SCRS/2024/141

TECHNICAL MSE DEMONSTRATION FOR ATLANTIC BLUE SHARK

T. R. Carruthers[1]

SUMMARY

Management Strategy Evaluation (MSE) may be perceived as a technically complex process that necessarily takes months or even years of coding and technical development time. Recent advances in open-source MSE software have substantially reduced this technical overhead. I provide a demonstration of the technical components of MSE for Atlantic Blue Shark including operating model specification, management procedure (MP, a.k.a. 'harvest strategy') design, MP derivatives, MP tuning, closed-loop MSE calculations, performance metrics, presentation of MSE results and exceptional circumstances protocols. This demonstration is intended to underline the relative ease, accessibility and flexibility of software designed to facilitate rapid and efficient development of MSE frameworks.

KEYWORDS

Management strategy evaluation, operating model, management procedure, harvest strategy.

[1] Blue Matter Science Ltd. 2150 Bridgman Ave, N. Vancouver. V7P 2T9.

Introduction

The time taken to develop management strategy evaluation (MSE) frameworks varies substantially among stocks.

Some of the development time can be attributed to the technical aspects of MSE which typically includes the following components (see Carruthers 2024 for how these fit into the broader MSE framework, Figure 1):

1) specifying operating models (OMs),
2) coding closed-loop simulation frameworks,
3) developing management procedure (MP, a.k.a. 'harvest strategy') archetypes (e.g., index-target, index-ratio, model-based etc),
4) defining MP derivatives (e.g., maximum TAC change, maximum TAC, etc.),
5) tuning MPs (e.g. to achieve a particular probability of overfishing),
6) running MSE calculations,
7) calculating performance metrics
8) presenting MSE results
9) defining exceptional circumstances protocols.

Recently, regional MSE processes such as those for Chilean northern hake and anchovy, and Canadian groundfish in B.C. have made use of modern MSE software that is more efficient and powerful, allowing rapid progress in the technical aspects of MSE listed above, and the adoption of management procedures in a matter of months (Haggarty et al. 2022a).

This step-change in the technical accessibility of MSE is demonstrated in this paper (see also Huynh et al. 2020) with an example MSE framework for Atlantic blue shark in which all the technical aspects listed above were completed using the open-source package *OpenMSE* (Hordyk et al. 2024a), the MSE presentation app *Slick* (Hordyk et al. 2024b) and the ECP exploration app *ECP* (Carruthers 2024).

Methods

All code for completing the following MSE steps is available on the public GitHub repository 'blue-matter/Blue_Shark_MSE' and in Appendix A.

1. Specifying operating models

Operating models were specified using Run 6 (alternative index and length composition weighting) of the 2015 Stock Synthesis assessment for Atlantic blue shark (Anon 2015) that was fitted to data up to and including 2013.

The use of an older assessment is deliberate and reinforces that this analysis is a technical demonstration and is not relevant to current policy making. More recently, operating models have been developed for Atlantic blue shark using RCM (Rapid Conditioning Model) of OpenMSE but these are still in development.

The important take-home message is that existing data for blue-shark are available to condition defensible operating models and that it is a single function to convert these to an OpenMSE operating model:

```
> OM = SS2OM('C:/shark_assessment')
```

Or

```
> OM = RCM(stock_parameters, data)@OM
```

These functions include check that the OpenMSE operating model exactly matches the dynamics of the estimation model (Stock Synthesis, RCM). This has proven a significant issue for bespoke custom-coded MSE frameworks elsewhere.

A reference grid of operating models was specified with three factors: natural mortality rate ('M'), steepness of the Beverton-Holt stock-recruitment curve ('h') and current stock depletion ($SSB_{2013}/SSB_{unfished}$, 'Depln'). Alternative levels of M were arbitrarily set as 3/4 and 4/3 of the stock assessment M-at-age vector, alternative levels of steepness were set at 0.6 and 0.9 (base assessment value was 0.73) and depletion was set at 2/3 and 3/2 of that estimated by the base assessment (the reference OM grid is summarized in Table 1). This OM grid in intended to encompass the three most important aspects of stock uncertainty in the determination of relative MP performance (productivity, resilience and status, respectively). Such axes are typical in other MSE processes such as that of Atlantic bluefin tuna (Carruthers et al. 2020 and North Atlantic swordfish (Hordyk et al. 2021).

2. Coding closed-loop simulation frameworks

Rather than code the MSE from scratch, the R package OpenMSE (Hordyk et al 2024, OpenMSE 2024) was used to do the age-structured stock and fishery calculations.

Observation error models for catches and indices are derived automatically from the historical fit of the operating models to the observed data and can include imprecision, autocorrelation and hyperstability / hyperdepletion (only imprecision and autocorrelation were selected in these simulations).

3. Developing MP archetypes

Three MP archetypes were developed that broadly follow the concepts of candidate MPs developed for Atlantic bluefin tuna and North Atlantic swordfish:

- Index target (*It*) - reduces TAC when index is below the target level, increases TAC when index is above target level (tuned by adjusting the index target level)
- Index ratio (*Ir*) - fishes at a constant multiplier of the recent index level, i.e. a constant F policy (tuned by adjusting the ratio)
- Index slope (*Is*) - aims to achieve a constant slope in the index and reduces TAC when slope is below target and increases TAC when slope is above target (tuned by adjusting target slope).

In this demonstration all MPs used assessment index 9 that had an observation error of approximately 0.25 (coefficient of variation) and low lag-1 autocorrelation in residuals (~0.2). Index target and index ratio MPs calculated recent index (for comparison with target and calculation of the TAC based on ratio) as the mean index over the last three years. The index slope MP used the slope in the index over the last 5 years (index standardized to mean 1, see Appendix A for MP code).

These MPs were assumed to have a 1-year lag in the index data and provided new TAC advice every year. MP advice was assumed to be taken exactly (perfect implementation).

4. Defining MP derivatives

Three derivatives of each MP archetype were developed:

- Max TAC change of 10% (*It_10*, *Ir_10*, *Is_10*)
- Max TAC change of 30% (*It_30*, *Ir_30*, *Is_30*)
- Max TAC change of 30%, maximum TAC of 30kt (*It_M30*, *Ir_M30*, *Is_M30*)

Although these derivatives are chosen somewhat arbitrarily in this case, they broadly reflect those requested by ICCAT managers for the Atlantic bluefin and North Atlantic swordfish MSE processes.

5. Tuning MPs

The purpose of MP tuning is to better reveal performance differences among MPs by controlling for one of the major performance axes: catch or biomass conservation. In this case MPs were tuned (adjusted index target, index ratio, index slope) to achieve probability of green kobe (F<FMSY & SSB > SSBMSY) of 60% (all 50 projection years, all operating models). The tuned versions of each were labelled with '_t' (e.g., *Ir_30_t*).

Tuning is achieved using the openMSE function tune_MP() in which the user defines the MP, the tuning parameter, the operating models and a function to be minimized (in this case the squared difference in PGK from that achieved at the given tuning parameter level and the desired 10k). PGK tuning was used in both North Atlantic swordfish and Atlantic bluefin tuna MP development.

6. Running MSE calculations

The OpenMSE libraries conduct the age-structured stock and fishery calculations using C++ code that is much faster than native R code, leading to relatively fast computation times. Calculations are divided into historical and projection phases, meaning that reference points and historical stock dynamics only have to be calculated once, and not each time a projection of that operating model is conducted for a new management procedure.

7. Calculating performance metrics

The North Atlantic swordfish MSE currently summarises top-level results according to five metrics:

- *AvTAC_short*, *AvTAC_med*, *AvTAC_long* – the mean TAC set over projection years 1-10, 11-20 and 21-30, respectively
- *nLRP* – probability of not being below the biological limit reference point of 60% SSBMSY over the first 30 projection years
- *PGK*, *PGK_short*, *PGK_med* – probability of green kobe (F<FMSY & SSB>SSBMSY) over all 50 projection years (North Atlantic swordfish is 30 years), projection years 1-10 and 11-20, respectively.
- *PNOF* – probability of not overfishing (F<FMSY) over all 30 projection years
- *VarC* – absolute change in TAC among years

Recognizing that an MSE for Atlantic blue shark would necessarily require a process of stakeholder and manager engagement to identify appropriate performance metrics specific to blue shark, these metrics for North Atlantic swordfish encapsulate the primary performance attributes of MPs: what is caught now (*AvTAC_short*, *AvTAC_med*), the biomass that is left over (*nLRP*, *PGK*, *PGK_med*), what can be caught later (*AvTAC_long*) and how much catch advice varies (*VarC*). These metrics also include overfishing metrics that are relevant to various stakeholder groups (*PNOF*).

8. Presenting MSE results

Results were summarized by the Slick app (Hordyk et al. 2024), a dedicated R package and online app (also can be run locally) for presenting MSE results across the key MSE axes: operating models, management procedures and performance metrics.

9. Defining exceptional circumstances protocols

When an adopted MP is in use, exceptional circumstances protocols (ECP) are an empirical check that new observations of data are consistent with those predicted by the operating models. For example, if observed indices used by the adopted MP are declining fast and to lower levels than any predicted by the OMs, then this may be considered exceptional and require a review of the operating model dynamics.

In this case it is assumed that the Ir_10 MP was adopted and instead of real data observations, a single simulated data set is compared with the data projected by the operating model to demonstrate ECP design and diagnostics using the ECP R package and app (Carruthers 2024). For demonstration purposes indices 8 and 9 were used to investigate ECP attempting to minimize overall Type I error (probability of falsely triggering ECP) while maximizing power (probability of correctly identifying problematic simulations). Here ‘problematic’ was arbitrarily defined as a simulation where SSB falls below 75% of SSBMSY at some point in the projection.

A note on simulation frequency

This demonstration MSE was based on relatively few simulations to allow the demonstration code to be run quickly by new users. In this case, only 12 simulations per operating model (96 in total) were specified allowing all of the code to run in less than 20 minutes on a laptop. Typically, more than 150 simulations per operating model (1200 in total) would be necessary to obtain the required precision in calculated performance metrics (taking closer to 5 hours in total). Since most of the calculations can be run in parallel, computation times can be reduced dramatically by using cluster computing.

Results

The projected stock status varied strongly among operating models (Figure 2). Operating model 2 (high M, low steepness, depleted) was the most challenging for the MPs and the stock crashed (median) for all of the MPs.

The index ratio MP provided the most consistent performance across OMs (Figure 2) crashing the stock in only operating model 2. Index ratio MPs provided the highest yields in the medium and long terms with the lowest TAC variability and a probability of green Kobe greater than 70% (Table 2).

In general, MP derivatives provided comparable performance outcomes (much bigger differences were seen among MP archetypes) (Figures 3 and 4, Table 2). There appeared to be little cost of imposing a 10% limit in TAC change (given annual updates this is perhaps understandable).

While achieving the same 10k mean catch over all OMs, the index ratio MP (Ir_10) provided outcomes closer to SSBMSY and FMSY (Figures 5-8).

The tuned MPs never dropped below the biological limit reference point of 60% SSBMSY (Table 2).

The example ECP (using a simulation of data rather than real data) shows data broadly consistent with posterior predictions although index 9 falls out of the 97.5% interval (Type I error = 2.5% per index per year) at the upper bound in the first projection year (2014) (Figures 9 and 10). The alternative set of simulations where problematic conditions occur (SSB below 75% SSBMSY at some point in the projection) overlapped strongly with those where this did not occur (Figure 11). For index 9 (used by the MP) problematic conditions were indicated by index observations that were relatively high in early years (2014 and 2015) and low in later years (2017 – 2019) (Figure 11) suggesting that evaluating an interval would be more powerful than just the lower bound (which would miss indicative data early in the projection). Posterior predicted data were somewhat correlated with each other and along years and again, the distribution of null (non problematic) and alternative (problematic) simulations strongly overlapped (Figure 12).

To obtain a 60% power to detect problematic simulations over 6 years (see plotted data for 2019, Figure 13) an ECP protocol using indices 8 and 9 would incur a cumulative type I error of 40% (a 4 in 10 chance of triggering ECP when data were consistent with the OM simulations). The power of the indicator and the relative power to Type I error was not improved by using only the lower tail of the data only (Figure 14), only using index 9 (Figure 15) or by using a higher annual Type I error rate (5% instead of 2.5%, Figure 16).

Discussion

Clearly this demonstration is focused on coding and calculation and does not alleviate other technical tasks associated with defining and selecting operating models. For example, this demonstration does not consider the suitability of OM fitting, OM weighting or the inclusion of robustness OMs. Other technical discussions over performance metrics and simulation of data were also avoided in this demonstration by simply adopting the same protocol as other ICCAT MSEs.

Nevertheless, while the coding and testing of MSE methods have previously taken several years in other settings, by using freely available open-source software these were implemented for Atlantic blue shark in a matter of weeks.

The demonstration of exceptional circumstances protocols confirmed the need for a principled approach to ECP design (selection of indices, type I error, selection of tails) based on the calculation of implied overall error and statistical power.

Code

The code for reproducing these analyses can be found at https://github.com/Blue-Matter/Blue_Shark_MSE

Acknowledgments

Many thanks to Adrian Hordyk whose code from the swordfish MSE process was used to populate the Slick object.

This work was funded by The Ocean Foundation.

References

Anon. 2015. Report of the 2015 Blue Shark Stock Assessment Session. Collect. Vol. Sci. Pap. ICCAT, 72(4): 866-1019 (2016)

Carruthers, T.R. 2020. Reference Set Operating models for Atlantic Bluefin Tuna assuming priors for Area-Specific Scale and western Stock Mixing. SCRS/2020/018. Retrieved from https://www.iccat.int/Documents/CVSP/CV077_2020/n_2/CV077020078.pdf [August 2022]

Carruthers, T.R. 2023. ECP: an R package and online app for the investigation and specification of exceptional circumstances protocols. https://github.com/Blue-Matter/ECP.

Carruthers, T.R. 2024. An updated roadmap for MSE development. SCRS/2024/XXX

Haggarty, D.R., Huynh, Q.C., Forrest, R.E., Anderson, S.C., Bresch, M.J., and Keppell, E.A. 2022a. Evaluation of potential rebuilding strategies for Inside Yelloweye Rockfish (Sebastes ruberrimus) in British Columbia. DFO Canadian Science Advisory Secretariat Research Document.

Haggarty, D.R., Siegel, M.R., Litt, M, A., Huynh, Q. 2022b. Quillback Rockfish Fishery and Conservation Objectives Workshop Summary Report. https://waves-vagues.dfo-mpo.gc.ca/library-bibliotheque/41059116.pdf

Huynh Q.C., Carruthers T., Mourato B., Sant'Ana R., Cardoso L.G., Travassos P., and Hazin, F. 2020. Demonstration of a MSE framework for western Skipjack tuna, including operating model conditioning. SCRS/2020/140. Collect. Vol. Sci. Pap. ICCAT 77:121-144.

Hordyk, A., Schirripa, M., Rosa, D. 2021. Updates to the operating models uncertainty grid for the North Atlantic swordfish MSE. SCRS/2021/099. https://iccat.int/Documents/CVSP/CV078_2021/n_7/CV078070240.pdf

Hordyk, A.R., Huynh Q., Carruthers, T.R.. 2024. OpenMSE: open-source libraries for conducting management strategy evaluation. www.openMSE.com

Punt, A.E., Butterworth, D.S., de Moor, C.L., De Oliveira, J.A.A., and Haddon, M. 2016. Management strategy evaluation: Best practices. Fish Fish. 17(2): 303–334. doi:10.1111/faf.12104.

Figures

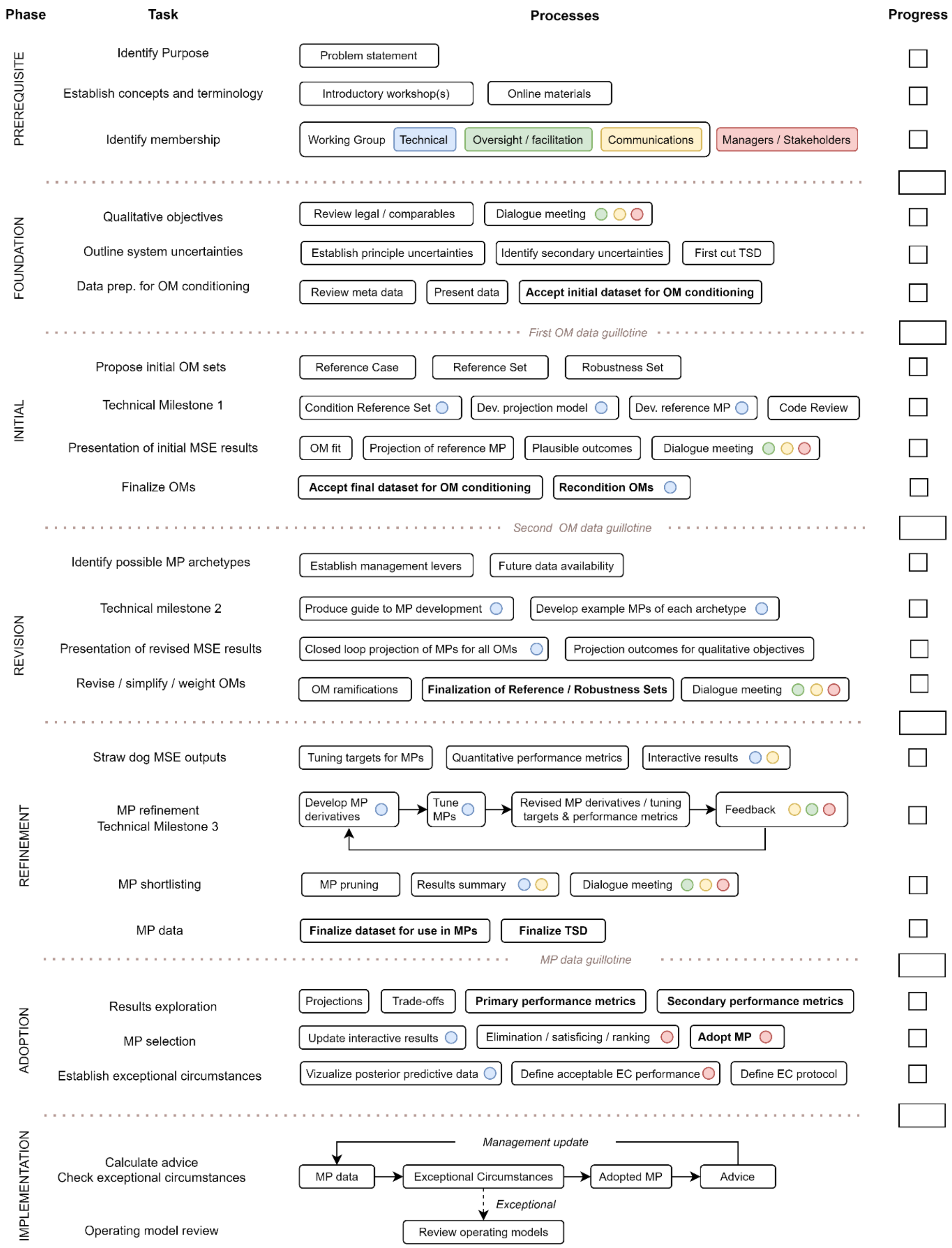

Figure 1. The updated MSE roadmap showing technical components (denoted by blue dots) in the context of the broader MSE process (Carruthers 2024). Unless specified by arrows, the process runs to the right and then downwards. Note that unless a specific group (colour) is assigned to a process (just a white box), all members of the working group are invited to participate.

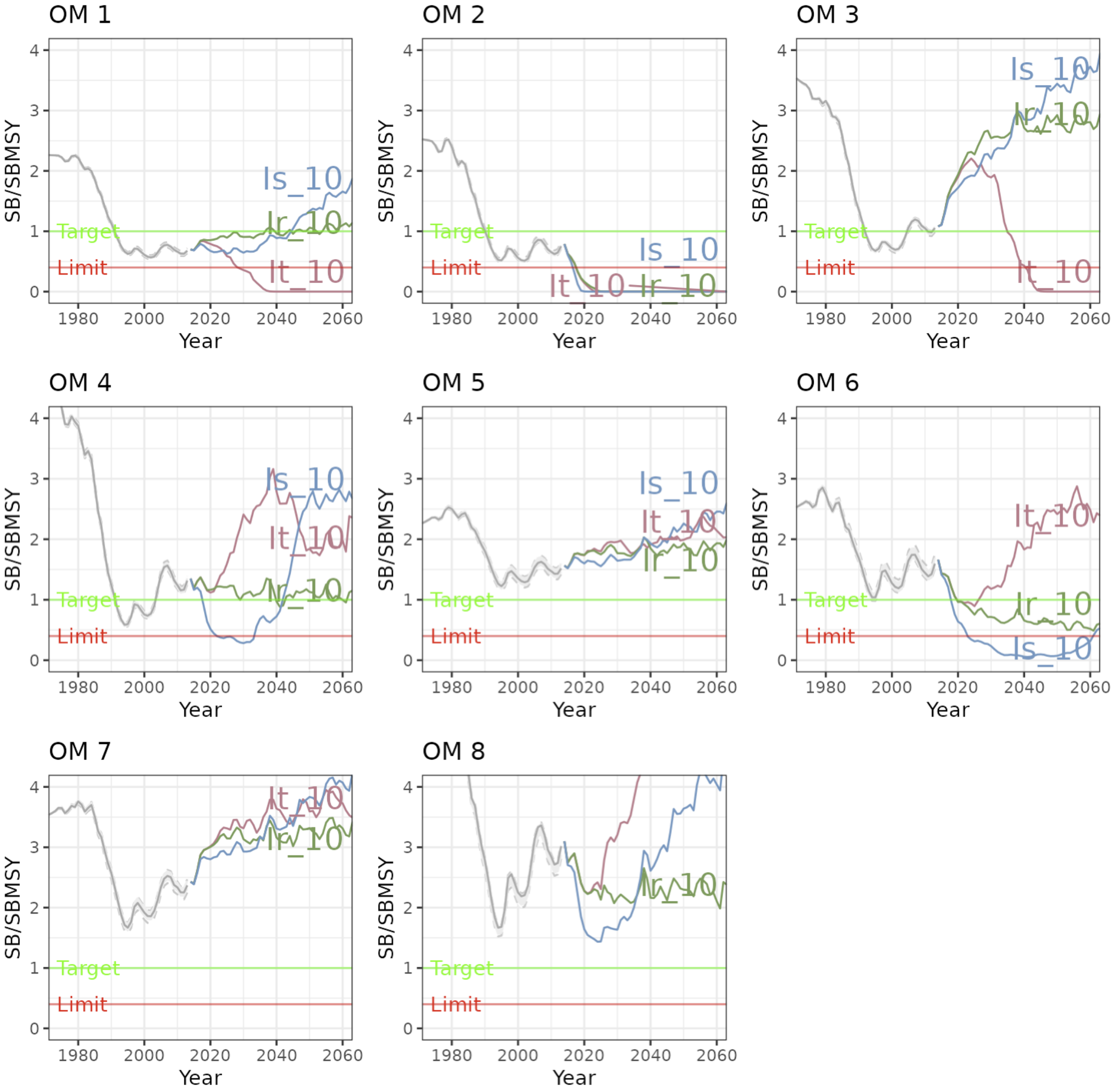

Figure 2. Impact of operating model on the median biomass performance of the index target (It_10), index ratio (Ir_10) and index slope (Is_10) MPs with a maximum TAC change of 10%.

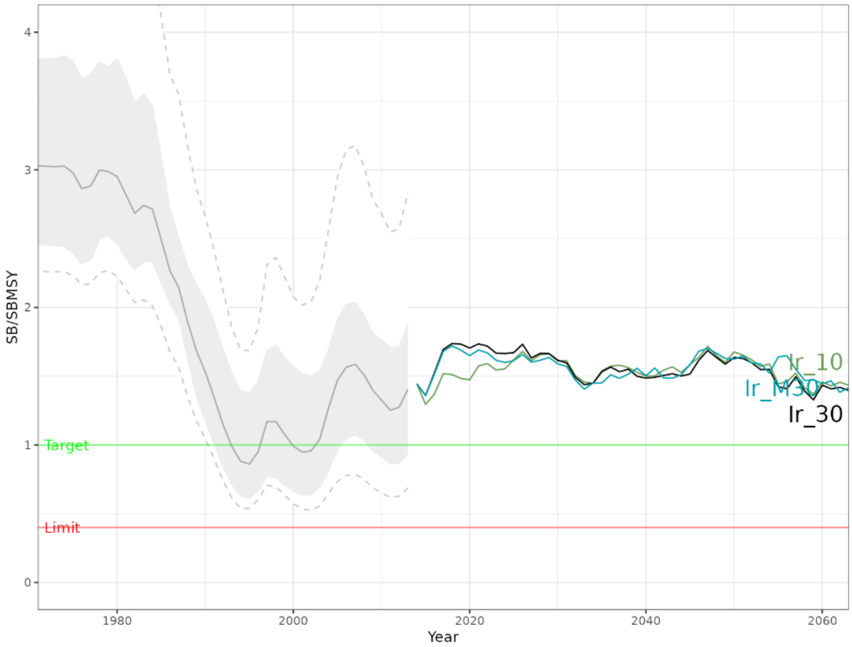

Figure 3. Projection of median SSB/SSBMSY for the three derivatives of the index ratio (Ir) management procedure that include a maximum 10% change in TAC (Ir_10), a maximum 30% change in TAC (Ir_30) and a maximum 30% change in TAC and a 30kt max TAC constraint (Ir_M30). Historical grey lines and shaded areas are the median, 50% and 80% intervals.

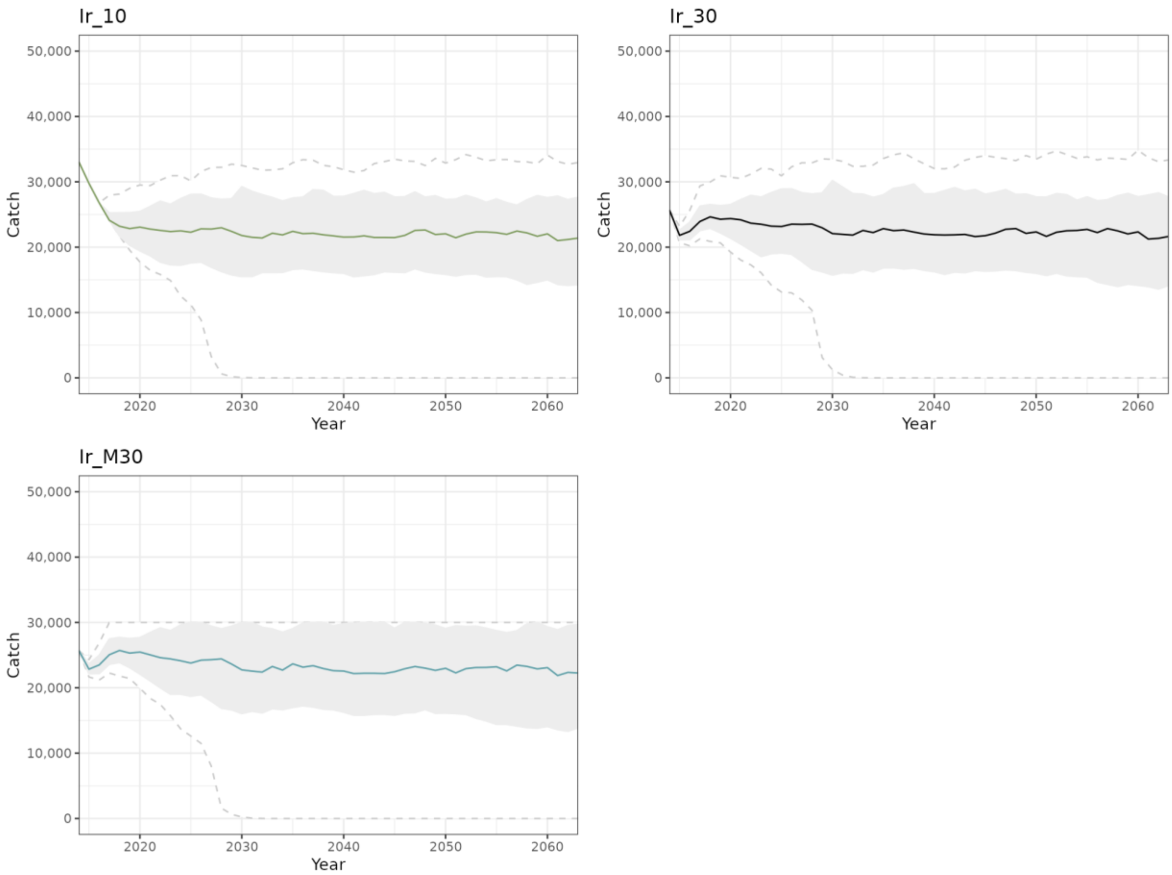

Figure 4. As Figure 3 but split into a panel per MP showing the median, 50% and 80% intervals for projected catches.

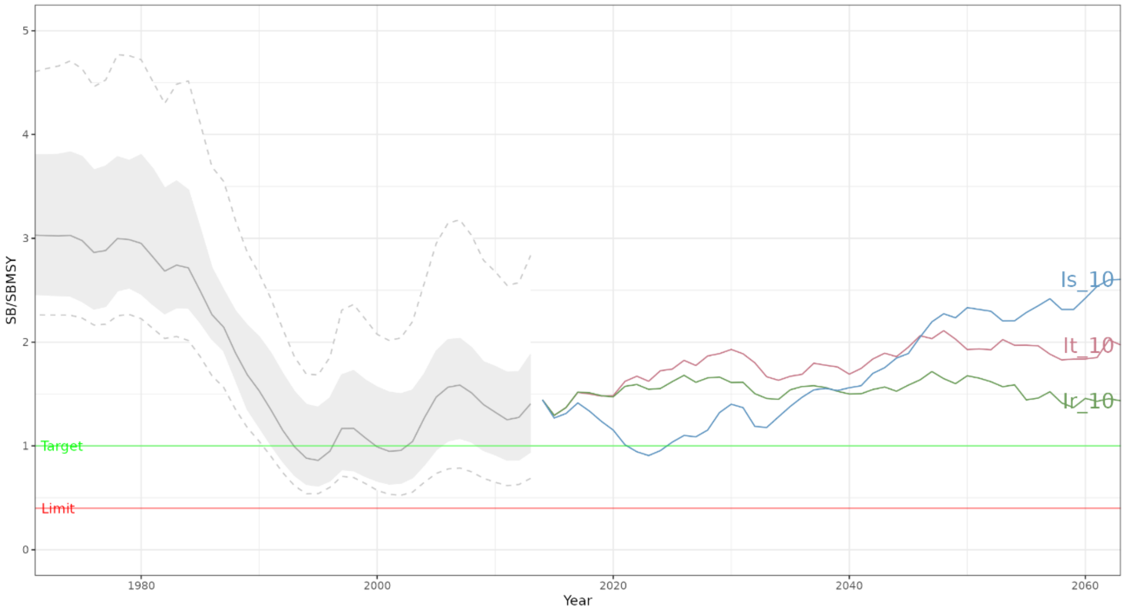

Figure 5. Projection of median biomass (across all OMs) among the various MP archetypes given the 10% TAC constraint.

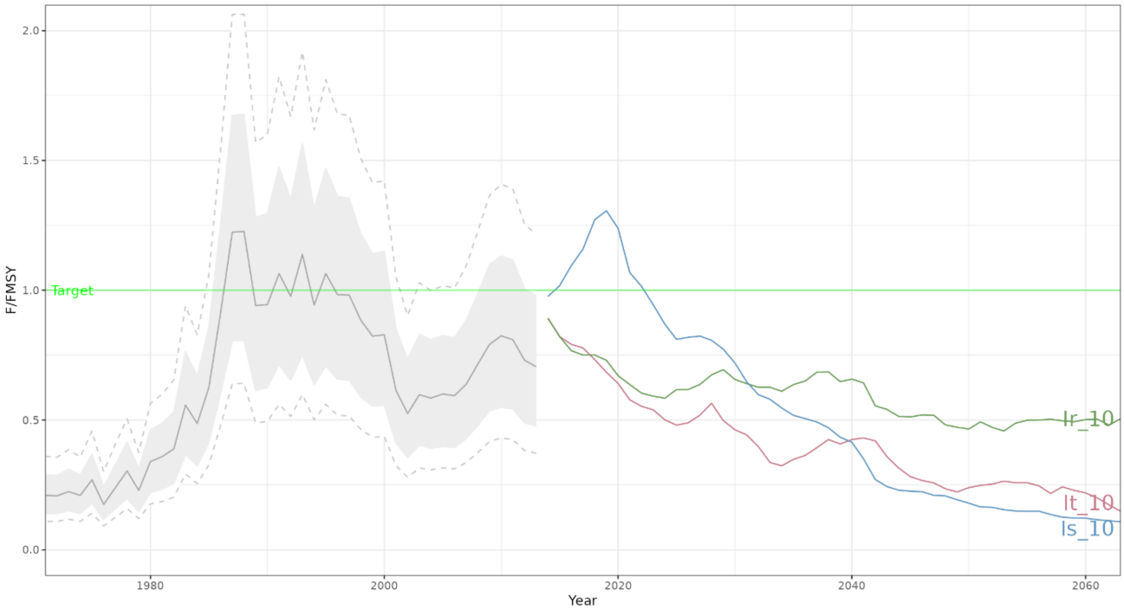

Figure 6. As Figure 5 but for F/FMSY.

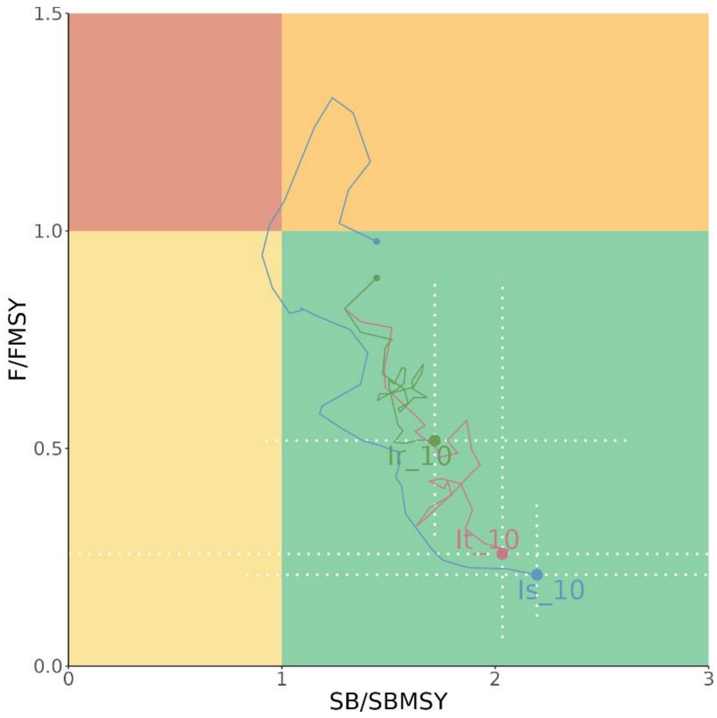

Figure 7. Kobe plot for the various MP archetypes given the 10% constraint in TAC change among years. Small points are the start of the projected time period, large labelled points are the end of the projected time series. White dotted lines represent the 50% interval.

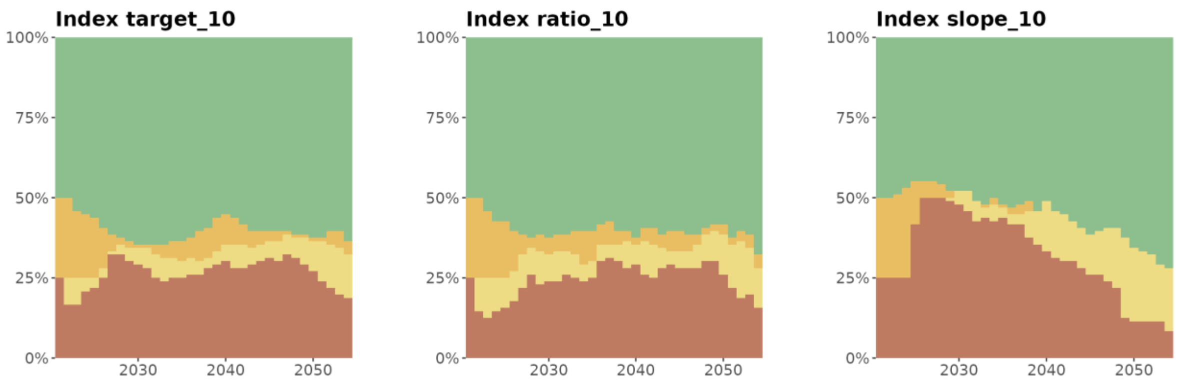

Figure 8. Kobe time plot showing the fraction of simulations in each of the Kobe quadrants (Figure 7) over projected years for the various MP archetypes given the 10% constraint in TAC change among years.

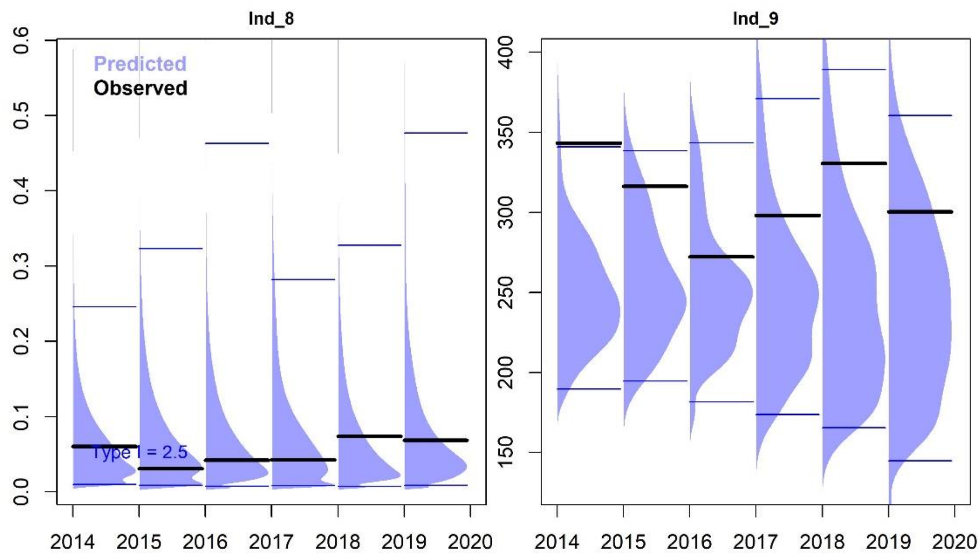

Figure 9. Posterior predicted distributions (blue shaded areas) and 97.5% interval (annual Type I error = 2.5%) (blue horizontal lines) compared to observed values (black horizontal lines) for indices 8 and 9.

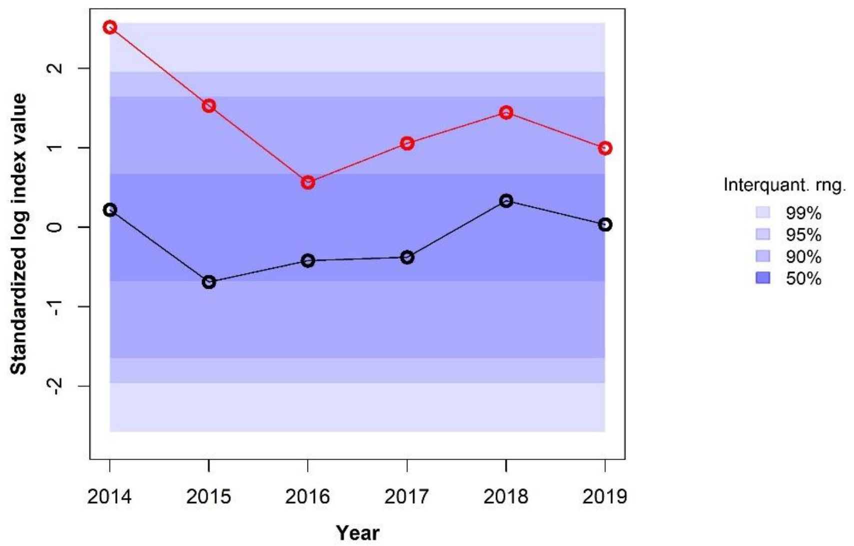

Figure 10. Standardized density plots (the data of Figure 9) assuming standard normal distribution (Index 9 is red, Index 9 is black).

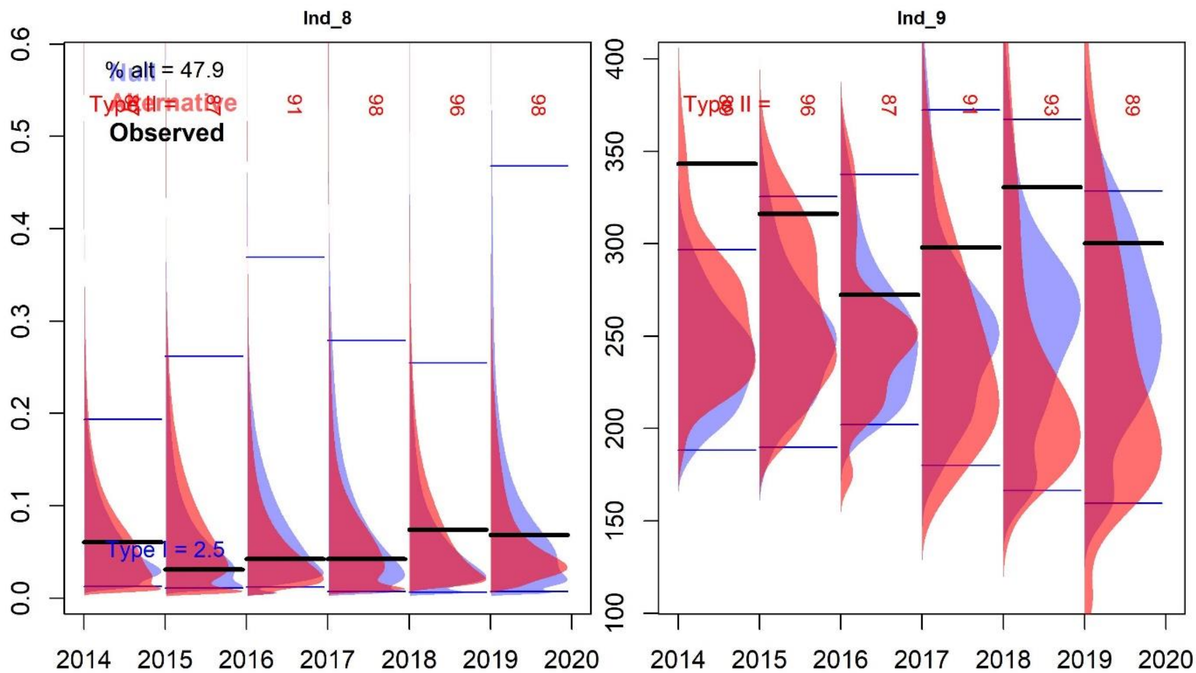

Figure 11. As Figure 9 but plotting the simulations where spawning biomass never falls below 75% SSBMSY (blue, null simulations) and those where projected spawning biomass does fall below 75% of SSBMSY at some point in the projection (red, alternative simulations). The figures at the top show the annual type II error (probability of not triggering ECP even though the simulations lead to the alternative scenarios of less than 75% SSBMSY).

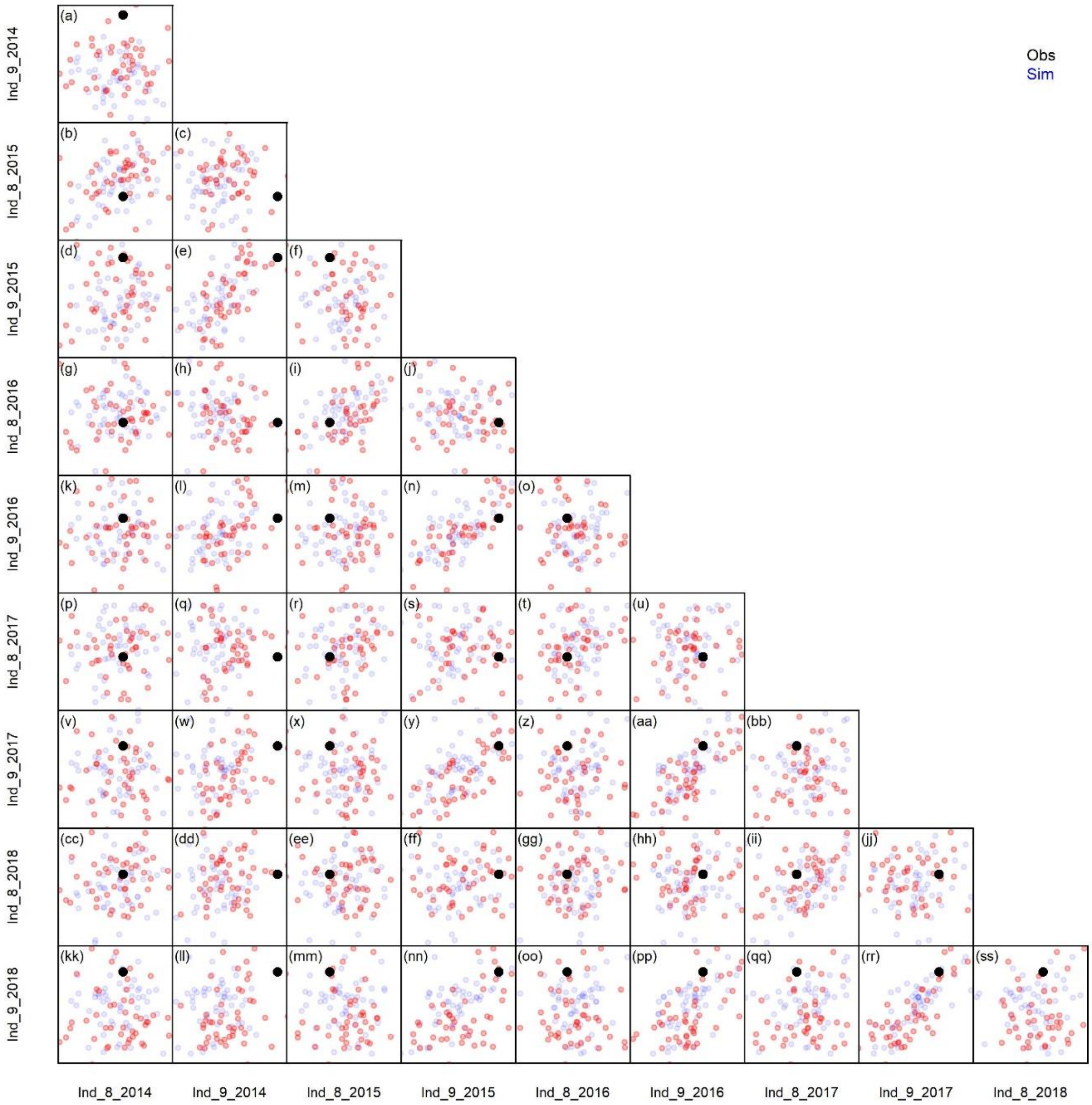

Figure 12. Posterior cross-correlation plots showing simulated data (coloured points, n=96) versus the observed data (black points) over multiple years. As Figure 11, the blue shaded points are posterior predicted data where SSB never falls below 75% SSBMSY, red points are where SSB does fall below 75% SSBMSY at some point in the projection.

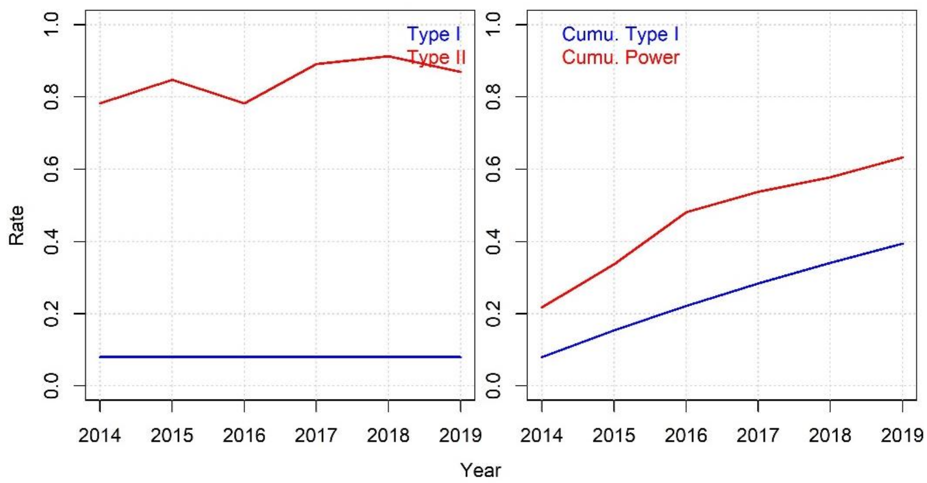

Figure 13. A cumulative power analysis corresponding to the distributions for indices 8 and 9 in Figure 11. This analysis accounts for correlations among projected data.

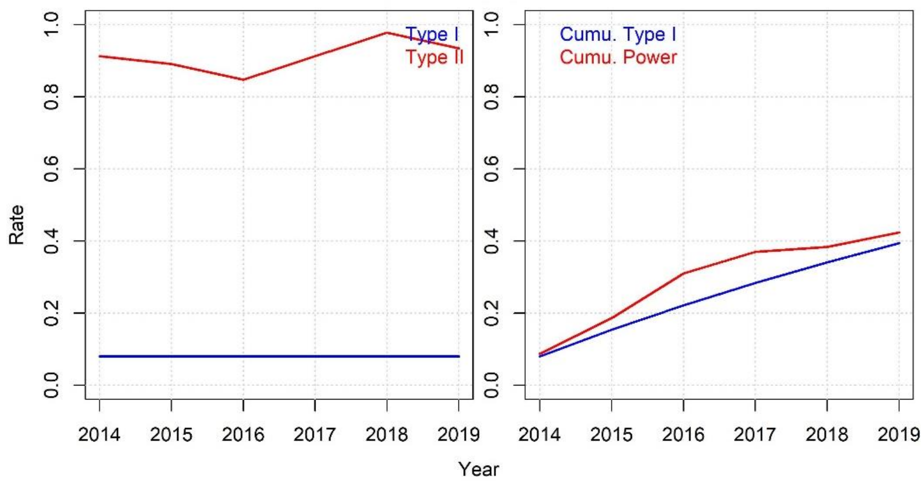

Figure 14. As Figure 13 but with ECP triggered only for the lower tail of the indices.

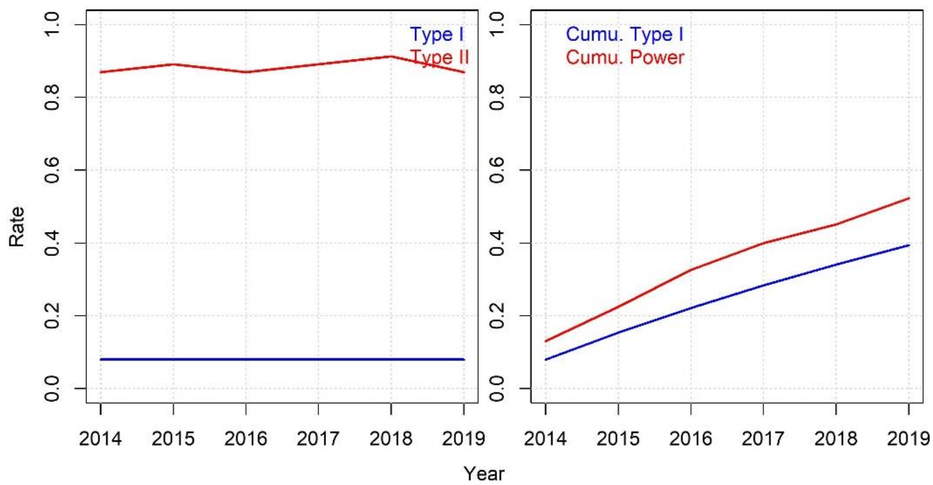

Figure 15. As Figure 13 but with ECP triggered only using index 9 (alpha is doubled to 5% since number of indices is halved).

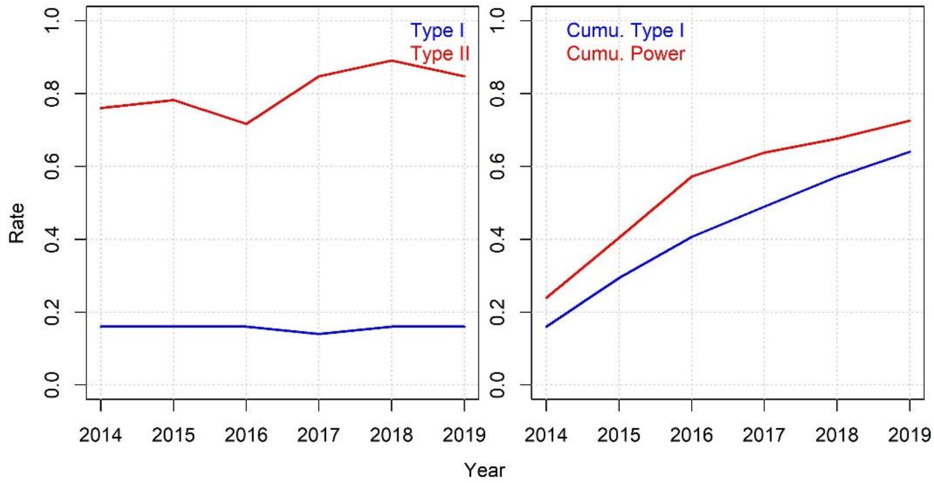

Figure 16. As Figure 13 but with type I error doubled to 5% per index per year.

Tables

Table 1. The reference grid of operating models that is a full cross (2 x 2 x 2) of two factor levels for each of three factors: natural mortality rate (M – a multiplier of the base assessment M-at-age vector), steepness of the Beverton-Hold stock recruitment curve (h – the base assessment value was 0.73) and current stock depletion (SSB_{2013} / $SSB_{unfished}$) (Depln – a multiplier of the base assessment estimate of stock depletion).

	M	h	Depln
1	3/4	0.6	2/3
2	4/3	0.6	2/3
3	3/4	0.9	2/3
4	4/3	0.9	2/3
5	3/4	0.6	3/2
6	4/3	0.6	3/2
7	3/4	0.9	3/2
8	4/3	0.9	3/2

Table 2. Median values over OMs as tabulated by Slick. Note that an updated version of Slick will include mean values that, for example, would match the PGK tunings of the MPs.

MP	AvTAC_long	AvTAC_med	AvTAC_short	nLRP	PGK	PGK_med	PGK_short	PNOF	VarC
It_10	16,400	13,600	21,000	1	0.697	0.750	0.833	0.796	0.0846
It_30	7,500	12,000	15,800	1	0.825	0.846	0.946	0.843	0.1440
It_M30	17,100	13,400	17,300	1	0.950	1.000	0.971	0.957	0.0790
Ir_10	22,200	23,000	23,600	1	0.761	0.729	0.833	0.849	0.0466
Ir_30	22,500	23,500	23,600	1	0.775	0.721	0.900	0.828	0.0467
Ir_M30	22,800	24,100	24,500	1	0.701	0.729	0.833	0.875	0.0414
Is_10	13,400	18,700	29,600	1	0.551	0.558	0.383	0.711	0.0511
Is_30	13,900	18,800	30,100	1	0.546	0.542	0.367	0.704	0.0544
Is_M30	19,800	22,000	28,600	1	0.711	0.750	0.512	0.939	0.0388

Appendix A. Code to complete all simulation work including OM specification, MP specification, derivatives and tuning, and MSE projections.

```
# ====================================================================================
# === A Demonstration MSE for Atlantic Blue Shark ===================================================
# ====================================================================================

# Tom Carruthers
# August 2024

# Notes
# Follows the technical components of the MSE roadmap (SCRS/2024/103) 'the roadmap'

# === Prerequisites ==========================================================================

library(openMSE)
library(r4ss)
setwd("C:/GitHub/Blue_shark_MSE")
source('Source/MP_tuning.R')


# === Technical Milestone 1 ====================================================================

# --- Condition Reference Set -----------------------------------------------

OM_RefCase = SS2OM('Assessment/Preliminary_Run_6_input',nsim=12)               # sample var-covar to make OpenMSE class OM
Data = SS2Data('Assessment/Preliminary_Run_6_input')                         # convert SS3 input data to OpenMSE class Data
Data@CAL = array(NA,c(1,1,1))                                    # don't simulate CAL data
Data@MPrec = Data@Cat[1,ncol(Data@Cat)]                                    # assume that the recent catch observation is the current TAC
OM_RefCase@cpars$Data = Data                                               # add real data to OM

OM_grid = expand.grid(Mfac = c(3/4,4/3), h = c(0.6,0.9), dep_fac = c(2/3,3/2))   # reference operating model grid
nOM = nrow(OM_grid)                                                # 8 total

OM_mod = function(OM, Mfac = 1, h = 0.73, dep_fac = 1, DCV = 0.05){ # OM modifier
  OM@cpars$M_ageArray = OM@cpars$M_ageArray * Mfac
  OM@h = h
  OM@cpars$qs = NULL  # catchability estimated to match depletion
```

```
  OM@cpars$D = trlnorm(OM@nsim,OM@D[1] * dep_fac, DCV)
  OM
}

for(i in 1:nOM){
  OM = OM_mod(OM_RefCase, OM_grid$Mfac[i], OM_grid$h[i], OM_grid$dep_fac[i])     # make reference case OM
  saveRDS(OM,paste0("OMs/OM_",i,".rds"))                                  # save OM
  saveRDS(runMSE(OM,Hist=T),paste0("OMs/Hist_",i,".rds"))                 # save historical OM dynamics (inc ref pts etc)
}


# --- Develop Reference MP --------------------------------------------------

Ref_MP = FMSYref75                                    # for now, just use 75% FMSY (perfect info) as reference


# === Technical Milestone 2 ======================================================================

# --- MP archetypes ---------------------------------------------------------

Data = readRDS('OMs/Hist_1.rds')@Data; x = 1                 # Data for designing MPs

# calculates a TAC from a TAC modifier, maximum TAC changes and maxTAC
doRec = function(MPrec, mod, maxchng, maxTAC){
  if(mod > (1+maxchng))mod = 1+maxchng
  if(mod < (1-maxchng))mod = 1-maxchng
  Rec = new('Rec')
  Rec@TAC = min(MPrec*mod, maxTAC)
  Rec
}

# Index target MP
I_targ = function(x, Data, reps = 1, targ = 2, nyrs = 3, maxchng = 0.3, maxTAC = 5E5, Ind = 9){
  I = Data@AddInd[x,Ind,]/mean(Data@AddInd[x,Ind,39:43],na.rm=T)
  recI = mean(I[length(I)-((nyrs-1):0)])
  mod = recI/targ
  doRec(Data@MPrec[x], mod, maxchng, maxTAC)
}

# Index ratio MP
I_rat = function(x, Data, reps = 1, targ = 0.5, nyrs = 3, maxchng = 0.3, maxTAC = 5E5, Ind =9){
  CpI = mean(Data@Cat[x,39:43]) / mean(Data@AddInd[x,Ind,39:43],na.rm=T)
  I = Data@AddInd[x,Ind,]
  recI = mean(I[length(I)-((nyrs-1):0)])
  PropTAC = recI * CpI * targ
  mod = PropTAC / Data@MPrec[x]
  #if(ncol(Data@Cat)==50)saveRDS(Data,"C:/temp/Data.rds")
  doRec(Data@MPrec[x], mod, maxchng, maxTAC)
}

# Index slope MP
I_slp = function(x, Data, reps=1, targ = 0.025, nyrs = 5, fac = 1, maxchng = 0.3, maxTAC = 5E5, Ind = 9){
  I = Data@AddInd[x,Ind,]/mean(Data@AddInd[x,Ind,39:43],na.rm=T)
  slp = lm(y~x,data=data.frame(x=1:nyrs,y=I[length(I)-((nyrs-1):0)]))$coefficients[[2]]
  mod = exp((slp-targ)*fac)
  doRec(Data@MPrec[x], mod, maxchng, maxTAC)
}

class(I_targ) = class(I_rat) = class(I_slp) = "MP"


# === Technical Milestone 3 ======================================================================

# --- MP derivatives --------------------------------------------------------

It_10 = It_30 = It_M30 = I_targ
Ir_10 = Ir_30 = Ir_M30 = I_rat
```

```
Is_10 = Is_30 = Is_M30 = I_slp

formals(It_10)$maxchng = formals(Ir_10)$maxchng = formals(Is_10)$maxchng = 0.1 # set max TAC change
formals(It_M30)$maxTAC = formals(Ir_M30)$maxTAC = formals(Is_M30)$maxTAC = 3E4 # set max TAC

class(It_10) = class(It_30) = class(It_M30) =
  class(Ir_10) = class(Ir_30) = class(Ir_M30) =
    class(Is_10) = class(Is_30) = class(Is_M30) = "MP"

allMPs = paste(rep(c("It","Ir","Is"),each=3),c("10","30","M30"),sep="_")


# --- Demo MSE --------------------------------------------------------------

Hist_1 = readRDS('OMs/Hist_1.rds')
initMSE = Project(Hist_1,c("It_30","Ir_30","Is_30","FMSYref"))
Pplot(initMSE)
matplot(t(initMSE@Catch[,4,]),type="l")
saveRDS(initMSE,"MSEs/initMSE.rds")


# --- MP Derivatives -------------------------------------------------------

derivMSE = Project(Hist_1, allMPs)
Pplot(derivMSE)
saveRDS(derivMSE,"MSEs/derivMSE.rds")


# --- MP tuning --------------------------------------------------------------

for(i in 1:nOM) assign(paste0("Hist_",i),readRDS(paste0("OMs/Hist_",i,".rds")))
Hist_list = list(Hist_1, Hist_2, Hist_3, Hist_4, Hist_5, Hist_6, Hist_7, Hist_8)

# A function that calculates the squared difference between obtained and target mean PGK
minfunc = function(MSE_list){
  PGKm = sapply(MSE_list,function(X){mean(X@SB_SBMSY>1 & X@F_FMSY < 1)})
  PGKw =  mean(PGKm) # ! this should really be mean() but this way it matches default slick table
  cat(paste0("PGKw = ",round(PGKw,6),"\n"))
  (PGKw - 0.6)^2
}

setup(cpus=8)     # do 8 MSE calcs in parallel (one per OM)
sfExport('doRec') # export any functions used by MPs

# Index target MP tuning

It_30_t = tune_MP(Hist_list,"It_30","targ",c(0.8,1.6),minfunc, tol=1E-3, parallel=T)
It_10_t = tune_MP(Hist_list,"It_10","targ",c(0.8,1.6),minfunc, tol=1E-3, parallel=T)
It_M30_t = tune_MP(Hist_list,"It_M30","targ",c(0.8,1.6),minfunc, tol=1E-3, parallel=T)

saveRDS(It_30_t,"MPs/It_30_t.rda")
saveRDS(It_10_t,"MPs/It_10_t.rda")
saveRDS(It_M30_t,"MPs/It_M30_t.rda")

# Index ratio MP tuning

Ir_30_t = tune_MP(Hist_list,"Ir_30","targ",c(0.5,0.65),minfunc, tol=1E-3, parallel=T)
Ir_10_t = tune_MP(Hist_list,"Ir_10","targ",c(0.5,0.65),minfunc, tol=1E-3, parallel=T)
Ir_M30_t = tune_MP(Hist_list,"Ir_M30","targ",c(0.6,0.85),minfunc, tol=1E-3, parallel=T)

saveRDS(Ir_30_t,"MPs/Ir_30_t.rda")
saveRDS(Ir_10_t,"MPs/Ir_10_t.rda")
saveRDS(Ir_M30_t,"MPs/Ir_M30_t.rda")

# Index slope MP tuning

Is_30_t = tune_MP(Hist_list,"Is_30","targ",c(0,0.05),minfunc, tol=1E-3, parallel=T)
```

```
Is_10_t = tune_MP(Hist_list,"Is_10","targ",c(0,0.05),minfunc, tol=1E-3, parallel=T)
Is_M30_t = tune_MP(Hist_list,"Is_M30","targ",c(0,0.05),minfunc, tol=1E-3, parallel=T)

saveRDS(Is_30_t,"MPs/Is_30_t.rda")
saveRDS(Is_10_t,"MPs/Is_10_t.rda")
saveRDS(Is_M30_t,"MPs/Is_M30_t.rda")


# --- Run all tuned MPs on all OMs -----------------------------

allMPs_t = paste0(allMPs,"_t") # MP names

# Load MPs
for(MP in seq_along(allMPs_t))assign(allMPs_t[MP],readRDS(paste0("MPs/",allMPs_t[MP],".rda")))
for(i in 1:nOM) saveRDS(Project(get(paste0("Hist_",i)), allMPs_t),paste0("MSEs/MSE_",i,".rds"))


# --- Slick script ---------------------------------------------

# --- ECP script -----------------------------------------------


# ==== END OF CODE ====================================================================
```