

# Explanation of Mathematical Methods Behind the Second Half of the Project

AUTHOR

Emma Bowen

## Introduction

What mathematical methods will I explain in this document that have not been explained in [Explanation of Mathematical Methods from First Half of the Project?](#)

- The basics of the MixME model
  - Creating the operating model
  - Building the MixME input object
  - Running the simulation
  - Analysing the results of the simulation
  - HCR parameters
  - Additions or differences in the shortcut model
- Calculating the risk
- Parallelisation

HIGHLIGHT MORE THAT FTARGETS ARE STAYING THE SAME THROUGHOUT THE SIMULATION - found in MixME wiki Fixed fishing mortality management strategy

## Context

The aim of this half of the project is to apply the methods from the first half of the project to mixed fisheries using the MixME R package. I have used the methods from the first half of the project to write code that follows a very similar process, but where the aim is now to find the  $F_{target}$  for each stock that is precautionary but maximises total catch over the years 2030 to 2039. I set this goal because MixME is designed to run projections into the future, and looking at catch in the long term will minimise the chance that a simulation where a stock fails is chosen ([Pace et al. 2025a](#)),([Pace 2024](#)). The 20 year projection from 2020-2039 was kept consistent with examples on the MixME documentation and follows guidelines from ICES to create long-term projections based on the biology of the stocks ([Pace 2024](#)),([ICES 2019a](#)). It also follows ICES guidelines to only calculate catch and risk for the last ten years for long-term projection, to allow time for a recovery period ([ICES 2019a](#)).

I focused on the datasets from the Fixed fishing mortality management strategy example (`mixedfishery_MixME_om`) and the Exploring simulation outputs example (`mixedfisherv_MixME_input`) which are both in the MixME documentation ([Pace 2024](#)).

(`mixedfishery_MixME_input`), which are seen in the MixME documentation ([see 2.2.1](#)).

However, for the second dataset `mixedfishery_MixME_input` I was given a shortcut method by a researcher at Cefas which takes a more direct approach to the simulation. The code for these datasets is `Optimising_ftarget_in_MixME_mult_points_parallel.R` and `Two_stocks_Optimising_ftarget_in_shortcut_model.R` respectively.

Both of these datasets have two stocks (cod and haddock) and two fleets ([Pace 2024](#)). I was told by the same researcher at Cefas that the stocks are North Sea cod and Celtic Sea haddock, but using citations I can only back up that they are Atlantic cod and haddock ([Pace 2024](#)), ([ICES, n.d.](#)), ([ICES, n.d.](#)). This allows me to use the same methods as in the first half of the project by replacing  $F_{target}$  with  $F_{cod}$  and  $B_{trigger}$  with  $F_{had}$ , where  $F_{cod}$  and  $F_{had}$  are the fishing mortalities for cod and haddock respectively.

For this half of the project, we have switched to modelling SSB directly instead of calculating the risk. This is due to the simulation being deterministic as both datasets only have one iteration and the noise for this is pre-calculated ([Pace et al. 2025b](#)). These conditions simplify the simulation but unfortunately mean that we cannot use the standard ICES definition of risk to calculate  $Risk = P(SSB < B_{lim})$  ([Pace et al. 2025b](#)), ([ICES 2019a](#)). Instead, we extract  $\min(SSB)$  for each sampled point from the years 2030-2039 in the simulation and then model these as a GP ([Spence 2025](#)), ([ICES 2019a](#)). This allows us to predict the distribution of  $\min(SSB)$  values at each point in the design space ([Spence 2025](#)). For each point, we then see how many of these predicted values fall below  $B_{lim}$  to calculate  $P(\min(SSB) < B_{lim})$  at that point ([Spence 2025](#)). We do this for each stock. This method of calculation satisfies the ICES precautionary standard ([ICES 2019a](#)).

Due to the change in stocks being modelled, I have decided to keep the GP prior for catch modelled by the `~.^2` function in R for every round of the optimisation process. Despite the more complicated prior used in later rounds in `case_study8.R` being designed to approximate yield curves, it may not be appropriate in a mixed fisheries context where the catch of one species is affected by the catch of another ([Spence 2025](#)), ([Pace et al. 2025a](#)), ([Ulrich et al. 2012](#)). Leaving the GP more general avoids mis-specification, ensuring that the GP can be appropriately fitted to the points ([Williams and Rasmussen 2006](#)).

The process for the first half of the project adapted to our new situation is outlined below. We take a Bayesian History Matching (BHM) approach ([Spence 2025](#)). Firstly, to get some initial data, we randomly sample our first set of  $n - 1$  points, where  $n$  is the number of cores the system we are on has ([Spence 2025](#)). Then, for each round we do the following:

- We set up or update the Gaussian Processes (GPs) to model the  $\min(SSB)$  from 2030-2039 for each stock and the GP to model the total catch from 2030-2039
- We can then use the  $B_{lim}$  for each stock as a threshold so that we only consider the values of  $F_{cod}$  and  $F_{had}$  that have  $P(\min(SSB) > B_{lim}) > 0.05$  for all of the last ten years of the simulation (2030-2039)
- We use the GP which is modelling the total catch to predict the value for the total catch at every point in the sample space, which will have some uncertainty
- We use BHM to remove any points that are implausible (that have a probability less than 0.01% of being higher than the current best total catch)

0.01% of being higher than the current best total catch)

- We use the Knowledge Gradient (KG) acquisition function to select  $n - 1$  plausible points to sample in the next round, as per the discussion in [Deciding which acquisition function is best](#)

We repeat this process until there is only one plausible point left and then we will accept this as being the  $F_{cod}$  and  $F_{had}$  that maximise the catch whilst keeping the  $SSB$  above  $B_{lim}$ .

## Calculating the risk

MixME defines risk as the proportion of iterations in each year where  $SSB$  falls below  $B_{lim}$  ([Pace 2024](#)). However, due to only having one iteration in each of my datasets, I have experimented with using a different method to measure the risk ([Pace 2024](#)).

In contrast to the first half of the project, we now calculate the risk using the  $SSB$ . We have described it above briefly but will go into more detail here. We should quickly note before our calculations that our new sample space is a grid with all the combinations of  $F_{cod}$  and  $F_{had}$  both ranging from 0 to 0.6 in 0.02 increments.

This range ensures that we are able to model stock collapse for cod by modelling values above  $F_{lim}$  and that we can model stock recovery for cod by including very low values ([ICES 2019c](#)). It also allows us to model unsafe fishing for haddock by modelling values above  $F_{MSY}$  ([ICES 2019b](#)). Recalling that cod is the choke stock for both of our datasets, this sample space is appropriate ([Pace 2024](#)).

We also have a  $B_{lim}$  for each stock, taken from ICES advice ([ICES 2020b](#)), ([ICES 2020a](#)). The  $B_{lim}$  for cod is 107,000 tonnes and the  $B_{lim}$  for haddock is 9,227 tonnes.

Firstly, we extract the  $\min(SSB)$  for each stock from the result of the simulation we have run using the tracking object ([Pace 2024](#)), ([Pace et al. 2025a](#)). We then put these results into the GPs we will use to model  $\min(SSB)$  for each stock.

These GPs have remained modelled by the  $\cdot^{\wedge}2$  function, the same as in the GP for risk in the first half of the project to avoid mis-specification ([Williams and Rasmussen 2006](#)), ([Roustant, Ginsbourger, and Deville 2012](#)). They have also kept the same estimation method, nugget and kernel ([Williams and Rasmussen 2006](#)). - JUSTIFY MORE

Next, we predict the  $\min(SSB)$  for every point in the sample space and calculate  $Risk = P(SSB < B_{lim})$  for each stock. Then, we set the KG of any unsafe points to be zero so that they will not be chosen as points to be sampled in the future.

By ensuring that  $P(\min(SSB) < B_{lim}) < 0.05$  we ensure that  $P(SSB < B_{lim}) < 0.05$  for every year in the long term forecast (2030-2039). This guarantees that the maximum annual risk in this period remains below the 5% threshold, which is exactly what is required to meet the ICES precautionary standard due to their definition of  $Prob3$  in ([ICES 2019a](#)). This means that this risk calculation could be used in policy documents to set official catch limits in countries that have agreed to this standard ([ICES 2019a](#)).

## The basics of the MixME model

Going to write down everything and then remove areas Gustav said I don't need to worry about.

- Explaining the tracking object as this how I get catch ?? - MAYBE IN A SECTION SUMMARISING FULL CODE PROCESS?

### Starting $F_{cod}$ and $F_{had}$

We started with  $F_{cod} = 0.28$  and  $F_{had} = 0.353$  based upon the values given in the Fixed fishing mortality management strategy example from the MixME wiki ([Pace 2024](#)). These were drawn from the North Sea Cod and Celtic Sea Haddock advice published in 2020 respectively ([ICES 2021](#)),([ICES 2020b](#)).

### Creating the Operating model

We first create the operating model `mixedfishery_MixME_om` which contains the true data for the stocks and fleets in the dataset ([Pace et al. 2025a](#)). We have data for North Sea Cod from 1963-2019 and for Celtic Sea Haddock from 1993-2019 ([Pace 2024](#)). For the stocks, this includes the numbers, natural mortality, stock mean individual weight and proportion mature split by age and for fleets this includes landing numbers, landing mean individual weight, discard numbers, discard mean individual weight, selectivity and catchability ([Pace 2024](#)). - **DISCUSS CATCHABILITY?**

In both of our files, this is loaded with the dataset ([Pace 2024](#)). This true data is generated using standard age-structured equations to model the dynamics of the fleets and the stocks ([Pace 2024](#)). For every year, it has the catch (in terms of landings and discards) from each fleet and the survivors from that year ([Pace 2024](#)). It also contains a stock-recruitment model and recruitment is done at the beginning of each time step ([Pace 2024](#)).

The steps for fully assembling the operating model for input into the MixME model are:

1. Estimate historic quota-share for the two fleets
2. Project stocks n years into the future
3. Calculate numbers of both stocks in initial year
4. Generate an observation error model

### Estimate historic quota-share for the two fleets

Firstly, we use `mixedfishery_MixME_om` to determine the quota share of the catch for each fleet. This is done by assuming for each stock that the quota share corresponds to the proportional share of landings and is carried out by the `calculateQuotashare` function ([Pace 2024](#)).

```
out <- calculateQuotashare(stks = mixedfishery_MixME_om$stks,
  flts = mixedfishery_MixME_om$flts, verbose = TRUE)
```

IS THIS IT?

## Project stock $n$ years into the future

To carry out our 20 year projection, we need to extend the stock and fishery structures forward from 2019 into 2039. There are three categories of parameters that are not estimated dynamically and so need to be extended ([Pace 2024](#)). These are:

1. All stock parameters, landings and discards mean individual weights and landed fraction
2. Catchability and catch selection
3. Quota-share

We project the parameters in each of these categories from 2019-2039 using the average from the last three years ([Pace 2024](#)).

## Calculate numbers of both stocks in initial year

To be able to do our projections of catch, we need to know the number of each stock in each age class at the beginning of the first projection year, 2020. This requires us to do a 1-year short term forecast from our starting point of 2019 using the FFlasher package ([Pace 2024](#)). We set an arbitrary forecast target for this forecast because the 2020 numbers are calculated from existing numbers (rather than a defined target) ([Pace 2024](#)). - ? FIND OUT MORE

## Creating the Observation Error Model

This is another important component of the MixME model ([Pace 2024](#)). We create the observation error model `stk_oem` where we apply pre-sampled noise to the catch from each fleet (which we obtained earlier from the `mixedfishery_MixME_om` object) ([Pace 2024](#)). We generate future stock and management advice from this object ([Pace 2024](#)). - MAY NEED TO REFERENCE MORE COMPLICATED TUTORIAL FOR FULL EXPLANATION OF THIS

## Build the MixME input object

We then use this in the `makeMixME` function to make the MixME input object, which can then be used to run the simulation. We have five arguments for this function. Two of these are where we input our Operating Model and our Observation Error model. The next two specify how many time steps (in our scenario, years) it takes for the management advice to be enacted and what type of management we are using respectively ([Pace 2024](#)). The last argument - NOT DESCRIBED YET

## Running the simulation

We update some of the management and simulation settings and then run the simulation ([Pace 2024](#)).

We update the arguments for when catches occur, the age range for calculating average fishing mortality in all the places it is needed and the target fishing mortality we are using for this round of the simulation ([Pace 2024](#)). - FIND OUT MORE ABOUT WHY NEEDED AND IMPORTANT IF POSS

It is important to note that the  $F_{cod}$  and  $F_{had}$  we choose remain constant throughout the simulation (Pace 2024). - SURELY IN SUPP OR PAPAER TOO?

## Analysing results of the simulation

The `tracking` object records summary statistics for the modelled stock and fleet dynamics, as well as metrics describing the observed state of the system by the management procedure. It also contains simulation performance and diagnostics statistics, which is what we will focus on here (Pace 2024).

We check for management advice failure, effort optimisation failure and the message given if there was any failure (Pace 2024). Effort optimisation failure can be sued to see if we are over-fishing the stocks to a point that they will go extinct (Pace 2024).

Other results we can see are the over-quota catches and the quota uptake (Pace 2024). Lastly, we can check which stock is the choke stock (Pace 2024).

## HCR parameters

We add  $B_{lim}$  into the HCR to help us calculate the risk later on (Pace 2024). It is helpful for calculating it in our current way, or the way MixME defines it as: "the proportion of iterations (read replicates) in each year where stock spawning biomass falls below the biological limit reference point ( $B_{lim}$ )" (Pace 2024).

ARE THERE OTHER hcrpars ALREADY DEFINED?

## Additions or Differences in the Shortcut Model

NEED CITATIONS FROM SOMEWHERE BUT MAY BE VERY DIFFICULT AS NOT PUBLIC YET

This is the `Two_stocks_Optimising_ftarget_in_shortcut_model.R` file. We use the same data here as in the `Optimising_ftarget_in_MixME_mult_points_parallel.R` file, except that we have now also loaded in the Operating Model and Observation Error Model with the data (MixMewiki?).

## Zero-catch advice

CHECK DEFINETLY NOT IN OLD METHOD

In this simulation, we have a condition meaning we return zero-catch advice if the  $SSB$  is below  $B_{lim}$  in the year after the advice year. We are checking the year after the advice year due to our management lag of one year. Firstly, we identify any simulation where  $SSB < B_{lim}$  and re-run these simulations specifically targeting  $B_{lim}$ .

However, if this still fails, we identify these runs in the `zeroTAC` variable and manually set the total annual catch (TAC) to zero.

MAYBE NOT ENOUGH INFO ON THESE TWO BELOW TO INCLUDE THEM?

## ICES\_HCR function



This is a new way to create the HCR.

### forecast\_fun function

This function projects forwards into the next year to set an appropriate TAC. We call this for every year of the simulation and it is similar to the short term forecast we used in the last file.

### Setting forecast arguments

We set up the forecast arguments in a similar way to before, but make sure to set them separately for each stock.

## Parallelisation

---

This process has been set up so that it is easy to run in parallel. We are able to load in the data we need once and then run the simulations for each point we have chosen to sample in the round in parallel. As they run, each process only gathers the values we need and removes the rest of the results of the simulation to save memory. Here, the values we need are the  $\min(SSB)$  for cod, the  $\min(SSB)$  for haddock and the total catch of cod and haddock all over the years 2030-2039.

Then, we collect these results and set up the GPs for each of them. After this, we run through the process of removing any implausible points and selecting the points to sample in the next round in a very similar way to the first half of this project. We iterate this process until the optimal point(s) are found. - TACKLE WHEN I AM ABLE TO TACKLE WHY I AM GETTING MULTIPLE POINTS AS AN ANSWER (LIKELY DUE TO HAVING SAME FCOD AND COD BEING THE CHOKE?)

The process above makes up one run of the script. To ensure that we are consistently receiving the same point(s) as our optimal point(s), we want to follow on from the original paper and run this script 1000 times ([Spence 2025](#)). This requires a new kind of parallelisation. We were able to accomplish this by creating an array of jobs which will run when there is free space on the Viking HPC used by the Univeristy of York ([York, n.d.](#)). This allowed us to submit one job that would run the script 1000 times, making it a lot easier to do this than having to submit 1000 separate jobs ([York, n.d.](#)). - SAY ABOUT MY 50 THROTTLE AS WELL?

## Further areas for Development

### Moving to a more complicated dataset

---

Mention the one in the MixME paper.

### Doing a proper risk calculation

---

I now meet precautionary, so would it be much of an improvement to use probabilities instead?

## References

- ICES. 2019a. "Workshop on Guidelines for Management Strategy Evaluations (WKG MSE2)," January. <https://doi.org/10.17895/ices.pub.5531>.
- . 2019b. "Haddock (*Melanogrammus aeglefinus*) in divisions 7.b–k (southern Celtic Seas and English Channel)," June. <https://doi.org/10.17895/ices.advice.4785>.
- . 2019c. "Cod (*Gadus morhua*) in Subarea 4, Division 7.d, and Subdivision 20 (North Sea, eastern English Channel, Skagerrak)," November. <https://doi.org/10.17895/ices.advice.5640>.
- . 2020a. "Cod (*Gadus morhua*) in Subarea 4, Division 7.d, and Subdivision 20 (North Sea, eastern English Channel, Skagerrak)," June. <https://doi.org/10.17895/ices.advice.5891>.
- . 2020b. "Haddock (*Melanogrammus aeglefinus*) in divisions 7.b–k (southern Celtic Seas and English Channel)," October. <https://doi.org/10.17895/ices.advice.5897>.
- . 2021. "Benchmark Workshop on North Sea Stocks (WKNSEA)," April. <https://doi.org/10.17895/ices.pub.7922>.
- . n.d. "ICES-FishMap Cod." <https://www.ices.dk/about-ICES/projects/EU-RFP/EU%20Repository/ICES%20FishMap/ICES%20FishMap%20species%20factsheet-cod.pdf>.
- Pace, Matthew. 2024. "MixME Wiki." <https://github.com/CefasRepRes/MixME/wiki>.
- Pace, Matthew, José Oliveira, Simon Fischer, and Paul Dolder. 2025a. "MixME: An r Package to Simulation-test Fisheries Management Robustness to Mixed-fisheries Interactions." *Methods in Ecology and Evolution* 16 (February): 698–706. <https://doi.org/10.1111/2041-210X.70005>.
- . 2025b. "Supplementary Material a: Conditioning a Multi-Stock, Multi-Fleet Operating Model for Celtic Sea Cod, Haddock and Whiting," February. <https://besjournals.onlinelibrary.wiley.com/action/downloadSupplement?doi=10.1111%2F2041-210X.70005&file=mee370005-sup-0001-Supinfo1.pdf>.
- Roustant, Olivier, David Ginsbourger, and Yves Deville. 2012. "DiceKriging, DiceOptim: Two r Packages for the Analysis of Computer Experiments by Kriging-Based Metamodeling and Optimization." *Journal of Statistical Software* 51 (1): 1–55. <https://doi.org/10.18637/jss.v051.i01>.
- Spence, Michael A. 2025. "Using History Matching to Speed up Management Strategy Evaluation Grid Searches." *Canadian Journal of Fisheries and Aquatic Sciences* 82: 1–14. <https://doi.org/10.1139/cjfas-2024-0191>.
- Ulrich, Clara, Douglas C. K. Wilson, J. Rasmus Nielsen, Francois Bastardie, Stuart A. Reeves, Bo S. Andersen, and Ole R. Eigaard. 2012. "Challenges and Opportunities for Fleet- and Métier-Based Approaches for Fisheries Management Under the European Common Fishery Policy." *Ocean & Coastal Management* 70: 38–47. <https://doi.org/10.1016/j.ocecoaman.2012.06.002>.
- Williams, Christopher KI, and Carl Edward Rasmussen. 2006. *Gaussian Processes for Machine Learning*. Vol. 2. 3. MIT press Cambridge, MA.
- York, University of. n.d. "Viking Documentation." <https://vikingdocs.york.ac.uk/index.html>.