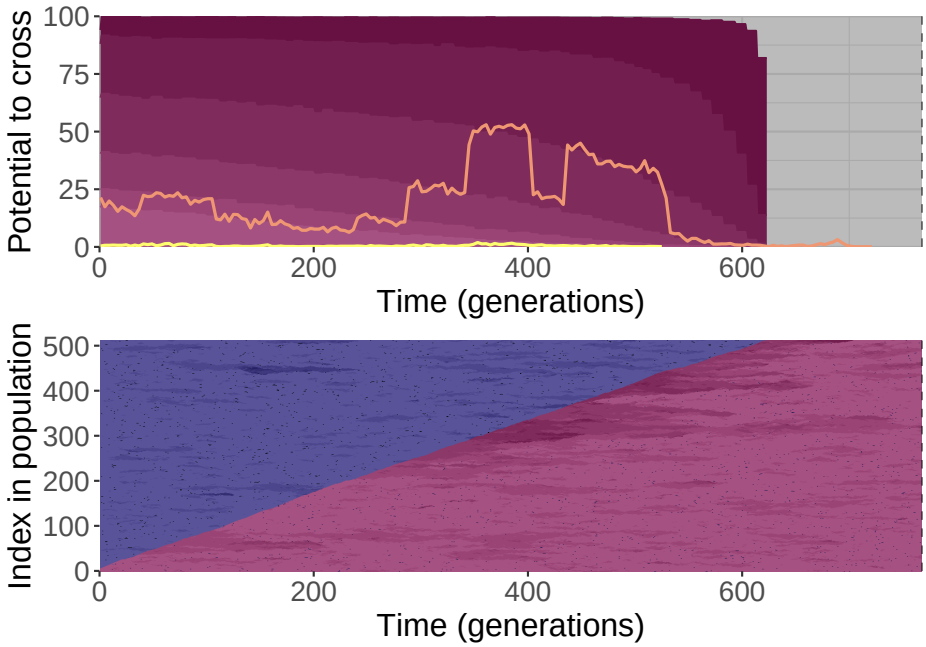
2.2.5 Seed 252



2.2.6 Seed 339

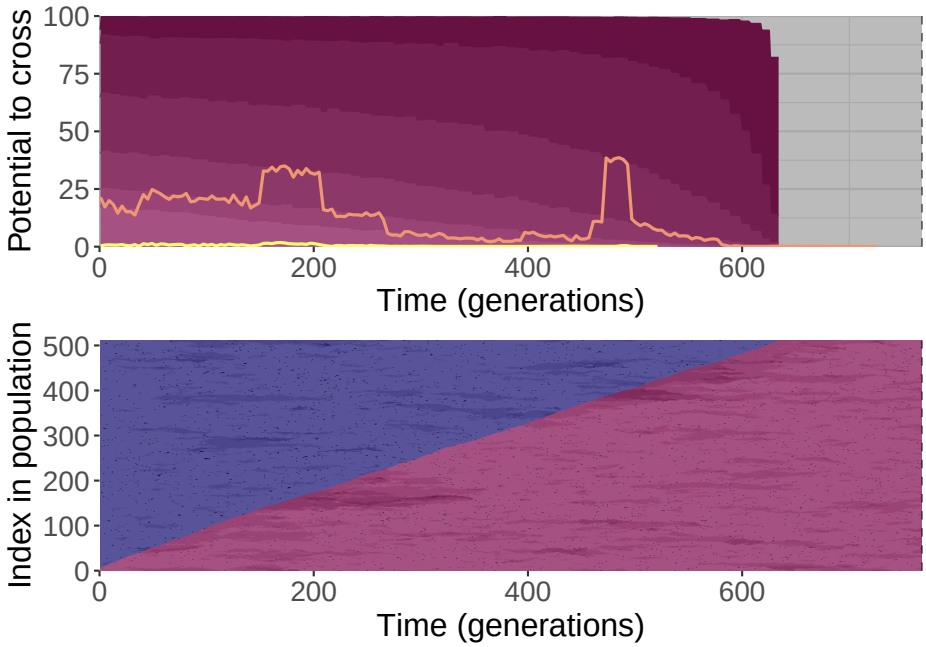
Note that this is the seed in Figure 6 of the paper.

```
plot_replay_with_adjusted_benchmark('339')
```



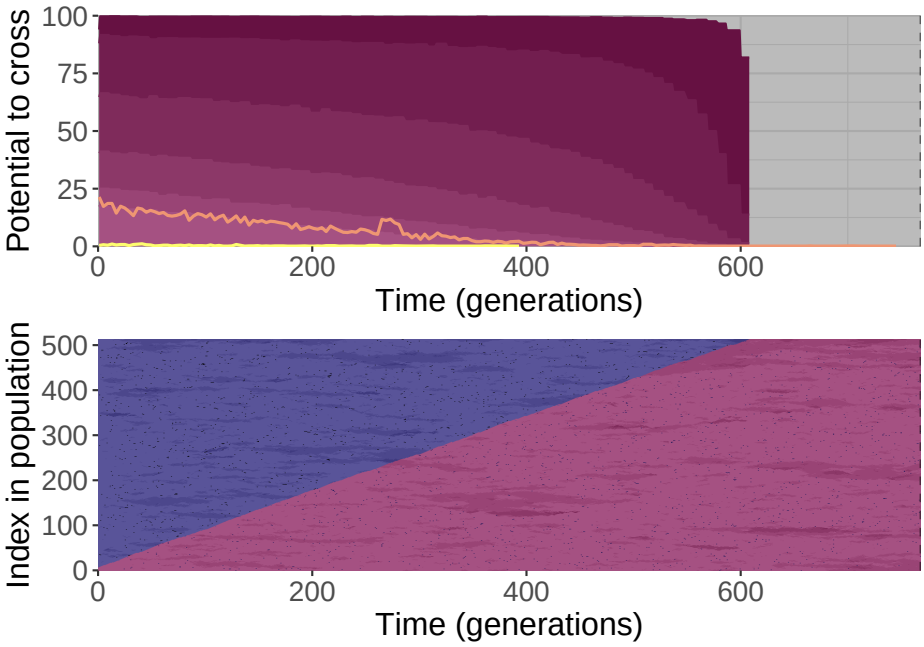
2.2.7 Seed 365

```
plot_replay_with_adjusted_benchmark('365')
```



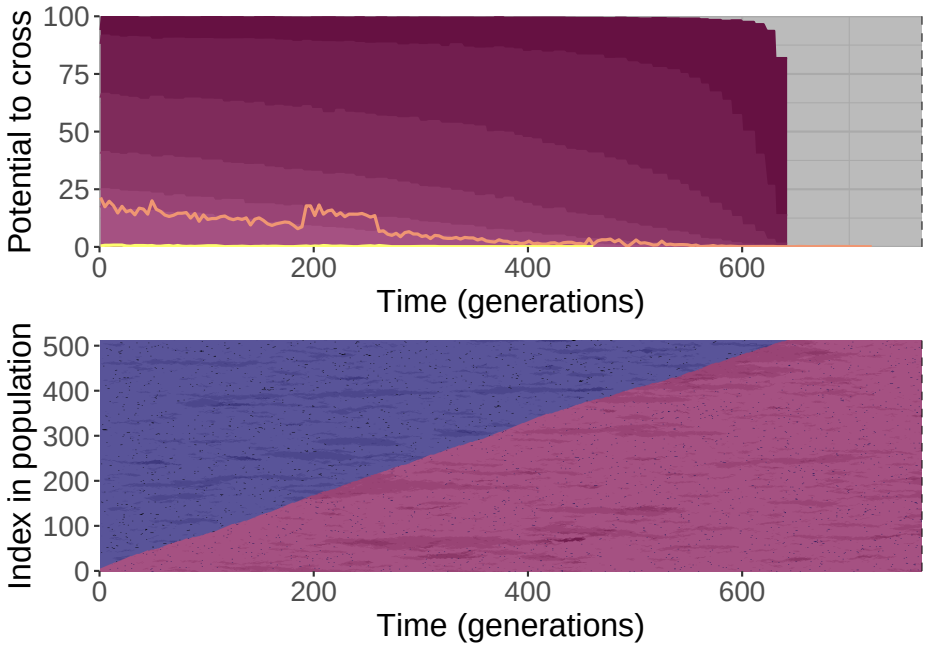
```
plot_replay_with_adjusted_benchmark('394')
```

2.2.8 Seed 394



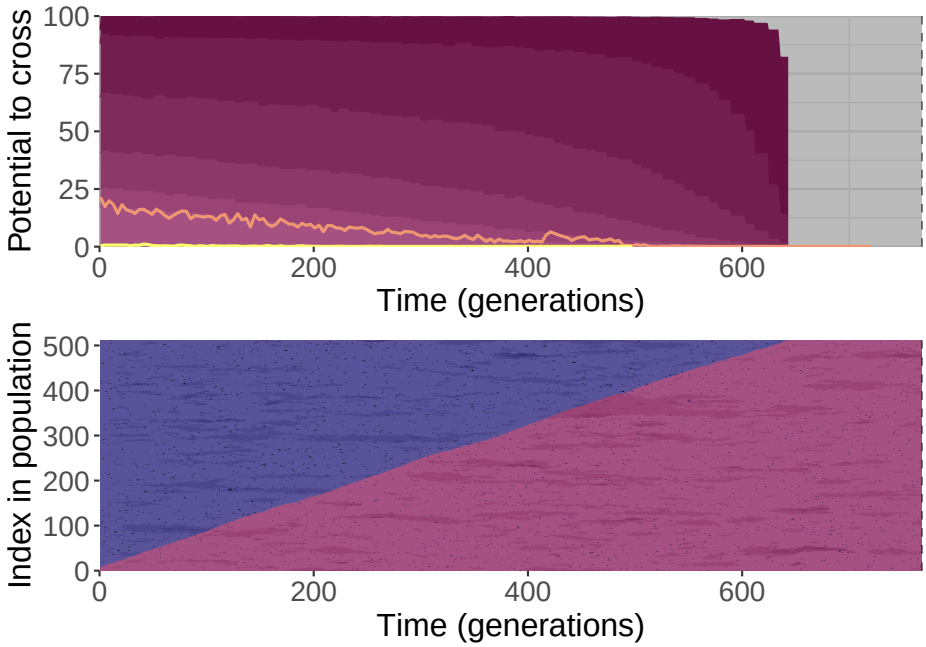

```
plot_replay_with_adjusted_benchmark('446')
```

2.2.9 Seed 446



2.2.10 Seed 450

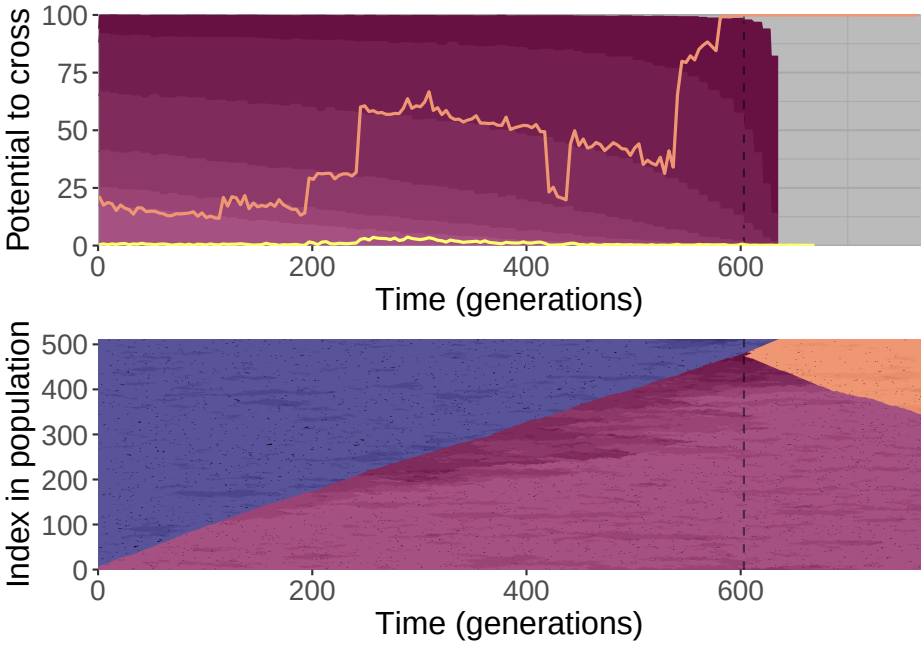
```
plot_replay_with_adjusted_benchmark('450')
```



2.3 Populations that crossed once

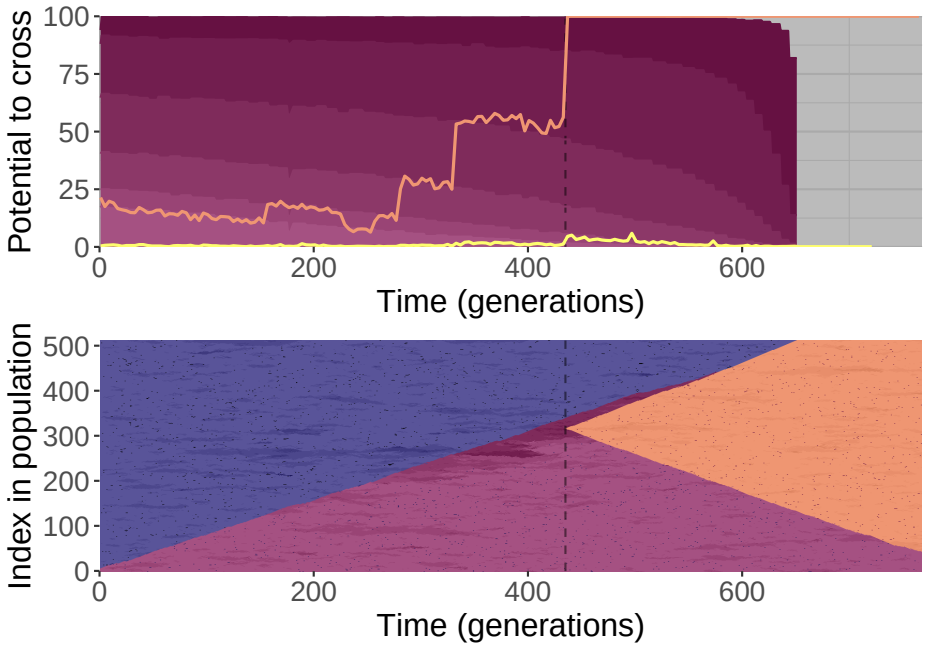
```
plot_replay_with_adjusted_benchmark('011')
```

2.3.1 Seed 011



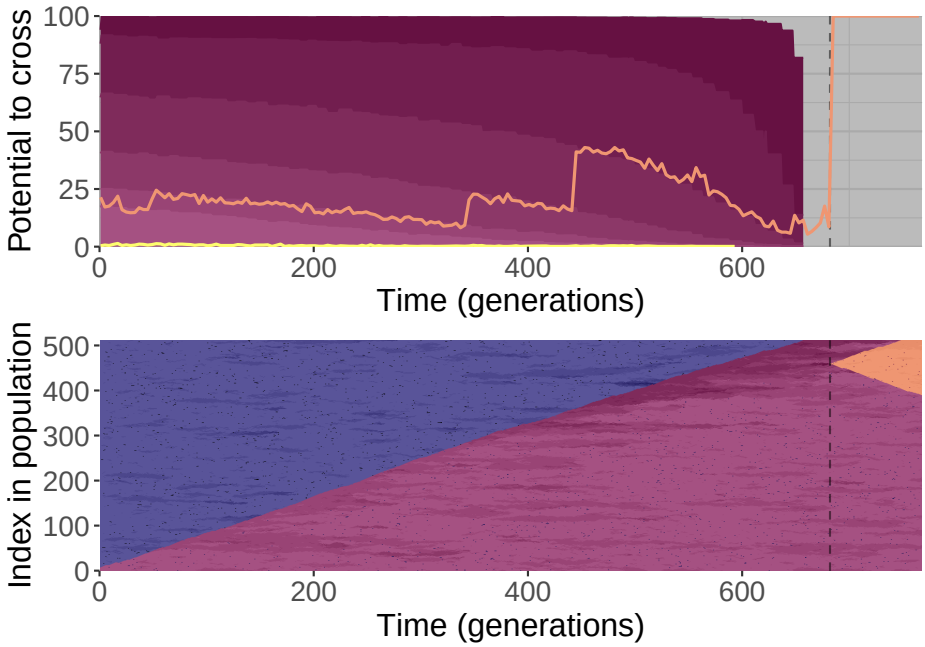

```
plot_replay_with_adjusted_benchmark('050')
```

2.3.2 Seed 050



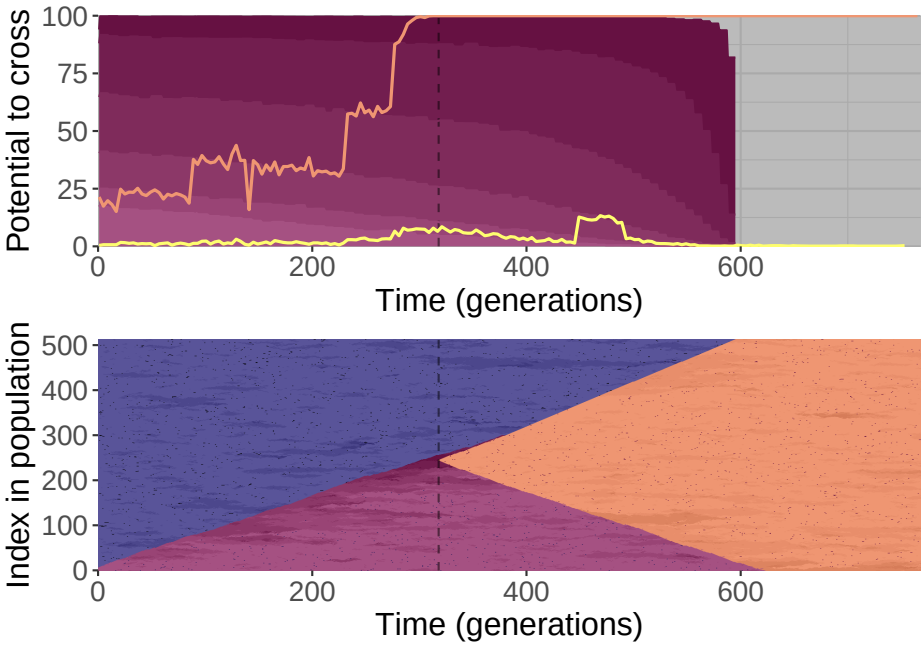
2.3.3 Seed 075

```
plot_replay_with_adjusted_benchmark('075')
```



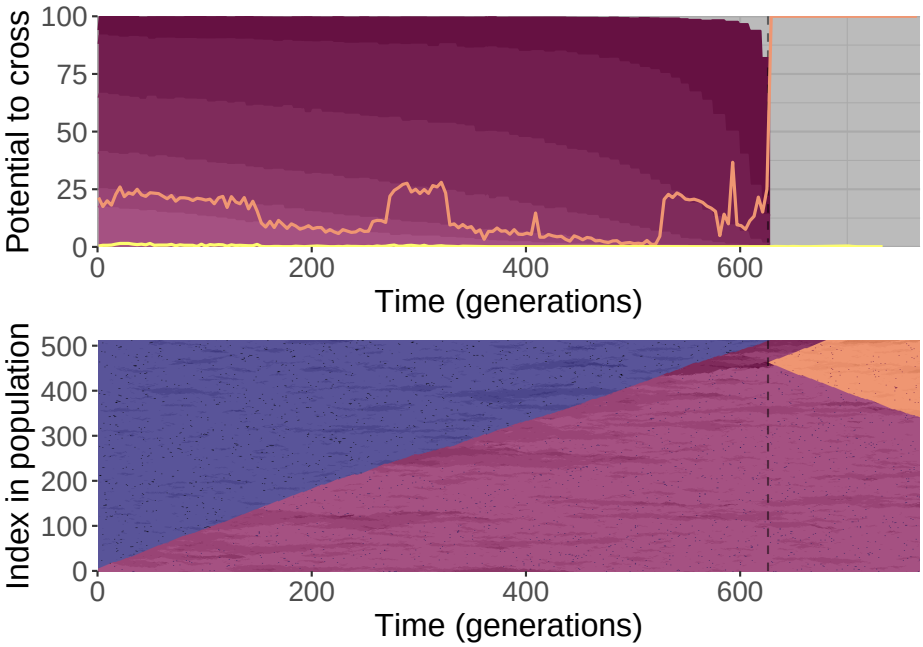
```
plot_replay_with_adjusted_benchmark('083')
```

2.3.4 Seed 083



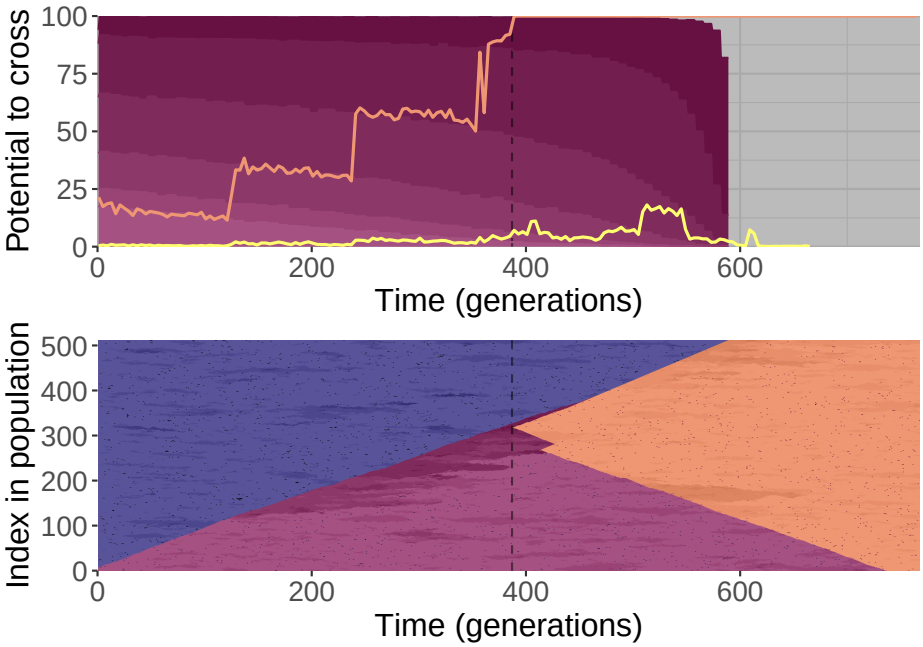

```
plot_replay_with_adjusted_benchmark('105')
```

2.3.5 Seed 105



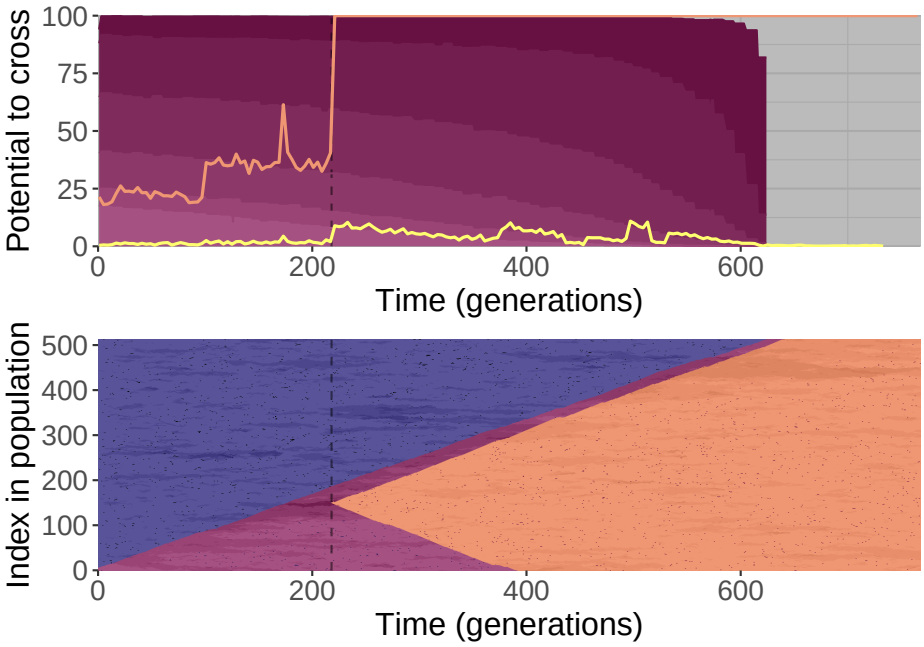
2.3.6 Seed 282

```
plot_replay_with_adjusted_benchmark('282')
```



```
plot_replay_with_adjusted_benchmark('343')
```

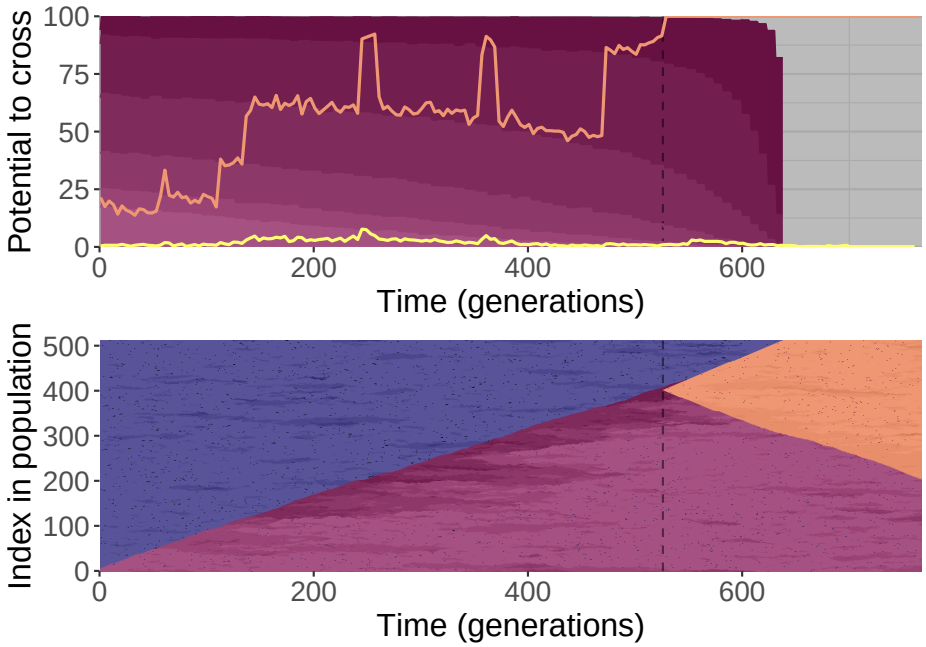
2.3.7 Seed 343



2.3.8 Seed 400

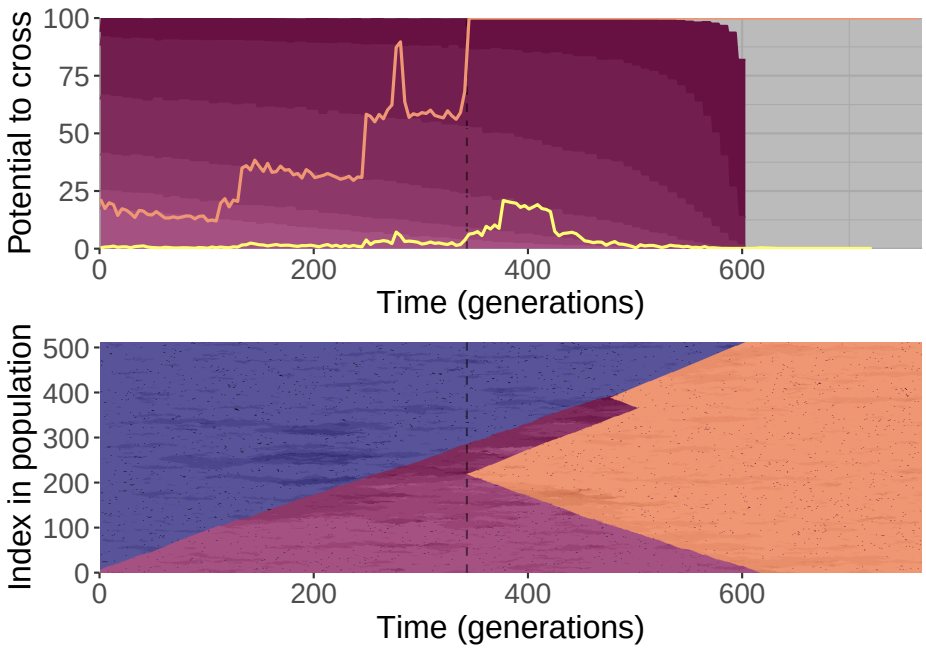
Note that this is the seed in Figure 4 of the paper.

```
plot_replay_with_adjusted_benchmark('400')
```



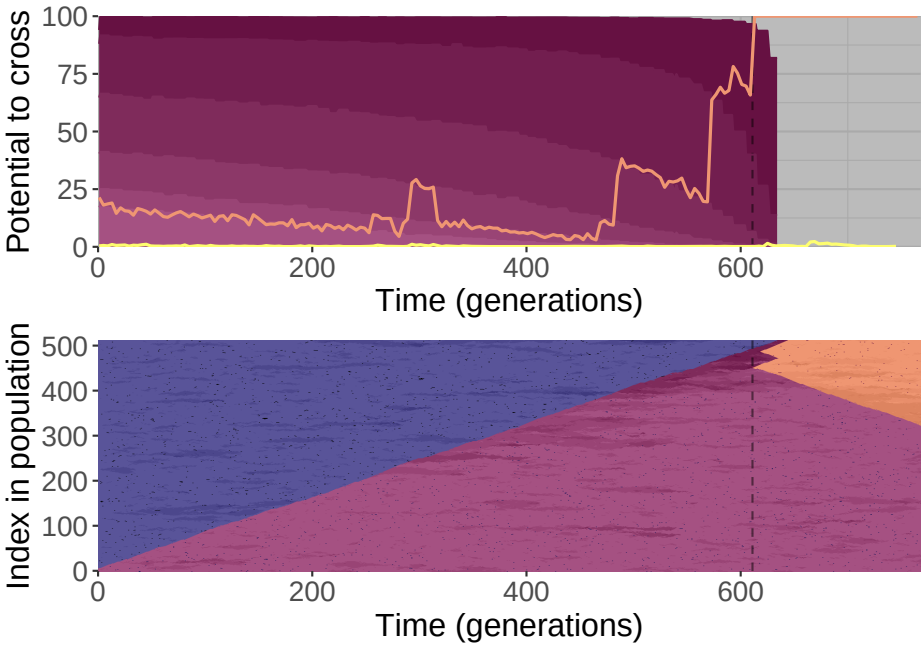
2.3.9 Seed 408

```
plot_replay_with_adjusted_benchmark('408')
```



```
plot_replay_with_adjusted_benchmark('415')
```

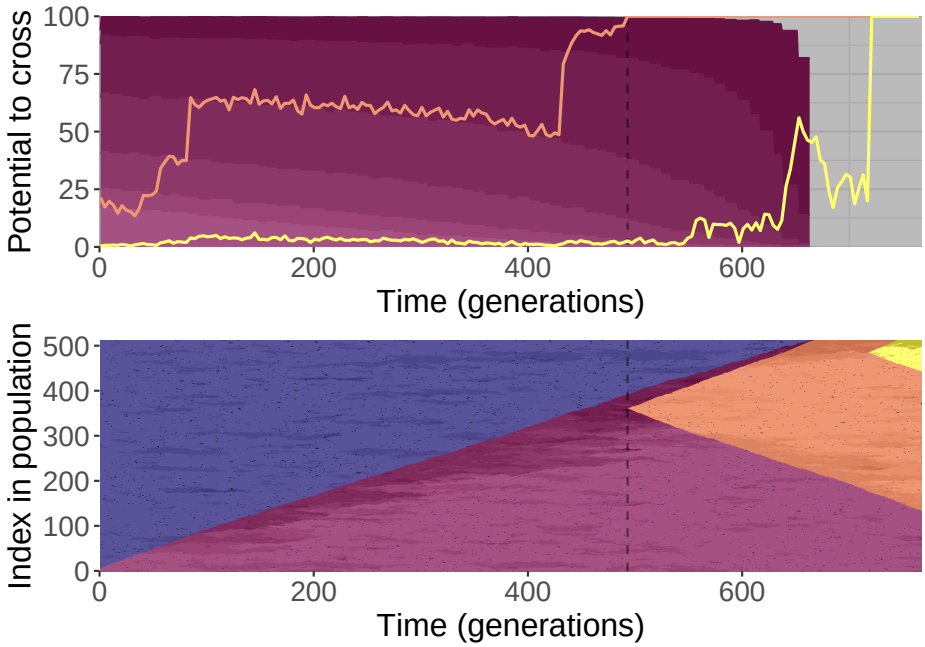
2.3.10 Seed 415



2.4 Populations that crossed twice

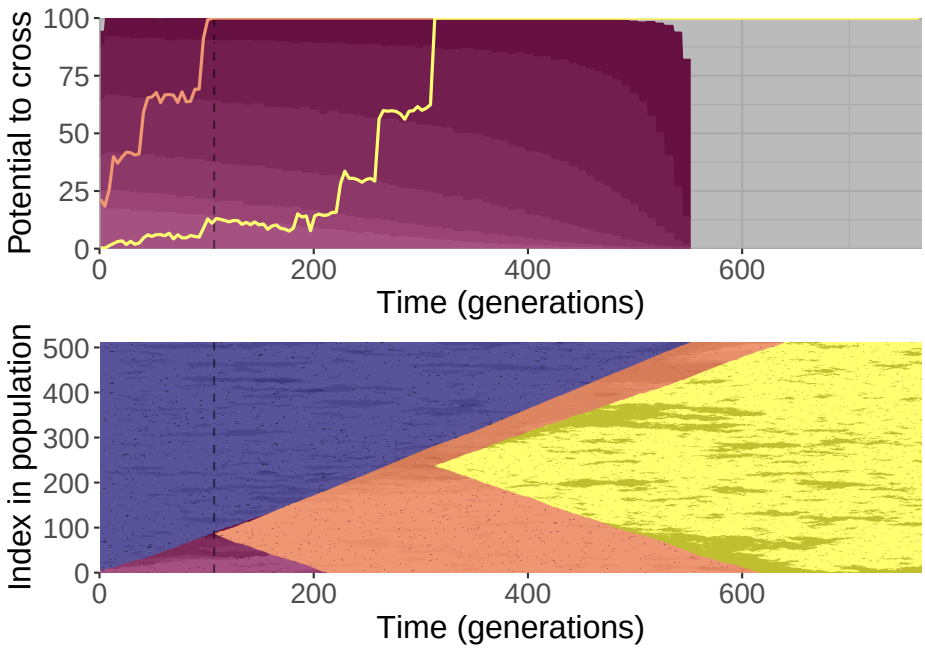
```
plot_replay_with_adjusted_benchmark('093')
```

2.4.1 Seed 093



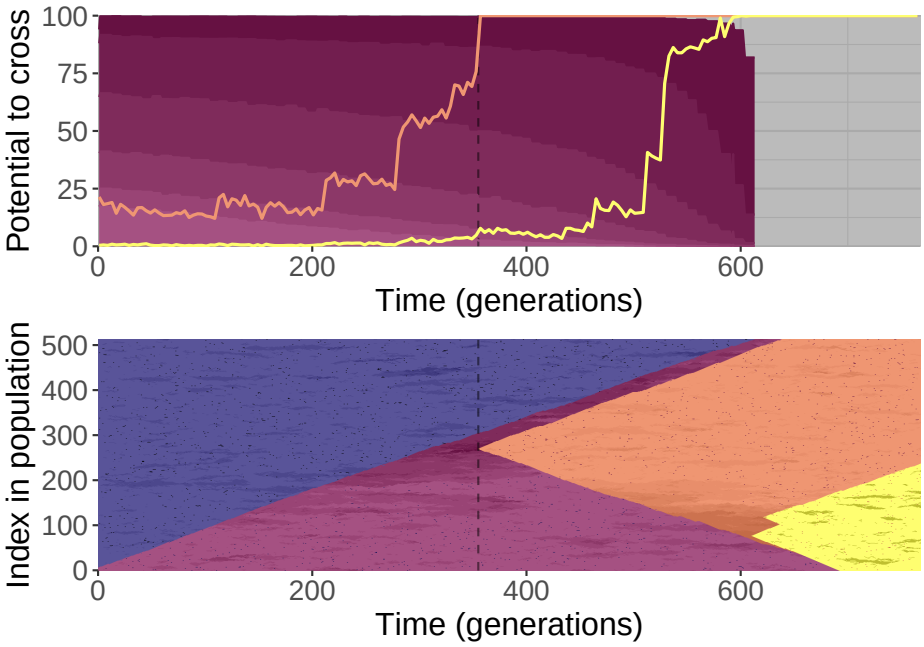
2.4.2 Seed 124

```
plot_replay_with_adjusted_benchmark('124')
```



```
plot_replay_with_adjusted_benchmark('138')
```

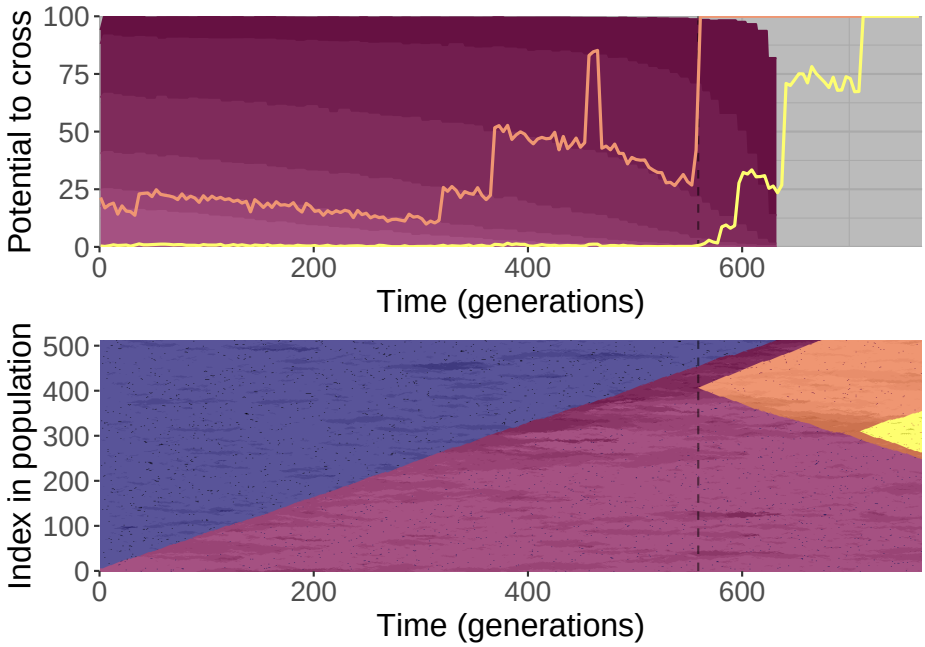
2.4.3 Seed 138



2.4.4 Seed 263

Note that this is the seed in Figure 5 of the paper.

```
plot_replay_with_adjusted_benchmark('263')
```


Potential to cross
100
75
50
25
0
0
200
400
600
Time (generations)
Index in population
500
400
300
200
100
0
0
200
400
600
Time (generations)

Chapter 3

Replaying equilibrium populations

The paper shows fine-grained replays for one replicate that was NOT experiencing adaptive momentum at the start. The replicate shown in the paper crossed two valleys, and was the only one to do so. We also replayed 10 randomlyselected replicates that crossed one valley and 10 randomly selected replicates that failed to cross any valleys. We show those here.

The functions to generate the plots are not shown here. They can be viewed in ‘/experiments/2024_03_22_01___no_am_replays/analysis

For each plot, the top subplot shows the potential to cross the first valley (red line) and the second valley (orange line). The background of the top subplot shows the expectation based on the position of the leading edge, as in the paper. The bottom subplot shows the state of the population at each generation of the original replicate.

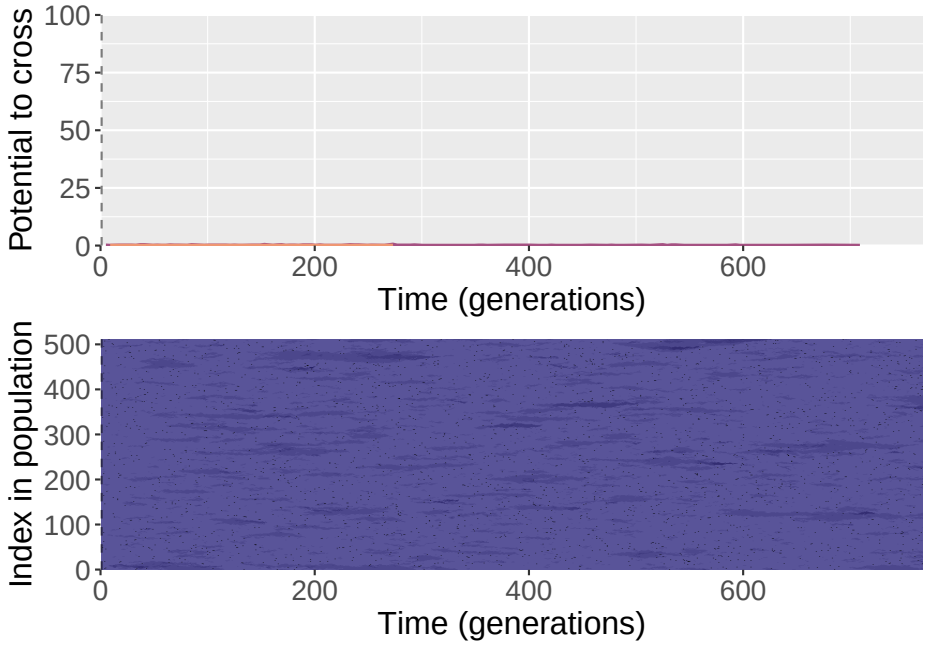
3.1 Dependencies

```
# External
library(ggplot2)
library(dplyr)
library(cowplot)

base_repo_dir = '../..'
exp_dir = paste0(base_repo_dir, '/experiments/2024_03_22_01__no_am_replays/')

# Internal
source(paste0(base_repo_dir, '/global_shared_files/global_analysis_variables.R'))
```

3.2 Populations that did not cross

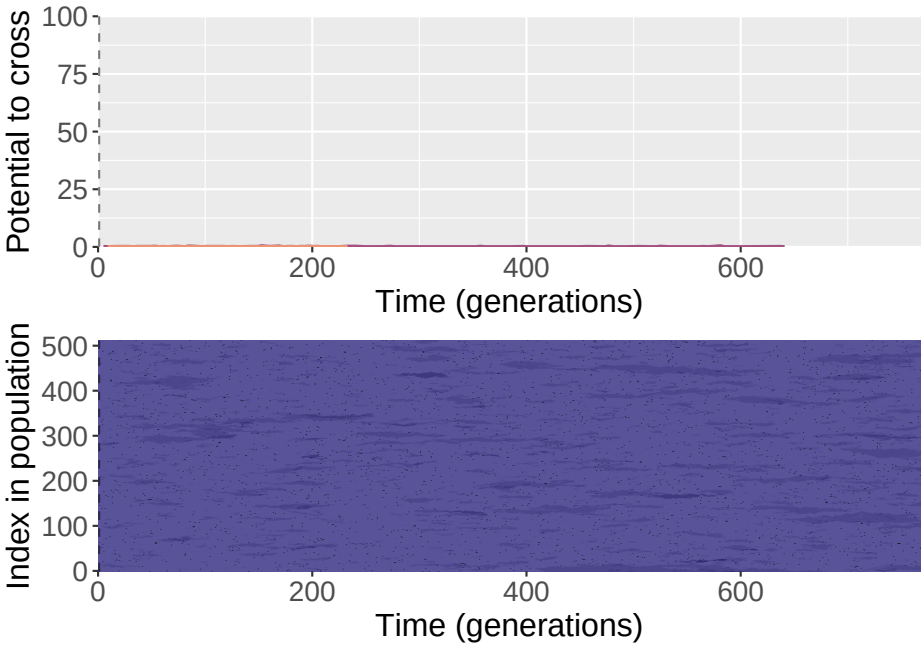
```
plot_replay_with_adjusted_benchmark('01164')
```

3.2.1 Seed 01164



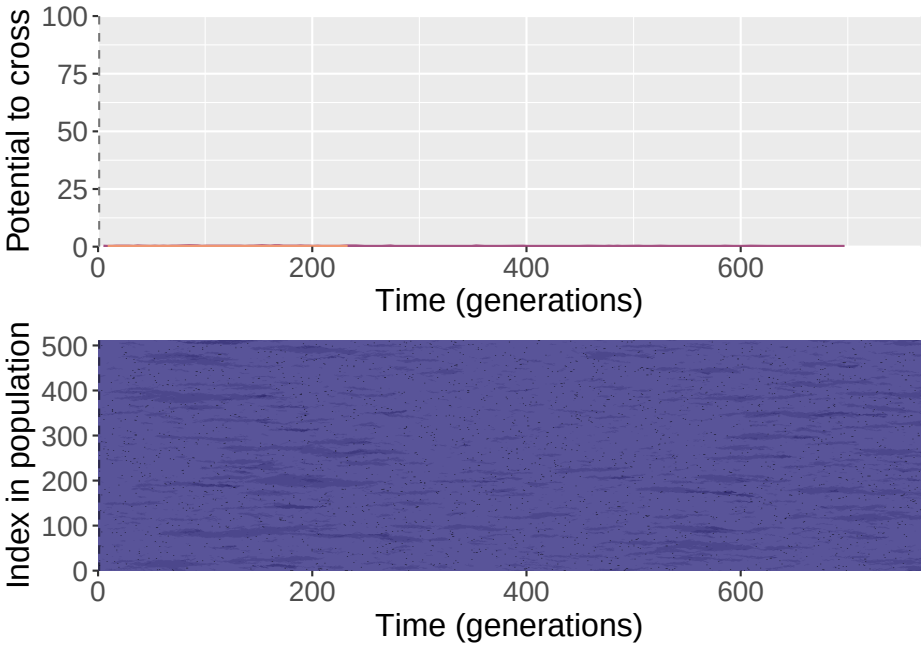

```
plot_replay_with_adjusted_benchmark('01435')
```

3.2.2 Seed 01435



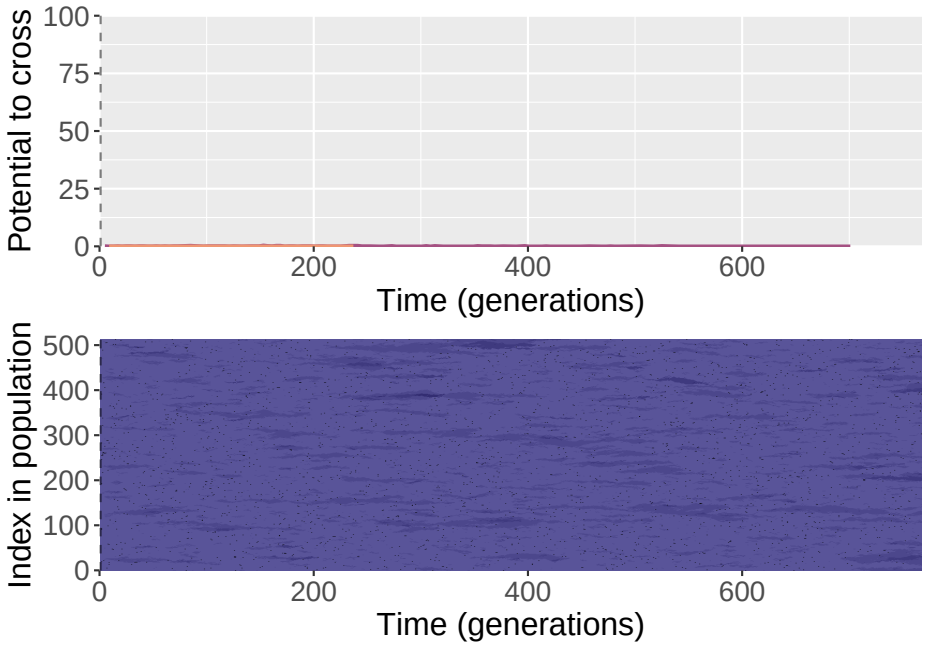
3.2.3 Seed 01572

```
plot_replay_with_adjusted_benchmark('01572')
```



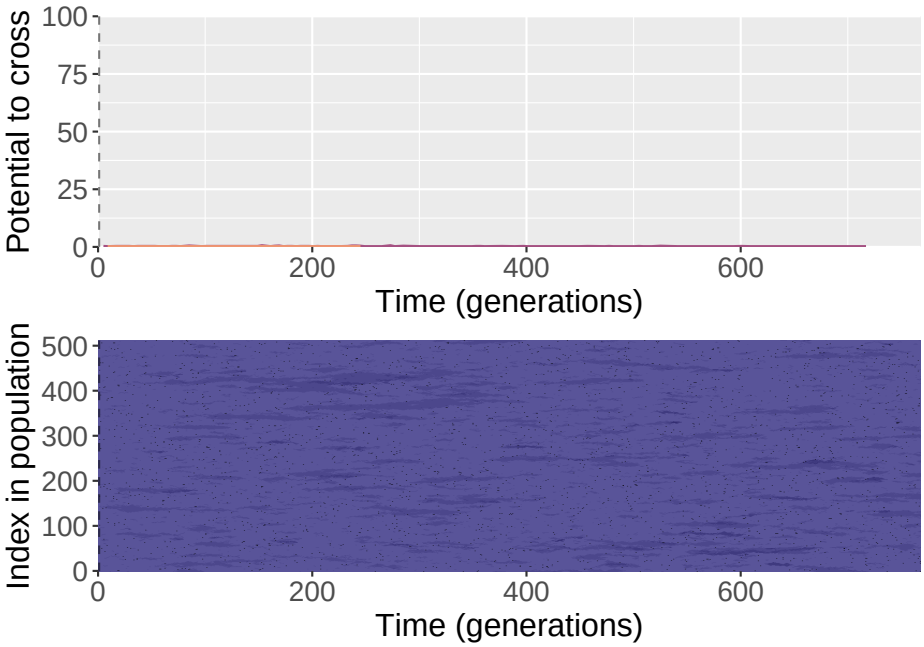
```
plot_replay_with_adjusted_benchmark('02581')
```

3.2.4 Seed 02581



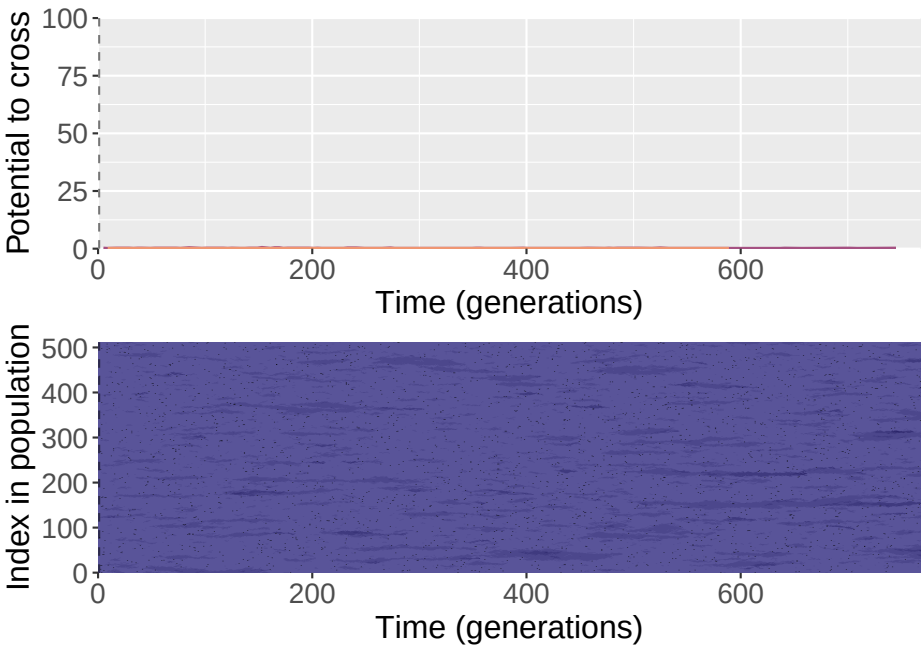

```
plot_replay_with_adjusted_benchmark('02711')
```

3.2.5 Seed 02711



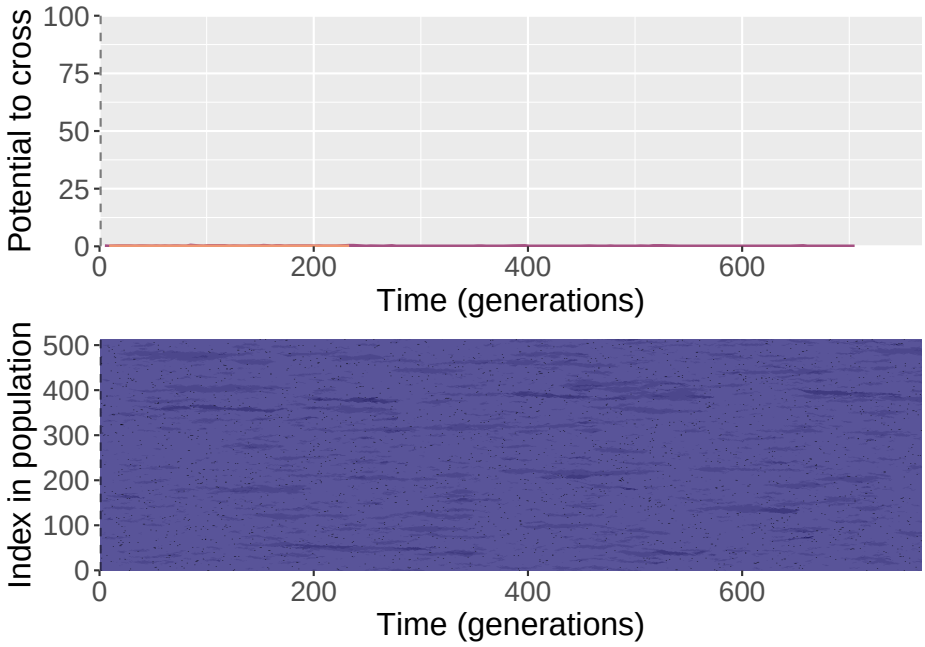
3.2.6 Seed 02961

```
plot_replay_with_adjusted_benchmark('02961')
```



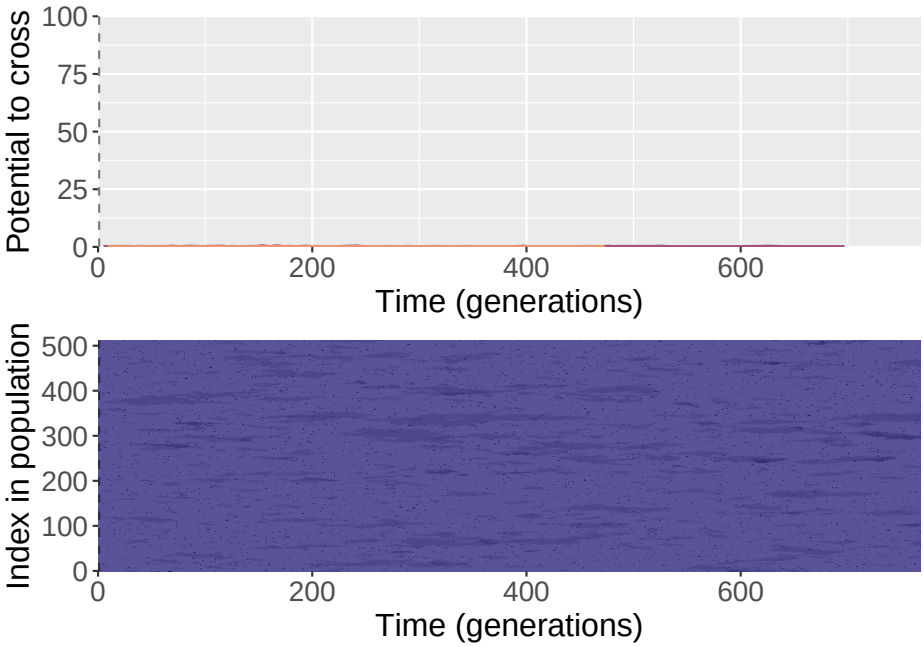
```
plot_replay_with_adjusted_benchmark('04390')
```

3.2.7 Seed 04390



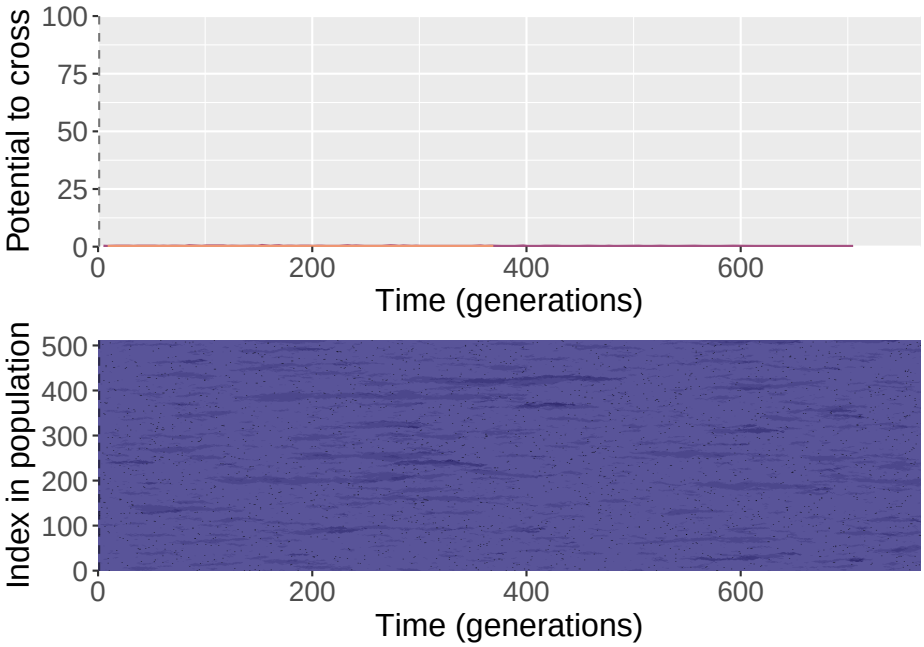

```
plot_replay_with_adjusted_benchmark('06116')
```

3.2.8 Seed 06116



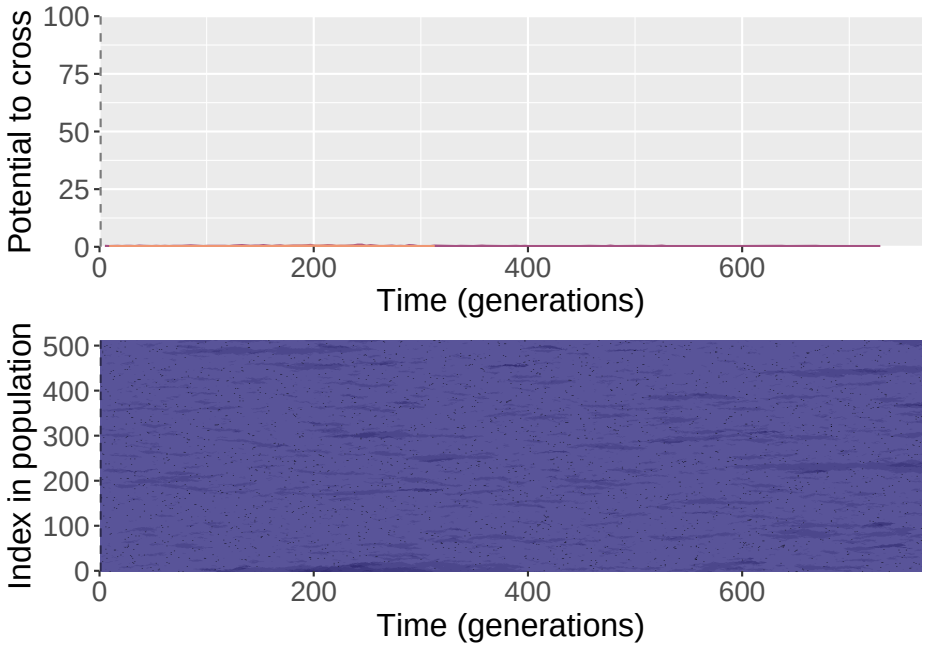
3.2.9 Seed 06583

```
plot_replay_with_adjusted_benchmark('06583')
```



```
plot_replay_with_adjusted_benchmark('08366')
```

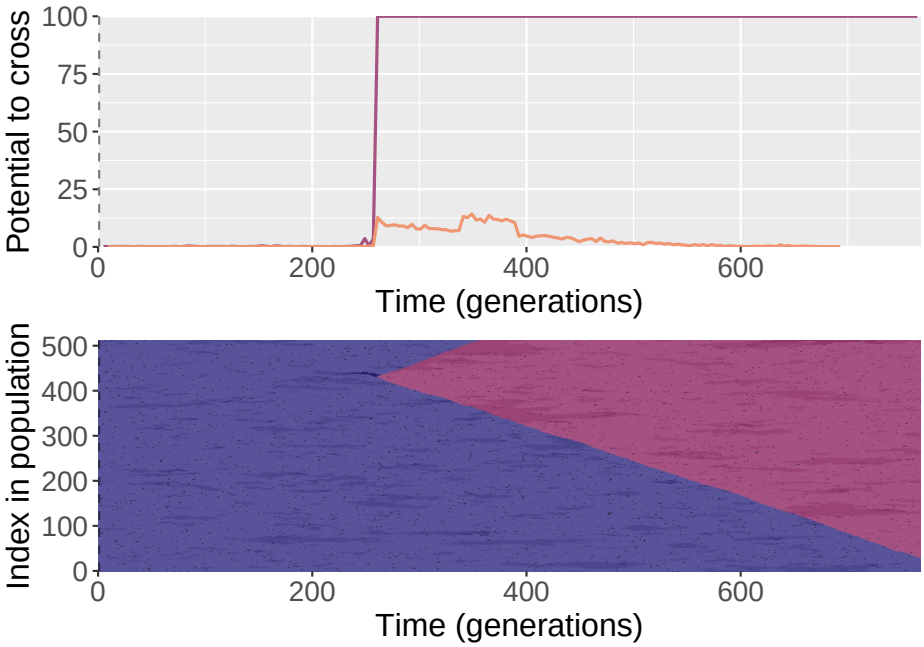
3.2.10 Seed 08366



3.3 Populations that crossed once

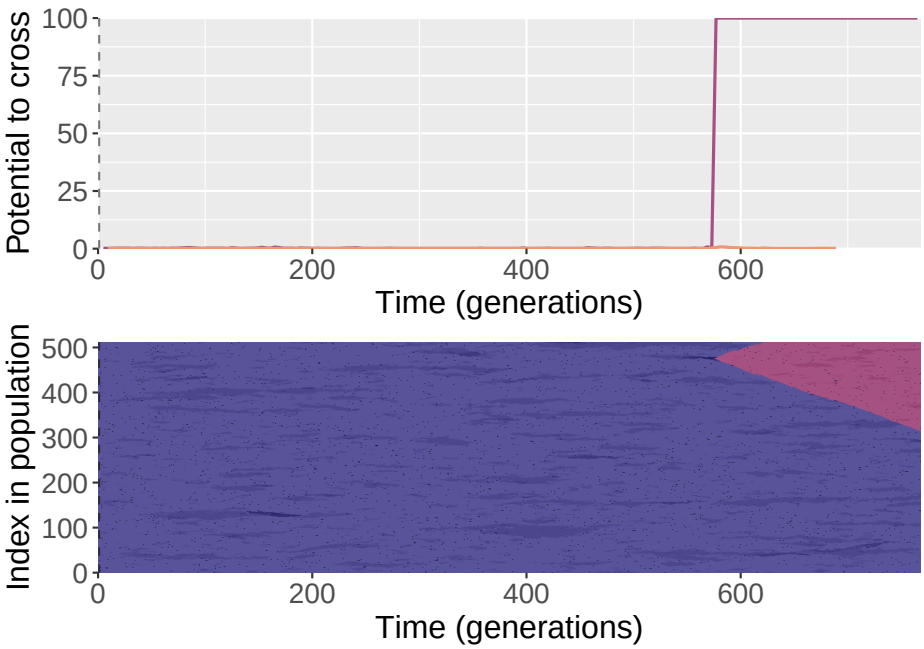
```
plot_replay_with_adjusted_benchmark('00833')
```

3.3.1 Seed 00833



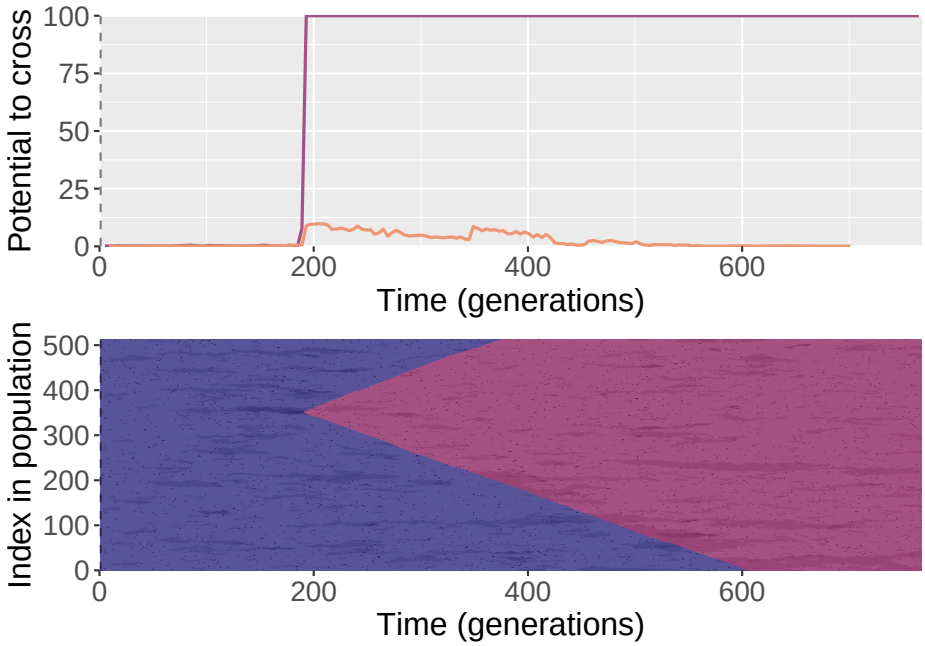
3.3.2 Seed 01357

```
plot_replay_with_adjusted_benchmark('01357')
```



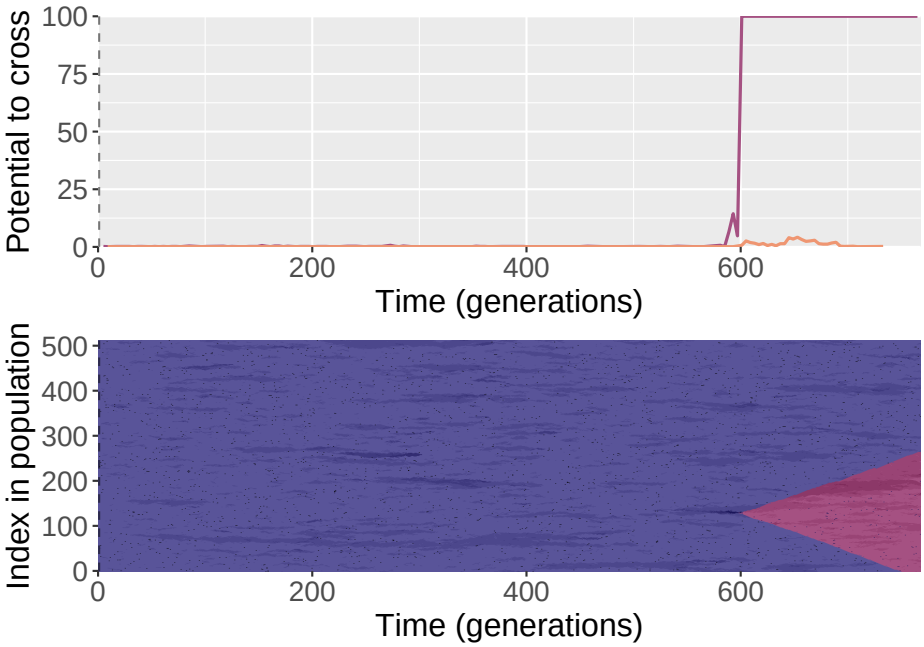
```
plot_replay_with_adjusted_benchmark('02290')
```


3.3.3 Seed 02290



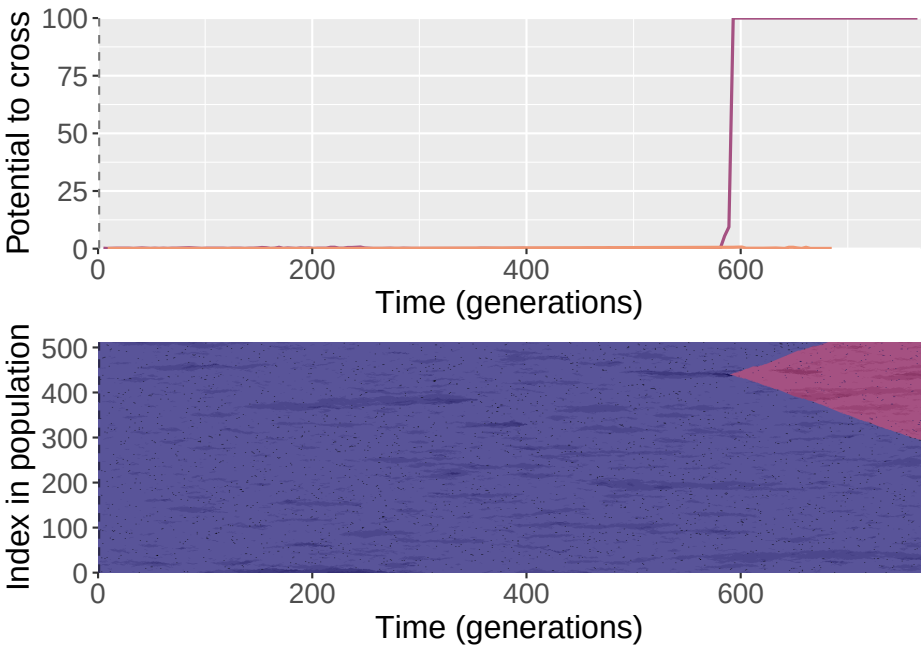

```
plot_replay_with_adjusted_benchmark('02359')
```

3.3.4 Seed 02359



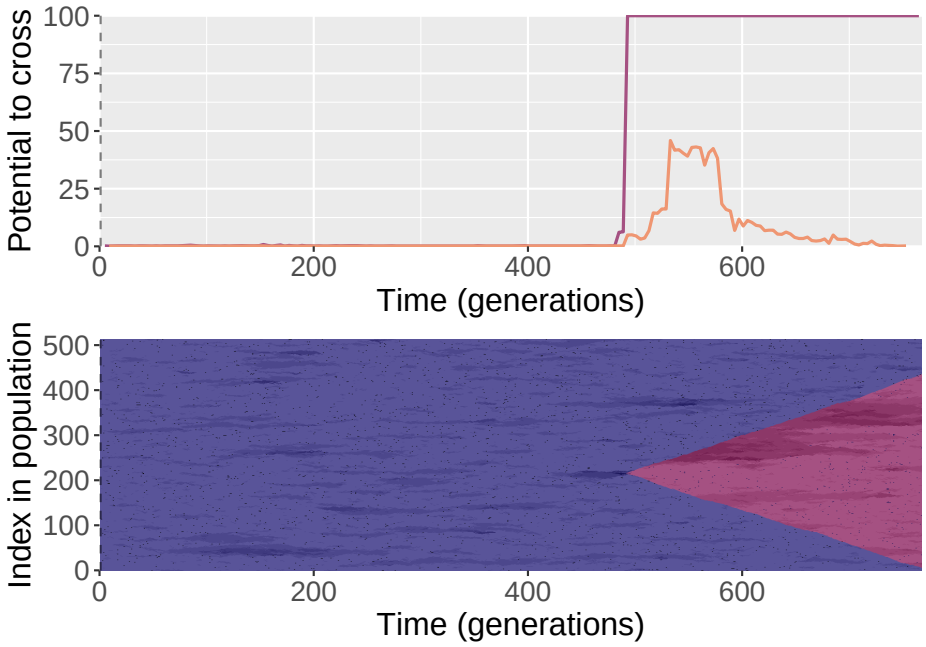
3.3.5 Seed 03149

```
plot_replay_with_adjusted_benchmark('03149')
```



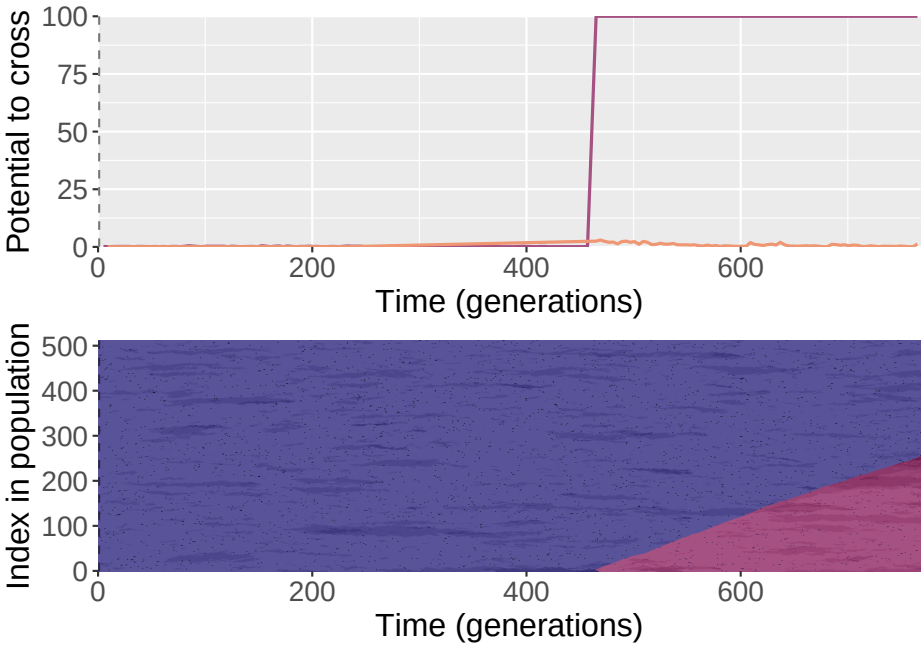
```
plot_replay_with_adjusted_benchmark('05295')
```


3.3.6 Seed 05295



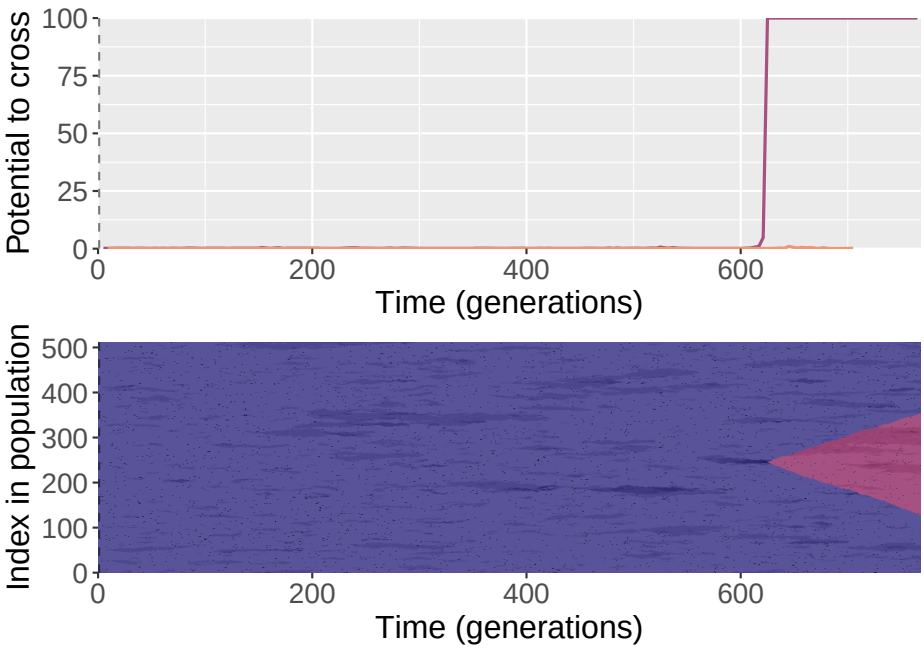

```
plot_replay_with_adjusted_benchmark('07051')
```

3.3.7 Seed 07051



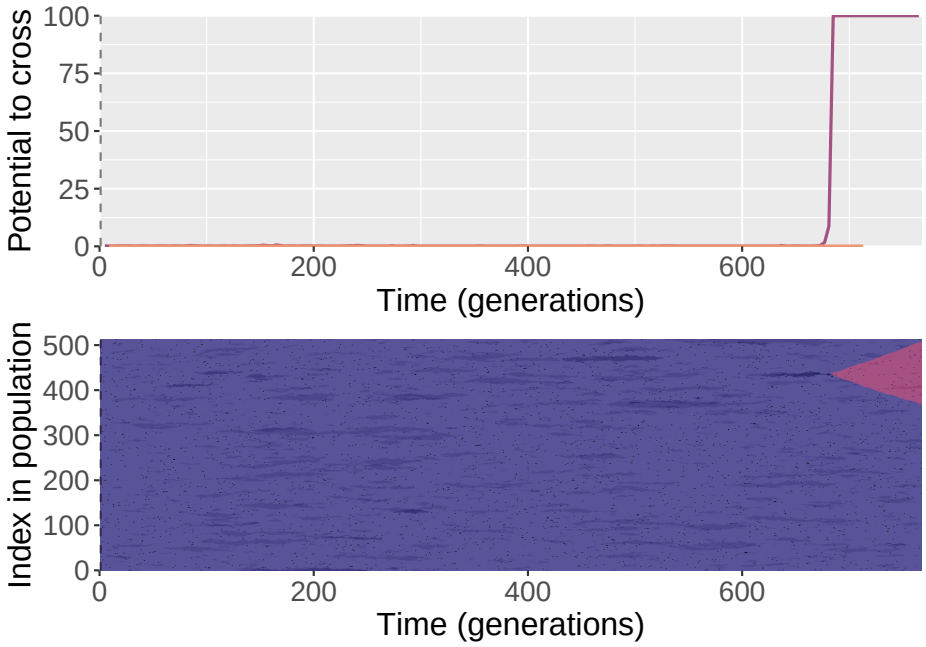
3.3.8 Seed 07605

```
plot_replay_with_adjusted_benchmark('07605')
```



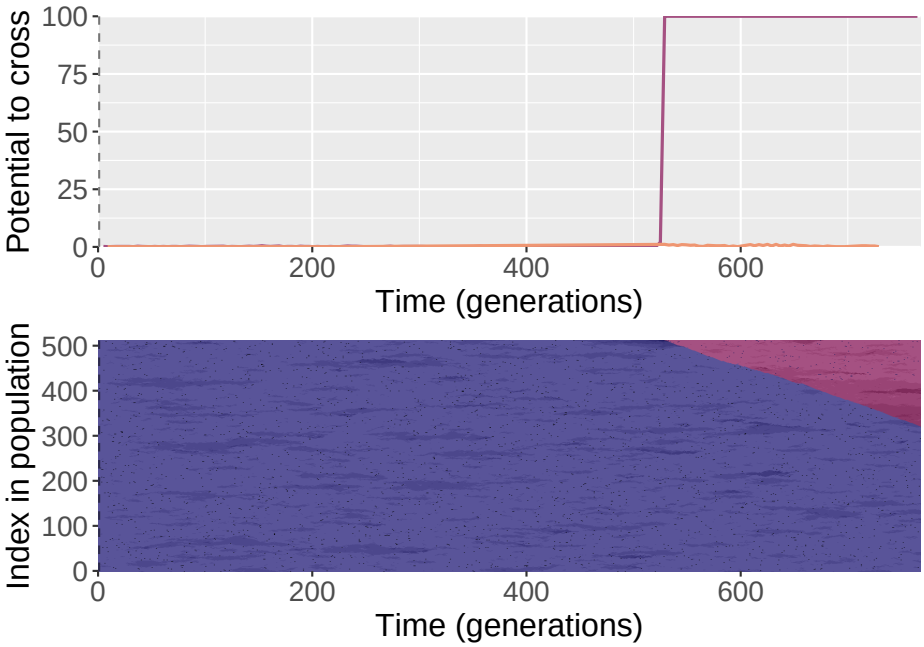
```
plot_replay_with_adjusted_benchmark('07916')
```

3.3.9 Seed 07916




```
plot_replay_with_adjusted_benchmark('09839')
```

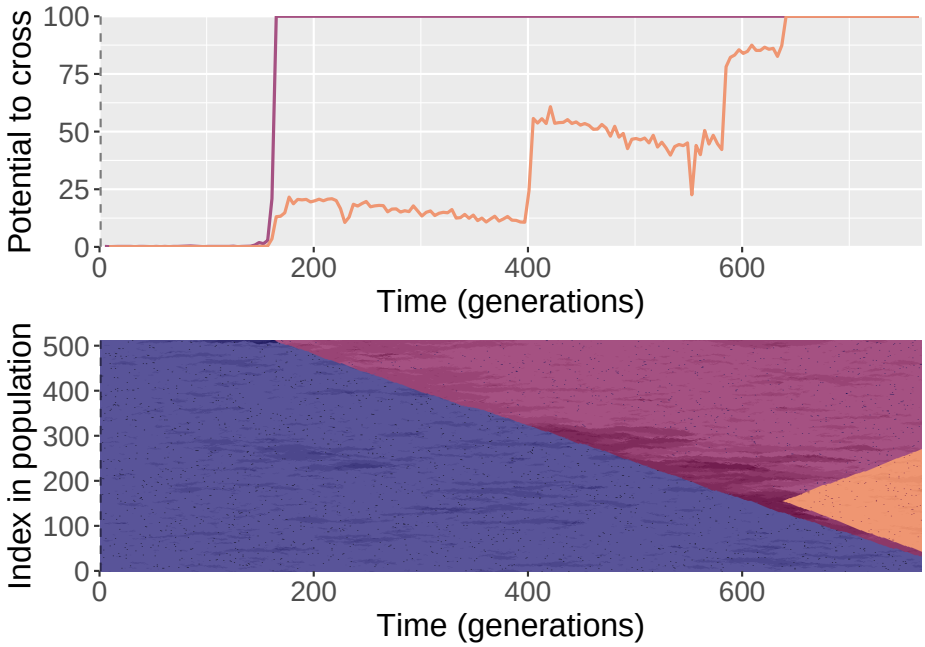
3.3.10 Seed 09839



3.4 Populations that crossed twice

Note that this is the seed in Figure 7 of the paper.

```
plot_replay_with_adjusted_benchmark('05501')
```


Potential to cross
100
75
50
25
0
0
200
400
600
Time (generations)

Index in population
500
400
300
200
100
0
0
200
400
600
Time (generations)

Chapter 4

Replaying shuffled populations

The paper shows one replay replicate that was shuffled prior to evolution. We actually ran 10 shuffled replays, and all of those files are included here.

To clarify, for each replayed generation we took the population snapshot at that generation like normal. But for each replicate, we shuffled the organisms in the population prior to starting evolution. This destroyed any existing population structure, though new population structure could evolve again after.

The functions to generate the plots are not shown here. They can be viewed in `/experiments/2024_03_18_01__replays/analysis`

For each plot, the top subplot shows the potential to cross with the normal replays (orange line) and the shuffled replays (black line). The bottom subplot shows the state of the population at each generation of the original replicate.

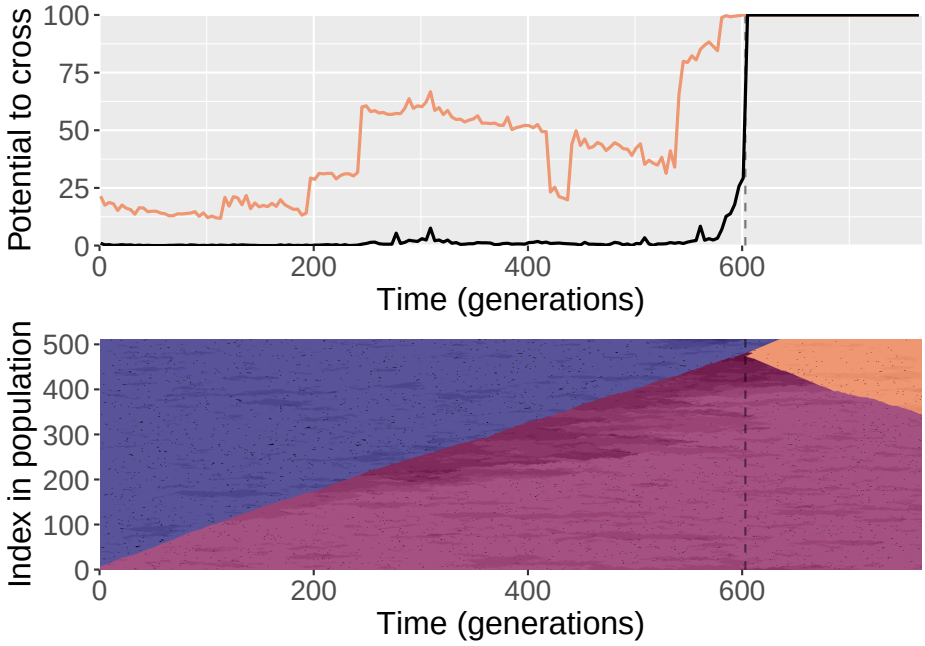
4.1 Dependencies

```
# External
library(ggplot2)
library(dplyr)
library(cowplot)

base_repo_dir = '../..'
exp_dir = paste0(base_repo_dir, '/experiments/2024_03_18_01__replays/')

# Internal
source(paste0(base_repo_dir, '/global_shared_files/global_analysis_variables.R'))
```

4.2 Populations that crossed once

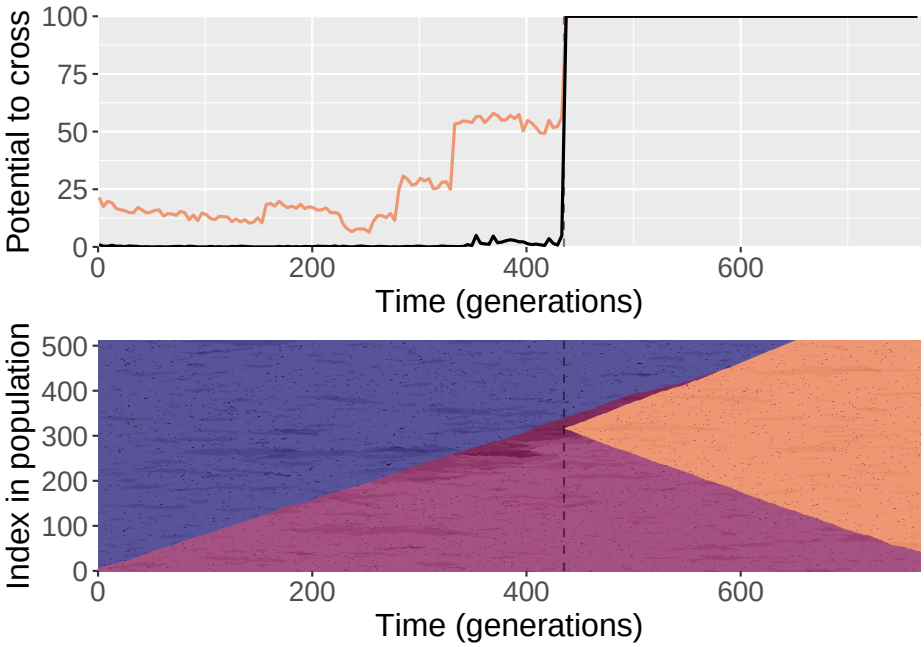
```
plot_shuffled_replay('011')
```

4.2.1 Seed 011



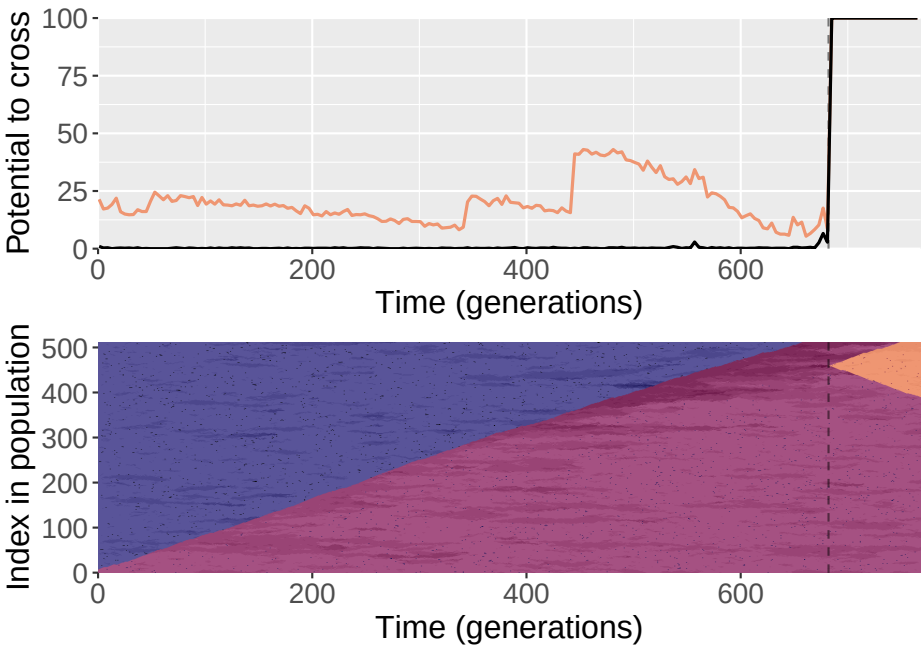

```
plot_shuffled_replay('050')
```

4.2.2 Seed 050



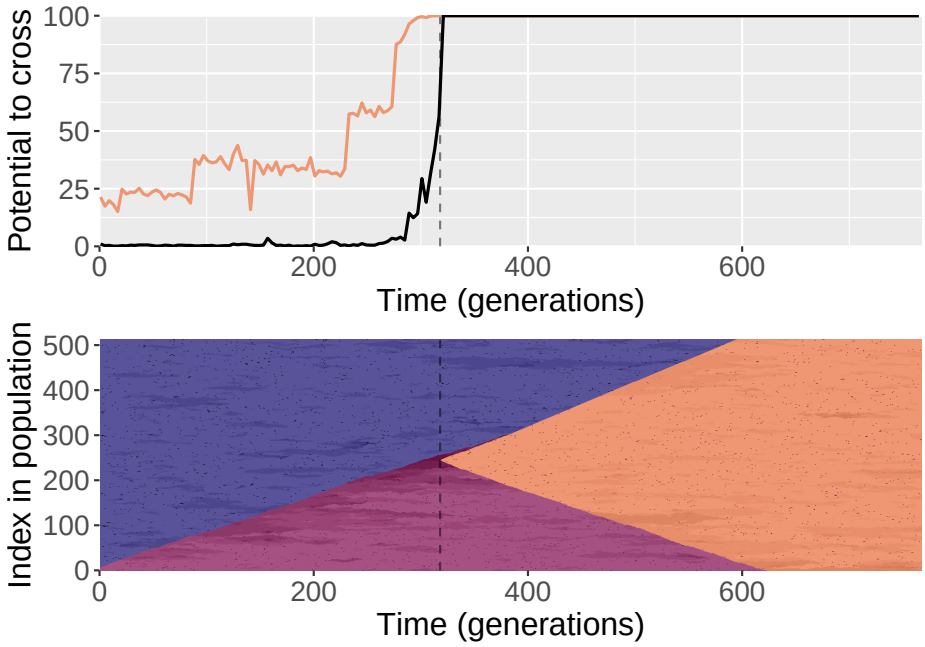
4.2.3 Seed 075

```
plot_shuffled_replay('075')
```



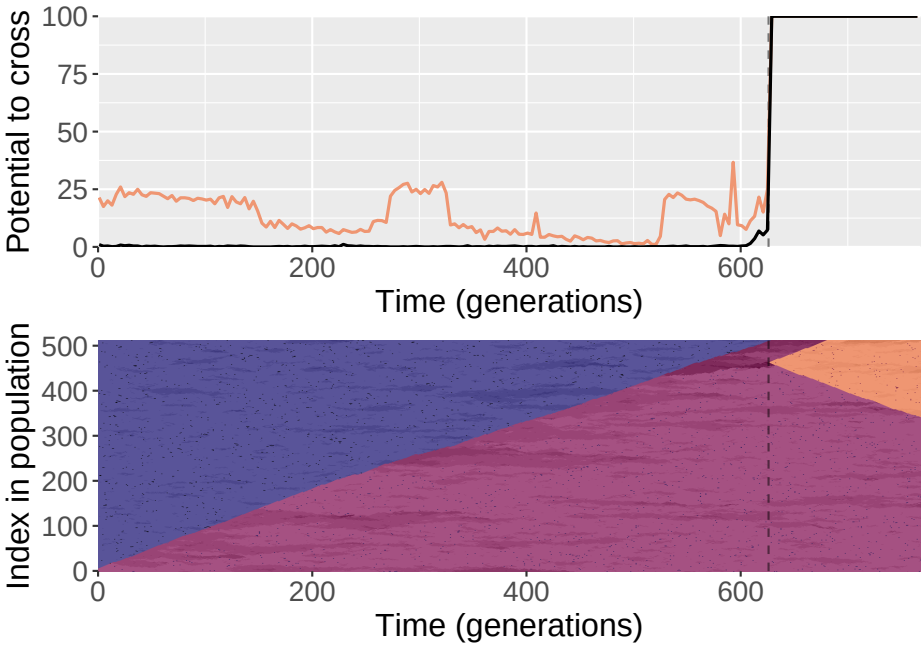
```
plot_shuffled_replay('083')
```

4.2.4 Seed 083



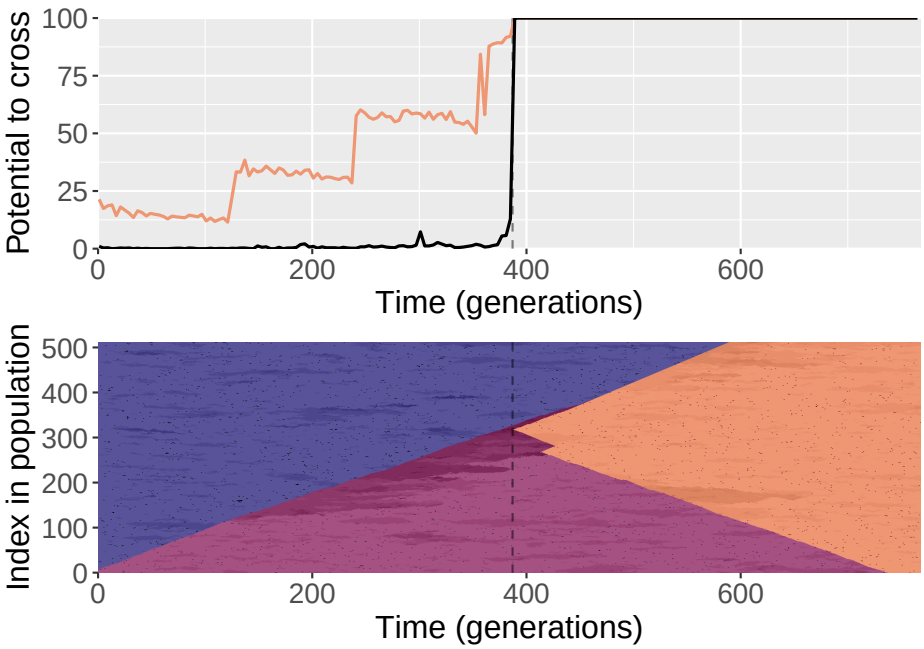

```
plot_shuffled_replay('105')
```

4.2.5 Seed 105



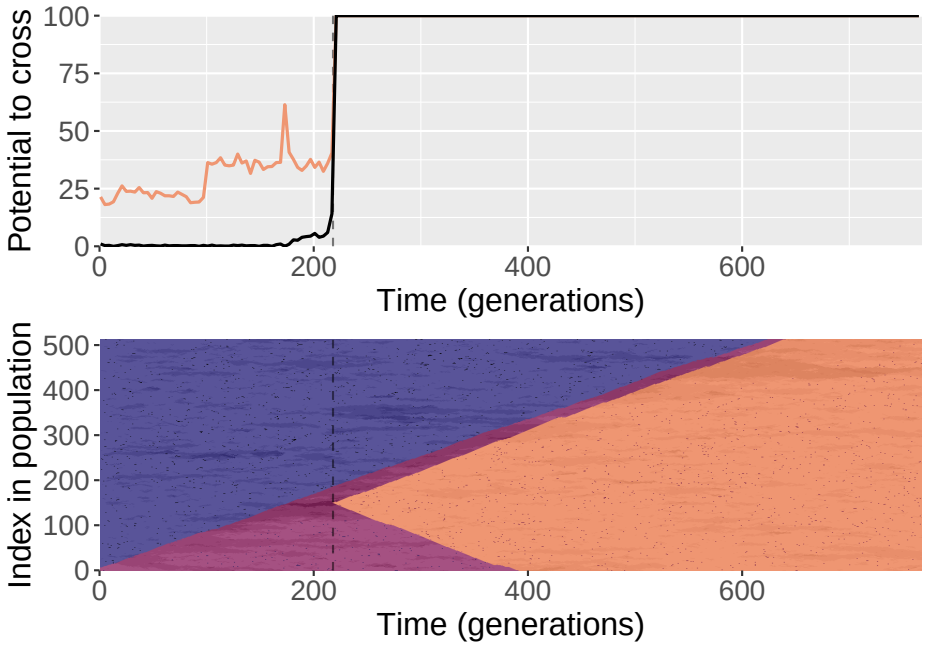
4.2.6 Seed 282

```
plot_shuffled_replay('282')
```



```
plot_shuffled_replay('343')
```

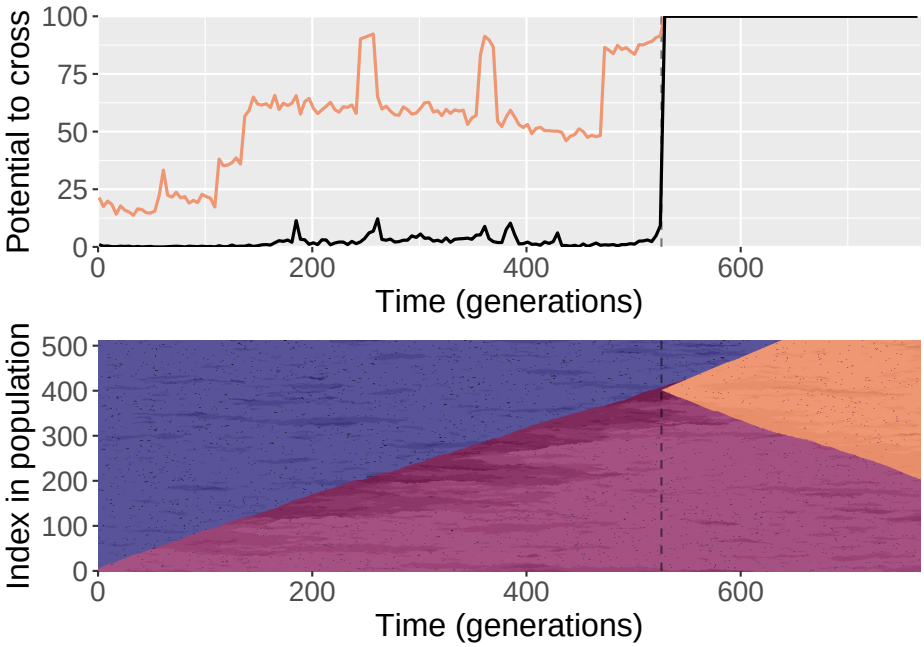
4.2.7 Seed 343



4.2.8 Seed 400

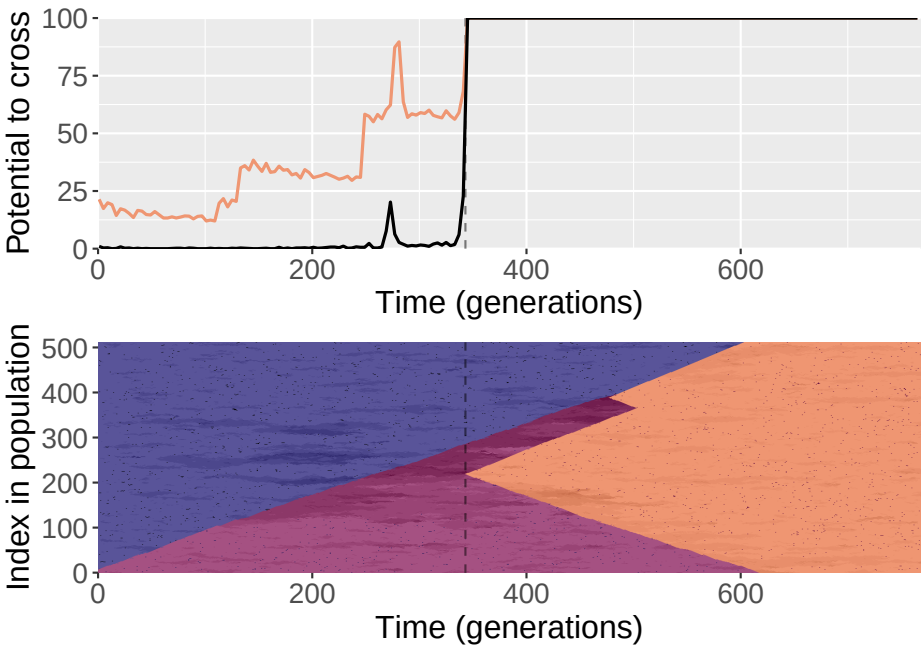
Note that this is the seed in Figure 8 of the paper.

```
plot_shuffled_replay('400')
```



4.2.9 Seed 408

```
plot_shuffled_replay('408')
```



4.2.10 Seed 415

```
plot_shuffled_replay('415')
```