

### Example Code Tutorial:

Description of the example code used in the set up and simulation of each treatment type.

Appilineni Kushal ([akushal@ucdavis.edu](mailto:akushal@ucdavis.edu))

All files and Matlab codes to our model is given here:

There are 2 aspects to our model, description of codes for each of them are given separately:

- 1) **Web generation and Fishing free part** – where we generate the webs using Niche model, run it for 4000 time steps with no fishing.
- 2) **Fishing under a certain management setting** – where we introduce fishing and a certain policy setting, select target and bycatch and run the system for further 1825 time steps.

### Web generation and Fishing free part

All codes for this are in the folder *Web generation and Fishing Free*.

First we generate webs using Niche Model. To do that we use *webgeneration.m*. It uses the *NicheModel.m* and creates **n** food webs, each with **spe** number of trophic species, with connectance **con** with an error **err**. See the code for more details. *webgeneration.m* can be used to generate new foodwebs. For each web, we store the foodweb (predator prey connections in form of a matrix), fish species of the web and initial biomasses. We generate 3000 webs to begin with, stored in *Webs3000.mat*.

These webs are then run for 4000 time steps with no fishing. We do that by using the *newwebfishingfree.m*. It takes *Webs3000.mat* and uses the following functions:

- 1) *webproperties.m* – Takes a foodweb, gives out the trophic level and species connected to basal species.
- 2) *newsetup\_default.m* – Takes foodweb and fish labels, initializes all the parameters (based on Table 1 in the manuscript).
- 3) *newdifferentialresp.m* – Generates the timeseries based on Equation 1 and 2 of the manuscript.
- 4) *isConnected.m* – Checks if the resulting foodweb after 4000 timesteps is still connected.

We use this to find out how many of those initially generated foodwebs (from *Webs3000.mat*) are “conserved”, after 4000 timesteps. The conserved foodwebs are stored in *newwebsUT4Fa4Fm4H12Hb2data.mat*. We use the first 400 food webs for our analysis.

### **Fishing under a certain management setting**

We considered 3 scenarios i.e. Open Access, Shared target policy and Shared target and bycatch policy. All files pertaining to each of those are in those respective folders. To generate timeseries, we use the following functions - *discreteeffortsoopenaccess.m*, *discreteeffortssharedtarget.m* and *discreteeffortssharedtargetbycatchquota.m*. Each of these functions takes the following inputs:

- 1) Simhbdata - set of all conserved foodwebs
- 2) k - index of the particular foodweb, who's timeseries is required
- 3) N – number of fishers
- 4) costratio – scaling parameter used in marginal costs
- 5) cat – catchability of target species
- 6) cab – catchability of bycatch species
- 7) cutofft – cutoff threshold for target species, below which fishery will be closed
- 8) cutoffb – cutoff threshold for bycatch species, below which fishery will be closed
- 9) nharvest – number of target species to be harvested
- 10) nbycatch – number of bycatch species
- 11) inE – maximum effort
- 12) nseasons – number of seasons (in multiple of 365 days)
- 13) hill – Hill's coefficient
- 14) hsb – Half saturation biomass
- 15) fa – fraction of assimilated carbon used for production
- 16) fm – factor for maintenance respiration
- 17) target – index indicating the target species, among all species in food web
- 18) bycatch – index indicating the bycatch species, among all species in food web

Each of them also take inputs, the following functions:

- 1) *newsetup\_default.m* – Takes foodweb and fish labels, initializes all the parameters (based on Table 1 in the manuscript).
- 2) *d\_seasonalcatchV2.m* - Generates the timeseries based on Equation 1 and 2 of the manuscript, when fishing is still profitable.
- 3) *newseasonclosure.m* - Generates the timeseries for the rest of the season based on Equation 1 and 2 of the manuscript, after fishery is closed.

*dataforrandomtarget.mat* – gives the target and bycatch indexes for each foodweb. It stores information (in form of indexes) for all possible target and bycatch combinations in each foodweb.

*minbynetworkindex.mat* – which indexes (among all possible target bycatch combinations from *dataforrandomtarget.mat*), correspond to most vulnerable bycatch.

Each of the timeseries generation functions - *discreteeffortsopenaccess.m*, *discreteeffortssharedtarget.m* and *discreteeffortssharedtargetbycatchquota.m* – returns the timeseries for 5 years with fishing (1826 time steps). First 30 columns refer to the biomasses of the species in the foodweb, 31 - 35 refers to the effort of the 5 fishers, 36 - 40 are the cumulative target yield of each fisher (in order i.e. 36 refers to the cumulative target yield (over time) of the fisher who's efforts are given by column 31 and so on), 41 – 45 are the cumulative bycatch yield of each fisher (in order).

After the time series for each of the 400 webs is generated (for both all both scenarios - random bycatch and vulnerable bycatch) , the properties of each web i.e. target survivability, bycatch survivability, biomass change, target extinctions, bycatch extinctions, secondary extinctions, target yield, bycatch yield, profits and fishing season was calculated using the following commands:

1. *simulationresults\_survextinct.m* – Calculated mean, median and mode for all the above observables for the scenario where bycatch is chosen randomly. The output of these are stored in the form of *newmedian\_openaccess\_data.mat* (for open access time series), *newmedian\_survST\_cutoff* (when only Shared target policy is in place) and *newmedian\_survSTB\_cutoff\_cutoffb* (when both Shared target and bycatch policy is in place).
2. *simulationresults\_MinBysurvextinct.m* - Calculated mean, median and mode for all the above observables for the scenario where bycatch is chosen as the most vulnerable species. The output of these are stored in the form of *newmedian\_minby\_openaccess\_data.mat* (for open access time series),

newmedian\_MinBy\_survST\_cutofft (when only Shared target policy is in place) and Newmedian\_minby\_survSTB\_cutofft\_cutoffb (when both Shared target and bycatch policy is in place).

Henceforth, I'll refer to these files as newmedian data files.

We used this newmedian data to compile information about all relevant observables for different  $esc_{target}$  and  $esc_{bycatch}$  in newmedian\_policydata\_randombycatch.mat and newmedian\_policydata\_vulnerablebycatch.mat. Each cell entry contains data for all escapement levels used in this study.  $esc_{target}$  are selected between [0.15, 0.3, 0.45, 0.6] and changes along the columns.  $esc_{bycatch}$  are selected between [0, 0.15, 0.3, 0.45, 0.6] and changes along the columns.

Description of each cell entry of the abovementioned policydata files and how it was obtained from newmedian data:

- 1) Median fraction of Target surviving (year 5 of fishing/penultimate fishing free year, 3<sup>rd</sup> entry of newmedian data)
- 2) Median fraction of Bycatch surviving (year 5 of fishing/penultimate fishing free year, 6<sup>th</sup> entry of newmedian data)
- 3) Mean target extinctions (since fishing started, 10<sup>th</sup> entry of newmedian data)
- 4) Mean bycatch extinctions (since fishing started, 11<sup>th</sup> entry of newmedian data)
- 5) Mean secondary extinctions (since fishing started, 12<sup>th</sup> entry of newmedian data)
- 6) Mean target yield (only year 5 of fishing, 13<sup>th</sup> entry of newmedian data)
- 7) Mean bycatch yield (only year 5 of fishing, 14<sup>th</sup> entry of newmedian data)
- 8) Mean profits (only year 5 of fishing, 15<sup>th</sup> entry of newmedian data)
- 9) average fishing year (only year 5 of fishing, 16<sup>th</sup> entry of newmedian data)

E.g. newmedian\_policydata\_randombycatch.mat{5} would give the mean secondary extinctions for each target (bycatch) escapements along the column (row).

Corresponding data for open access was stored in policy\_openaccessdata\_randombycatch.mat and policy\_openaccessdata\_vulnerablebycatch.mat

*heatmapcommands.m* was used to interpolate the policydata and plot heatmaps Figure 3 and 4.