

# IOTC Yellowfin and Bigeye Tuna Management Strategy Evaluation: Phase 1 Technical Support Project Final Report

Dale Kolody ([Dale.Kolody@csiro.au](mailto:Dale.Kolody@csiro.au)), Paavo Jumppanen

June 2016



European Union



# Contents

<b>1</b>	<b>Executive Summary</b> .....	<b>1</b>
<b>2</b>	<b>Acronyms and definitions</b> .....	<b>3</b>
<b>3</b>	<b>Introduction</b> .....	<b>3</b>
3.1	<i>Background</i> .....	4
3.2	<i>Contract Objectives</i> .....	8
3.2.1	<i>Software Distribution</i> .....	9
<b>4</b>	<b>MSE Software</b> .....	<b>9</b>
4.1	<i>Software design Principles</i> .....	10
4.2	<i>R-based MSE Control and Projection software</i> .....	10
4.3	<i>C++ Projection Sub-routine</i> .....	14
<b>5</b>	<b>Operating Model Parameterization</b> .....	<b>14</b>
5.1	<i>Demonstration Yellowfin Tuna Operating Models</i> .....	17
5.1.1	<i>Yellowfin Conditioning Assumptions</i> .....	17
5.1.2	<i>Additional Yellowfin Projection Assumptions</i> .....	29
5.2	<i>Demonstration Bigeye Tuna Operating Model</i> .....	35
5.2.1	<i>Conditioning Assumptions</i> .....	35
5.2.2	<i>Additional Bigeye OM Projection Assumption</i> .....	43
<b>6</b>	<b>Candidate Management Procedures</b> .....	<b>47</b>
<b>7</b>	<b>Performance Measures and Graphics</b> .....	<b>52</b>
<b>8</b>	<b>Demonstration MSE results</b> .....	<b>54</b>
8.1	<i>R and C++ projection performance comparison</i> .....	54
8.2	<i>Demonstration yellowfin MSE results</i> .....	60
8.3	<i>Demonstration bigeye MSE results</i> .....	66
<b>9</b>	<b>Critique of the Indian Ocean Yellowfin and Bigeye MSE software, demonstration case Oms and candidate MPs</b> .....	<b>71</b>
9.1	<i>Projection software</i> .....	71
9.2	<i>OM Parameterization process</i> .....	73
9.3	<i>Yellowfin demonstration Operating Model OMyft2</i> .....	74
9.4	<i>Bigeye demonstration Operating Model OMBet1</i> .....	76
9.5	<i>Candidate MPs</i> .....	76

<b>10</b>	<b>Discussion .....</b>	<b>77</b>
<b>11</b>	<b>Acknowledgements .....</b>	<b>78</b>
<b>12</b>	<b>References .....</b>	<b>79</b>
<b>13</b>	<b>Appendix 1. User Manual for Indian Ocean Yellowfin and Bigeye MSE Software</b>	

# 1 Executive Summary

This report summarizes technical development undertaken in support of the Indian Ocean Tuna Commission (IOTC) Management Strategy Evaluation (MSE) process for yellowfin and bigeye tunas from Jan2015 – Jun2016, as defined under the IOTC/FAO-CSIRO Letter of Agreement GCPRAF466EC. The primary objectives included the development, documentation and distribution of MSE software for the IOTC community, including demonstration case Operating Models (OMs) to represent the dynamics and uncertainty of Indian Ocean yellowfin and bigeye tuna populations, and the evaluation results for some candidate Management Procedures (MPs). This project was conducted with the oversight of the IOTC Working Party on Methods (WPM), with additional feedback from the IOTC Working Party on Tropical Tunas (WPTT). The following points summarize key deliverables from the project:

1. MSE control and projection software – the project adapted the R-based software developed by Carruthers et al. (2014) for Atlantic bluefin tuna to meet the needs of Indian Ocean yellowfin and bigeye tuna. The modified software includes features for: i) defining operating models on the basis of weighted combinations of Stock Synthesis (Methot and Wetzel 2013) assessment models, ii) conducting stochastic projections which simulate fishery dynamics, including the fish population, fishery, simulated data collection, and management, and iii) summarizing and comparing candidate MPs in graphical and tabular form on the basis of the IOTC management performance measures. The projections are designed to evaluate catch-based or effort-based management (or a combination of the two, where each fishery is assigned only one type), including fishery-specific management implementation error. A C++ based projection sub-routine was also developed, as an independent check on the integrity of the R-based code, and a more computationally efficient option for evaluating MPs.
2. Conditioned yellowfin tuna Operating Models (OMs) – demonstration OMs are provided, including i) very simple, deterministic dynamics that are useful for model testing and MP development and ii) a more "realistic" OM parameterized with an ensemble of Stock Synthesis model estimates derived from the most recent (Langley et al. 2015) assessment. Additional options for forward projections are defined. Key features include:
  - Conditioning period 1950-2014, with models fit to total catch, CPUE, size composition and tags
  - Age-structured with quarterly dynamics
  - 21 fisheries (19 with stationary historical selectivity)
    - Projected selectivity has temporal variability
  - 4 areas, with non-seasonal, age-dependent movement
  - Stationary biology (size-at-age, maturity, fecundity, migration, stock-recruit relationship)
  - Beverton-Holt stock-recruitment with auto-correlated log-normal deviates
  - An equally-weighted grid of models representing 54 combinations of assumptions:
    - 3 levels of Beverton-Holt stock recruit steepness

- 3 levels of natural mortality ( $M$ )
    - 3 levels of tag likelihood weighting
    - 2 levels of assumed CPUE catchability trend
  - Estimated parameters include virgin recruitment, fishery selectivity, CPUE catchability, recruitment deviates and movement rates
  - Constant catch projections are employed from 2014-2017, with MP-based management beginning in 2018
  - "Recent" catch distributions are defined as in the assessment (most recent 9 quarters)
3. Conditioned Bigeye tuna Operating Models – a demonstration BET OM is provided, derived from an ensemble of SS models derived from the most recent (Langley et al. 2013) assessment. Key OM features:
- Conditioning period 1950-2012, with model fits to total catch, CPUE, and size composition data
  - Age-structured with quarterly dynamics
  - 12 fisheries with stationary selectivity
    - Projected selectivity has temporal variability
  - 1 area
  - Stationary biology
  - an equally-weighted grid of models representing 18 combinations of assumptions:
    - 3 levels of Beverton-Holt stock recruit steepness
    - 3 levels of natural mortality ( $M$ )
    - 2 levels of assumed CPUE catchability trend
  - Estimated parameters include virgin recruitment, fishery selectivity, CPUE catchability and recruitment deviates
  - Constant catch projections are employed from 2013-2017, with MP-based management beginning in 2018
  - "Recent" catch distributions are defined as in the assessment (mean of 2008-11)
4. Candidate Management Procedures (MPs)– in addition to simple constant catch and effort MPs used for testing and quality control, two categories of feedback-based MPs are implemented and evaluated. The first is a simple “empirical” MP, which seeks to regulate the Total Allowable Catch (TAC) to stabilize (longline) catch rates near a desirable target level. A second “model-based” MP involves fitting a Pella-Tomlinson surplus production model, then allocating a TAC based on the estimated stock status, Maximum Sustainable Yield (MSY), and a “40:10”-type Harvest Control Rule (HCR).
5. Demonstration MSE results compare a suite of candidate MPs on the basis of the agreed performance measures developed by the IOTC Working Party on Methods, spanning a broad range of the performance trade-off space that the Commission is likely to consider of interest.

The yellowfin management performance trade-offs do not appear to be straight-forward because the demonstration OM suggests an overfished state with recent overfishing. The bigeye MSE results are simpler and more intuitive. The demonstration MPs can probably be improved with more careful consideration of the management objectives (e.g. currently there are conflicting objectives put forth by the Commission -  $B_{MSY}$  is a target reference point, while a high probability of remaining above  $B_{MSY}$  is considered desirable). In the case of yellowfin, an MP should probably be developed which considers the nuances of rebuilding objectives, i.e. greater catch reductions in the short-term can be expected to lead to greater catches in the long-term. In contrast, the bigeye models suggest that more aggressive harvesting could be sustained, such that it may not be sensible to aim for similar objectives for both species. None of the MPs examined made use of size composition data.

6. Software User Manual –The user manual provides a description of the software including the dynamics equations, demonstration OM specifications, candidate MPs, and R scripts and documentation for reproducing all of the results shown in this report.
7. The open source software is publicly available from Github: <https://github.com/pjumpnanen/MSE-IO-BET-YFT>, and is intended to encourage multiple IOTC users to engage with the MP development process, both to help ensure the integrity of the code, and to support independent exploration of OMs and MPs.

We emphasize that this project does not constitute a completed MSE, but delivers the first phase of the technical support required for the broader MSE process to move forward with the active engagement of the Commission, Scientific Committee and relevant technical working groups. This document explains a number of interim decisions that were made to complete the phase 1 project, and includes a substantive critique section, primarily aimed at scientists, to assist in the review of the software functionality and demonstration case operating models. It is expected that additional OM specification changes and software modifications will be required before the MSE process is completed. Noting that the Commission has proposed an ambitious timeline with external peer review of the MSE in 2017/2018, some of the expected future technical requirements are described.

## 2 Acronyms and definitions

A list of acronyms and definitions used in this report is provided in the user manual (Appendix 1).

## 3 Introduction

This report provides an overview of the state of development of the technical tools required to support IOTC yellowfin and bigeye Management Strategy Evaluation (MSE). The main text focuses on a high level description of the MSE software features with demonstration results, and justification for decisions made during software development, including contrasts with other MSE

initiatives in the IOTC and elsewhere. It is assumed that the reader has a general familiarity with stock assessment concepts, while we attempt to introduce the concepts of Management Strategy Evaluation and Management Procedures in the Indian Ocean context. A technical user manual is included (Appendix 1) which provides precise mathematical equations for the projection software dynamics and instructions for reproducing and modifying the results presented in this report.

Following the introduction, the main text is arranged in sections loosely corresponding to the software workflow outlined in Figure 1, including i) MSE control software, ii) stochastic projection software, iii) specification of demonstration case Operating Models (OMs), iv) candidate Management Procedures (MPs), v) management performance indicators, and vi) demonstration results for candidate MPs. A critique section highlights issues that may require further consideration from the IOTC technical working parties or Commission. The discussion attempts to place this body of work in the context of the broader IOTC MSE process.

### 3.1 Background

In pursuing the objectives of achieving conservation and optimum utilization of tuna stocks, the Indian Ocean Tuna Commission (IOTC) committed to pursuing Management Strategy Evaluation (MSE) for the key target species of swordfish and albacore, skipjack, bigeye and yellowfin tunas (IOTC 2011).

MSE is a process in which a fishery system, including the fish population, fishery, and management decisions, are simulated over a medium to long term time horizon, and performance of the management system is evaluated with respect to explicit management objectives (e.g. see Punt et al 2014 and references therein). A computer simulator (Operating Model, OM) is intended to describe the main uncertainties in the system, including the current state of the fish population and stochastic future events. The Management Procedure (MP) is the algorithm that recommends a unique management action given the perceived state of the fishery, and is applied at pre-determined intervals. The MP should use feedback control, to change the management action in response to new information about the changing state of the fishery. Simulation-tested MPs offer many advantages over the traditional cycle of stock assessment and ad hoc decision making, including: i) MPs should be robust to the main uncertainties in the system (i.e. provide reasonable management performance regardless of the true underlying dynamics), ii) MPs are evaluated against multiple explicit management objectives, and iii) pre-agreement on data collection, analytical methods and harvest control rules pre-empt disagreements about management actions arising from a failure to reach a consensus assessment.

In this project, we have aimed to evaluate MPs using the *sensu stricto* definition, in which the MP explicitly includes the specification of the data collection and analytical methods to be used, in conjunction with a Harvest Control Rule (HCR), e.g. as used in the southern bluefin tuna fishery (Hillary et al. 2016). In some other applications, the MP does not include the internal specification of data collection and analysis. For example, the Indian Ocean skipjack MP prescribes an HCR that assumes a sensible stock assessment will always be available. In this latter case, the simulation testing involved simulating stock status outputs with a known degree of accuracy and precision. Given that the former approach requires assumptions about simulated assessment data observation errors, the distinction between the two approaches may appear subtle. But

assessment model inferences are often biased in ways that are difficult to anticipate, particularly when there are substantial structural errors in the model. Simulating the assessment process is probably the only way to reliably represent these potential biases. However, there is a more important operational distinction between the two approaches that is critical when assessment bodies are unable to reach a consensus view of the stock status (and indeed adoption of the MP approach was motivated by this problem in some international fisheries organizations). The *sensu stricto* MP approach explicitly pre-empts the problem of conflicting assessments, because the MP data and analyses are agreed in advance (and simulation tested to ensure that performance is robust to alternative plausible assessment interpretations). The stock assessments for the IOTC fisheries have undergone substantial changes in recent years, and it would not be surprising if they continue to evolve in the foreseeable future, such that consensus is not inevitable. The *sensu stricto* approach may have the further advantage that resources required for the traditional stock assessment process should not be required every time that the MP is evaluated. If the internal MP "assessment" is a straightforward mechanical calculation, this potentially frees up assessment resources for other strategic research needs.

The MSE process can be partitioned into a series of steps (represented schematically in Figure 1):

1. Identification of management objectives and quantifiable performance measures
2. Development of a range of Operating Models (OMs) to represent the uncertainty in the fishery
3. Development of candidate Management Procedures (MPs)
4. Simulation testing of candidate MPs using the OMs
5. Selection of a preferred MP on the basis of the simulated performance with respect to the management objectives (performance measures)
6. Implementation of the MP

The process is rarely a linear sequence, as individual steps tend to be iteratively revisited as information is exchanged among participants, and decision makers come to understand the possible performance trade-offs. Following adoption of an MP, it should not be expected that the MP will continue to manage the fishery in perpetuity, on "auto-pilot". Part of the broader MP specification should include periodic performance reviews, to ensure that the MP is meeting the management objectives, and that the management objectives remain appropriate. It would be optimistic to expect that MPs will always perform well, and there should be regular scientific oversight of the fishery to check whether the system has moved into a space that was not encompassed by the original simulation process. It is not possible to anticipate all of the "exceptional circumstances" which could arise, but likely problems include: i) new observations may fall substantially outside of the envelope of uncertainty that the MP was designed to handle (e.g. unrecognized biological uncertainties become evident), ii) the fisheries data may cease to be as informative as expected (e.g. longline CPUE may no longer be available on the spatial and temporal scales used historically), iii) management actions may not be as effective as expected (e.g. due to large increases in efficiency for effort-managed fleets, or increasing IUU catches). If exceptional circumstances arise, they should be examined to see if they affect the management recommendation, and the MP should be temporarily suspended until the issues can be resolved, or a new MP evaluated and implemented.



To date, the IOTC MSE process is most advanced for skipjack (Bentley and Adam 2015) and albacore (Mosqueira and Scott 2015), and has not begun for swordfish. This report describes the current state of development of IOTC yellowfin and bigeye MSE. Essentially, the preliminary technical work has been completed, which allows OMs to be defined, MPs to be tested and performance measures to be summarized and presented. The overall MSE process is far from complete, but is now supported at a level that allows the technical working parties to review and further develop the OM specifications, and will allow the higher level decision makers to clarify their preferences in terms of management actions and performance trade-offs. Note that this project is completely independent of some prior work on Indian Ocean bigeye tuna MSE (Tong et al. 2011, Zhang et al. 2013). Those earlier studies were more academic in nature, instigated before the IOTC MSE process was formalized, and independent of feedback from the Commission or technical working parties.

The IOTC meeting reports identify several MSE timeline expectations, though the parallel path for scientist, manager and stakeholder engagement in the process is not clearly defined. IOTC Resolution 15/10 states "[MSE] Assessments for yellowfin tuna, bigeye tuna and swordfish to be completed by 2017 and presented to the Commission meeting in 2018." IOTC Scientific Committee (IOTC SC 2015) notes in the program of work with respect to yellowfin and bigeye MPs "Final model with MPs" - Dec 2016, while the budget identifies the need for external peer review of bigeye and yellowfin MSE in 2017 and 2018 (with an equal budget for both species in both years). We conclude this report with a brief discussion of some of the technical support requirements that will likely be required to achieve the overall MSE objectives.

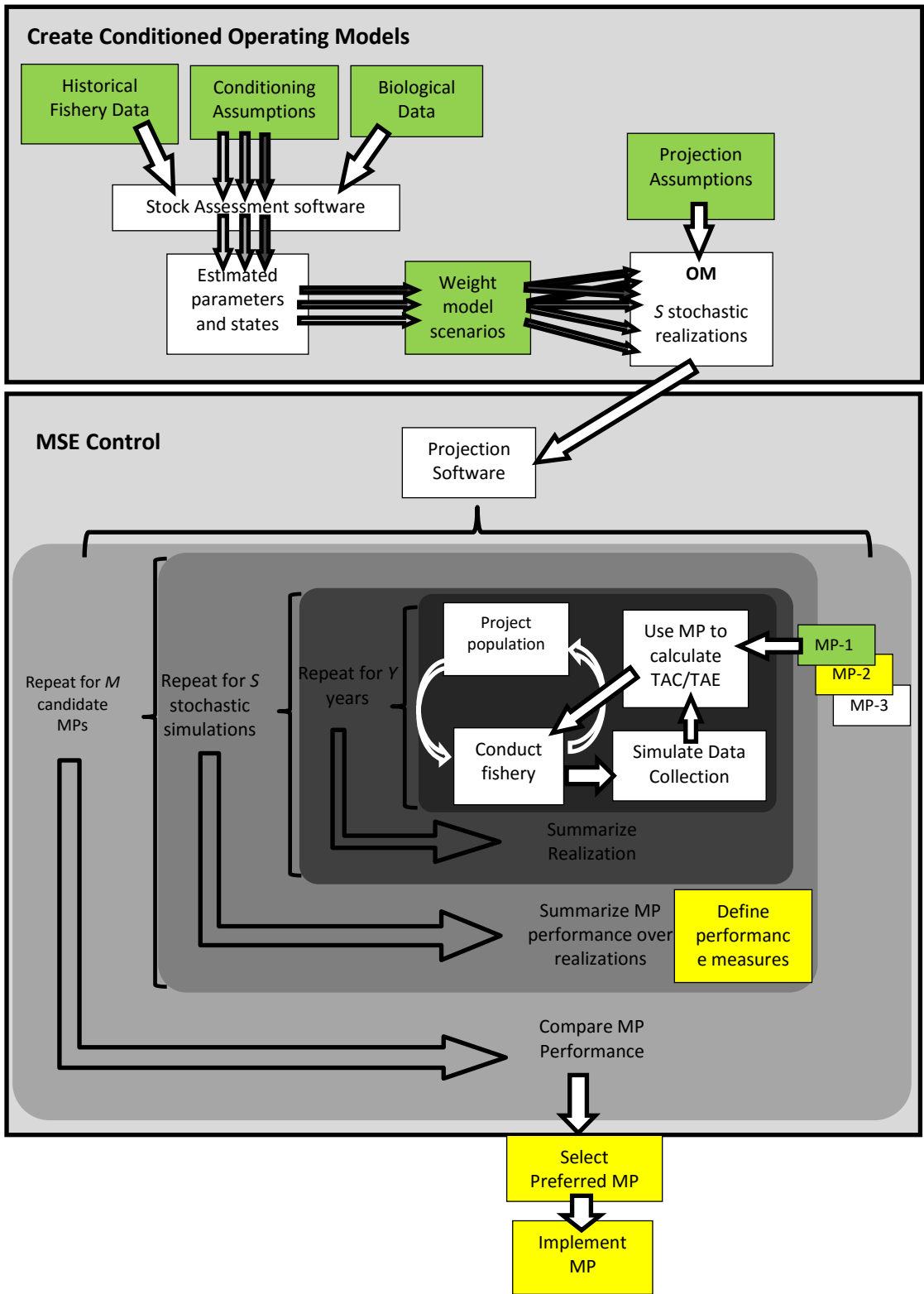


Figure 1. Schematic flowchart of the MSE process, emphasizing the technical elements as implemented in this project. Key points for integrating broader scientific input are highlighted in green, while other stakeholder and manager (Commission) inputs are highlighted in yellow.

## 3.2 Contract Objectives

The IOTC/FAO-CSIRO Letter of Agreement GCPRAF466EC itemized the following deliverables, recognizing that many details would need to be resolved during the project execution, subject to oversight by a technical steering committee, which in practice consisted of the feedback from the IOTC Working Party on Methods (including the informal sub-group on MSE), with input from the IOTC Secretariat:

- 1) Develop an operating model (a suite of models) that can simulate the key features of the population dynamics, mixed fleet characteristics, data collection process, and harvest control rule implementation for Indian Ocean bigeye and yellowfin tuna.
- 2) Use the operating model to evaluate a suite of candidate Harvest Control Rules with respect to the performance objectives defined by the Commission.
- 3) Preparing a user manual and documentation for the code in the OM, including a description for third parties of how to use the code to evaluate HCRs.

Project milestones included attending a number of meetings and providing progress reports for technical review (available from the IOTC website or from the authors). The interim reports are not reproduced here, because the content of this report supersedes all previous results. Project feedback was received at the following meetings:

- IOTC informal Working Party on Methods (WPM) sub-group on MSE 2014 (Ispra, Italy)
  - Scoping exercise prior to project commencement
- Informal WPM sub-group on MSE 2015 (Ispra, Italy)
  - Proposed approach to software design presented and reviewed
- WPM 2015 (Montpellier, France)
  - Review of MSE software progress
- Working Party on Tropical Tunas (WPTT) 2015 (Montpellier, France)
  - Presentation of preliminary yellowfin OM
  - Discussion of bigeye OM options
- Informal WPM sub-group on MSE 2016 (Tokyo, Japan)
  - Presentation and discussion of preliminary yellowfin and bigeye OMs
  - Presentation and discussion of candidate MPs
  - Presentation and discussion of preliminary demonstration case MSE results
  - Final workplan endorsed to end of contract

Release of this report and the accompanying software concludes CSIRO's obligations under the current contract. The authors may not be able to provide any additional support for the software, including feature extensions, bug fixes, user tutorials, or practical application and engagement with the relevant IOTC bodies, unless additional funding can be sourced. However, the authors will endeavour to document and archive relevant feedback into the software repository to ensure that the overall MSE process can move forward.

### 3.2.1 Software Distribution

All of the documentation, source code, compiled executables, and scripts required to reproduce the results described in this report are publicly available for unrestricted use from Github (<https://github.com/pjumppanen/MSE-IO-BET-YFT>). The documentation consists of this final report and a user manual (Appendix 1), and additional documentation on the use of the ADT software (see below). The user manual is intended to be a "living document", which should be periodically updated as the MSE progresses.

There was an unexpected challenge in the development/distribution of the C++ projection sub-routine arising from a change to a third-party software license. The code was designed to use the TAPENADE utility, developed and maintained by INRIA (National Institute for Research in Computer Science and Control, France, <http://www-sop.inria.fr/tropics/tapenade.html>), to provide automatic differentiation. In this project, this allows for efficient solving of the Baranov catch equations. Mid-project, we reported that INRIA had changed the licensing arrangements, such that a substantial license fee was introduced for the sort of TAPENADE usage that had originally been planned. This decision has since been reversed. An annual license is still required, but it is once again free for academic use (as of Sep 2016). We provide the C++ source code, including the CSIRO-developed ADT supporting software (Automatic Differentiation with TAPENADE), and the fully functional DLL. Any change to the C++ code involving the objective functions will require updated differentials (most conveniently obtained using TAPENADE). In part because of the license fee issue, the initial release includes a parallel (and mathematically somewhat different) implementation of the projection code in R. The R/C++ performance comparison (section 8.1) concludes that neither implementation has a substantive advantage at this time. A real advantage of C++ could be realized in the future, if more of the code were to be implemented in C++ to further improve performance.

## 4 MSE Software

The MSE software consists of several inter-related components, not all of which are necessary for every application. In this report, we provide a qualitative description of the rationale and functionality of the different software components, with emphasis on important issues that would benefit from review by a broad range of IOTC scientists. The user manual (Appendix 1) provides a precise mathematical description of the OM projection dynamics and MPs (and technical details required to install and operate the software to reproduce the results shown in the main text).

The term Operating Model (OM) can be used in somewhat different ways, and for the purposes of this document, we describe the OM in two distinct parts as shown in Figure 1: i) the structural relationships defined in the projection software (e.g. how many areas are there, what is the stock recruitment relationship, etc.), and ii) the specific parameter values (or distribution of values) adopted to represent the parameter estimation uncertainty (e.g. Beverton-Holt stock recruit steepness = 0.7 - 0.9, Francis 1992). This section refers to the structure of the projection code, while the following sections refer to the process used to parameterize demonstration cases for yellowfin and bigeye OMs using stock assessment model parameter estimates (conditioning).

## 4.1 Software design Principles

This project was initiated with the over-arching design objective of keeping the code accessible to IOTC member scientists through the use of freely available software, familiar to the fisheries science community, and ideally re-using and extending established MSE code to the extent possible. Most of the software is R-based (R Core Team 2015), designed to run on MS Windows operating systems on personal computers, and depends on packages freely available from the CRAN repository.

Most of the parameters of the demonstration case operating models are estimated using the freely available Stock Synthesis (SS) software (Methot and Wetzel 2013), and SS output files are processed in part using *r4ss*, the R-based SS support package. The conditioning is intended to be compatible with, and evolve in accordance with, the insights from the IOTC stock assessment process. However, the OM structure and initialization process would require modification if substantially different SS specifications were adopted.

Graphics for presenting MP performance in the format agreed by the IOTC WPM members are provided (some of which are modified from the Fisheries Library in R, e.g. <http://www.flr-project.org/>).

This project was pursued in the context of the *sensu stricto* definition of MP (i.e. the data and analytical needs of the MP are specified in an attempt to pre-empt the ambiguity that could arise due to contested assessments). However, we recognize that the Commission may opt for a less restrictive definition for the MSE process, as has been adopted for skipjack.

For anyone with a modest familiarity with R, the end product should be relatively straightforward to install, reproduce the results presented here, and modify or extend to analyse different combinations of conditioning assumptions, projection assumptions or MP algorithms. However, the original objective of re-using existing MSE tools was not as successful as originally planned, because of the case-specific needs of the Indian Ocean tropical tuna applications. Substantive code modifications, for qualitatively different analyses, will probably not be easy for non-programmers to undertake.

## 4.2 R-based MSE Control and Projection software

For the purposes of this report, we refer to the R-based control software as the large lower box of Figure 1. The MSE control framework:

- Accepts an argument list including a single OM and one or more MPs to be evaluated.
- Co-ordinates the simulation testing of each MP for the dynamics described by the OM. Each MP is tested with many stochastic realizations using the projection sub-routine (below), subject to the range of conditions defined by the uncertainty in the OM.
- Collates and summarizes the management performance statistics.

We refer to the projection sub-routine as the innermost (darkest shaded) region of Figure 1, which represents the most computationally intensive core of the MSE process. This sub-routine simulates the fishery dynamics with an iterative series of events:

- Biological states are updated at each time-step, including: spawning-recruitment, migration, mortality and age/growth increment. These processes may differ in each simulation, depending on the structural and parameter uncertainty represented in the OM, and stochastic error (i.e. variability that cannot be explained or predicted).
- The collection of fisheries data is simulated with observation error, subject to a realistic time lag: total catches, longline CPUE (relative abundance indices), size composition data.
- The MP is applied, i.e. the MP algorithm inputs the historical and simulated fishery data and outputs a management action (e.g. TAC). This happens at user-defined intervals (triennial frequency adopted for the results described here). The MP algorithm could involve fitting a stock assessment model to the data, to make statistical inferences from the simulated data. However, the MP does not have access to any other information from the OM which would reduce the uncertainty about the underlying dynamics.
- The fishery catches are removed from the population, according to the management action prescribed by the MP, and subject to stochastic implementation error (i.e. it is difficult to manage catch or fishing mortality precisely).

This project produced two parallel versions of the projection sub-routine, an R-based version described here, and a C++ version described in the subsequent section. The implementation equations for the projection sub-routines are provided in the user manual (Appendix 1). The basic feature set includes the following (species-specific details of parameterization and the initial state are provided in subsequent sections):

- Population and fishery dynamics resemble standard fisheries stock assessment models:
  - Order of events: migration, recruitment, natural and fishing mortality, age-class graduation.
  - Age-structured (with seasonal dynamics for growth, recruitment, fisheries). Most summary reporting is annual.
  - Multiple areas (if required).
  - Single sex.
  - Multiple fisheries - each assigned to a single area.
  - Age-based fishery selectivity, with the option to introduce temporal variability parameterized as a function of long period sine wave oscillations.
  - Beverton-Holt stock-recruitment relationship, with auto-correlated log-normal stochastic variability.
  - Instantaneous, bulk-transfer migration (potentially season-specific, though the conditioning to date assumed all seasons were identical).
- Data simulation:
  - Total catch in mass, by fishery, with log-normal error.
  - Aggregate annual CPUE with auto-correlated log-normal errors.
    - Derived from region-specific, quarterly, fishery-selected CPUE.
  - Multinomial size composition samples by fishery (variance is increased by decreasing the sample size). Sampling bias is not implemented, but non-stationary selectivity can be used to ensure size composition is not overly informative.

- Management:
  - Following a realistic data collection delay, MPs have access to observations of catch by fishery, size composition samples and annual aggregate CPUE indices (an overall delay of 3 years between data collection and implementation of each MP was assumed for the results in this report).
  - Management actions consist of fishery-specific effort multipliers and/or an aggregate quota for fisheries that are not effort-managed.
  - Fishery-specific quota allocations and baseline effective effort are defined from an arbitrary recent period (the definitions from the recent stock assessments were used for the results in this report). These can be season-specific or averaged over seasons.
  - Log-normal implementation errors for the effort or TAC, applied independently for each fishery.
- Computational considerations:
  - The random number is seeded such that the OM reproduces the same stochastic variability for each MP evaluated.

Carruthers et al. (2014) developed MSE software for Atlantic Bluefin Tuna (ABT), with the support of the ICCAT Atlantic-Wide Research Programme on Bluefin Tuna (GBYP), which appeared to meet most of the needs of the yellowfin/bigeye MSE project. In preliminary scoping, it was evident that the ABT MSE software was well organized, coded entirely within R, very easy to install, and it reproduced the expected results without any assistance other than the existing documentation. It was also designed for a fishery of greater complexity than most single species fisheries MSE applications, and largely encompassed the anticipated needs of the yellowfin and bigeye fisheries.

It was hoped that this software could be easily adapted for the current project, and that the shared software development would facilitate testing and extensions that were useful for both IOTC and ICCAT in the long run. Although modifications were more extensive than anticipated, much of the general structure of the ABT software was retained, including some attractive features that have not been used to date, but which may prove useful in the future (e.g. multiple stocks, hyperstability- hyperdepletion in CPUE parameters, ability to randomly sample some parameters rather than adopting the fixed values from the SS conditioning models, amongst others). For the benefit of users familiar with the ABT code, we note some of the substantive changes:

- The original ABT MSE implementation achieved reasonable computational efficiency in R through the use of very large multidimensional arrays. However, the base R *array* objects do not generalize very well because dimensions of length 1 (e.g. area, population) are automatically dropped in many calculations, which causes subsequent errors. A replacement R package, *keep* (uploaded to CRAN), was developed as part of this project, to increase the control over dimension management. This elegantly solved this specific problem with minimal recoding. However, the memory consumption also proved impractical for the Indian Ocean yellowfin applications, such that the dimensionality of the array structures had to be reduced (largely achieved by dropping the year dimension, which is managed through conventional loops).

- The ABT OM represented uncertainty by randomly sampling parameters (e.g. growth, natural mortality, etc.) from independent, user-defined prior distributions, while the IOTC application imports most initial states and parameters from an ensemble of SS models explicitly conditioned to historical data. This is discussed in the following section on OM conditioning.
- The ABT-OM used quarterly dynamics for the fisheries and migration, but other biological features were described on an annual basis (which is probably adequate for this relatively slow-growing species). The tropical tuna applications required quarterly dynamics for recruitment and growth (i.e. including calculation of quarter-specific catch-at-length distributions), and for consistency with SS there was a further disaggregation into beginning of time-step sizes (for spawning-related calculations) and mid-time-step sizes (for catch-related calculations).
- The ABT code used a large number of age classes, assuming that negligible numbers surviving beyond the final age could be ignored. The IO-MSE code uses a conventional “plus group” accumulator, assuming that all fish in the final age class are biologically identical.
- The ABT application partially implemented independent movement parameters for mature and immature individuals. The IOTC application uses age-specific migration (independent of maturity status).
- The ABT application includes a dynamic fishing effort distribution model, in which effort is a function of vulnerable biomass and a targeting efficiency parameter. The IOTC version adopted the stock assessment convention of assigning each fishery to exactly one region. All MSY and MP projection calculations are determined by a pre-defined “recent” distribution of catch (and/or effective effort). MPs can regulate the fishery-specific effort distribution.
- ABT MPs are all based on quota management. The yellowfin/bigeye approach can evaluate candidate MPs that use a mix of catch and effort controls (each fishery is managed by only one of these). Effort can be managed independently by fishery, while quotas are distributed among fisheries that are not effort-managed, according to the recent catch distributions.
- Neither the ABT model nor the R-based yellowfin/bigeye application uses the standard Baranov catch equations for quota-based fishery dynamics. The ABT code sequentially applies natural mortality, followed by catch extraction. The R version of the yellowfin/bigeye MSE uses the usual Baranov equations for purely effort-based management, and a hybrid approach of splitting the time-step for quota extraction. This involves applying continuous natural mortality and effort-based fishing mortality for the first part of the time-step, instantaneously extracting quota-based catch at a user-defined point within the time-step, and then applying natural mortality and effort-based fishing mortality for the remainder of the time-step; i.e. if only quotas are used, and the catch is extracted halfway through the time-step, this is equivalent to Pope’s approximation. If only effort-based management is used, this is the usual Baranov solution. In part because of concerns about this approach raised at the IOTC WPM, the optional C++ projection sub-routine was implemented with the Baranov equations.



- The sequence of events from the ABT projection model was changed to conform to the SS conditioning model (spawning, movement, mortality, age-class graduation).
- The ABT code distributes recruits spatially in relation to the spawning biomass distribution. The BET/YFT MSE code adopts the SS approach of pooling all spawning biomass for the purpose of stock-recruitment, and apportioning recruits spatially according to a distribution that is independent of spawning biomass (and which in turn may have stochastic spatial deviations).

Completely self-contained R-code was achieved through the use of computationally-efficient array manipulation functions (to minimize the use of loops). Unfortunately, this programming style is difficult to understand for inexperienced R-users, and difficult to debug and modify for programmers accustomed to working in low level languages.

### 4.3 C++ Projection Sub-routine

The purely R-based projection code is reasonably efficient in so far as R allows, however, it was also desirable to develop a C++ projection sub-routine for several reasons: i) C++ is faster, easier to understand and debug, and has more efficient memory usage, ii) the C++ code provided an independent check on the R-based code (bugs were identified and fixed in both implementations as a result), and iii) the C++ implementation uses the standard Baranov catch equations, which are consistent with the SS assessments used for OM conditioning. The C++ and R implementations are compared in terms of computational efficiency and numerical consistency in section 8.1.

## 5 Operating Model Parameterization

Describing uncertainty in the context of MSE OMs (and stock assessments) is a challenging and imperfect process (see Punt et al 2014). Understatement of the uncertainty leads to overconfidence (and an under-appreciation of the real risk) because there is a high probability that management outcomes will fall outside of the range of dynamics that were tested. In contrast, overstatement of uncertainty may lead to lost economic opportunities from overly precautionary management, i.e. quotas may be unnecessarily conservative if the probability of adverse outcomes is exaggerated. The approach we have adopted admits the importance of “model uncertainty” i.e. potentially including different structural assumptions or alternative data series, but in practice also includes parameter uncertainty for a range of key parameters that are known to be difficult to estimate. This approach is generally expected to provide a broader representation of uncertainty than the statistical uncertainty conditional on any individual model (though see section 5.1.2).

There are different ways to parameterize operating models. We adopted an explicit conditioning approach, similar to that used for Indian Ocean albacore (Mosqueira and Scott 2015). As outlined in the top box of Figure 1, this involves estimating parameters using statistical stock assessment models and ensures that the simulation dynamics are reasonably consistent with the historical data, and current perceptions of stock status derived through the stock assessment process. A suite of assessment models are fit to the fishery/biological data and assumptions, and the

Maximum Posterior Density (MPD) estimates (i.e. the best point estimates) are retained. The MPD estimates for the parameters and initial population state provide a joint distribution that describes the covariance in the system. The alternative model formulations typically include differing assumptions about how data should be weighted, different structural assumptions and/or different values for parameters that might be estimable in principle, but are notoriously difficult to estimate in practice (e.g.  $M$ , stock recruit steepness). Typically a "grid" of models is run in which key assumptions are evaluated with a range of options in a balanced design, such that all possible combinations of options are evaluated to describe the potential interactions among assumptions (in contrast, stock assessments typically emphasize sensitivities from a reference case model formulation in terms of single assumption changes relative to a reference case, which may ultimately provide the main assessment advice).

Carruthers et al. (2014) used a different approach for ABT, in which operating model parameters were randomly sampled from independent distributions. Bentley and Adam (2015) used a similar approach for IOTC skipjack, with an additional suite of qualitative criteria to judge the consistency of the stochastic OM realizations against the historical data and perceived stock status (e.g. rejecting ~95% of simulations as not sufficiently consistent with expectations). In general, we would expect that the explicit conditioning approach should result in historical inferences that are more consistent with the stock assessment process, and might be more suitable for MSE when the data are of a reasonable quality, and there is a well-developed stock assessment process. In contrast, the latter approach might be more appropriate when there are lower quality data, a weak stock assessment, and/or one is seeking to make broad generalizations about MP performance across a range of hypothetical situations. However, the distinction may be somewhat artificial in the sense that the two approaches fall on a continuum, and may converge depending on how the priors and retention criteria are formulated.

Whichever approach is used, it is essential to examine the model results to ensure that they are biologically plausible. The conditioned approach ought to ensure that OMs are consistent with the historical data and assessment assumptions (provided that there is a sensible assessment, and appropriate measures are taken to consider the uncertainties associated with conflicting data). But unexpected dynamics sometimes occur, e.g. population declines may be explained by recruitment trends that are independent of fisheries (as Mosqueira and Scott 2015 observed for IOTC albacore). It may prove desirable to reject some models, or differentially weight the models within the grid, so that the OM provides a better representation of the perceived uncertainty. Likelihood-based weighting is an obvious tool to employ in this context, and it might be argued that Bayesian posteriors represent the natural extension to this approach. However, we have found that Bayesian methods are often impractical for complicated stock assessment models (e.g. Langley 2015 noted this for the Indian Ocean yellowfin assessment). Additionally, some inferences often appear to be determined with much more certainty than seems believable, given the type and amount of data available. This latter problem presumably results because the likelihoods do not really conform to statistical ideals, and biases arise due to the constraining assumptions required to produce tractable estimators (e.g. these problems may be evident when parameter estimates appear to be very precise, but are sensitive to alternative assumptions, see Schnute and Richards 2001). Furthermore, likelihood-based weighting may not be possible if models are structurally different (e.g. fit to different data).

Some R tools are provided for automating the process of setting up, fitting and examining a grid of SS models as detailed in the user manual. The following states and parameters are adopted from the SS outputs (whether they were estimated or adopted as a fixed input):

- Initial numbers-at-age in the final time-step of the assessment model
- Stock-recruitment relationship functional form, steepness and virgin recruitment parameters
- Fishery selectivity
- Natural mortality
- Catchability for the abundance index fleets (i.e. longline CPUE), including speculative temporal trends
- Age-specific length, mass and maturity schedules
- The seasonal distribution of effective catch and effort (from a suitable recent period)

In addition to the conditioning, flexibility for adding additional uncertainty to the OM projections has been added in terms of:

- Initial numbers at age at the commencement of projections - younger ages should be more uncertain because they are observed fewer times in the fishery (additionally, IOTC assessments have recent recruits deterministically derived from the mean stock-recruit relationship). In the IOTC yellowfin and bigeye OMs, this additional error is implemented with two parameters: a lognormal error CV for the youngest age, and an exponential decay parameter that describes a decrease in the CV with increasing age (see user manual).
- CPUE observation error is parameterized in terms of a CV and lag(1) auto-correlation. These can be estimated from the fit between predictions and observations in the conditioned assessment model, or they can be externally imposed. i.e. The CPUE fit may appear better than could reasonably be expected from opportunistic commercial fishery data, and there is rarely any independent means to evaluate the relationship between abundance and CPUE). There is also a CPUE hyperstability parameter retained from the ABT implementation (not yet used in the IOTC context).
- Future recruitment variability CV and lag(1) auto-correlation are defined by the user. The auto-correlation reflects the fact that there can be extended periods of good or poor recruitment (and partially compensates for the systematic lack of fit that would arise in the short-medium term if the stock recruitment relationship was mis-specified).
- The option of adding non-stationary fishery selectivity was added. This was parameterized using random long period sine wave oscillations, causing the selectivity to i) become more concentrated/diffuse, and/or ii) shifting towards younger or older age classes. This is intended to introduce temporally-structured noise to the size composition distributions to prevent these data from being unrealistically informative for the MP. It is difficult to defend the specific parameterization, except to note that the alternative of truly stationary selectivity is probably naively optimistic.

We note some additional concerns about the SS conditioning:

- Recent yellowfin and bigeye assessments have relied on an atypical SS configuration in which quarters are represented using SS years. This is inconvenient in the sense that timescale labels are confusing, and rate processes (M, F, growth etc.) must be redefined in quarterly, rather than annual, units. The option exists to define a seasonal SS configuration that uses natural time-steps, however there are drawbacks, notably i) tags are assigned integer ages (i.e. the seasonal model does not admit different quarterly ages of tags within a year), and ii) there is a single annual spawning biomass calculation (which should be adequate provided that the population does not have very fast dynamics). However, the year/season configuration would allow some seasonal parameters to be estimated (e.g. migration, recruitment, catchability).
- In spatial models, SS reports spatially-aggregated fishing mortality, and not the region-specific fishing mortality that is required to estimate the local effective effort for effort-based management. In principle, it should be possible to back-calculate fishing mortality by region from the SS numbers-at-age outputs, but initial attempts to do so suggested that the relevant outputs are not intuitively defined.

## 5.1 Demonstration Yellowfin Tuna Operating Models

The demonstration case is designed to maintain general consistency with the insights from the most recent assessment (Langley 2015), and the associated discussion at the IOTC WPTT 2015. However, the objective of assessment and operating models are usually perceived to be substantially different. Stock assessment traditionally has a greater emphasis on identifying the “best estimates”, e.g. the Langley (2015) assessment emphasizes the results from a single model specification, even though several models were developed and compared. It is an objective of MSE to test that MPs are robust to uncertainties that are plausible, but not necessarily described in the assessment. Accordingly, the OM should encompass a greater degree of uncertainty.

### 5.1.1 Yellowfin Conditioning Assumptions

Yellowfin OM demonstration case OMyft2 is based on the MPD estimates from an ensemble of 54 conditioned models, derived from the assessment with different combinations of input assumptions. We do not repeat all of the historical arguments for the assessment assumptions (refer to Langley 2015), but attempt to explain why we have deviated from the assessment in certain respects. The OMyft2 ensemble is derived from fitting SS configurations with the following shared characteristics (differences are described subsequently):

- Structure:
  - 4 regions (Figure 2)
  - Conditioned on data from 1950-2014, with quarterly dynamics
  - Single Stock
  - Age-structured
  - Sex-aggregated
  - 21 fisheries (19 with stationary selectivity; 2 with step-function temporal changes in selectivity), Table 1

- Beverton-Holt stock-recruit relationship
  - Spawning biomass aggregated across regions
  - Log-normal quarterly recruitment deviates from the spatially aggregated stock-recruit relationship, except for the early years and most recent 8 quarters
  - Recruits distributed among regions with (renormalized) log-normal deviates
- Baranov catch equations (implemented with the SS "hybrid" approach)
- Fixed parameters (though some vary among SS specifications as indicated below) include:
  - Size-at-age (Figure 3)
  - Maturity-at-age (with fecundity proportional to mass) (Figure 3)
  - Natural mortality (Figure 3)
  - Stock-recruit steepness
  - Recruitment temporal and spatial variance ( $\sigma_{Rt}$ ,  $\sigma_{Rr}$ )
  - CPUE observation error variance ( $\sigma_{CPUE}$ )
  - Size composition sampling variance (i.e. assumed sample sizes for multinomial likelihood, which tend to be far lower than true sample sizes, and derived in the assessment with some consideration to iterative re-weighting principles)
  - Weighting factor ( $\lambda$ ) for tag data influence in the likelihood
- Estimated states and parameters include:
  - mean unfished recruitment
  - recruitment deviation time series (temporal and spatial)
  - fishery selectivity (shared among some fisheries)
  - CPUE catchability (for informative longline fisheries, shared among fisheries)
  - Numbers-at-age (estimates for the final quarter included in the stock assessment are projected forward 1 quarter for the starting point of the MSE)
  - Age-dependent migration
- Conditioning consists of estimating the parameters that result in the best fit to the following data:
  - Total catch in mass or numbers (depending on the fleet)
  - 4 CPUE series (fishery-selected relative abundance indices) corresponding to the distant water longline fisheries (1 per area)
  - Catch-at-size composition for all fisheries
  - Tag recaptures (for some scenarios)

Stock Synthesis version 3.24Y was used for the conditioning. The assessment used version 3.24F, which has a bug in the biomass calculations related to the complicated growth curve. The net effect on the assessment reference case has not been examined, but it is likely to be on the order of <10% difference.

The assessment reference case included migration rates that were parameterized as a function of environmental co-variates. While there is no doubt that the environment influences migration, the mechanism remains poorly understood. We have not used this approach for OM conditioning at this time, because i) it is a new approach that has not been seriously explored (e.g. it is unclear the extent to which the data can actually distinguish between migration and seasonal catchability, or the degree to which the environmental variables are partitioning seasonal and inter-annual effects), ii) this introduces the problem of forecasting environmental variables in the MSE projections, and iii) the overall stock status estimates with and without the environmental variables did not appear to be very different in the assessment. The yellowfin assessment suggests that migration may be important in determining localized depletion (i.e. very high longline fishing mortality is estimated in the north-east region), and it is unclear whether this is a robust conclusion or an artefact of migration, selectivity and/or catchability assumptions.

The 54 SS models in OMyft2 included a balanced factorial grid of 2-3 options in each of 4 dimensions (i.e.  $3 \times 3 \times 3 \times 2 = 54$ ). Option labels (e.g. used for file management and labels in Figure 4 - Figure 6) and values are listed below; reference case assessment options are indicated in bold:

- 3 levels of Beverton-Holt stock recruit steepness ( $h$ ):
  - $h7 = 0.7$
  - **$h8 = 0.8$**
  - $h9 = 0.9$
- 3 levels of natural mortality (defined as a scaling parameter, see Figure 3):
  - $M06 = 0.6$
  - $M08 = 0.8$
  - **$M10 = 1.0$**
- 3 levels of tag likelihood ( $\lambda$ ) weighting (for the SS multinomial and Poisson likelihood terms):
  - $t00 = 0.0$  (i.e. tags omitted)
  - $t01 = 0.1$
  - **$t10 = 1.0$**  (i.e. tags not weighted, as in the assessment)
- 2 levels of assumed CPUE catchability trend (% increase per annum):
  - **$q0 = 0.0$**
  - $q1 = 1.0$

Stock-recruit steepness is notoriously difficult to estimate and the 3 levels adopted here have been entertained as equally plausible alternatives in recent IOTC yellowfin assessments.

The lower levels of  $M$  were introduced because the assessment indicated that there was substantial support for low  $M$  in the likelihood, particularly influenced by the tags. Tags are in principle the most effective means for estimating  $M$ , however when unconstrained, the preferred  $M$  values were too low to be credible.

Two scenarios with down-weighted tags were included because it is not clear that the tag assumptions can be adequately met for Indian Ocean yellowfin (Langley and Million 2012). In

addition to the strong influence on  $M$  noted above, other analyses have suggested that mixing problems are likely for many tropical tuna stocks and may introduce large estimator biases (e.g. Kolody and Hoyle 2013, 2015). There is a further concern that the default likelihood formulation used in the SS model might be over-weighting the tags, through a double-accounting process, such that down-weighting might be merited even if other tag assumptions are valid. The options represent a scalar multiplier for the tag (negative-log) likelihood (i.e. the 0 option removes the tag influence completely).

The q1 CPUE scenario assumed that longline catchability has increased at a rate of 1% per annum (compounded quarterly) in a manner that cannot be accounted for through the CPUE standardization process. This is a small but arbitrary amount, not included in the 2015 yellowfin assessment, but consistent with the assumption used for bigeye tuna in the last assessment (Langley et al 2013). To fit this scenario in the SS model, the catchability trend was imposed with an artificial CPUE data series (note that if multiple catchability options are included in the OM in this way, the MPs must not have any information to indicate which CPUE series was used in conditioning). The catchability trend is projected forward to bias future CPUE observations, but does not affect future effective effort assumptions for the purpose of MP effort-based implementation errors.

A structural option of reducing the model to 2 areas (west and east), with the same fishery definitions was explored in the early stage of the project (Kolody et al. 2015). These formulations tended to be more optimistic than the assessment in terms of equilibrium production dynamics (higher  $MSY$ ). But they were questionable because they suggested that declining CPUE trends were predominantly a function of declining recruitment, rather than fishery depletion. Given the numerical dominance of the west and east equatorial fisheries in the 4 area model, it remains unclear why the 2 area model estimated substantially different dynamics. An alternative formulation for representing this uncertainty would be to retain the 4 area structure, but set east-west migration to 0.

Key features of OMyft2 are summarized graphically in Figure 4-Figure 7, from which we note:

- 53 of the 54 SS model configurations converged adequately (on the basis of the maximum gradient plots). The one marginal case (gradient  $\sim 2$ ) was extreme in terms of the assumptions, but was retained in the OM ensemble because the corresponding dynamics were consistent with the other models.
- The total likelihood values are compared across model assumption options in Figure 4. However, these likelihoods are not comparable across tag weighting options. If only the fully-weighted tag option (t10) is considered, the likelihood favours the lower  $M$  values.
- The quality of fit to the CPUE data does not favour any particular assumption options on the basis of the CPUE Root Mean Squared Error (RMSE).
- None of the grid models estimate substantial trends in recruitment deviation over time.
- Figure 5 shows the quality of fit between predicted and observed size composition data for 4 of the larger fisheries (ESS =Effective Sample Size output from SS). For each fishery, the difference between the best and worst fit is probably not large enough to justify discarding models. For the fisheries shown, the  $M$  options have the greatest effect on the

post-fit ESS. The lower  $M$  options fit the longline data better, while the higher  $M$  options fit the purse seine data better.

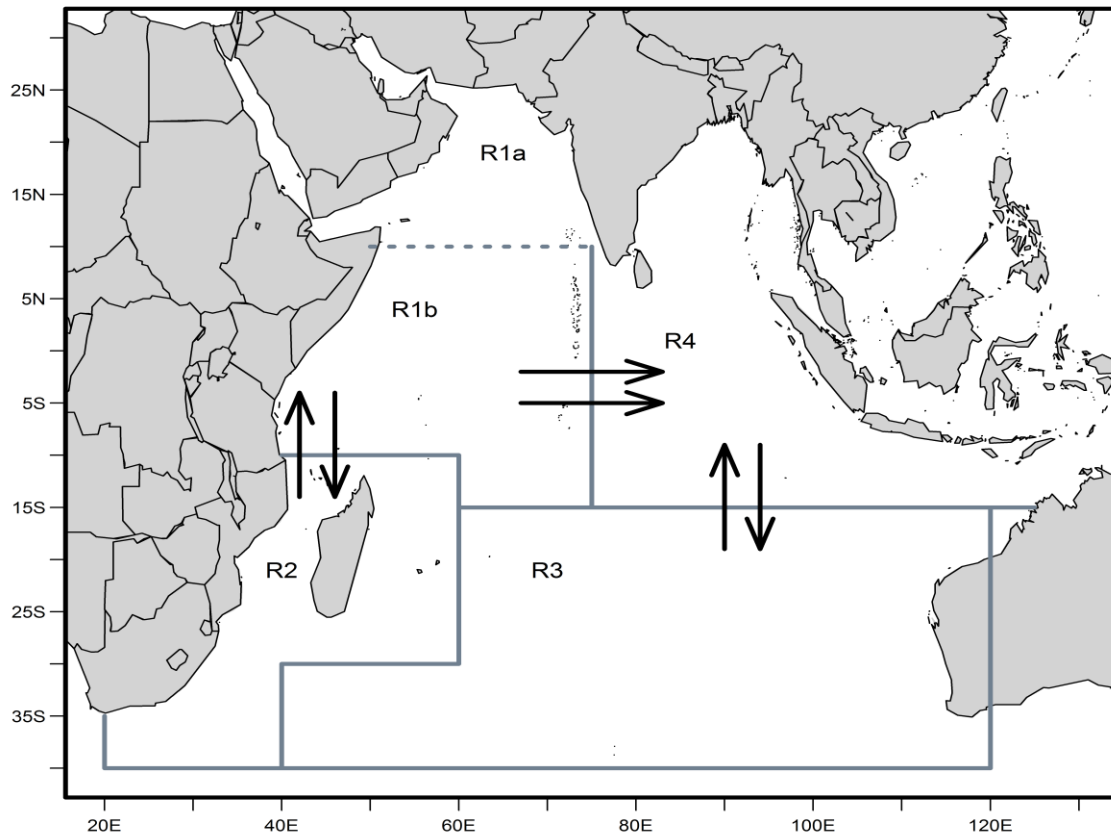
- Despite the variation in  $M$  and steepness, all of the grid models seem to have very similarly-shaped production curves (e.g.  $SB_{MSY}/SB_0 \sim 0.37$ ). The grid  $MSY$  estimates are very consistent with the assessment reference case, and the lower bound of the assessment sensitivity trials, while a minority of grid models are more optimistic than the upper end of the assessment sensitivity trials.
- The time series of  $SB/SB_{MSY}$  and  $F/F_{MSY}$  show that the OM brackets the reference case assessment, with a substantial envelope of uncertainty.

Figure 8 compares the biomass by region and migration dynamics from three contrasting SS specifications. The relative biomass by region is reasonably similar in all cases, but the absolute biomass and migration estimates are substantially different and appear to depend primarily on the tags. We would expect the tags to provide the most useful information about movement, but as with  $M$ , there are reasons to be concerned about the tag-based estimates; i.e. in addition to mixing assumptions: few fleets have reliable tag return rate estimates, the tag release design was very unbalanced, and there were very few returns from the eastern Indian Ocean. Additionally, independent spawning populations or migration route fidelity will undermine the assessment assumption that all fish of a given age and region share the same movement probabilities.

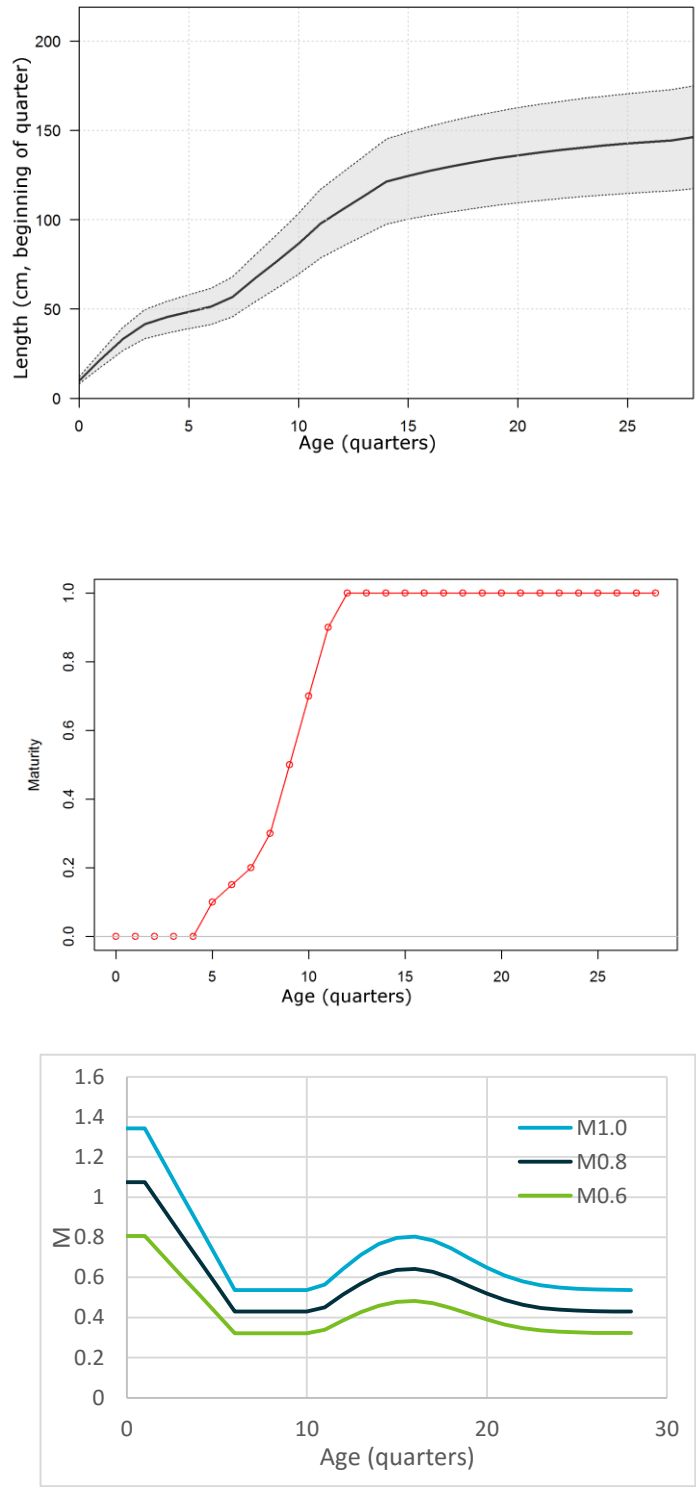


**Table 1. Fishery definitions for the Indian Ocean yellowfin operating models (from Langley 2015).**

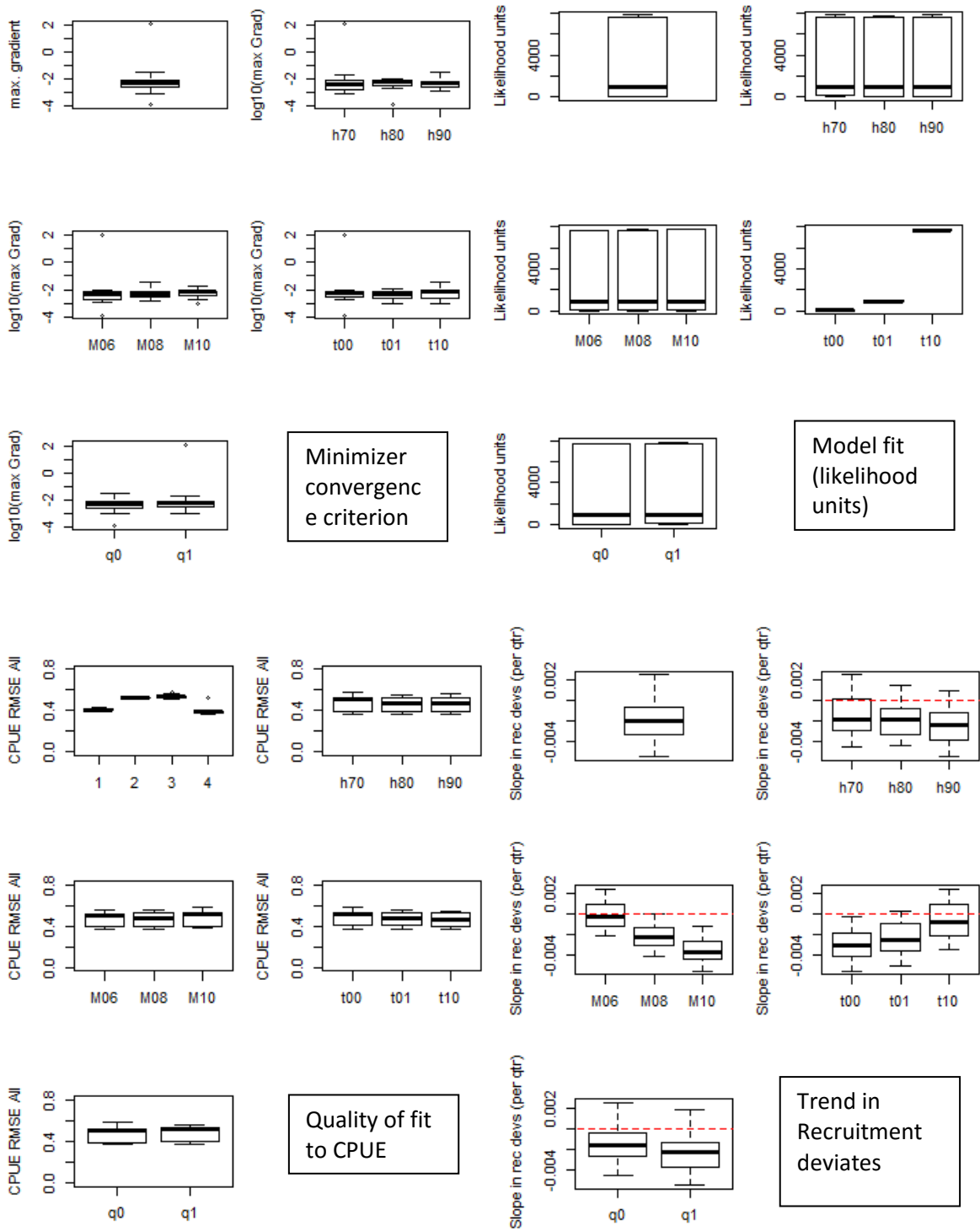
<b>Fishery</b>	<b>Nationality</b>	<b>Gear</b>	<b>Region</b>
1. GI 1a	All	Gillnet	1a
2. HD 1a	All	Handline	1a
3. LL 1a	All	Longline	1a
4. OT 1a	All	Other	1a
5. BB 1b	All	Baitboat	1b
6. PS FS 1b 2003-06	All	Purse seine, school sets	1b
7. LL 1b	All	Longline	1b
8. PS LS 1b 2003-06	All	Purse seine, log/FAD sets	1b
9. TR 1b	All	Troll	1b
10. LL 2	All	Longline	2
11. LL 3	All	Longline	3
12. GI 4	All	Gillnet	4
13. LL 4	All	Longline (distant water)	4
14. OT 4	All	Other	4
15. TR 4	All	Troll	4
16. PS FS 2	All	Purse seine, school sets	2
17. PS LS 2	All	Purse seine, log/FAD sets	2
18. TR 2	All	Troll	2
19. PS FS 4	All	Purse seine, school sets	4
20. PS LS 4	All	Purse seine, log/FAD sets	4
21. PS FS 1b pre 2003	All	Purse seine, school sets	1b
22. PS LS 1b pre 2003	All	Purse seine, log/FAD sets	1b
23. PS FS 1b post 2006	All	Purse seine, school sets	1b
24. PS LS 1b post 2006	All	Purse seine, log/FAD sets	1b
25. LF 4	All	Longline (fresh tuna)	4



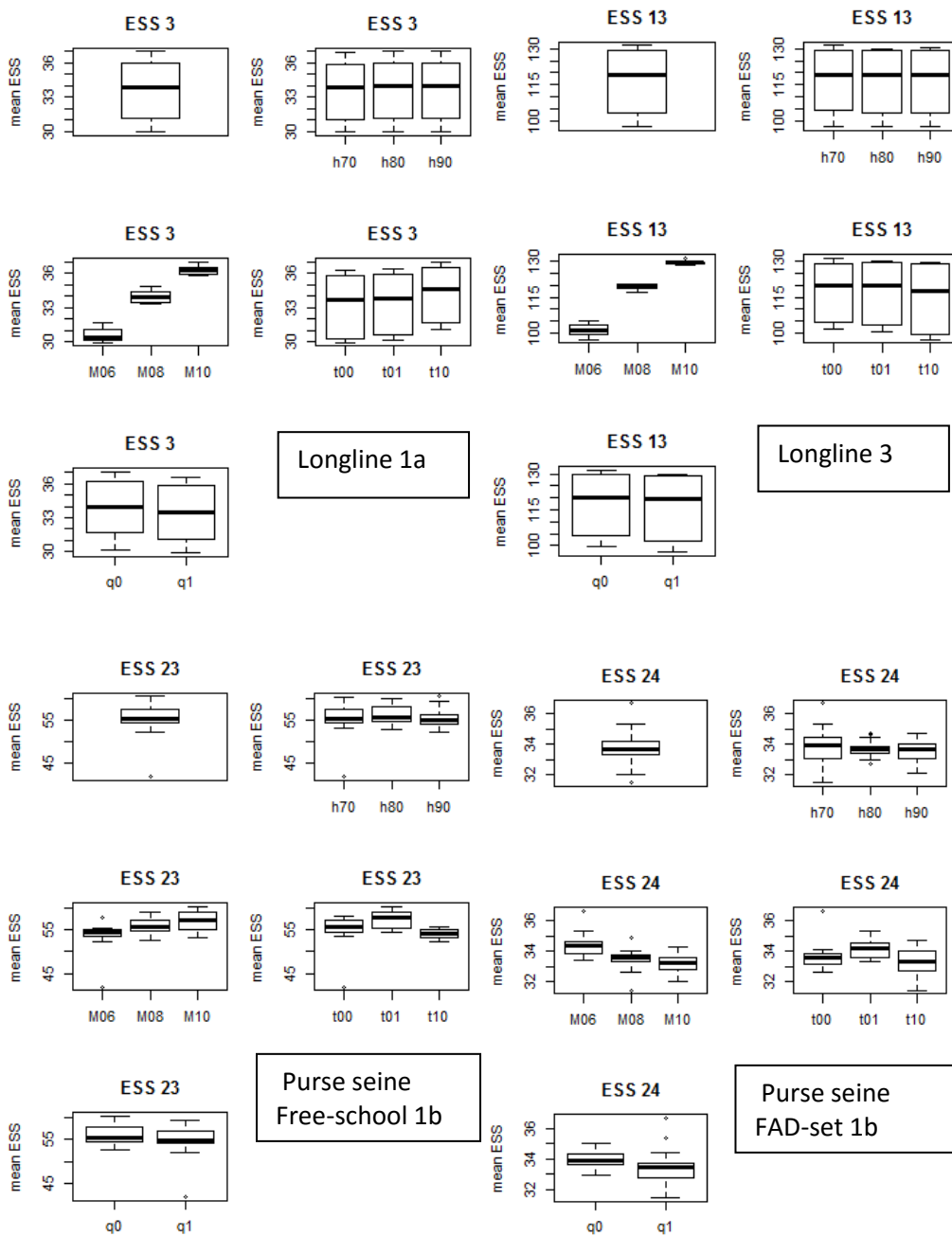
**Figure 2. Four area spatial structure used in yellowfin operating model OMyft2, from Langley (2015). The 1a and 1b regions are used for fishery definitions, but the fish population is not disaggregated between these two regions.**



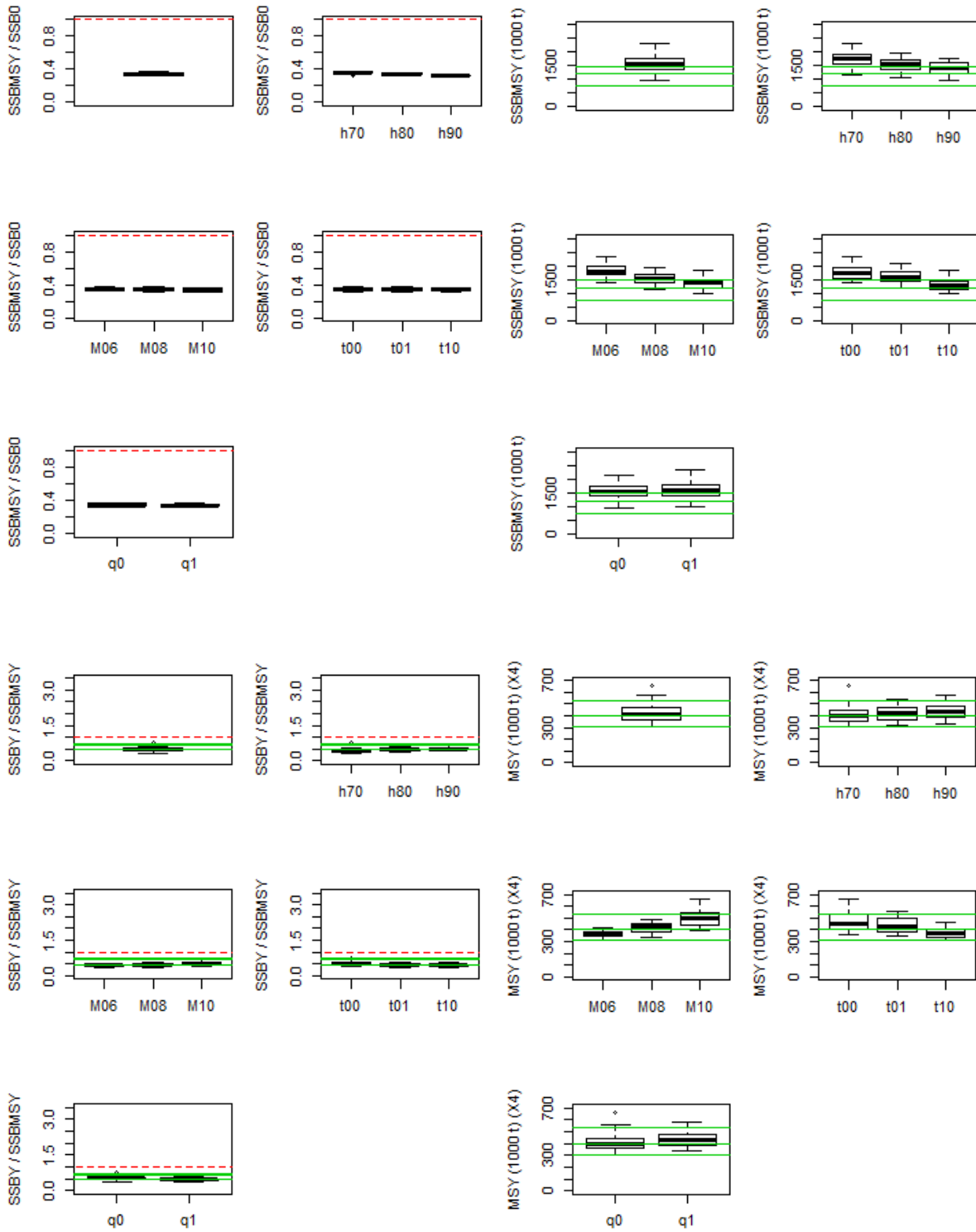
**Figure 3. Key biological assumptions in the yellowfin operating model OMyft2: length-at-age (top panel), maturity (middle panel) and three *M* vector options (bottom panel).**



**Figure 4.** Boxplots summarizing diagnostics of SS outputs for the ensemble of 54 models in OMyft2. Multiple boxplots within a panel represent all 54 values, partitioned according to the model assumption option indicated on the x-axis, and marginalized over the other options. The unlabeled panels represent all options combined. The numbered CPUE RMSE panel represents all options combined, disaggregated by region; the other CPUE RMSE panels are disaggregated by option, and aggregated over regions. Option labels are defined in the text.



**Figure 5. Boxplots summarizing diagnostics of SS outputs for the ensemble of 54 models in yftOM2. Multiple boxplots within a panel represent all 54 values, partitioned according to the model assumption indicated on the x-axis, and marginalized over the other assumption options. These panels represent the post-fit effective sample size (indicative of the quality of fit between predicted and observed catch length frequency distributions) for some key fisheries. Option labels are defined in the text (unlabeled panels represent all options combined).**



**Figure 6.** Boxplots summarizing stock status MPD estimates for the ensemble of 54 models in OMyft2. Multiple boxplots within a panel represent all 54 values, partitioned according to the model assumption indicated on the x-axis, and marginalized over the other assumption options. The green reference lines represent the value from the Langley 2015 assessment reference case, and the most extreme of the sensitivity trials. The unlabeled panels represent all options combined. Option labels are defined in the text.

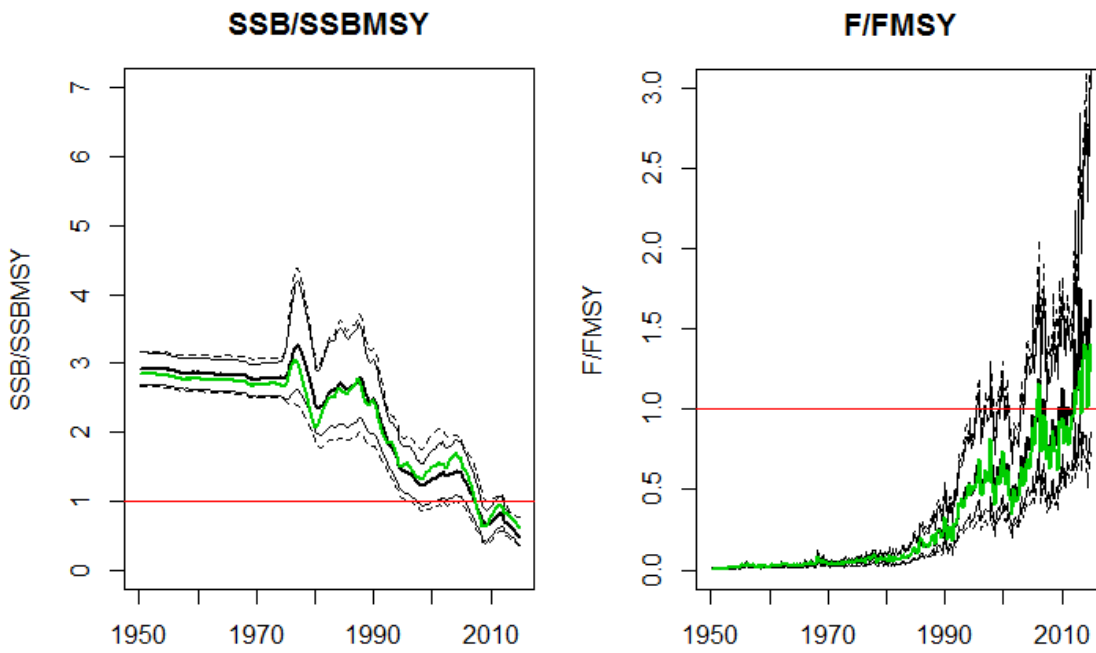


Figure 7. Time series plots summarizing  $B/BMSY$  and  $F/FMSY$  MPD estimates for the ensemble of 54 SS models in OMyft2. Black lines represent the median (bold), 95th percentiles (solid) and range (broken), green lines represent the value from the Langley 2015 assessment reference case.

### 5.1.2 Additional Yellowfin Projection Assumptions

While the conditioning process facilitates the development of OMs that are plausibly consistent with the historical data, there are additional decisions about future dynamics that need to be made. Many of these decisions can be consistent with the stock assessment input assumptions, however, there are reasons why this may not be desirable, including:

- The assessment input assumptions may not be consistent with the model outputs e.g. deviations between predictions and observations often indicate that the assumed variance is too low. But even when the magnitude of the variance is acceptable, as might be achieved through a conscientious iterative reweighting approach, strong temporal patterns in the residuals might be evident.
- It is desirable to simulation-test MPs against situations that are more challenging than the assumptions used in the assessment (e.g. the assumed variance on commercial CPUE, and the assumption that selectivity is stationary with unbiased catch-at-length sampling, may be unrealistically informative). One may also want to be prepared for specific events like recruitment failures, or non-stationary biology (e.g. changes in  $M$  or size-at-age).
- There are specifications required that are not informed by the assessment (e.g. how accurately will a TAC be implemented, and what is the catch likely to be in the period between the last available data and the first MP implementation?)

Table 2 summarizes the additional projection assumptions made for the two demonstration case yellowfin OMs discussed in this report (additional details can be found in the OM definition files and user manual). OMyftNE was defined to have minimal process and observation error (in the projections) and is primarily included for rapid development and error checking of OMs and MPs.

The OM calculates and records CPUE by region and quarter, however, to simplify the interpretation, annual spatially-aggregated indices are reported and supplied to the MPs, and CPUE observation error characteristics are based on the annual indices.

We defined the “recent” period for the purposes of MSY calculations, TAC distributions and effort multipliers by fishery as it is in the assessment (i.e. the mean over the 9 most recent quarters included in the assessment, 2012-14). For consistency with the assessment, all seasons were treated equally, but the OM has the option to describe the seasonal pattern of catch and effort.

The level of auto-correlation in CPUE and recruitment for OMyft2 was arbitrarily specified to be a plausible level that is high enough to matter, and somewhat higher than the values estimated from the residuals in typical model formulations examined. Further consideration may be warranted in specifying these values, noting that there are likely seasonal patterns that have not been adequately quantified in the current SS configurations.

At 300K t, the bridging catch levels are considerably lower than recent annual catches (e.g. > 400K t). Many of the models suggest that recent catches simply cannot be sustained given the current stock size. Forcing the MP to become active following extreme depletion may not be realistic. However, if catches after 2014 remain above 400K t, without obvious indicators of further decline or industry distress, this will provide evidence that the recent assessment and demonstration case OM are too pessimistic.

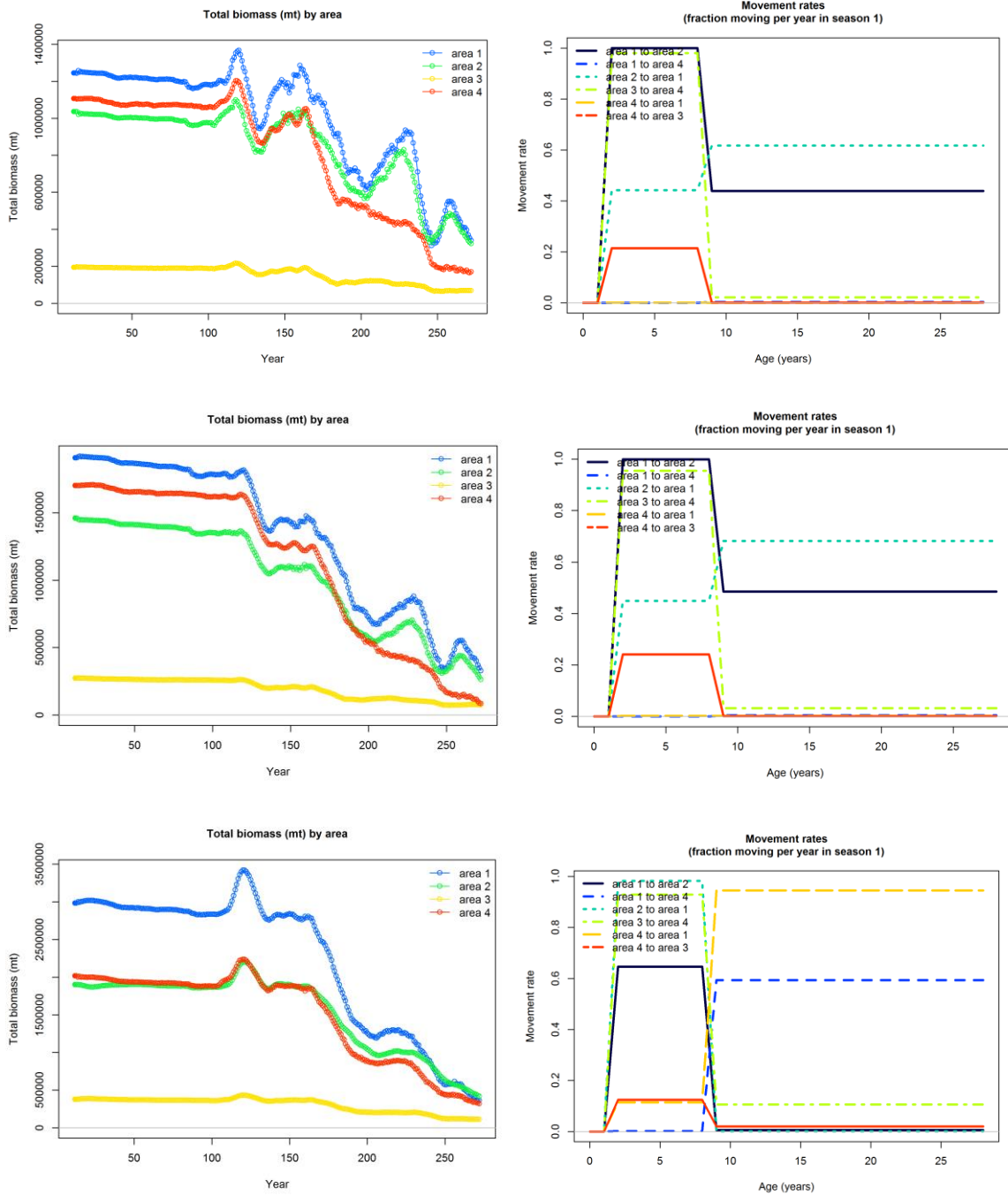


Figure 9 shows time series summaries from the ensemble of models in OMyftNE projected with fishing stopped after 2013. The pre- and post- exploitation biomass and spawning biomass appear to be consistent. Half of the future CPUE series resemble historical levels, while half exceed the historical levels (which is consistent with the catchability trend scenarios). Figure 10 is the equivalent for OMyft2, indicating the effects of the variability due to the full ensemble of SS models, and stochastic recruitment and CPUE observation error.

Figure 11 illustrates the reference case constant catch projection (80% of recent catch) from the Langley (2015) assessment. Figure 12 is an analogous projection from 2160 realizations of OMyft2. The central tendency of OMyft2 is more optimistic than the assessment. The level of uncertainty in the assessment projections (inverse Hessian-delta approximation) appears to be very high, with a greater probability of very pessimistic outcomes relative to the OMyft2 ensemble. This was not expected given that the assessment result is based on a single model specification with a large number of influential parameters assumed to be known without error (and at a level that is more optimistic than many of the OMyft2 grid options). It is not obvious whether the assessment projections should be considered to have unrealistically high uncertainty, or whether OMyft2 represents a serious understatement of the uncertainty due to the limitations of reliance on the MPD grid to capture parameter uncertainty.

**Table 2. Assumptions for yellowfin demonstration case operating models OMyftNE and OMyft2.**

OM feature	OMyftNE	OMyft2
Number of SS models	2	54 (4 dimensions)
First Projection year	2015	2015
First MP year	2018	2018
Bridging Catch (1000 t)	0	300
Bridging Catch CV (each fishery)	0.1	0.1
"Recent" period (9 quarters non-seasonal)	2012-14	2012-14
Projection period (years)	50	27
Initial N(a=0) CV	0	0.6
Initial N(a) CV decay	0	0.1
Recruitment CV (quarterly)	0	0.6
Recruitment CV (spatial)	0	0
Recruitment auto-correlation (quarterly)	0	0.5
CL sample size (each fishery)	1000	100
CPUE CV (annual aggregate)	0	0.3
CPUE auto-correlation	0	0.5
TAC/TAE Implementation error	0	0.1



**Figure 8. Total biomass by region, and migration rate estimates from 3 contrasting SS models in the yellowfin OMYft2 ensemble (note that time series and migration rates are quarterly). Top panels have high  $M$ , high steepness, no longline catchability trend and include tags; middle panels have low  $M$ , low steepness, 1% longline catchability trend and include tag data; bottom panels have low  $M$ , low steepness, 1% longline catchability trend but exclude tag data.**

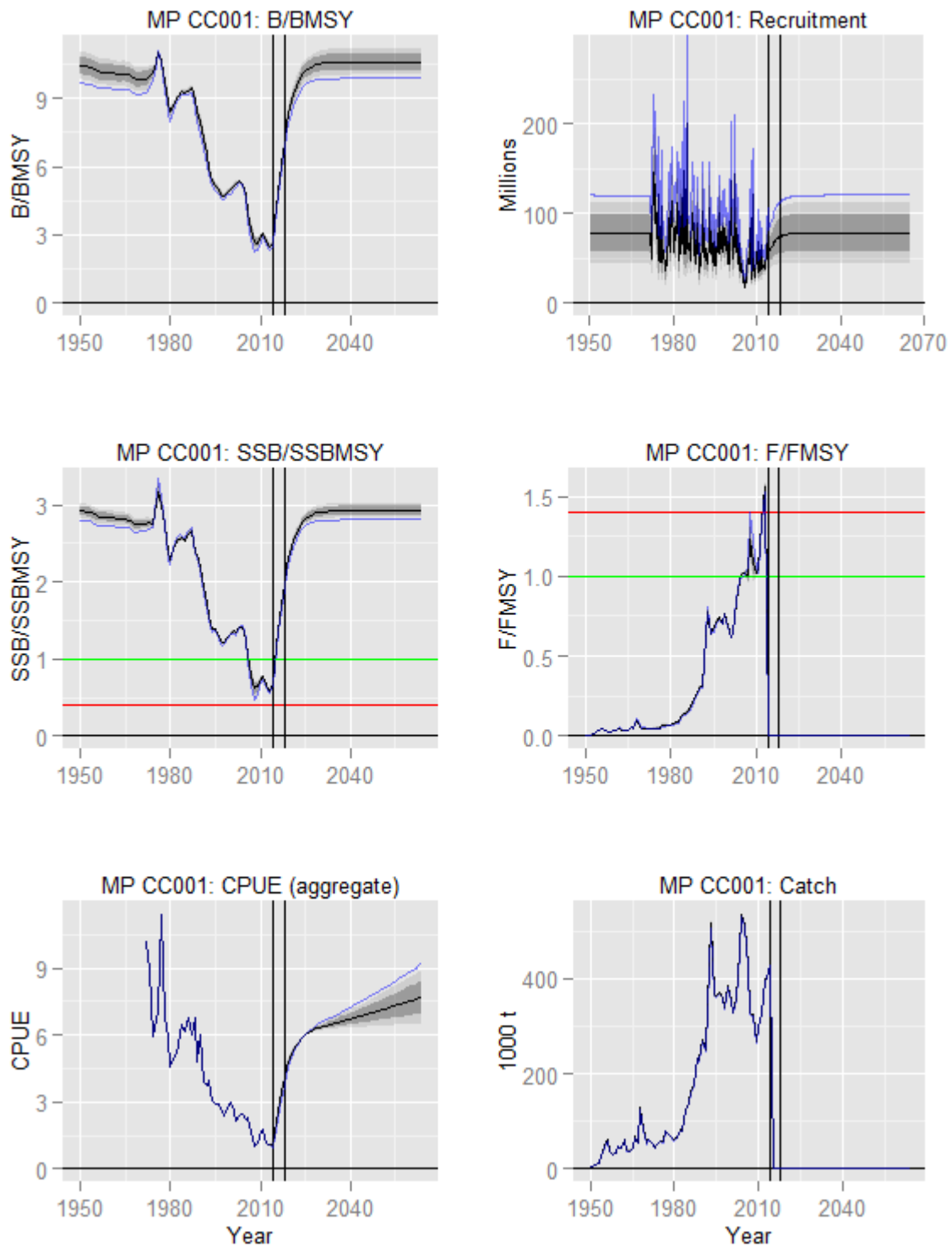


Figure 9. Fifty year projections for the operating model OMyftNE, with minimal process and observation error, and all fisheries shut-off in 2014.

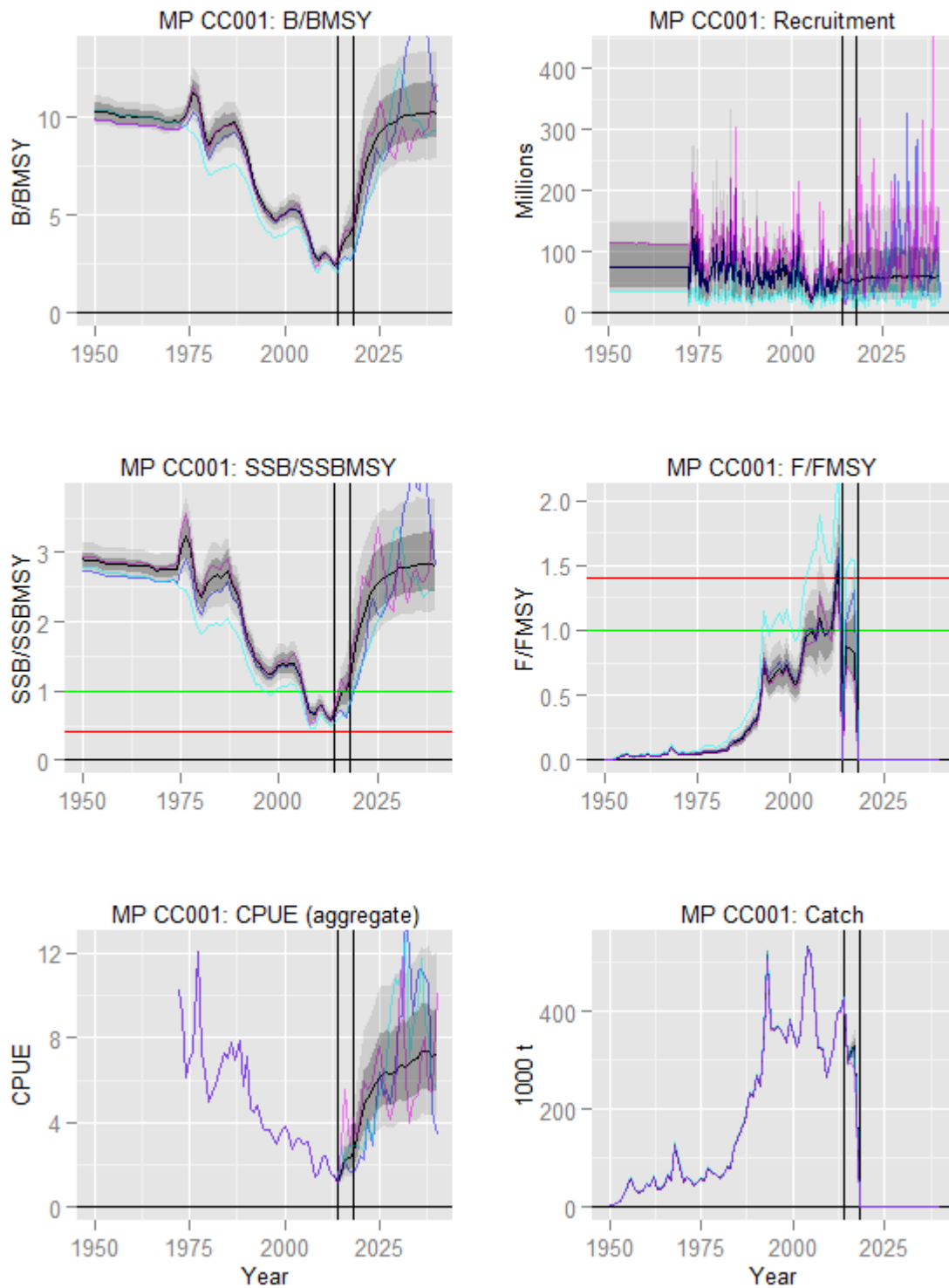


Figure 10. Twenty-five year projections for the operating model OMyft2 with all fisheries shut-off in 2018.

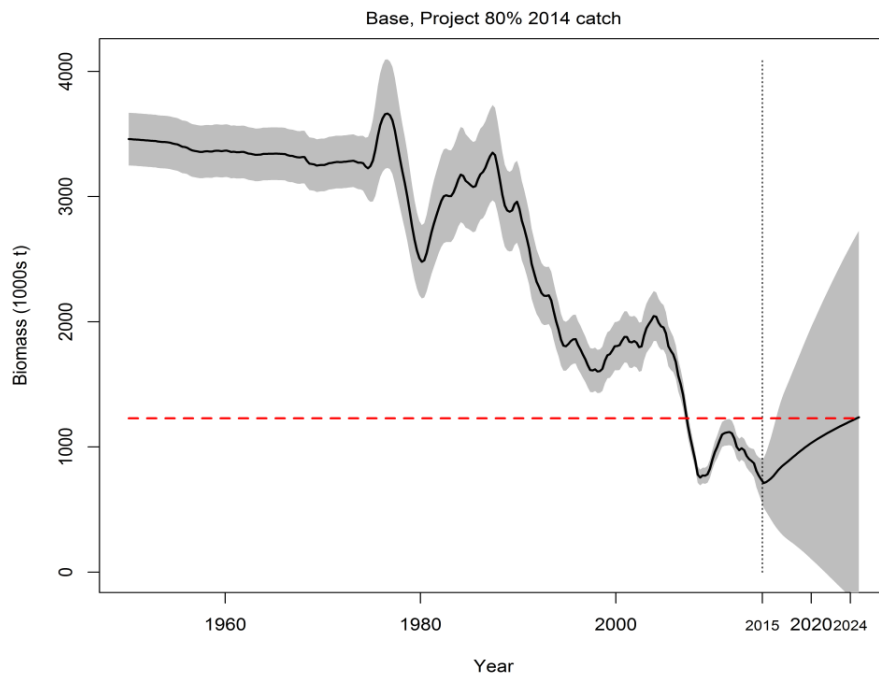


Figure 11. Constant catch projections at 80% of current from the reference case yellowfin assessment including the inverse Hessian error estimates (Langley 2015).

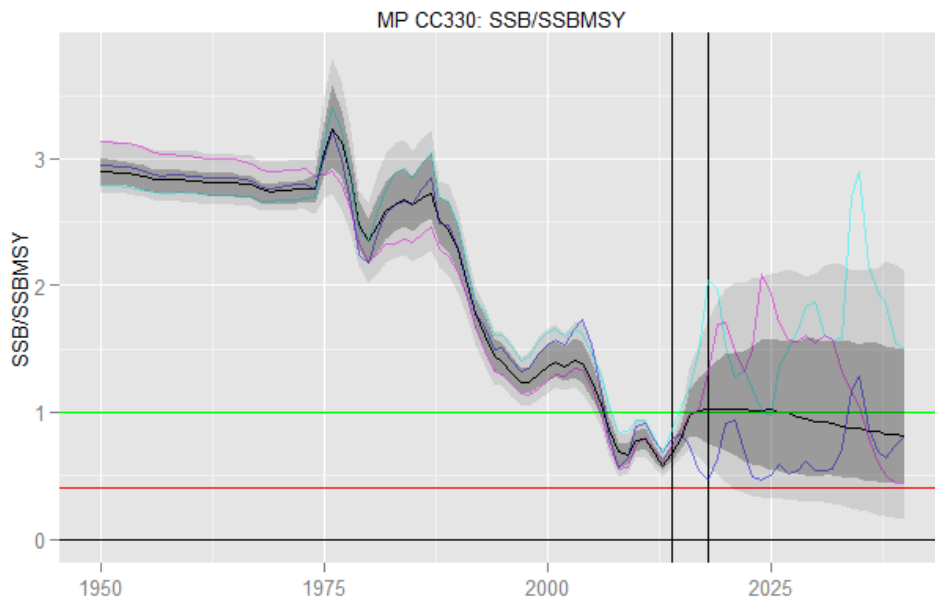


Figure 12. Constant catch projections at 80% of current from the demonstration case yellowfin operating model OMyft2r.

## 5.2 Demonstration Bigeye Tuna Operating Model

The Langley et al. (2013) assessment was very compatible with the development of an ensemble OM configuration because a grid-based approach was used to explore alternative assumptions and their interactions. This grid covered the most obvious sensitivity assumptions, and was adopted as the basis for the demonstration bigeye OM OMBet1 with only one addition (an intermediate  $M$  option).

### 5.2.1 Conditioning Assumptions

The OMBet1 ensemble of SS models included the following characteristics (differences are described subsequently):

- Structure:
  - 1 region (Figure 13)
  - Single Stock
  - Age-structured population
  - Conditioned with data from 1950-2011, with quarterly dynamics
  - Sex-aggregated
  - 12 fisheries with stationary selectivity (Table 3)
  - Beverton-Holt stock-recruit relationship
    - Log-normal quarterly recruitment deviates
  - Baranov catch equations (implemented with the SS "hybrid" approach)
- Fixed parameters (though some vary among specifications as detailed subsequently):
  - Size-at-age (Figure 14)
  - Maturity-at-age (with fecundity proportional to mass) (Figure 14)
  - Natural mortality (Figure 14)
  - Stock-recruit steepness
  - Recruitment variance ( $\sigma_{Rt}$ )
  - CPUE observation error variance ( $\sigma_{CPUE}$ )
  - Size composition sampling variance (i.e. assumed sample sizes for multinomial likelihood, which tend to be far lower than true sample sizes; adopted from the assessment which was specified with some consideration of iterative re-weighting principles)
- Estimated states and parameters include:
  - mean unfished recruitment
  - recruitment deviation time series
  - fishery selectivity (shared among some fisheries)
  - CPUE catchability
  - Numbers-at-age (the starting point for the MSE projection code corresponds to the last time-step in the assessment, projected forward one time-step)

- Conditioning consists of estimating the parameters which result in the best fit to the following data:
  - Total catch in mass or numbers (depending on the fleet)
  - CPUE (fishery-selected relative abundance indices) corresponding to the whole Indian Ocean Japanese longline fishery
  - Catch-at-size composition for all fisheries

Tags were excluded from the bigeye assessment and OM primarily because of concerns about tag mixing. Stock Synthesis version 3.24Y was used for the conditioning. The assessment used SS 3.24M (no obvious difference in the results was noted between versions).

The 18 SS models in OMbet1 ensemble included a balanced factorial grid of 2-3 options in each of 3 dimensions. Option labels (e.g. used for file management and Figure 15 - Figure 17 labels) and values are listed below:

- 3 levels of Beverton-Holt stock recruit steepness ( $h$ ):
  - $h7 = 0.7$
  - $h8 = 0.8$
  - $h9 = 0.9$
- 3 levels of natural mortality correspond to multipliers applied to the older ages (Figure 14):
  - $M06 = 0.6$
  - $M08 = 0.8$
  - $M10 = 1.0$
- 2 levels of assumed CPUE catchability trend (% increase per annum):
  - $q0 = 0.0$
  - $q1 = 1.0$

Key features of the OMbet1 grid model ensemble are summarized graphically in Figure 15 - Figure 18, from which we note:

- The OMbet1 grid is identical to the assessment grid, except for the addition of the intermediate  $M$  vector. Langley et al (2013) refer to vectors  $M1$ ,  $M2$ , which we refer to as  $M10$  and  $M06$  respectively.  $M08$  was added on the basis of Figure 17, in which it appears that  $M$  has a substantial influence on stock status estimates, such that using only the two levels from the assessment tends to result in clearly bimodal inferences.
- All of the models appeared to converge adequately (Figure 15).
- The likelihood consistently favours the more optimistic model assumptions (high  $M$ , high steepness and no catchability trend).
- The longline catchability option is the most important assumption influencing the likelihood (no trend fits better than the 1% trend). Even though different data series are used to implement the CPUE options, these likelihoods are comparable, because the data were simply transformed in a manner that is equivalent to adding a catchability trend

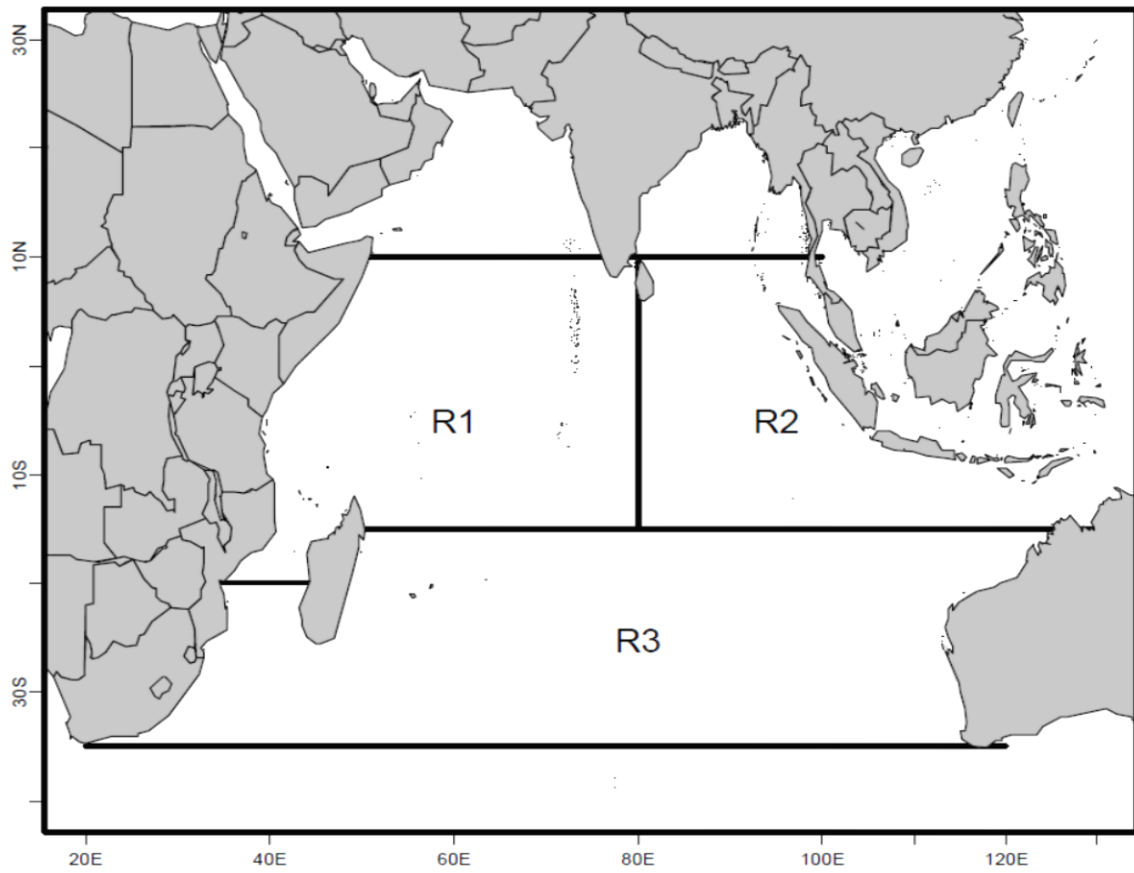
parameter. Qualitatively, however, it is doubtful that the fit to the CPUE series is substantially different based on the RMSE (Figure 15).

- There is no substantial trend in the recruitment deviates (Figure 15).
- As with yellowfin, it is not obvious that any of the model assumptions result in clearly better or worse fit to the size composition data (Figure 16) and the effects of the assumption are not consistent among fisheries.
- The bigeye equilibrium production curve is estimated to be more variable than yellowfin, with  $SB_{MSY}/SB_0$  ranging from  $\sim 0.22-0.32$ .
- Productivity in terms of  $MSY$  is qualitatively consistent with expectations, i.e. the highest  $MSY \sim 200\text{Kt}$  occurs with high  $M$ , high steepness and no catchability trend, while the opposite extreme yields  $MSY \sim 100\text{K t}$ .

**Table 3. Fishery definitions from the bigeye operating models (from Langley et al 2013).**

Code	Method	Region	Flag	Notes
FL2	Longline, fresh tuna fleets	2	All	
LL1	Longline, distant water	1	All	
LL2	Longline, distant water	2	All	
LL3	Longline, distant water	3	All	
PSFS1	Purse seine, free school	1	All	
PSFS2	Purse seine, free school	2	All	
PSLS1	Purse seine, associated sets	1	All	
PSLS2	Purse seine, associated sets	2	All	
BB1	Baitboat and small scale encircling gears (PSS, RN)	1	All	Primarily catch from the Maldives baitboat fishery. Predominantly catching small bigeye.
LINE2	Mixed gears (hand-line, gillnet/longline combination)	2	All	Gears grouped on the basis that primarily catch large bigeye.
OT1	Other (trolling, gillnet, unclassified)	1	All	
OT2	Other (trolling, gillnet, unclassified)	2	All	





**Figure 13. Spatial structure used to define the bigeye operating model betOM1 fisheries (from Langley et al 2013). The regions were used to define homogenous fisheries, but the population dynamics were spatially aggregated.**

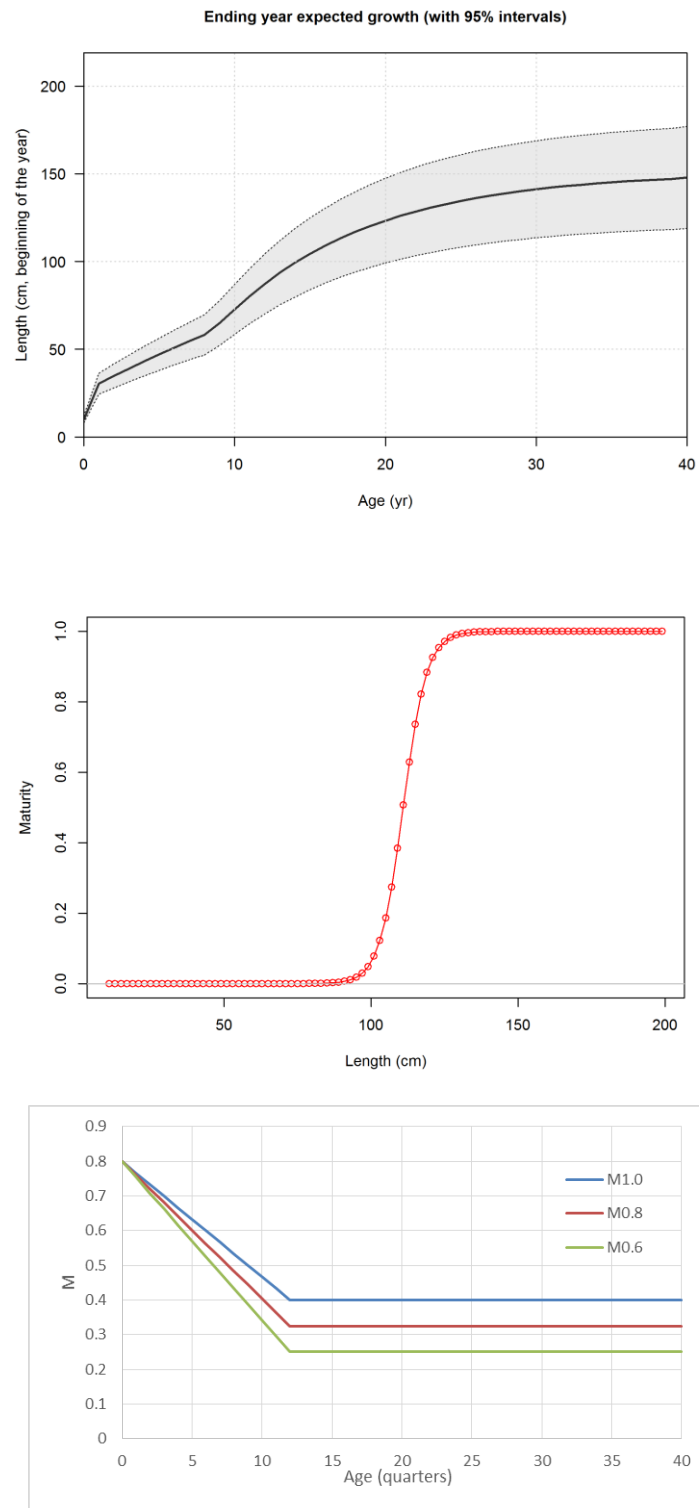
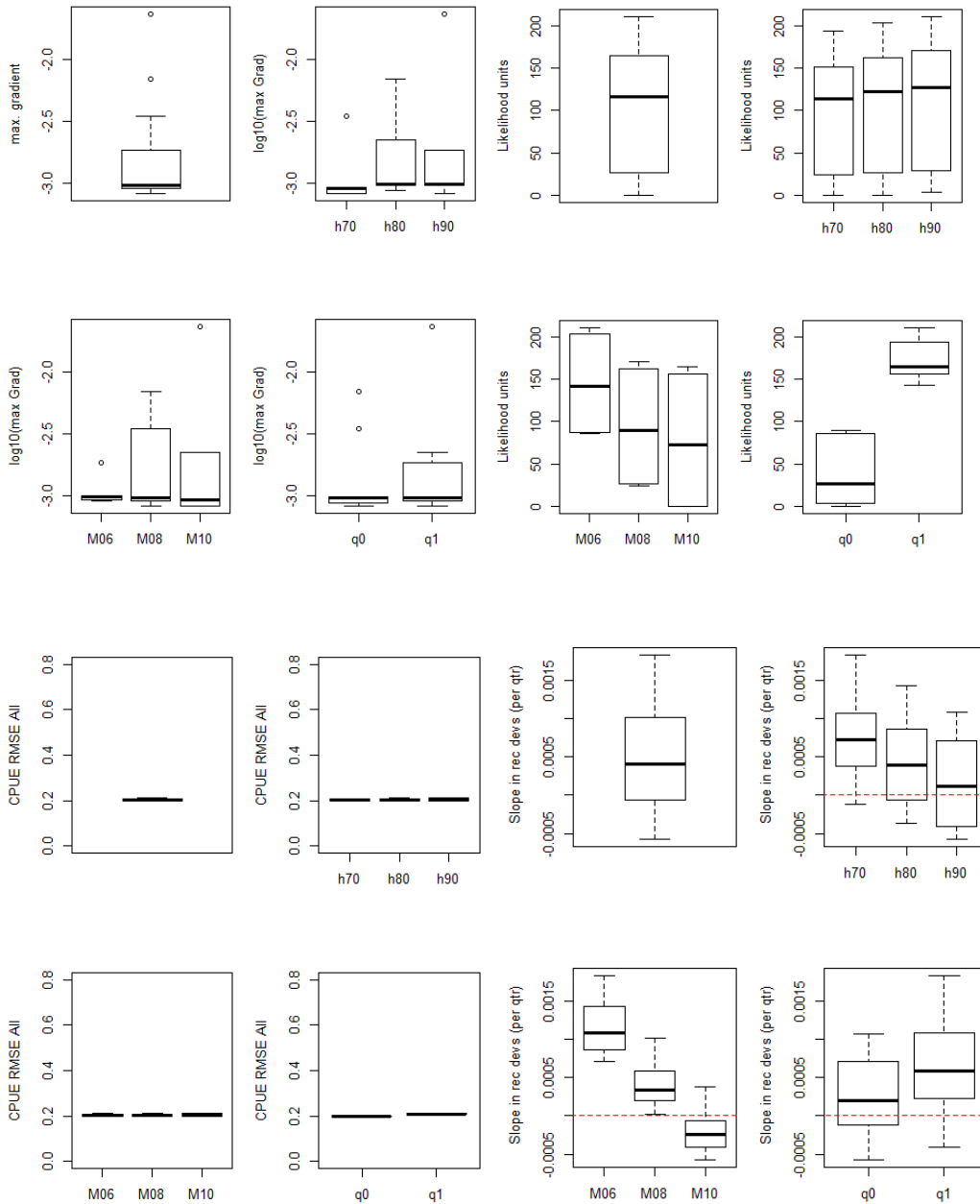
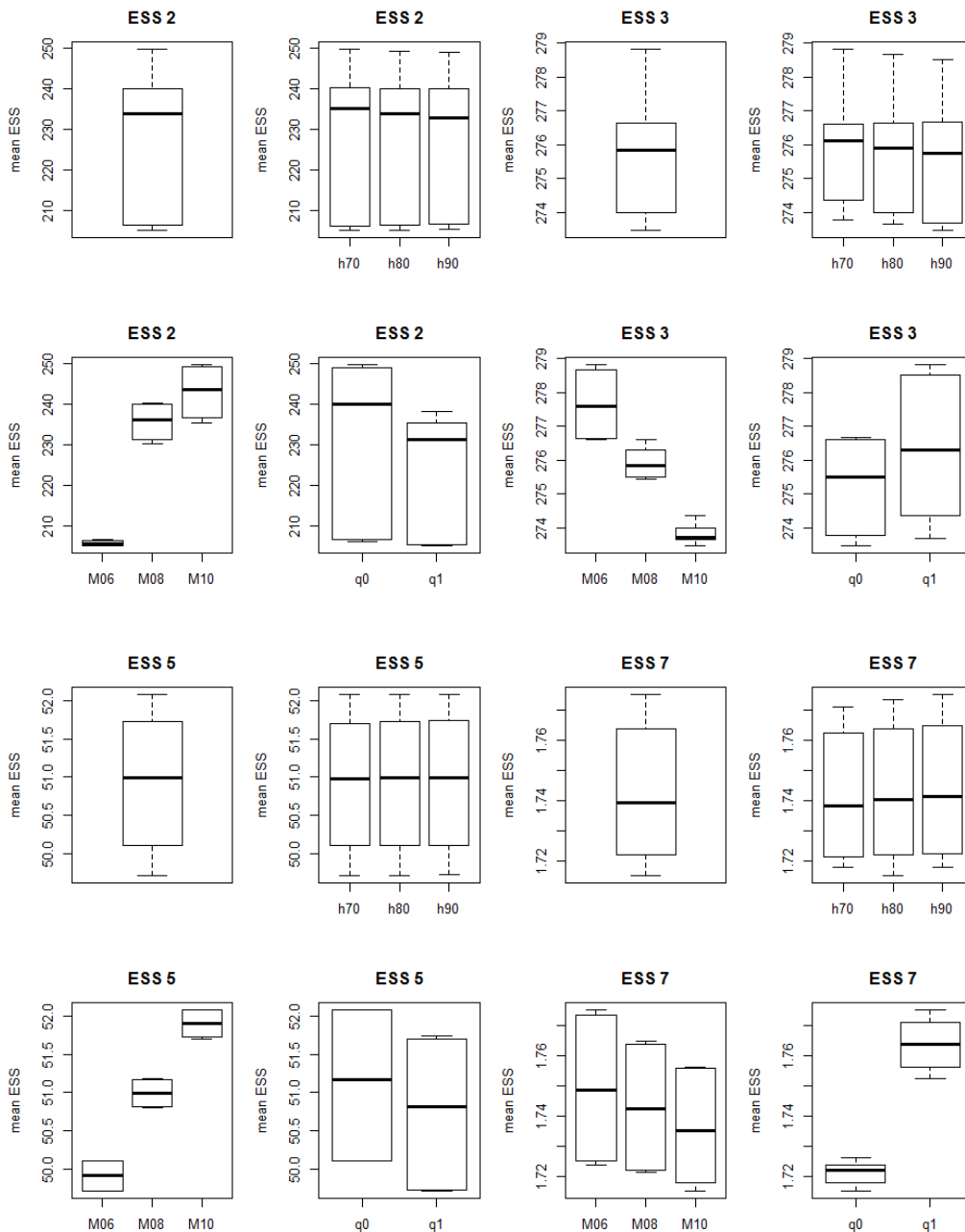


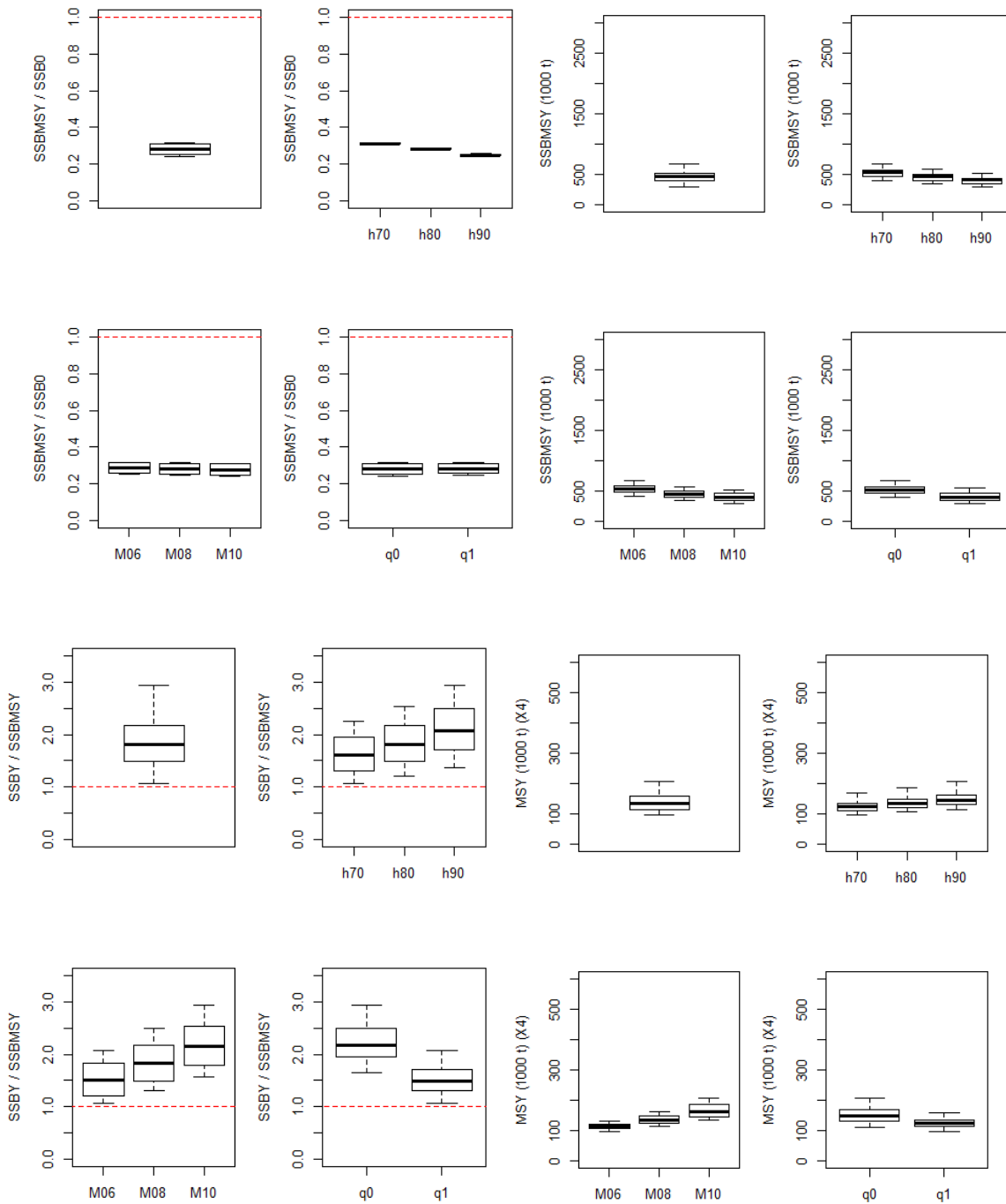
Figure 14. Key biological assumptions in the bigeye OM Ombet1: length-at-age (top panel), maturity (middle panel) and three  $M$  vector options (bottom panel).



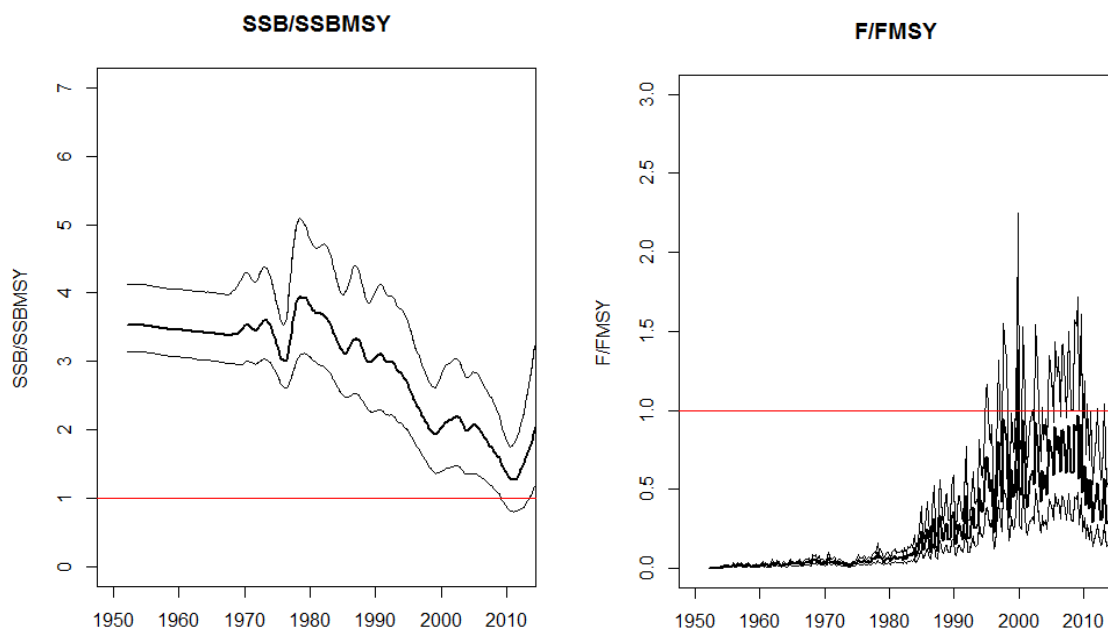
**Figure 15. Boxplots summarizing diagnostics of SS outputs for the ensemble of 18 models in OMbet1. Multiple boxplots within a panel represent all 18 values, partitioned according to the model assumption indicated on the x-axis, and marginalized over the other assumption options. Option labels are defined in the text (unlabeled panels represent all options combined).**



**Figure 16. Boxplots summarizing diagnostics of SS outputs for the ensemble of 18 models in OMbet1. Multiple boxplots within a panel represent all 18 values, partitioned according to the model assumption indicated on the x-axis, and marginalized over the other assumption options. Option labels are defined in the text (unlabeled panels represent all options combined).**



**Figure 17. Boxplots summarizing diagnostics of SS outputs for the ensemble of 18 models in OMbet1. Multiple boxplots within a panel represent all 18 values, partitioned according to the model assumption indicated on the x-axis, and marginalized over the other assumption options. Option labels are defined in the text (unlabeled panels represent all options combined).**



**Figure 18.** Time series plots summarizing the median (wide lines) and range (narrow lines) of  $B/B_{MSY}$  and  $F/F_{MSY}$  time series estimates from the ensemble of 18 models in OMbet1. The reference case assessment is not included for bigeye, because the bigeye assessment emphasized the whole grid of options (except for the additional M08 vector).

### 5.2.2 Additional Bigeye OM Projection Assumption

Table 4 summarizes the assumptions for the two demonstration case bigeye OMs discussed in this report (additional details are provided in the OM definition files and user manual).

We defined the “recent” period for the purposes of MSY calculations, TAC distributions and recent effective effort as in the assessment (2007-2010, mean over seasons).

Figure 19 and Figure 20 show time series summaries from the ensemble of models in OMbet1 projected without fishing. The pre- and post- exploitation  $B$ ,  $SB$  and CPUE all appear to be consistent (except that, as with yellowfin, half of the CPUE scenarios include a rising catchability trend, which results in future CPUE exceeding the historical levels).

**Table 4. Bigeye demonstration case operating model projection assumptions.**

OM feature	OMbet0	OMbet1
Number of SS models	2	18 (3 dimensions)
First Projection year	2012	2012
First MP year	2018	2018
Bridging Catch	Catch(y=2011)	Catch(y=2011)
Bridging Catch CV (each fishery)	0.1	0.1
"Recent" Catch-effort period (non-seasonal)	2007-2010	2007-2010
Initial N(a=0) CV	0	0.6
Initial N(a) CV decay parameter	0	0.1
Recruitment CV (quarterly)	0	0.6
Recruitment CV (spatial)	0	0
Recruitment auto-correlation (quarterly)	0	0.5
CL sample size	1000	100
CPUE CV (annual)	0	0.3
CPUE auto-correlation (annual)	0	0.5
TAC/TAE implementation error	0	0.1

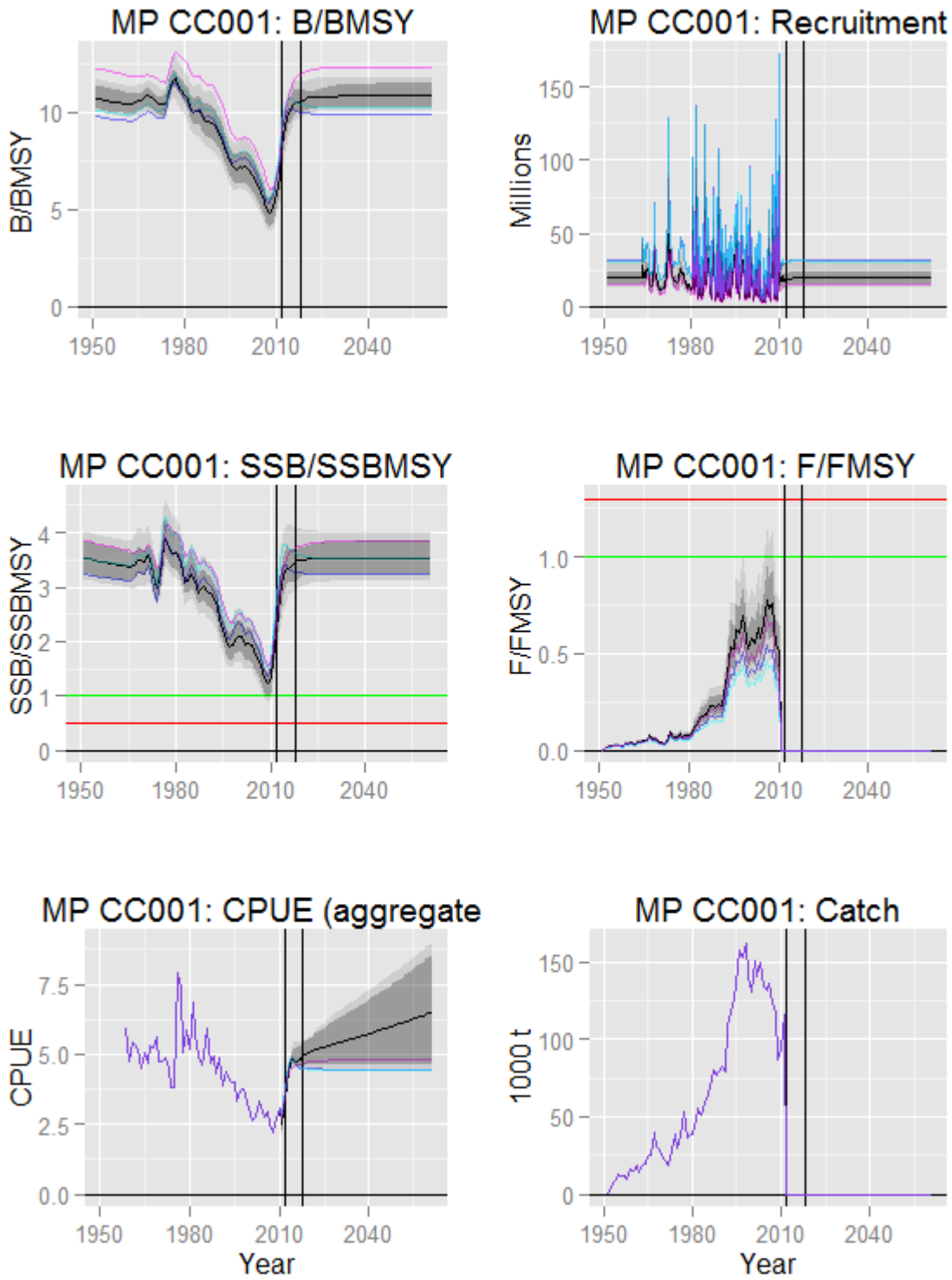


Figure 19. Fifty year projections for the operating model OMBet0, with minimal process and observation error, and all fisheries shut-off.



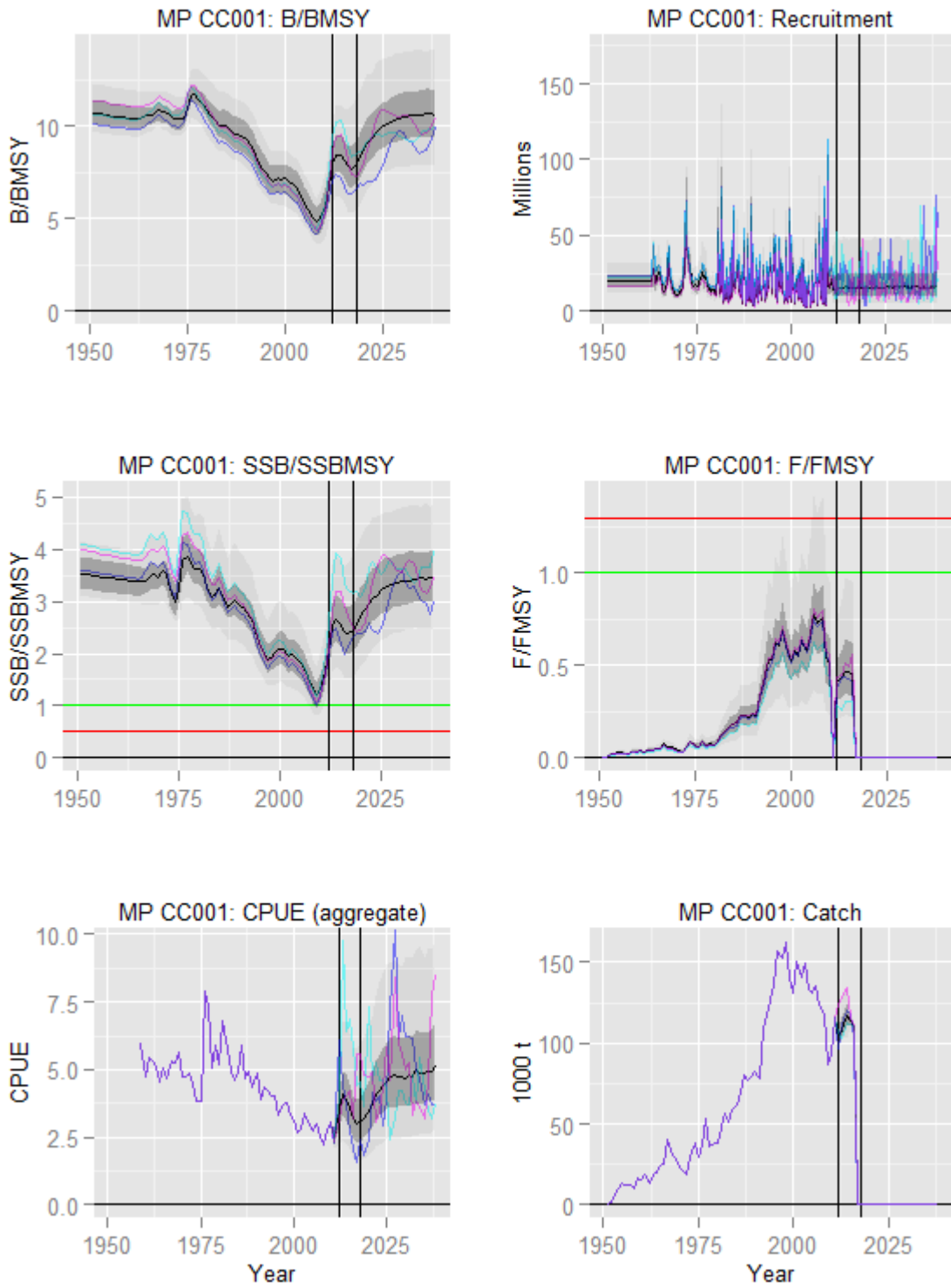


Figure 20. Projections for the operating model OMBet1, with all fisheries shut-off in 2018.

## 6 Candidate Management Procedures

Several candidate MPs are included with the software and have been applied in the demonstration cases included in this report. In pursuing the *sensu stricto* definition of MP, the MSE software attempts to reproduce key elements of the real assessment and management process. The OM admits considerable uncertainty about the underlying dynamics of the fishery by using different parameters in the simulations. However, as in the real world, the MP never receives any direct insight about the underlying dynamics of the OM, and must make inferences on the basis of the incoming fisheries data (the historical data are available as well, but are always the same). Feedback-based MPs attempt to extract a useful signal from these incoming data, and the level of analytical complexity used to accomplish this might vary considerably in different MPs. A simple empirical MP might calculate the TAC as a trivial function of the raw data. A complicated model-based MP could fit a statistical age-structured population model with an HCR that is based on the confidence bounds of parameter estimates. Computationally-demanding MPs may be impractical to simulation test because of time constraints (and it may be impossible to automate complicated model fitting and diagnostic checking). But it is only by directly simulating the full numerical algorithm in the MSE, that we are able to directly compare the performance of different MPs.

The included MPs fall into the 3 general categories listed in Table 5, and differ in terms of the specific control parameters (i.e. each different level of constant catch represents a different MP, but all fall into the general CCE category i.e. constant catch and/or effort). The CCE MPs are not intended to be viable candidate MPs because they lack the responsiveness of feedback-based control. These MPs are included primarily for testing, and to allow continuity with the constant catch projections from the Kobe-2 Strategy Matrices provided with recent IOTC stock assessments (e.g. Langley 2013, 2015). Empirical and model-based MPs are included, but they are not necessarily the best examples in terms of either the structure or control parameters. Control parameters for the different MP versions included in this report are included in Table 6 (the precise mathematical descriptions are included in Appendix 1). These parameters were haphazardly selected with the objective of illustrating the broad trade-off between catch and biomass performance indicators (over 10 and 20 year time periods).

The empirical "aim for CPUE target" (IT) MP is shown schematically in Figure 21. Only CPUE data are required for the TAC calculation. Conceptually, the MP assumes that i) CPUE is proportional to abundance, and ii) there is a target CPUE that represents target abundance, such that the population can be driven toward, and sustained near to, the target abundance by adjusting the catches. There is no traditional stock assessment model associated with this MP. The MP simply calculates the recent CPUE trend, compares current CPUE with a target CPUE and uses a simple algorithm to output a new TAC.

The model-based MP (PT) is shown schematically in Figure 22. This MP requires the CPUE and total catch time series. It includes a simple stock assessment model combined with an HCR. Every time the MP is called, it first estimates key parameters for a surplus production model by fitting to the latest catch and CPUE time series. The resulting point estimates of current depletion and MSY are then used with a 40:10-type HCR to set the new TAC.

In practice, a formal MP requires a number of additional specifications that have been included in the OM (e.g. date of first MP application, data availability delay). All simulations to date assumed that the MP would be applied triennially, but the MSE control software does allow this option to be changed.

Additional optional control parameters might be imposed on any MP, e.g., in the interest of industry stability and investment planning, we included constraints to the maximum change in TAC between consecutive settings. Limiting the allowed inter-annual increase in TAC can reduce the risk of rapid industry over-investment due to an unsustainable anomaly, while constraining the amount that the TAC can decrease should reduce fears about sudden, economically devastating fishery closures. But it is important to recognize that these measures could have substantial effects on MP performance and should be explicitly included in the simulation testing. A reduced ability to change the magnitude of the TACs from year to year comes with the benefit of increased industry stability, but it comes with the cost of reduced ability to react to negative (or positive) situations. Thus there is often a total catch vs: catch stability trade-off.

**Table 5. MP categories included in this report. See Appendix 1 for the mathematical definition.**

MP Category	Description
CCE	<p><b>Constant catch and/or effort:</b></p> <ul style="list-style-type: none"> <li>• Some or all fisheries are managed with a constant TAC, distributed among fisheries and seasons according to recent observed catch distributions.</li> <li>• Some or all fisheries are managed by a constant effort multiplier, defined relative to recent effective effort levels estimated in the OM conditioning.</li> <li>• Control parameters include an aggregate TAC, and fishery-specific effort multipliers (each fishery is managed by either TAC or effort multiplier, but not both).</li> </ul>
IT (Figure 21)	<p><b>Aim for a CPUE target (simplified version of Prince et al. 2011):</b></p> <ul style="list-style-type: none"> <li>• TAC is set based on recent CPUE trends, and the difference between recent CPUE and a desirable CPUE target (<math>CPUE_{target}</math>).</li> <li>• Increasing CPUE trend or <math>CPUE &gt; CPUE_{target}</math> tends to increase TAC.</li> <li>• Decreasing CPUE trend or <math>CPUE &lt; CPUE_{target}</math> tends to decrease TAC.</li> <li>• Control parameters include: <math>CPUE_{target}</math>, number of years to include in the CPUE trend calculation, TAC change responsiveness to increasing and decreasing CPUE, TAC change responsiveness to the difference between recent CPUE and CPUE target, maximum TAC change limits.</li> </ul>
PT (Figure 22)	<p><b>Pella-Tomlinson 40:10-type rule:</b></p> <ul style="list-style-type: none"> <li>• Fit a Pella-Tomlinson surplus production model to catch and CPUE time series and calculate TAC as a function of estimated depletion, estimated <i>MSY</i> and a 40:10-type HCR.</li> <li>• Control parameters include the thresholds of the 40:10 HCR, maximum TAC as a function of estimated <i>MSY</i>, Pella-Tomlinson shape parameter, maximum TAC change limits.</li> </ul>

**Table 6. Specific MPs, including control parameters. See Appendix 1 for the mathematical definition.**

MP	Control Parameters			
CCE	TAC (1000t) (for TAC fisheries) $k^{TAC}$	TAE (Effort Multiplier) (for TAE fisheries) $k_f^{TAE}$		
CC050	50	N/A		
CC100	100	N/A		
CC150	150	N/A		
CC200	200	N/A		
CC300	300	N/A		
CC400	400	N/A		
CE1.0	0	1		
IT	CPUE Target (relative annual units) $I^T$	CPUE difference responsiveness $k_1, k_2$	CPUE Slope responsiveness $k_3, k_4$	TAC change limits $\Delta^{\min}, \Delta^{\max}$
IT1.50	1.5	0.1	0.1	+/- 20%
IT2.50	2.5	0.1	0.1	+/- 20%
IT3.50	3.5	0.1	0.1	+/- 20%
PT	HCR slope Bounds (Bt/B0) ( $k_1, k_2$ )	Max TAC relative to MSY estimate ( $k_3$ )	TAC change limits $\Delta^{\min}, \Delta^{\max}$	PT Shape parameter ( $B_{MSY}/B_0$ )
PT41.100.2	0.1, 0.4	1.0	+/- 20%	0.33
PT41.100.5	0.1, 0.4	1.0	+/- 50%	0.33
PT41.100.9	0.1, 0.4	1.0	+/- 90%	0.33
PT42.125.2	0.2, 0.4	1.25	+/- 20%	0.33
PT82.150.2	0.2, 0.8	1.5	+/- 20%	0.33

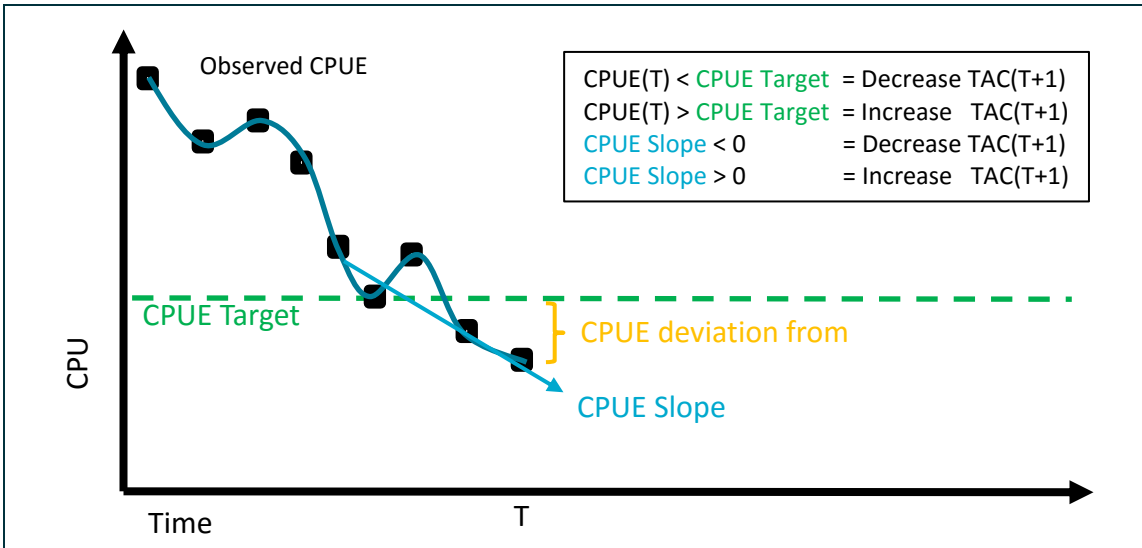


Figure 21. Cartoon representation of the IT MP category (aim for target CPUE). Implementation equations are included in the user manual.

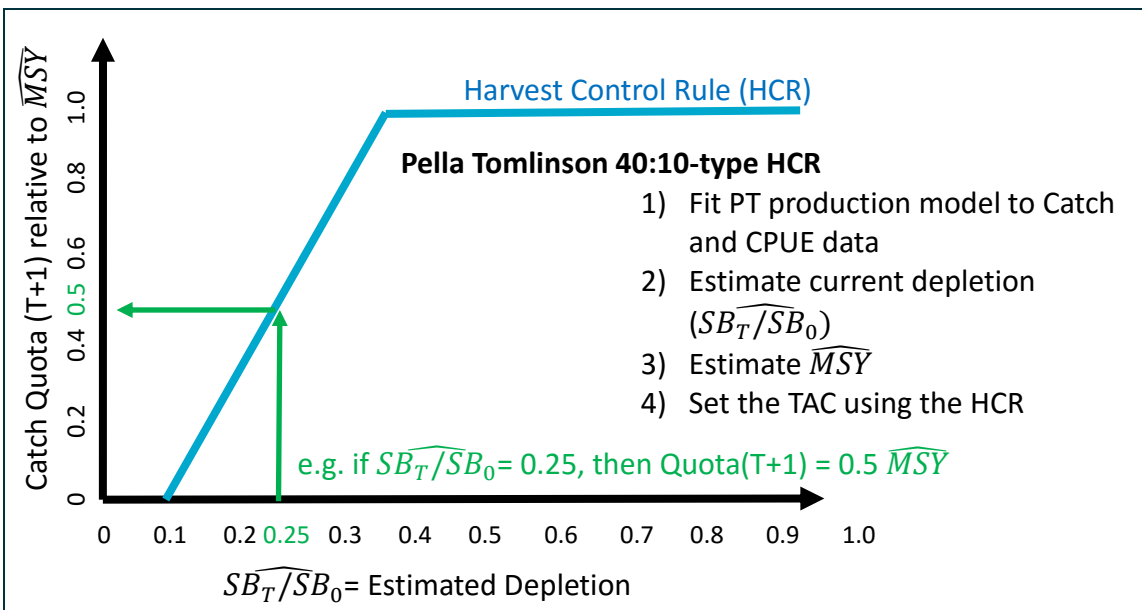


Figure 22. Cartoon representation of the PT 40:10-type MP category. Implementation equations are included in the user manual.

## 7 Performance Measures and Graphics

The MSE software reports a standard set of management performance measures as proposed by the IOTC Working Party on Methods, endorsed by the IOTC Scientific Committee (IOTC-SC 2015), and subsequently refined by the WPM informal sub-group on MSE (Table 7). The user manual provides a verbatim extract of the report section (in case of ambiguity). For one (non-spatial, single fishery) operating model, a full summary table comparing 10 MPs would include: 16 agreed statistics, 10 MPs, 4 summary time windows (1, 5, 10, 20 years) and 5 percentiles (10th, 25th, 50th, 75th, 90th) for a ponderous total of 3200 entries. While the tables will probably be useful when the MP options are whittled down, the important information can be more easily communicated with the agreed graphical outputs:

- Trade-off plots (Figure 26) show the relationship between two performance measures simultaneously (catch on one axis and stock status or catch variability on the other). The standard suite of plots has 4 panels for each of 2 time periods (10 and 20 years of projected MP management). In each plot, an individual MP is represented by a (possibly lop-sided) cross (i.e. a vertical line spans the 10th - 90th percentiles of the Y-axis performance measure, and a horizontal line spans the 10th - 90th percentiles in the X-axis performance measure, and the two intersect at the 50th percentiles). Many MPs can be conveniently overlaid. The plots consist of:
  - $SB/SB_{MSY}$  vs: Yield
    - performance measures 10 and 3 from Table 7
  - Pr (stock status in green Kobe phase plot quadrant) vs: Yield
    - 10 and 6 from Table 7
  - $SB > SB_{lim}$  vs: Yield
    - 10 and 9 from Table 7
  - $\text{mean}(1 - C_y/C_{y-1})$  vs: Yield
    - 10 and 14 from Table 7
- Confidence interval plots ("Skinny boxplots" or "Udon-Soba plots") (Figure 27) compare several MPs with respect to an individual performance measure. Only 3 of the 16 performance measures are shown in this plot, summarized over the 4 time windows. Each entry includes the median, 25th-75th percentiles as a thick line, and 10th-90th percentiles as a thin line.
- Time series plots show the distribution of historical and projected  $SB_y/SB_{MSY}$ ,  $F_y/F_{MSY}$ ,  $Catch_y$  and  $CPUE_y$  (Figure 28). The median is represented by a black line, a dark ribbon shades the 25th-75th percentile region and a light ribbon shades the 10th-90th percentile region. Horizontal lines indicate target (green) and limit (red) reference points. Three additional coloured lines show individual realizations (selected from the 25th, 50th and 75th percentiles of the biomass trajectories in the final year). These latter lines are included primarily to emphasize that individual trajectories are likely to be much more erratic than the summary percentiles. The standard time series plots also include

projected catch by fishery and CPUE by region (not shown). We include additional plots of total biomass and recruitment time series for quality control of the OMs (e.g. Figure 19).

**Table 7. Performance measures (the interim reference points from IOTC Resolution 15/10 differ by species:**

$SB_{target,YFT} = SB_{target,BET} = SB_{MSY}$ ;  $F_{target,YFT} = F_{target,BET} = F_{MSY}$ ;  $SB_{limit,YFT} = 0.4SB_{MSY}$ ;  $F_{limit,YFT} = 1.4F_{MSY}$ ;  $SB_{limit,BET} = 0.5SB_{MSY}$ ;  $F_{limit,BET} = 1.3F_{MSY}$ .

<b>Status : maximize stock status</b>		
1. Mean spawner biomass relative to pristine	$SB/SB_0$	Arithmetic mean over years
2. Minimum spawner biomass relative to pristine	$SB/SB_0$	Minimum over years
3. Mean spawner biomass relative to $SB_{MSY}$	$SB/SB_{MSY}$	Arithmetic mean over years
4. Mean fishing mortality relative to target	$F/F_{tar}$	Arithmetic mean over years
5. Mean fishing mortality relative to $F_{MSY}$	$F/F_{MSY}$	Arithmetic mean over years
6. Probability of being in Kobe green quadrant	$SB,F$	Proportion of years that $SB \geq SB_{tar}$ and $F \leq F_{tar}$
7. Probability of being in Kobe red quadrant	$SB,F$	Proportion of years that $SB < SB_{tar}$ and $F > F_{tar}$
<b>Safety : maximize the probability of remaining above low stock status (i.e. minimize risk)</b>		
8. Probability of spawner biomass being above 20% of $SB_0$	$SB$	Proportion of years that $SB > 0.2B_0$
9. Probability of spawner biomass being above $B_{Lim}$	$SB$	Proportion of years that $SB > SB_{limit}$
<b>Yield : maximize catches across regions and gears</b>		
10. Mean catch	$C$	Arithmetic mean over years
11. Mean catch by region and/or gear	$C$	Arithmetic mean over years
12. Mean catch relative to MSY	$C/MSY$	Arithmetic mean over years
<b>Abundance: maximize catch rates to enhance fishery profitability</b>		
13. Mean catch rates (by region and gear)		
(for fisheries with meaningful catch-effort relationship)	$I$	Arithmetic mean over years
<b>Stability: maximize stability in catches to reduce commercial uncertainty</b>		
14. Mean absolute proportional change in catch	$C$	Arithmetic mean over years of $\text{abs}(1-C_t/C_{t-1})$
15.* % Catch co-efficient of variation	$C$	100% X standard deviation catch over years / mean
16. Probability of shutdown	$C$	Proportion of years that $C < 0.1MSY$

\*the WPM recommended the variance in catch, which is scale-dependent.



## 8 Demonstration MSE results

The results presented in this section are selected to illustrate the general functionality of the MSE software, summary statistics and graphics. The first section provides comparisons between the R and C++ projection sub-routines (and SS). The following sections provide results for yellowfin and bigeye.

### 8.1 R and C++ projection performance comparison

Table 8 summarizes the time required to conduct various MSE analyses (times are approximate, as the comparisons were not necessarily a fair "all other things being equal" test). While the C++ projection sub-routine is multi-threaded and much more efficient than the R-based projection code, the iterative solution to the Baranov equations is a much more demanding calculation. Additionally, the C++ projection sub-routine uses many of the slow R calculations (simulating observations and applying MPs). Multi-threading was implemented for the MP calculations of the R code (i.e. MP calculations are farmed out across CPUs, while the actual projections are conducted serially on a single CPU). It appears that the computational overhead associated with setting up the R multi-threading results in a performance decrease, unless computationally-intensive MPs are being used. Furthermore, for reasons that are not clear, the parallel processing in R occasionally fails to assign cores properly and aborts the run. At this time, there is not a substantial time benefit to consider when choosing between the R and C++ code (though the C++ option has greater capacity for future expansion as discussed in the critique section below), and the current implementation of the multi-threaded R is not recommended.

We suggest that 100-200 stochastic replications would usually be sufficient for developers to understand and improve MP performance based on median performance measures, and at ~2-10 minutes per MP evaluation, this is readily achievable within an interactive workshop. However, we would recommend reporting ~2000 realizations for formal decision-making, particularly where 10th percentiles are involved. Timeframes of 45-105 minutes per MP would be cumbersome for an interactive workshop, but would enable several runs to be conducted overnight in a multi-day meeting (i.e. similar to the time taken to conduct assessment runs in a WPTT meeting). Trade-off plots comparing MP performance evaluations based on 216 and 2160 simulations (and 10th, 50th and 90th percentiles) show that they are very similar (Figure 23).

Table 9 compares MSY-related estimates for the R and C++ projection subroutines, with the estimates from the Stock Synthesis model ensembles in OMyft1 and OMBet1. The R and C++ implementations are identical to ~3 significant figures. Since they were implemented largely independently, this gives us confidence that they are consistent with each other and the design specifications. Furthermore, the R and C++ MSE projections based on TAC = MSY were confirmed to be consistent with the MSY calculations. This appears to be the case for the SS calculations as well (though this was difficult to confirm because there is a 100 year projection limit in the SS forecast software, which is only 25 years when quarters are defined as years, and does not necessarily allow the population time to equilibrate).

The R and C++ based  $B_0$  and  $SB_0$  values were observed to be within 1% of the SS estimates. The R and C++  $SB_{MSY}$  and  $MSY$  values were within 1.5% of SS for bigeye, however they deviated by up to 6% for yellowfin. The discrepancy in the yellowfin case arises because SS uses a simplified approximation for projections based on spatially aggregated  $F$ , which the OMs were not designed to reproduce. There is a further complication in that SS does not report the appropriately disaggregated  $F$  values to use in the definition of "recent" effective fishing effort, so the aggregate values are used by default at this time. This introduces an inconsistency with SS assessments for  $MSY$ -based reference points, and the relative distribution of effort for effort-controlled MPs. The difference in effort distribution is problematic for effort-controlled MPs, however it is irrelevant for the dynamics of the quota-regulated MPs that are emphasized here (quota-based management has been the emphasis of IOTC albacore and skipjack MSE initiatives to date as well).

Some inconsistencies between R and C++ projection sub-routines are evident in the performance trade-off plots for yellowfin (Figure 24), but are negligible for bigeye (Figure 25). The discrepancies are largest when the yellowfin fishing mortality is very high (i.e. when the TAC cannot be achieved, as in the 400K t constant catch MP and lower 10th percentile of yield for the 300K t MP). In the tests conducted, the discrepancy is small for bigeye, even when fishing mortality is high and the TAC cannot be extracted. The problem in the yellowfin model appears to arise when some regions and age-classes are extremely depleted, though other regions and age-classes may not be. In the problematic yellowfin simulations, the R-based catch equation approximation is closer to the Baranov implementation when the TAC is extracted at the beginning of the time-step rather than midway (Pope's approximation), so this is probably the better option to adopt.

The choice between the R and C++ projection sub-routines could potentially be sufficiently large to influence the process of yellowfin MP selection. The argument can be made that the C++ code is more consistent with SS. However, the more important question may be whether either implementation is likely to provide a realistic simulation of the dynamics under those circumstances. If the fishery is depleted to the extent that quotas cannot be achieved (and numerical constraints on very high mortality are relevant), then the economic viability of some fisheries is probably doubtful. Some fleets would probably stop fishing, target different species or move to different fishing grounds, all of which would undermine both sets of projection assumptions.

**Table 8. Observed computation time for selected MP evaluations, using a quad core Intel i7 processor on a laptop PC. The stochastic projections are based on 26 years of projections, with the indicated MPs applied every 3 years (R and C++ projections for MP PT41.100.2 both use R-based fitting of a surplus production model). Times are approximate only, as background processes and memory management were not consistent among all trials.**

MP evaluation scenario	R Projections 1 core	R Projections 4 cores	C++ Projections 4 cores
Minutes			
4 area YFT OM: 216 reps, MP: CC100	6	9	5
4 area YFT OM: 216 reps, MP: PT41.100.2	9	13	9
4 area YFT OM: 2160 reps, MP: CC100	62	67	55
4 area YFT OM: 2160 reps, MP: PT41.100.2	78	105	76
1 area BET OM: 216 reps, MP: CC100	2.4	3.7	3.2
1 area BET OM: 216 reps, MP: PT41.100.2	5.2	5.9	4.3
1 area BET OM: 2160 reps, MP: CC100	25	42	35
1 area BET OM: 2160 reps, MP: PT41.100.2	54	47	43

**Table 9. Comparison of OM and SS MSY calculations (note that OMyft1 was used in testing, but was superseded by OMyft2, which is presented in the results)**

MSY calculations	R	C++	Stock Synthesis
Mean (range) 1000 t			
OMyft1 $SB_0$	385 (303-502)	385 (302-502)	385 (303-501)
OMyft1 $SB_{MSY}$	130 (96.4-178)	130 (96.5-179)	132 (95.8-183)
OMyft1 MSY	392 (323-474)	392 (323-474)	375 (308-459)
OMbet1 $SB_0$	165 (123-211)	165 (123-211)	165 (123-211)
OMbet1 $SB_{MSY}$	470 (300-682)	479 (304-673)	467 (301-670)
OMbet1 MSY	137 (97.5-203)	137 (97.5-203)	139 (98.7-206)

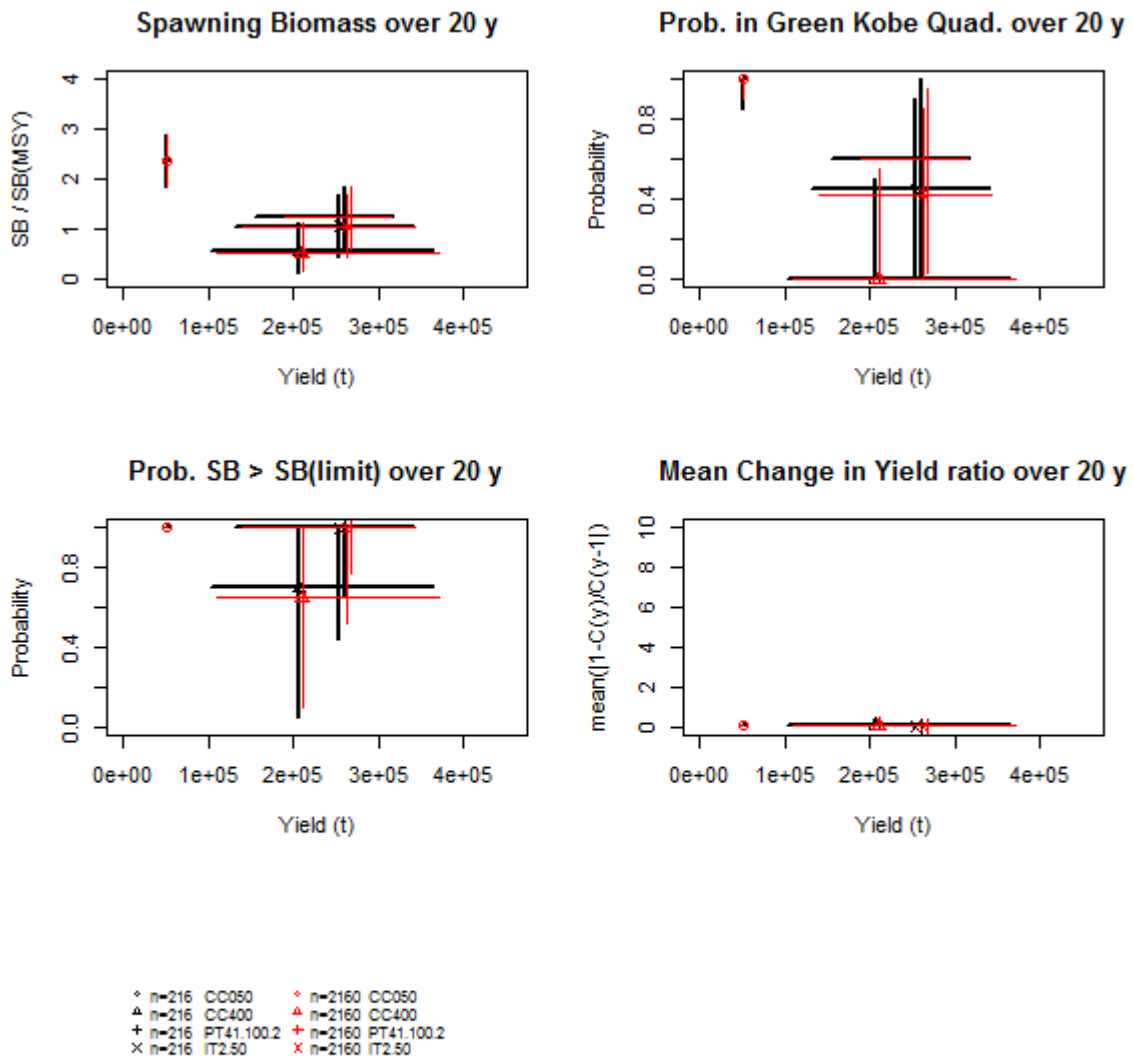
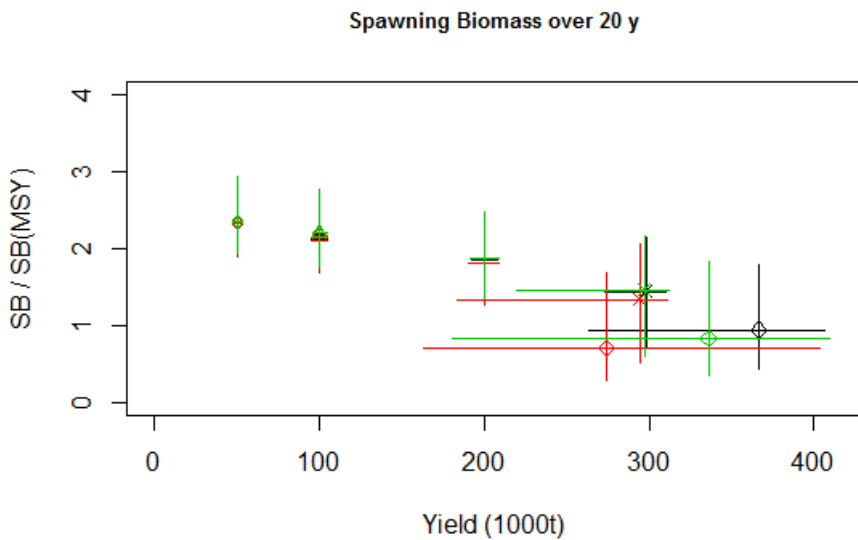
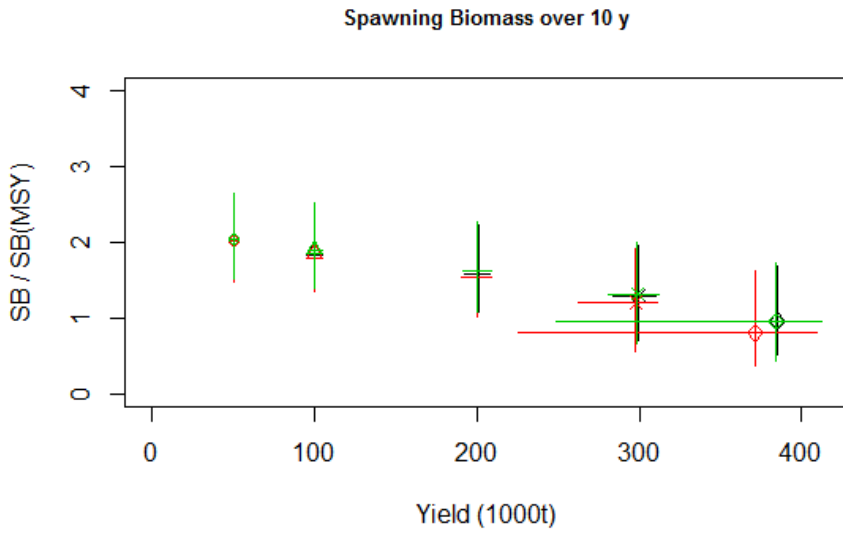
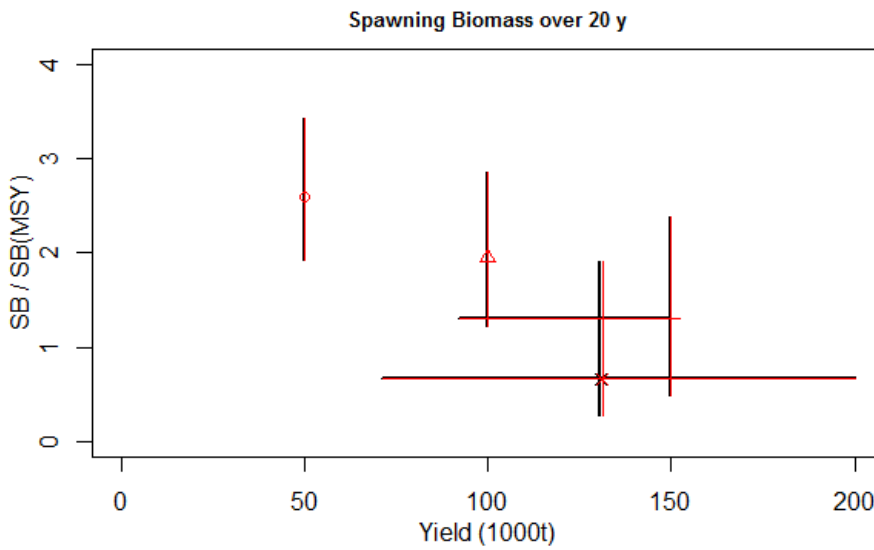
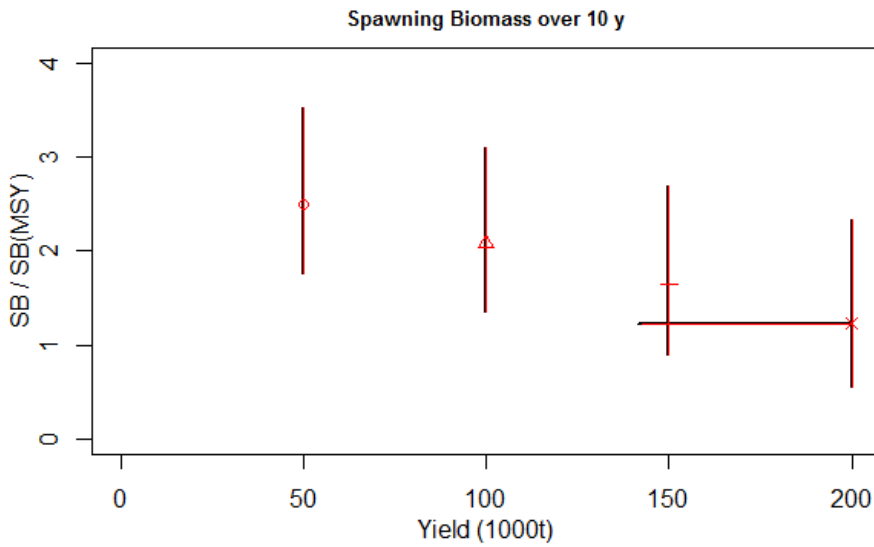


Figure 23. Trade-off plots comparing 4 yellowfin MPs evaluated using 216 and 2160 stochastic realizations with operating model OMyft1 (OMyft1 was subsequently superseded by OMyft2). Plots are explained in text section 7.



- ◇ C++ CC050   ◇ R50 CC050   ◇ R01 CC050
- △ C++ CC100   △ R50 CC100   △ R01 CC100
- + C++ CC200   + R50 CC200   + R01 CC200
- × C++ CC300   × R50 CC300   × R01 CC300
- ◇ C++ CC400   ◇ R50 CC400   ◇ R01 CC400

**Figure 24.** Trade-off plots comparing key performance measures for 5 constant catch yellowfin MPs evaluated using the C++ (Baranov) and R (instantaneous TAC extraction approximation) projection subroutines. R50 = TAC extracted mid-way through the time-step, R01 = TAC extracted at beginning of time-step. The key point is that results diverge appreciably only when fishing mortality is so high that the full TAC cannot be extracted ( $\geq 300K$  t). Plots are explained in text section 7.



- ◇ R50 CC050    ◇ C++ CC050
- △ R50 CC100    △ C++ CC100
- + R50 CC150    + C++ CC150
- × R50 CC200    × C++ CC200

Figure 25. Trade-off plots comparing key performance measures for 4 bigeye MPs evaluated using the R and C++ projection subroutines. Note that performance measures are almost identical, and both sets of results are visible only because the line widths differ. Plots are explained in text section 7.

## 8.2 Demonstration yellowfin MSE results

Figure 26 illustrates the standard trade-off plots (defined in section 7) comparing 10 MPs evaluated with OMyft2 (R-based projections). The 10 year summaries suggest a well-defined "frontier" illustrating that higher catch levels are generally associated with lower biomass and higher risk of violating biomass reference points. However the trade-off plots illustrate some results that may not be intuitive. For examples, the MP CC400 (400K t constant catch) only results in a small proportion of catches attaining 400K t, even though more than 50% of the OMyft2 ensemble has an  $MSY > 400K$  t. Furthermore MP CC300 (300 K t) has higher median catch than CC400 over a 20 year period. This occurs primarily because the population is depleted below  $B_{MSY}$  at the start of the projections, and tends to become more severely depleted with higher catches (though more that 10% of the CC400 results are able to sustain the full catch for the 20 year period). Additionally, the population may be locally depleted in some regions, such that the corresponding fisheries may be unable to reach their individual quotas even if the overall population could sustain those levels. The time-dependent rebuilding considerations are not easily visualized with this type of plot.

The boxplot summaries in Figure 27 provide similar information, though the inclusion of 1, 5, 10 and 20 year time horizon summaries emphasizes how large catches in the short term represent a greater risk. Lower initial catches may result in similar catches in the longer term with lower overall risk levels.

Figure 28 -Figure 31 illustrate the standard time series plots for some selected MPs, from which we note:

- MP CC400 illustrates that constant catch near the level of recent historical levels probably cannot be sustained for long. However, the overall population may not be driven to collapse very quickly because the spatial partitioning (combined with selectivity) protects some component of the population. The migration dynamics are not well enough understood to conclude whether this is a plausible extrapolation.
- MP PT41.100.2 tends to keep the median biomass near or above  $SB_{target}$ , and <10% of the realizations exceeded  $SB_{lim}$ . This seems reasonably consistent with the Commission objectives. The initial Pella-Tomlinson model estimates recognize the need to reduce quotas. However, the magnitude of the quota reduction is limited by the 20% change constraints. MP PT41.100.9 uses the same HCR as MP PT41.100.2, except that the 90% change constraint is not very restrictive. Accordingly, there is a more rapid drop in quota, which probably would not be very palatable to the fishing industry.
- MP IT2.50 demonstrates that the simple CPUE-based decision rule can achieve results that are very similar in character to the Pella Tomlinson model, perhaps better in the sense that median catch appears to be very stable over the 20 year horizon.

We conclude from these plots that:

- The empirical and model-based MPs both appear to be able to span the management performance trade-off space with the appropriate selection of control parameters. Performance can presumably be improved with more effort spent targeting a specific management objective, i.e. "tuning" can be used to find MPs that have identical

performance with respect to the highest priority management objective, and these can then be ranked with respect to secondary objectives).

- If OMyft2 is a reasonable representation of the dynamics, the MPs that appear to meet the Commission objectives will probably result in considerably lower mean catches than recent historical levels.
- Stock rebuilding may play an important role in the yellowfin MP performance. The short-term vs: long-term catch trade-off is probably most easily communicated with the time series plots.



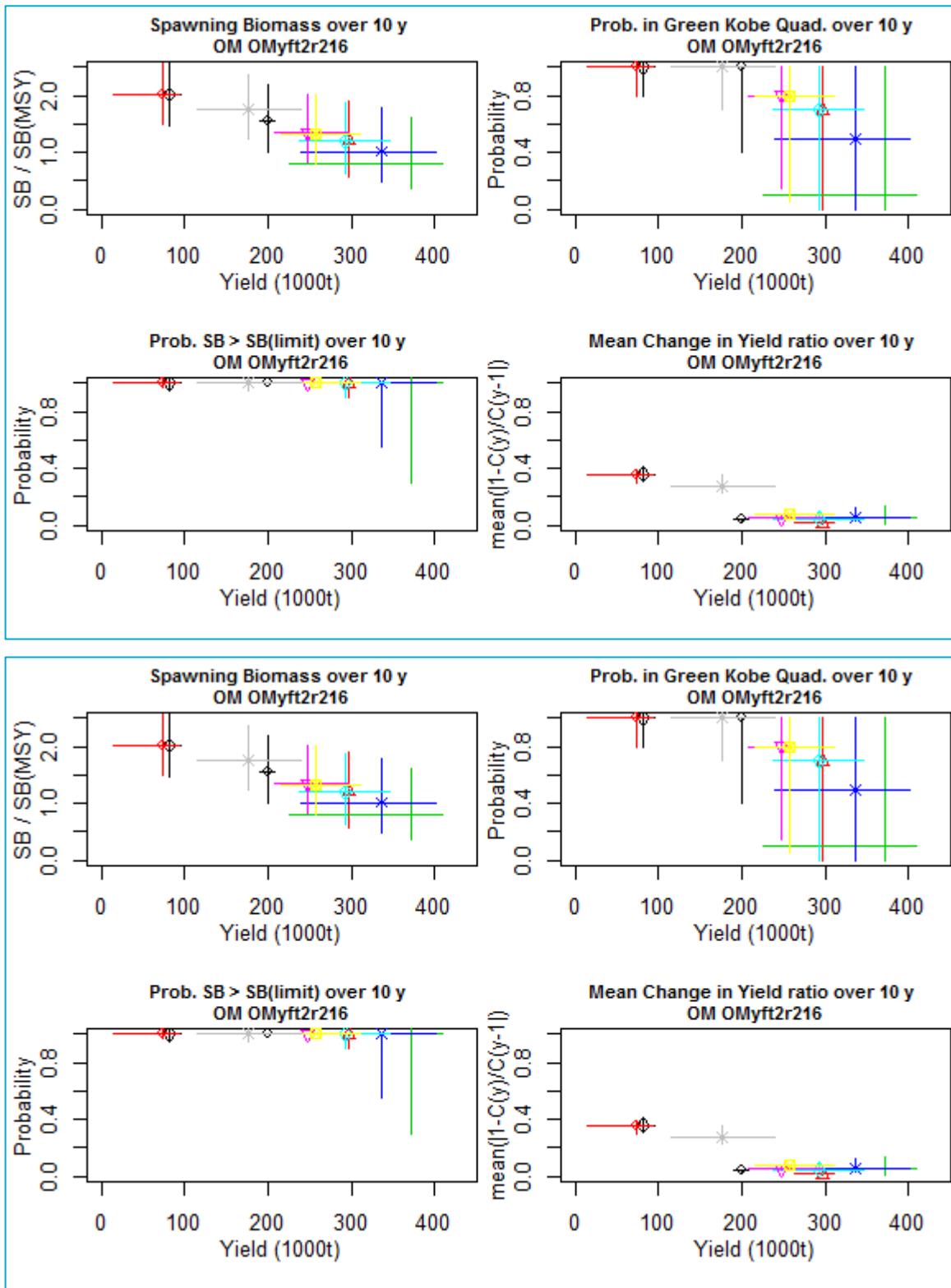


Figure 26. Trade-off plots comparing performance of 10 MPs evaluated against yellowfin operating model OMyft2 over projection periods of 10 years (top 4 panels) and 20 years (bottom 4 panels). Plots are explained in text section 7.

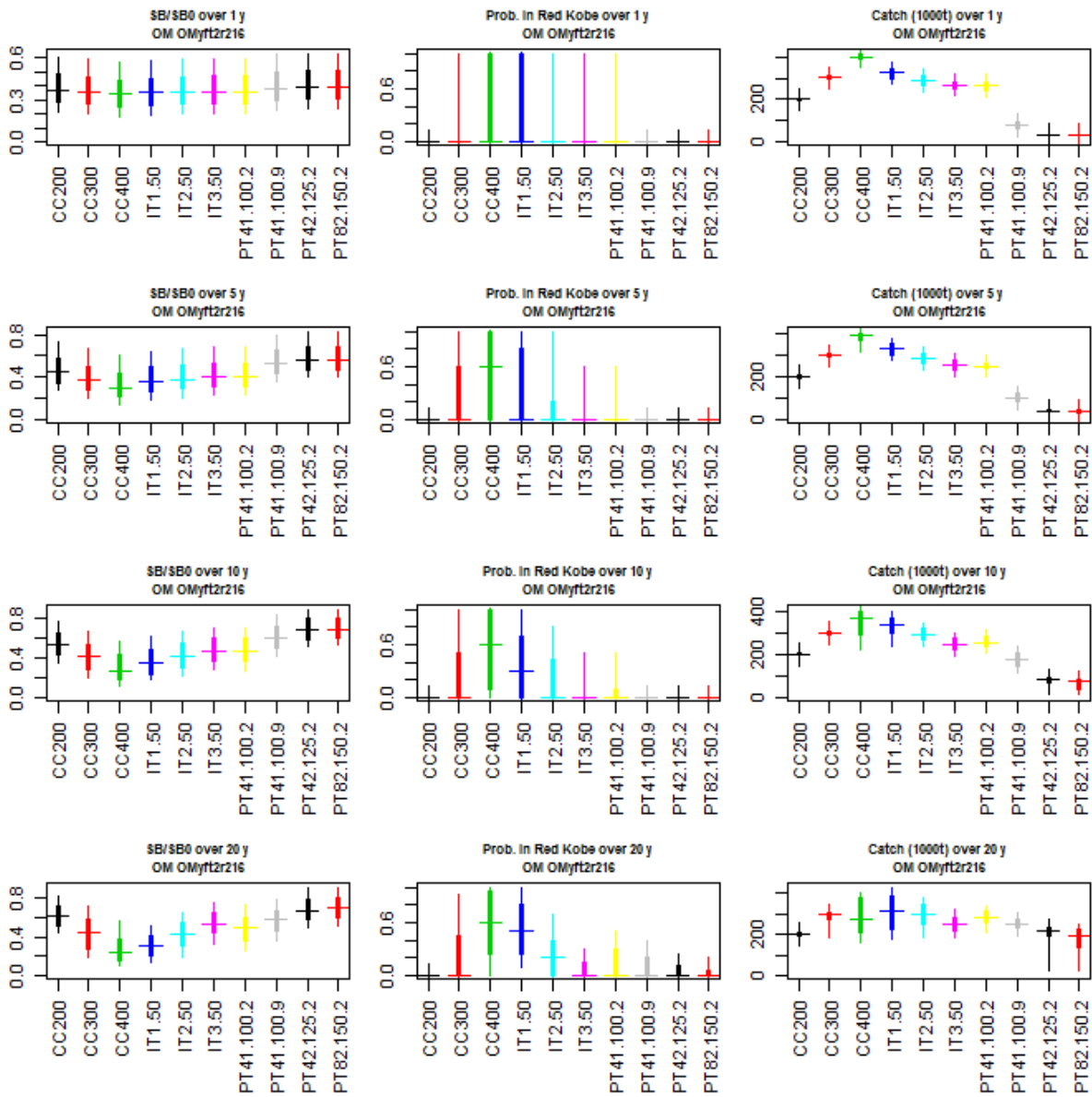


Figure 27. Boxplots (Udon-Soba) for 3 performance indicators for 10 MPs evaluated against yellowfin operating model OMMyft2r216, summarized over time periods of 1,5,10 and 20 years). Plots are explained in text section 7.

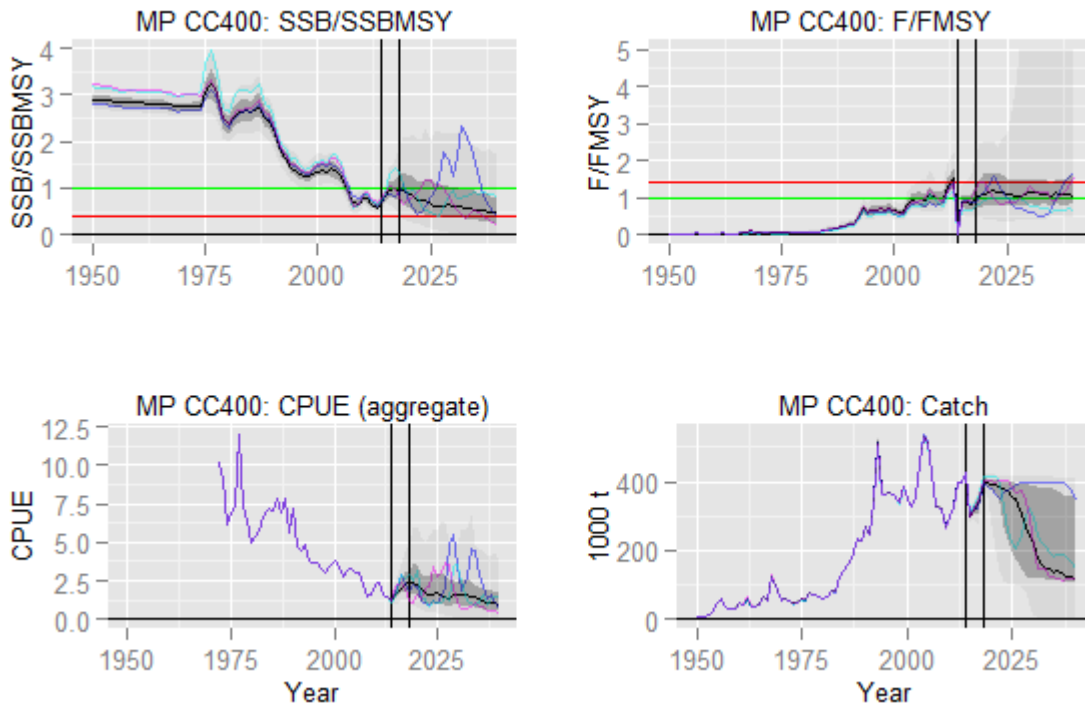


Figure 28. Time series (worm) plots for MP CC400 evaluated against yellowfin operating model OMyft2.

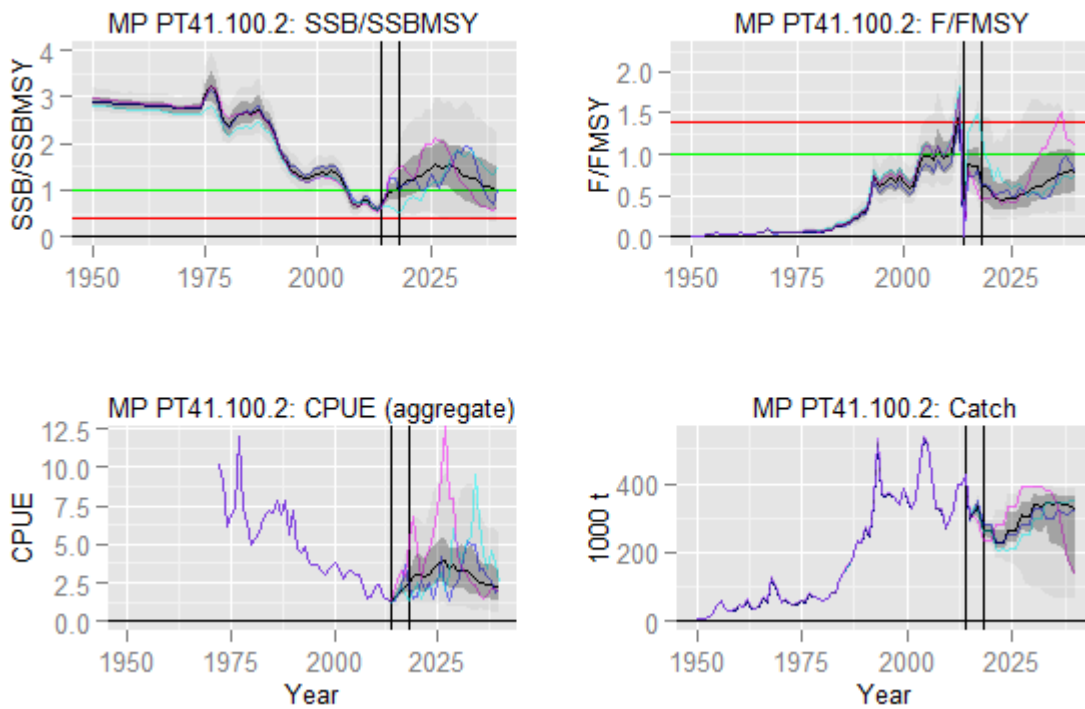


Figure 29. Time series (worm) plots for MP PT41.100.2 evaluated against yellowfin operating model OMyft2. Plots are explained in text section 7.

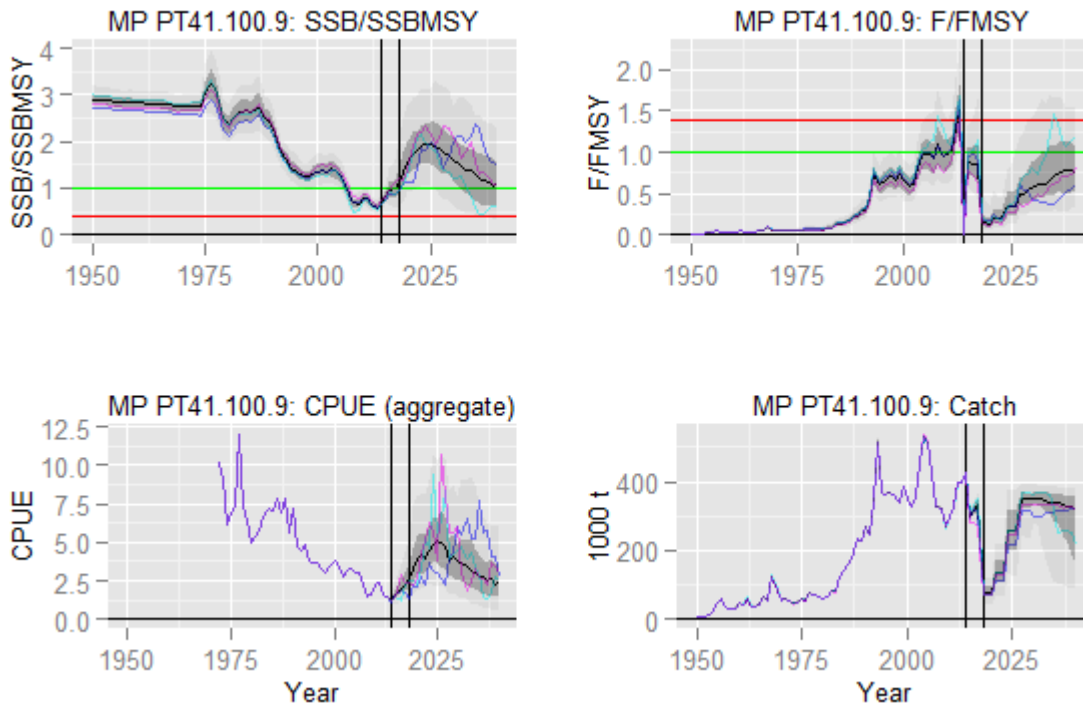


Figure 30. Yellowfin time series plots for MP PT41.100.9 evaluated against operating model OMyft2.

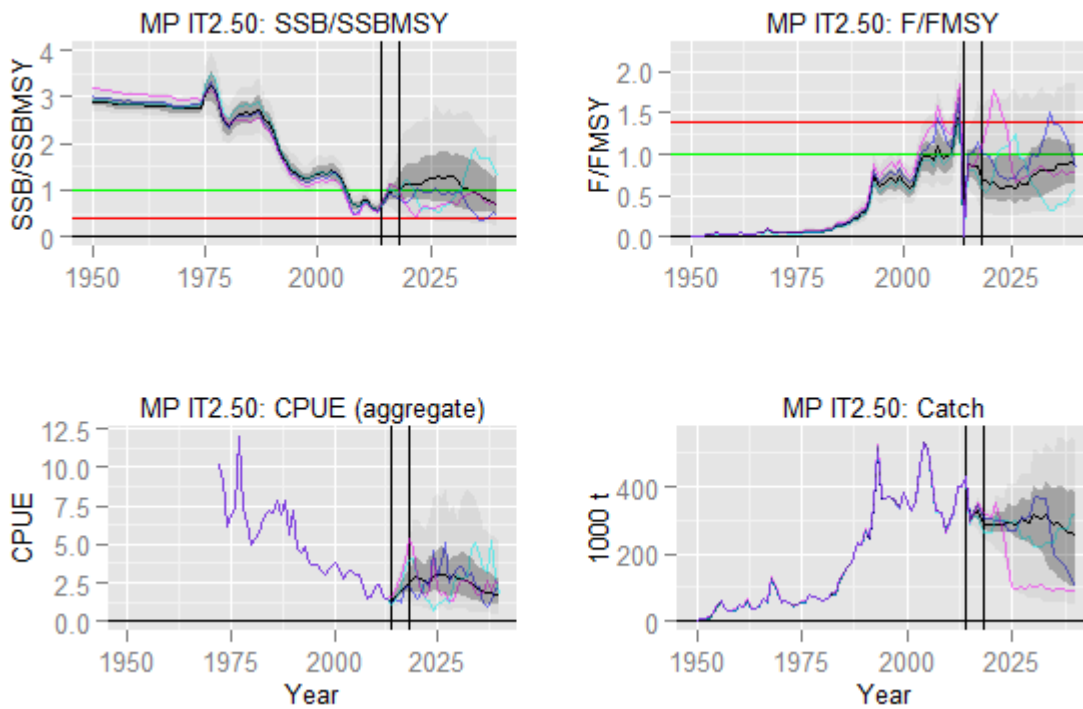
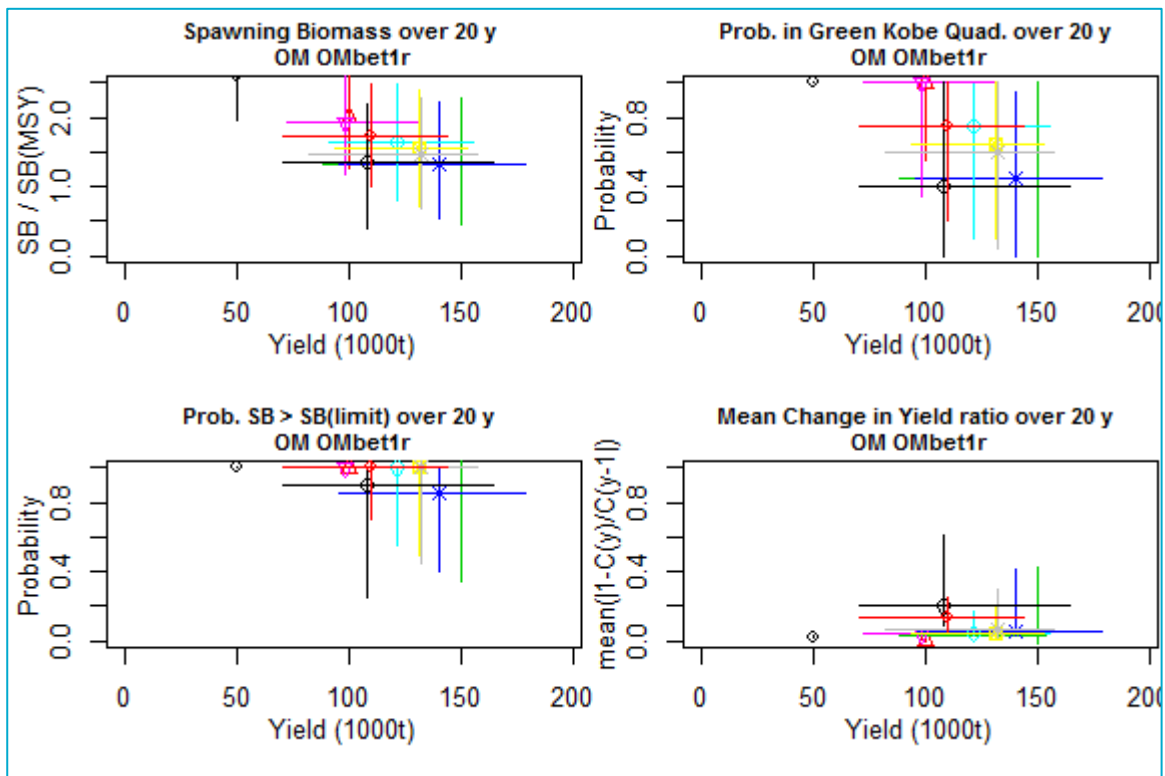
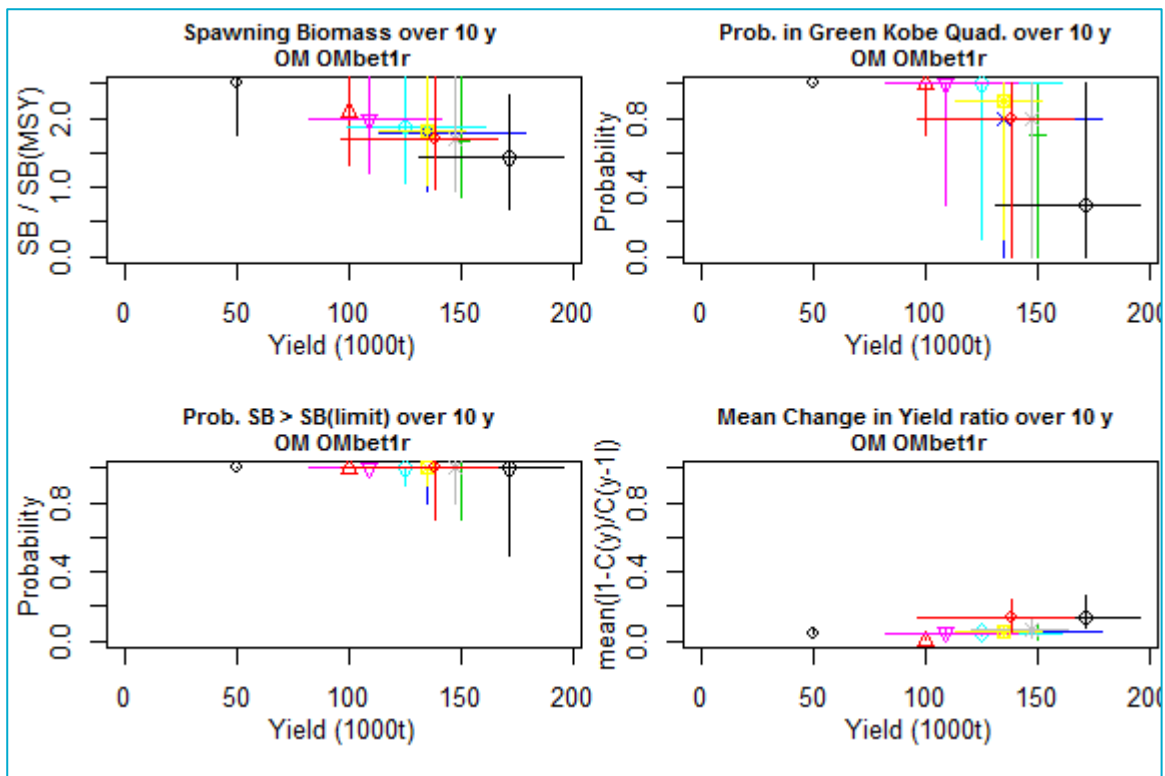


Figure 31. Yellowfin time series plots for MP IT3.5 evaluated against yellowfin operating model OMyft2r. Plots are explained in text section 7.

### 8.3 Demonstration bigeye MSE results

Figure 32 illustrates the standard trade-off plots (for 10 and 20 year projections) for bigeye (R-based projections). The corresponding boxplot summaries are in Figure 33, and selected time series plots are shown in Figure 34-Figure 37. The bigeye MPs appear to have simpler and more optimistic behaviour than the yellowfin MPs, because the stock is estimated to be much less depleted (substantially above  $B_{MSY}$ ). The suite of MPs shown presumably spans the Commission objectives.

In these examples, it is not obvious that the highest constant catch MP (CC150 = 150 000 t) is very problematic in the short-medium term. These catch levels are near the highest ever recorded, and near the median MSY of the OMBet1 ensemble. Since it takes about 20 years before the median  $SB$  declines to  $SB_{lim}$  in this MP (Figure 34), it suggests that feedback-based MPs do not necessarily offer better performance in the medium time horizon. More aggressive MPs, or a longer time horizon would be required to attain a median of  $SB_{target}$  and to make the advantage of feedback-based management clear.



- ◊ CC050    + CC150    ◇ IT2.50    ◻ PT41.100.2    ⊕ PT42.125.2
- △ CC100    × IT1.50    ▽ IT3.50    \* PT41.100.9    ⊕ PT82.150.2

Figure 32. Trade-off plots comparing performance of 10 MPs evaluated against bigeye operating model OMbet1 over projection periods of 10 years (top 4 panels) and 20 years (bottom 4 panels). Plots are explained in text section 7.

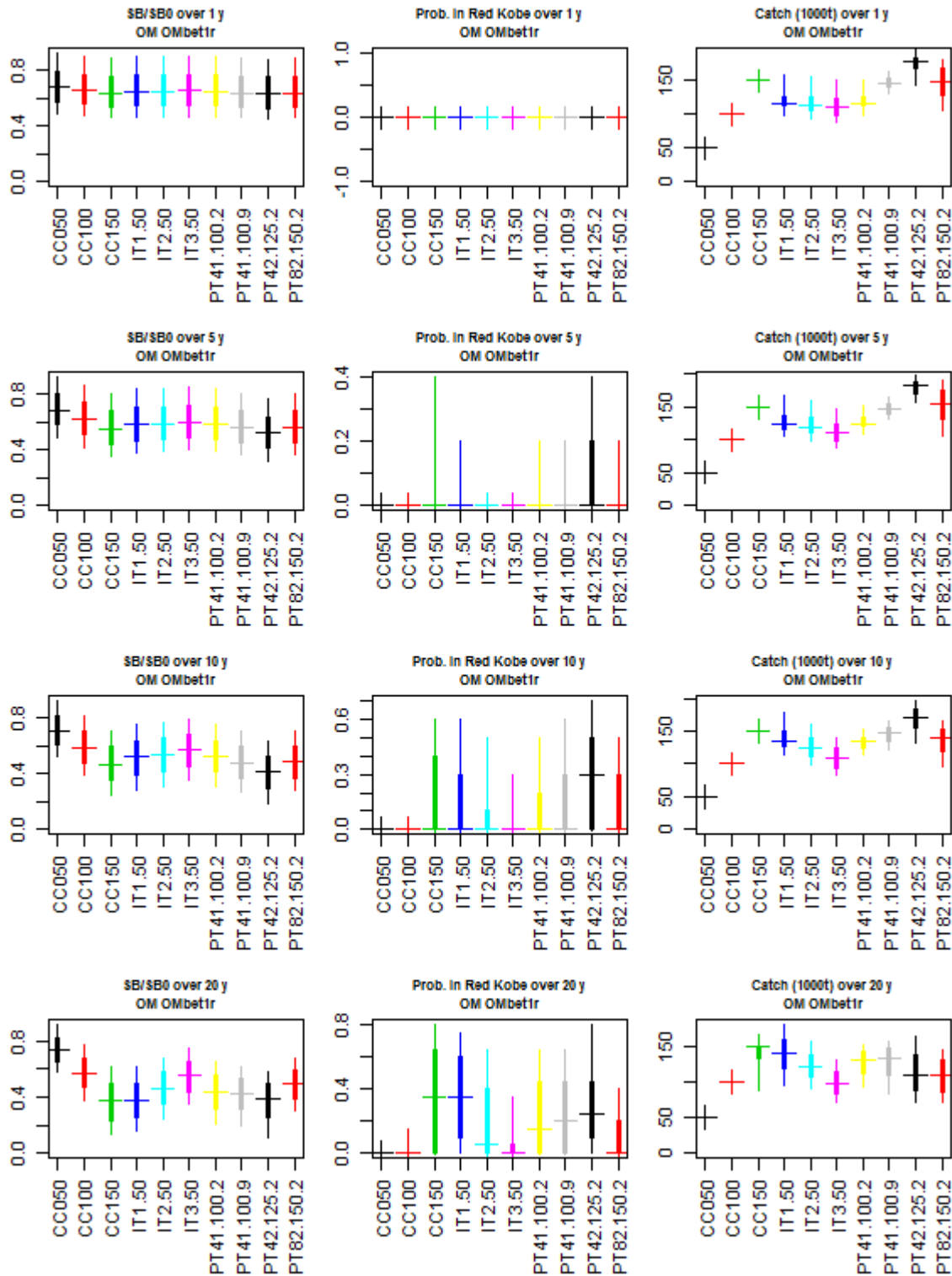


Figure 33. Boxplots (Udon-Soba) for 3 performance indicators for 10 MPs evaluated against bigeye operating model OMBet1, summarized over time periods of 1,5,10 and 20 years). Plots are explained in text section 7.

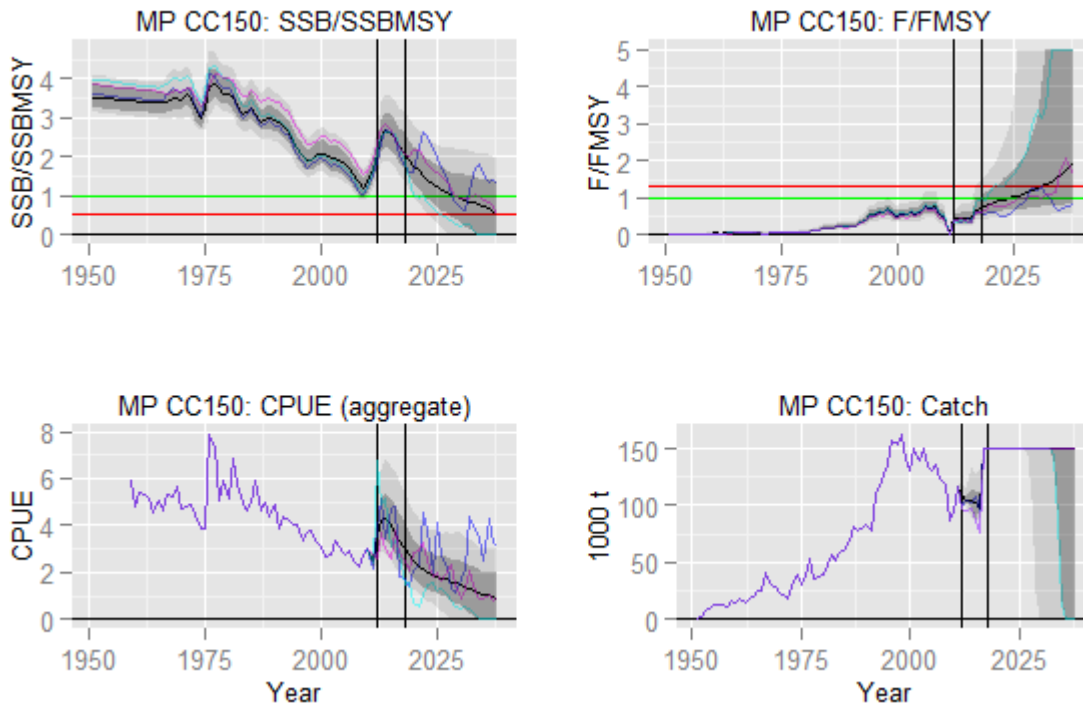


Figure 34. Time series (worm) plots for MP CC150 evaluated against bigeye operating model OMBet1.

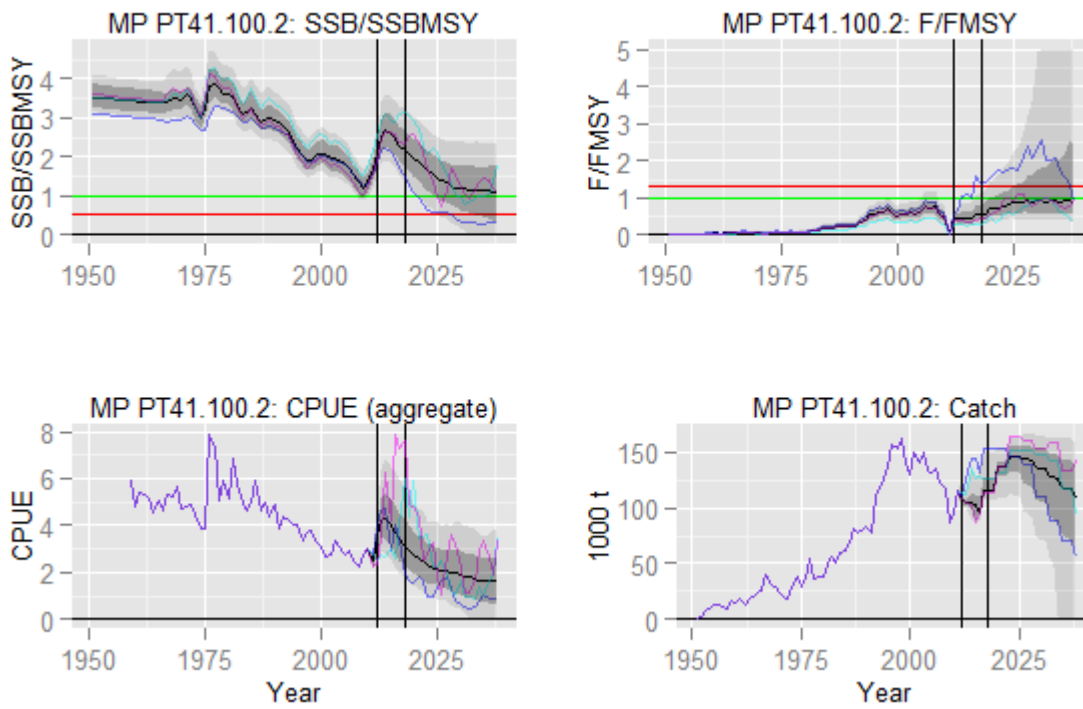


Figure 35. Time series (worm) plots for MP PT41.100.2 evaluated against bigeye operating model OMBet1. Plots are explained in text section 7.



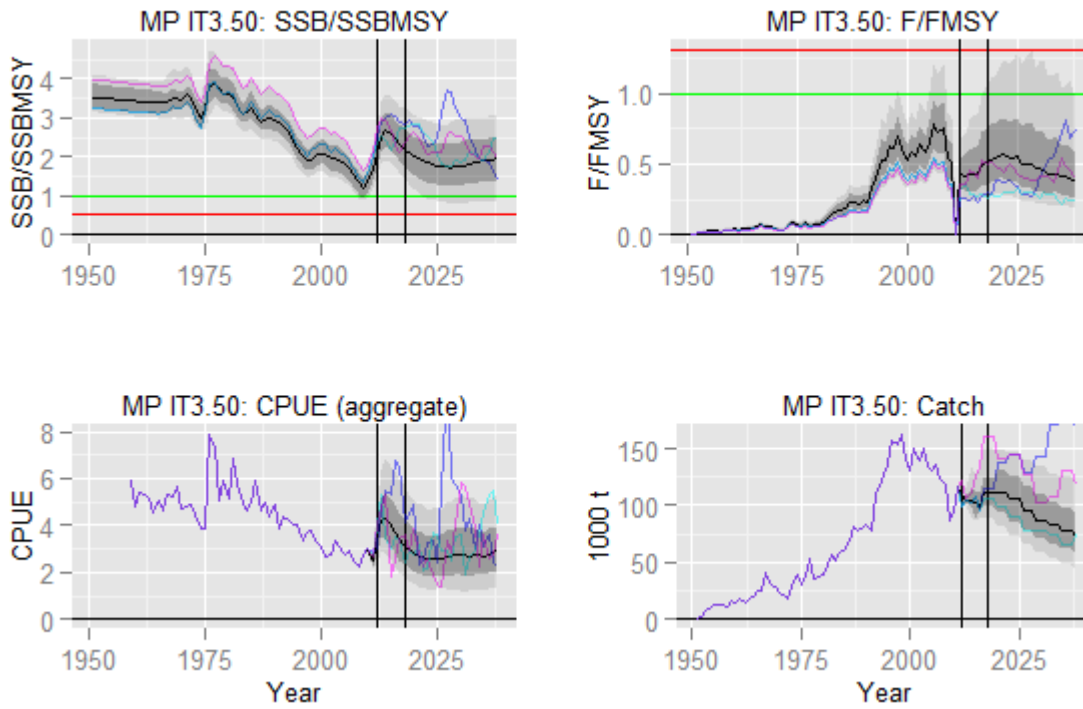


Figure 36. Time series (worm) plots for MP IT3.50 evaluated against bigeye operating model OMBet1. Plots are explained in text section 7.

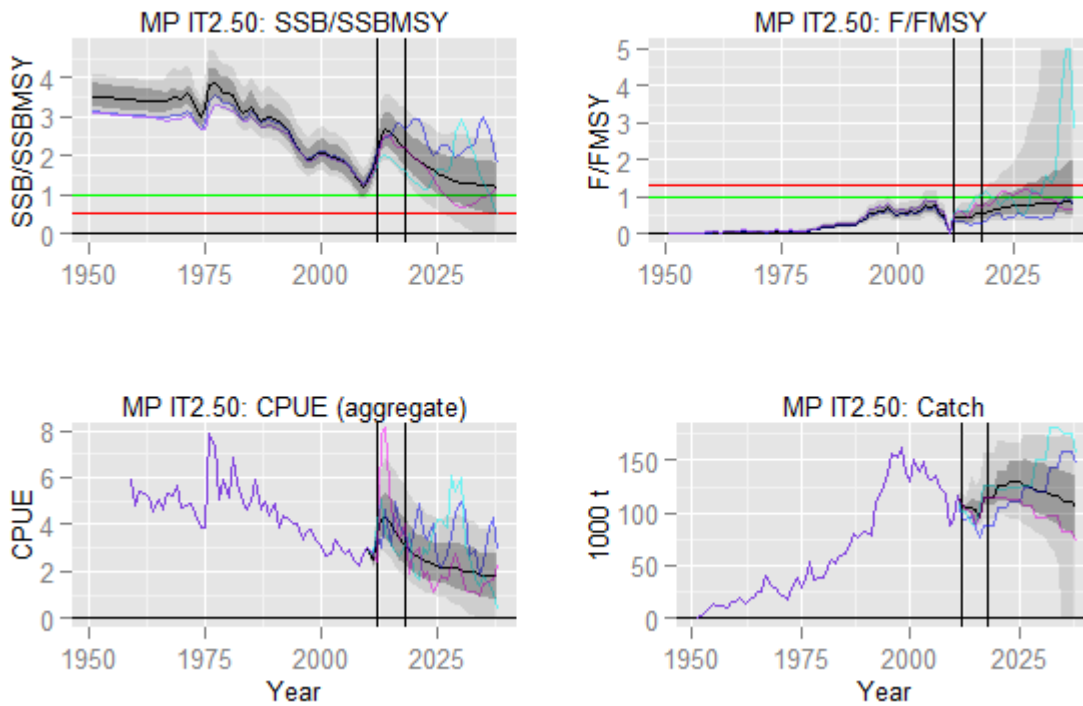


Figure 37. Time series (worm) plots for MP IT2.50 evaluated against bigeye operating model OMBet1. Plots are explained in text section 7.

# 9 Critique of the Indian Ocean Yellowfin and Bigeye MSE software, demonstration case Oms and candidate MPs

The following list is a compilation of concerns from the phase 1 project. It highlights decisions that should be reviewed, identifies limits to what the MSE can deliver, and suggests priorities for future work.

## 9.1 Projection software

- 1) The R-based and C++ based projection sub-routines use different implementations for the catch equations, and a decision will ultimately have to be made to select the MP on the basis of one or the other. The two implementations provide very similar MP evaluation results except when fishing mortality rates are very high in the yellowfin spatial model. Ideally this would not be an issue if reasonably conservative MPs were being tested. However, the yellowfin population is estimated to be substantially depleted now (especially the North-East) and the choice of sub-routine may have some influence on MP selection. It can be argued that the C++ code is more consistent with the SS conditioning model, and would be preferable on that basis. However, we would question whether either approach is very realistic when fishing mortality rates are so high that quotas cannot be met, catch rates are uneconomical, and incentives exist for fleets to stop fishing, move among areas and/or switch species targeting.
- 2) The projection software is slower than desired, but should be adequate for the purpose. Options exist for improving the speed. More of the MSE code could be implemented in C++ (including the simulated observation processes and any computationally demanding MP calculations), or the R code could be parallelized more effectively. Implementing the R-based catch equation approximation in C++ would be an option for dramatically improving speed. However, speed is probably not an urgent problem unless/until substantially more complicated OMs or MPs are required (e.g. multi-stock, multi-species or MPs based on fitting complicated assessment models).
- 3) The OM (and MPs) assume that an aggregate relative abundance index will be available for the stock, derived from a combination of CPUE series with exactly one from each area. If CPUE observations cease to be available from some regions, or additional series are added, this will require code changes. Additionally, the simulated error characteristics for the index are applied to the aggregate index at this time (i.e. to simplify considerations about how independent the error structure is by season and area). The MPs are provided with a small number of imputed CPUE observations from recent years.
- 4) Independent reviews for the Indian Ocean skipjack and albacore MSEs suggested that alternative stock-recruit functions (e.g. Ricker) should be considered. We are not aware of evidence from bigeye or yellowfin tuna population to suggest that recruitment initially increased with declining *SB* (as predicted by a Ricker function), or that there is a mechanism that would lead one to expect a Ricker function (e.g. high rates of cannibalism, redd

superimposition), so we have not considered this to be a priority. The Ricker function has been retained from the original ABT code, but has not been used in OM conditioning or tested in the projections.

- 5) Stationarity assumptions - there is a long list of stationary assumptions in most stock assessment models that are required to produce tractable estimators. Non-stationarity in  $M$ , growth and recruitment processes could have important implications for production dynamics. Non-stationarity in recruitment distribution and/or movement dynamics might introduce the need to reconsider fleet distribution dynamics.
- 6) We have included options for projecting non-stationary fishing selectivity (using parameterizations linked to long period sine wave oscillations). This is intended to introduce temporally-structured noise to the size composition distributions to prevent these data from being unrealistically informative for the MP (but affect CPUE as well). However, the parameterization was arbitrarily conceived from qualitative arguments and it would not even make sense to attempt to estimate the parameters.
- 7) MSY-based reference points depend on fishing selectivity and biology and are not stationary if the biology changes or selectivity changes (including a change in the relative fishing mortality among fisheries with stationary individual selectivity, as would be observed with fishery-specific quotas allocated with constant relative catches). In this case, the reported reference points are fixed on the basis of the "recent" effective effort distributions (i.e. estimated fishing mortality). We would suggest that the approach of using "proxy" reference points based on depletion would be preferable for bigeye and yellowfin (as adopted for skipjack).
- 8) There is a point of divergence between the bigeye/yellowfin MSE and the albacore and skipjack MSE related to the treatment of historical data (as discussed in the April 2016, Tokyo, WPM informal MSE workshop). We have adopted the perspective that the MPs should only have access to the historical data, regardless of which OM scenario is being projected forward. The alternative approach (used for albacore and skipjack, at least in April 2016) of simulating scenario-specific historical data for the MP evaluation amounts to providing extra insight to the MP that is not available in the real world, and hence the MP might perform better in the simulations than it would in the real world. Whether or not the difference is important is probably very case-specific. But one obvious argument for not simulating historical data relates to the communication of the first MP recommendation, i.e. since there is only one set of historical data, there can only be one recommendation. The first recommendation can be very important for industry cooperation in the MSE process, particularly in an overfishing/rebuilding situation where large catch reductions might be justifiable.
- 9) There is reason to think that Indian Ocean yellowfin and bigeye population connectivity and stock structure is more complicated than the homogenous mixing assumed in most assessments to date. With this in mind, the projection code partially supports the option of multiple stocks with independent biology, but the implementation has not been tested, and stock-specific population diagnostics are not reported at this time.
- 10) Ignoring the multi-species technical interactions in tropical tuna fisheries may limit the utility of MP evaluations (or at least increases the importance of management implementation errors). The projection code partially maintains the ABT feature of multiple

independent populations within a species (though it remains untested), and it should be reasonably straightforward to extend this feature to a multi-species context. However it is not a trivial modification, and conditioning would require the data for the different species to be supplied with compatible fishery definitions. Representing the joint uncertainty of multiple species could presumably be easily achieved by independently sampling the species-specific OMs.

- 11) Tag dynamics have not been included in the projection model. It may prove desirable in the future to simulate the collection and analysis of conventional or genetics-based mark-recapture methods to evaluate MPs based on fisheries-independent monitoring. Ideally this would be achieved by simulating tagging programmes within the MSE framework, and developing MPs that used simple tag-based indicators (as opposed to full statistical tag-based estimators that might be too computationally demanding to simulation test).

## 9.2 OM Parameterization process

- 1) We opted to weight all model specifications within the demonstration OMs equally at this time. Our rationale was based on the observation that none of the models showed egregious problems (in terms of fitting to the data or radical outlier behaviour), and the expectation that heavy-tailed OMs are probably going to require a more robust MP than centrally-weighted OMs. We would expect that other IOTC scientists have relevant insight about specific SS model formulations to add, remove or differentially weight in the OMs.
- 2) Ignoring parameter estimation uncertainty (for those parameters not included in the OM grid) might provide an understatement of some key uncertainties (see point 1). We have attempted to avoid understatement of uncertainty by using the grid-based approach, and adding a number of projection options, including user-defined parameters for some process and observation errors, (CV and auto-correlation), non-stationary selectivity, and error on the initial numbers-at-age. However, we recognize that this approach is subjective and other IOTC scientists may have good arguments for alternative assumptions.
- 3) The IWC and CCSBT MP processes found value in distinguishing between "reference" and "robustness" OMs (e.g. Punt et al 2014). The reference OMs were intended to provide a general description of the fishery with a reasonable description of uncertainty evident from the historical data. In contrast, robustness OMs comprised a series of OMs that were considered to have a low probability, but potentially very negative consequences (e.g. sustained recruitment failure, large IUU catches). We did not define any robustness scenarios for bigeye or yellowfin, but these may be worth considering.
- 4) Depending on how the Commission decides to manage the tropical tuna populations, it may be necessary to reconsider the fishery definitions used in the assessment and MSE. The MSE results currently do not provide nation-specific results (though the IOTC secretariat would have the information required to decompose the fisheries post-hoc if required). At this time bigeye, yellowfin and skipjack fisheries are defined differently, which may confuse communication and management decisions in the short-term, and makes multi-species MSE difficult to implement in the longer term.

## 9.3 Yellowfin demonstration Operating Model OMyft2

- 1) The Langley (2015) reference case assessment stock status reference points are near the central tendency of the OMyft2 ensemble of models. This is despite several formulation differences:
  - A biomass bug in the assessment SS version 3.24F (resolved in 3.24Y used here)
  - OMyft2 does not use environmental co-variates for movement dynamics
  - OMyft2 includes  $M$  and CPUE catchability options which are on the pessimistic side of the reference case, while down-weighting the tag data tends to be more optimistic than the reference case.

However, comparing the current catch projections in Figure 11 and Figure 12, shows that the central tendency from the Langley (2015) assessment is more pessimistic than OMyft2. The uncertainty in the assessment projections (inverse Hessian-delta approximation) appears to be very high, and higher than the OMyft2 ensemble (at least on the pessimistic tail of the distribution). We would not have predicted this outcome, given that the assessment result is based on a single model specification with a large number of influential parameters assumed to be known without error (and on the more optimistic side of some of the assumptions included in OMyft2). It is not clear whether the assessment projections should be considered to have unrealistically high uncertainty, or whether the grid-based approach in OMyft2 under-represents the pessimistic side of the stock status uncertainty.

- 2) The OMyft2 bridging catches are defined to be considerably lower than recent catches (300K t rather than 413K t), because a large number of simulations could not sustain the recent catches to the beginning of the MP period. The estimates of yellowfin stock status suggest that the stock could decline rapidly in the near future. Unfortunately, the MSE is sensitive to the assumptions about catches between 2015 and 2018 (the first MP year). More recent data and indicators might be available to help determine if the OMyft2 dynamics are too pessimistic.
- 3) The YFT assessment defined 9 quarters for the MSY reference period (but reported 8 in the text). This presumably introduces a minor seasonal imbalance, but was maintained in the demonstration case OM as part of the effort to identify and resolve inconsistencies. The OM has the option to define season-specific effort patterns, which may be important if seasonality is given more attention in the future (e.g. to avoid the issue of unrealistically high mortality in regions/seasons when the fish have migrated somewhere else).
- 4) SS does not provide information about relative fishing mortality by area for spatial configurations. In principle, it should be possible to back-calculate spatial  $F_s$  from the SS outputs, but an initial attempt to do this suggests an inconsistency in the SS code (or documentation) that has not yet been resolved. This contributes to the (relatively minor) MSY inconsistencies (Table 9) between SS and the MSE code, and creates a problem for simulating effort-based MPs for yellowfin. However, it does not affect quota-managed simulations, which have been the emphasis of IOTC MSE to date (i.e. including skipjack and albacore).

- 5) Given the diverse range of biological parameters estimated/assumed for yellowfin within the Indian Ocean (and in other oceans), it could be argued that additional uncertainties should be included in the OM grid, including:
- WPTT (2015) identified concerns about the total catch time series, particularly for artisanal fleets. While the secretariat would be in the best position to propose alternative catch data for inclusion in the conditioning, it would be relatively straightforward to come up with speculative scenarios (e.g. 25% under-reporting for the non-industrial fleets). This would not be for the purpose of questioning official statistics, but merely to test whether it matters for the MP performance. The main concern expressed by the WPTT relates to the potential misreporting of juvenile bigeye as yellowfin (given the relative size of the two populations, the effect would presumably be more of a concern for bigeye).
  - The assessment and conditioning process are critically dependent on the interpretation of CPUE series as relative abundance indices. The catchability relationship is always difficult to quantify because fleets change their temporal and spatial distribution, technology improves, and species targeting shifts in ways that cannot be properly quantified from the available data. It may be argued that the 1% CPUE catchability trend should be a robustness case, or perhaps it would be better to include a 1% increasing and decreasing catchability trend in the OM. The IOTC informal working group on longline catch rate standardization may be in the best position to propose alternatives scenarios. The assumption of shared (area-weighted) CPUE catchability among regions provides critical stability for assessment model fitting, but the weighting assumptions may merit further investigation.
  - Recent assessments have assumed that longline selectivity approaches an asymptote with increasing age. This is a common assumption in tuna assessments, that adds numerical stability to the assessment models, however, it may not have a strong justification in all cases. Given the uncertainty in  $M$ , and the model interaction between selectivity and  $M$ , admitting dome-shaped selectivity might be worthwhile. Given the diverse areas over which the longliners operate, and the potentially different tuna behaviour among areas, it may also be worth considering whether it is appropriate to share selectivity among regions.
  - The joint-tRFMO (Kobe process) MSE participants have expressed interest in comparing length-at-age estimation methods among species and oceans. We have suggested that this MSE case study would provide a valuable contribution in demonstrating whether or not the current levels of growth uncertainty are likely to make an important difference to management decisions and outcomes in an MP management framework.
  - Spatial processes. The SS models tend to estimate relatively low migration rates among some areas, such that some fisheries might be unable to attain their quotas despite very high fishing effort, while other fisheries easily achieve their targets. In the context of a single aggregate spawning population and stationary fishery distributions, this creates the possibility for spatial (and selectivity-based) refugia,

that may make it very difficult to recruitment over-fish the stock. As a consequence, it may be important to:

- Consider a range of other fixed input movement assumptions, to see if these parameters strongly affect the model fit, and whether the MPs are robust to the alternatives.
- Explore YFT scenarios with seasonal movement, e.g. environmental co-variates could be used as in the reference case assessment, using seasonal means in the projections (potentially with stochastic variability).

## 9.4 Bigeye demonstration Operating Model OMBet1

- 1) The most recent bigeye assessment, and hence the conditioned OMs, are several years out of date, and should be updated in parallel to the 2016 assessment, including any new insights on appropriate spatial structure, biological parameters, and catch (including speculation that juvenile bigeye may be mistakenly reported as yellowfin) and CPUE uncertainties. As noted in WPTT (2015), it would be useful if the next bigeye assessment coordinated the fishery definitions between yellowfin and bigeye, to simplify communication, and to enable consistent multi-species management considerations.
- 2) Sensitivity analyses in the assessment included multi-area models, but the preferred model structure was a single area, primarily because of the perception that tag mixing rates were too slow to be consistent with tag dynamics assumptions in the assessment model. While slow mixing dynamics provides a good justification for doubting the tag-based inferences, it also supports the idea that management sensitivities to alternative spatial structures may be worth exploring, even if the parameterization is based on speculative modelling.

## 9.5 Candidate MPs

- 1) There have been many MPs described in the literature in recent years, and some may provide better performance than those considered here. Time invested in improving MPs would probably be best spent when the performance objectives have been more narrowly defined (e.g. tuning objectives agreed). The MPs developed for IOTC albacore and skipjack were not evaluated for bigeye and yellowfin. If data requirements and MP performance are essentially equivalent, it might simplify the communication process within the Commission if fewer MPs were used among species.
- 2) The candidate MPs described in Table 5 did not attempt to use information from the size composition data. Provided that selectivity (and associated assumptions of growth and  $M$ ) remain stable, size data may provide useful information about incoming recruitment or spawning biomass. This was the intent of the size composition data in the Prince et al. (2011) empirical MP, however, the species-specific evaluations of that MP in Kolody et al. (2010) failed to demonstrate that the size composition data improved performance relative to the purely CPUE-based MPs.
- 3) From the demonstration case yellowfin and bigeye results described here, it appears that the situation for the two species is very different, and it may not be sensible to aim for the same generic management objectives. To bring the two species to a similar stock status

(and comparable risk level) in the medium term would appear to require some combination of i) aggressive catch reductions and rebuilding for yellowfin, and ii) an increased exploitation rate on bigeye. The rate of yellowfin rebuilding should be carefully considered as a management objective, while increasing bigeye exploitation rates may not be in the interest of industry, depending on the economics of declining catch rates (and possibly market responses to increased supply).

- 4) The MPs examined to date all prescribe quota changes relative to the previous quota. This may not be appropriate if there are large implementation errors. For example, in the case of bigeye, there may be no incentive for industry to catch the elevated quotas that would be required to bring  $SB$  down to  $SB_{target}$ . The MP algorithm may respond by recommending higher quotas and eventually the quotas and catches could become decoupled from one another. Of course, the same is true if catches greatly exceed quotas. Some facility for relatively minor carry-over of under-catches and over-catches from one year to the next would be worth considering, and provided that the over- and under- catches are small, there would probably be no need to explicitly evaluate this within the MP.
- 5) Model-based MPs, such as the PT models presented here, may have additional practical problems that are not a concern for the empirical MPs. Fitting an assessment model every time step can be prohibitively time-consuming in a simulation context, and may not be easy to automate. If the model does not always converge reliably to a global minimum, this may affect MP performance and the MP selection process. These were genuine problems in the CCSBT process, partly related to the fact that production models were difficult to fit to historical SBT dynamics. The Pella-Tomlinson models generally appeared to provide reasonable performance in this study (though the fit to the simulated bigeye and yellowfin data was often poor). It is important to avoid the awkward situation of creating an MP that offers the best performance subject to flawed model fitting.

## 10 Discussion

This document provides a largely qualitative description of the current state of the yellowfin and bigeye MSE projection software, demonstration case OMs and candidate MPs. The user manual and references provide specific technical details. It is hoped that this combination of products will be sufficient to allow the IOTC working parties and Commission to move the MSE process forward. However, it is also recognized that unanticipated challenges arose during the phase 1 project, and many somewhat arbitrary decisions had to be made. Some of these decisions were endorsed by the IOTC Working Party on Methods, but most were not reviewed by the broader IOTC Scientific Committee and associated technical working groups. We encourage scientists to provide feedback on technical concerns and priorities as soon as possible, and suggest that the preceding critique section is a good starting point. Ideally, the next phase of the technical process should include hands-on contributions from member scientists in terms of both the parameterization of OMs, and development/evaluation of candidate MPs.

As the MSE process moves forward in the next 2-3 years, we would expect the following key points of engagement between the MSE technical participants and the broader IOTC scientific and decision-making community:



- Attempts should be made to address some of the critical OM concerns identified in this report:
  - Most notably the uncertainty in the yellowfin OM should be reviewed, with at least qualitative consideration of new data (i.e. revised CPUE series, and updated catch histories).
  - The bigeye OM should be updated in relation to (ideally in parallel with) the 2016 assessment. Though if the current stock status perceptions are valid, this should be a lower priority than yellowfin.
- The yellowfin and bigeye OMs should be reviewed by the relevant IOTC scientists at the WPTT 2016, WPM 2016 and SC 2016, and modified accordingly.
- There should be peer review by external experts once the internal IOTC concerns are addressed (tentatively scheduled by the SC for 2017 and 2018, though expectations are not clearly defined).
- When the OMs have reached a sufficient consensus in the scientific community, preliminary MP results should be presented to the Scientific Committee and Management Procedures Dialogue meetings. Ideally a first iteration of this would happen at the SC 2016 or MPD 2017.
- The process should be recognized as iterative but with explicit target adoption dates. As commissioners are informed about the biological production and feedback control limitations of the system, the management objectives may evolve. As the management objectives become more specific, the scientists may be able to improve expected MP performance. But it must be recognized that the MP evaluations will never be perfect, and a substantive MSE process which has addressed the main concerns of the relevant scientific members should provide reasonable management outcomes. MSE technical problems should only be used to justify delays to the MP process if large uncertainties or errors are identified, i.e. of a magnitude that could reasonably be expected to cause the MP to perform worse than short-term ad hoc decision making.

One of the key scientific issues that needs to be addressed by the IOTC scientific community relates to how uncertainty and risk are being quantified for the different species. For presentation to the Commission, performance indicators are being communicated in a common format that includes percentiles of simulated outcomes that are likely to be interpreted as probabilities. This implies a consistency among species that may not be justified because of the very different and subjective approaches that are being used to quantify the uncertainty in the OMs.

## 11 Acknowledgements

This work was jointly funded by the European Union (through IOTC and the Indian Ocean Commission) and CSIRO. Technical oversight and advice was provided by the IOTC Working Party on Methods, notably Iago Mosqueira, Toshihide Kitakado, Nokome Bentley, Gerald Scott, Gorka Merino, Hilario Murua and Rishi Sharma. Operating model conditioning built upon the stock

assessment work of Adam Langley, Rishi Sharma and others at the IOTC, with useful advice from Ian Taylor. The R-based MSE code was adapted from the Atlantic Bluefin MSE work developed by Tom Carruthers (funded by the ICCAT GBYP project). Some of the summary result graphics were adopted from the Fisheries Library in R (FLR) code provided by Iago Mosqueira. Robin Thomson, Jemery Day, Hilario Murua and Gorka Merino provided useful comments on the draft report.

## 12 References

- Bentley, N and Adam MS. 2015. An operating model for the Indian Ocean skipjack tuna fishery. IOTC–2015–WPM06–09.
- Carruthers, T, Kell, L, Davies, C. 2014. Evaluating management strategies for Atlantic bluefin tuna. Report 1 – contract report to support to BFT assessment (GBYP 02/2014) of the Atlantic-wide research programme on bluefin tuna (ICCAT GBYP– Phase 4).
- Francis, RICC. 1992. Use of risk analysis to assess fishery management strategies: a case study using orange roughy (*Hoplostethus atlanticus*) on the Chatham Rise New Zealand Can. J. Fish. Aquat. Sci., 49 (1992), pp. 922–930.
- Hillary, RM, Preece, AL, Davies, CR, Kurota, H, Sakai, O, Itoh, T, Parma, AM, Butterworth, DS, Ianelli, J, Branch, TA. 2016. A scientific alternative to moratoria for rebuilding depleted international tuna stocks. Fish and fisheries 17: 469–482.
- IOTC 2011. Report of the fifteenth session of the Indian Ocean Tuna Commission. Colombo, Sri Lanka 18-22 March 2011.
- IOTC SC. 2015. Report of the 18th Session of the IOTC Scientific Committee. Bali, Indonesia, 23–27 November 2015. IOTC–2015–SC18–R[E].
- Kolody, D, Jumppanen, P, Carruthers, T, Langley, A. 2015. Indian Ocean Yellowfin and Bigeye Tuna Management Strategy Evaluation Development Framework – Draft Progress Update. IOTC-2015-WPM06-10.
- Kolody, D., S. Hoyle. 2013. Evaluation of tag mixing assumptions for skipjack, yellowfin and bigeye tuna stock assessments in the western Pacific and Indian Oceans. WCPFC Working Paper: WCPFC-SC9-2013/SA-IP-11.
- Kolody, D.S., A.L. Preece, C.R. Davies, J.R. Hartog, N.A. Dowling. 2010. Integrated evaluation of management strategies for tropical multi-species long-line fisheries. Final report for FRDC project 2007/017. 213p.
- Kolody, D., S. Hoyle. 2015. Evaluation of tag mixing assumptions in western Pacific Ocean skipjack tuna stock assessment models. Fisheries Research 163 (2015) 127–140.
- Langley, A, 2015. Stock assessment of yellowfin tuna in the Indian Ocean using stock synthesis. IOTC-2015-WPTT17-30.
- Langley, A, Herrera, M and Sharma, R. 2013. Stock assessment of bigeye tuna in the Indian Ocean for 2012. IOTC-2013-WPTT15-30 Rev\_1.

- Mosqueira, I and Scott, F. 2015. Operating model for Indian Ocean albacore tuna. IOTC–2015–WPM06–08.
- Langley, A., Million, J., 2012. Determining An Appropriate Tag Mixing Period for the Indian Ocean Yellowfin Tuna Stock Assessment. Indian Ocean Tuna Commission 14th Working Party on Tropical Tuna Working Paper IOTC-2012-WPTT14-31.
- Methot, R.D., Wetzel, C.R., 2013. Stock synthesis: A biological and statistical framework for fish stock assessment and fishery management. *Fish. Res.* 142: 86– 99.
- Prince, J, Dowling, N, Davies, C, Campbell, R and D. Kolody. 2011. A simple cost-effective and scale-less empirical approach to harvest strategies. *ICES J. Mar. Sci.* 68 (5): 947-960.
- Punt, AE, Butterworth, DS, de Moor, CL, De Oliveira, JAA and Haddon, M. 2016. Management strategy evaluation: best practices. *Fish and Fisheries*, 17: 303–334. doi: 10.1111/faf.12104.
- R Core Team (2015). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL <https://www.R-project.org/>.
- Schnute, JT, Richards, L, 2001. Use and abuse of fishery models. *Can. J. Fish. Aquat. Sci.* 58, 10–17.
- Tong, Y., D. Kolody, X. Chen, and Y. Chen. 2011. Evaluation of three harvest control rules for bigeye tuna (*Thunnus obesus*) fisheries in the Indian Ocean. IOTC–2011–WPTT13–INF15.
- WPTT 2015. Report of the 17th Session of the IOTC Working Party on Tropical Tunas. Montpellier, France, 23–28 October 2015.
- Zhang, Y, Chen, Y, Zhu, J, Tian. S, Chen, X. 2013. Evaluating harvest control rules for bigeye tuna (*Thunnus obesus*) and yellowfin tuna (*Thunnus albacares*) fisheries in the Indian Ocean. *Fisheries Research* 137 (2013) 1–8.

CONTACT US

**t** 1300 363 400  
+61 3 9545 2176  
**e** [csiroenquiries@csiro.au](mailto:csiroenquiries@csiro.au)  
**w** [www.csiro.au](http://www.csiro.au)

FOR FURTHER INFORMATION

**CSIRO Oceans & Atmosphere**  
Dale Kolody  
**t** +61 3 6232 5  
**e** [dale.kolody@csiro.au](mailto:dale.kolody@csiro.au)

AT CSIRO, WE DO THE  
EXTRAORDINARY EVERY DAY

We innovate for tomorrow and help improve today – for our customers, all Australians and the world.

Our innovations contribute billions of dollars to the Australian economy every year. As the largest patent holder in the nation, our vast wealth of intellectual property has led to more than 150 spin-off companies.

With more than 5,000 experts and a burning desire to get things done, we are Australia's catalyst for innovation.

CSIRO. WE IMAGINE. WE COLLABORATE.  
WE INNOVATE.

## **13 Appendix 1. User Manual for Indian Ocean Yellowfin and Bigeye MSE Software**

# User Manual for IOTC Yellowfin and Bigeye Tuna MSE software

Version 1.0 – June 2016

Paavo Jumppanen (Contact: [Paavo.Jumppanen@csiro.au](mailto:Paavo.Jumppanen@csiro.au))

Dale Kolody

CSIRO Oceans & Atmosphere  
GPO Box 1538, Hobart, TAS, 7001, Australia

## Table of Contents

1	Introduction.....	1
1.1	Acronyms and definitions used in this document .....	1
1.2	Documentation conventions.....	3
2	Obtaining and Installing the Software .....	3
3	Software Organization .....	4
3.1	OM Conditioning with Stock Synthesis.....	6
3.2	MSE Control and Projection software .....	8
3.3	C++ Projection Code.....	11
3.3.1	General Coding Details Regarding the Mseom C++ Library .....	12
3.3.2	R DLL and Interface Files .....	15
3.3.3	Loading the Library .....	15
3.3.4	Creating and Destroying an Object Instance.....	16
3.3.5	The MSY projection.....	16
3.3.6	The Managed Catch / Effort Forward Projection.....	18
3.4	Some issues to be aware of: .....	19
4	R Classes.....	28
	OMd.....	28
	Description.....	28
	Attributes .....	28
	Name .....	28
	Label .....	28
	Date.....	28
	Author .....	28
	Notes .....	28
	PrimarySource.....	28
	CppMethod .....	28
	UseCluster.....	28
	Npop.....	29
	Nfleets .....	29
	SSRootDir .....	29
	SBlim .....	29
	Flim.....	29
	OMList.....	29

nsimPerOMFile.....	29
proyears.....	29
targpop.....	29
seed.....	29
recentPerFirst.....	29
recentPerLast.....	29
seasonCEDist.....	29
nsubyears.....	30
lastSeas.....	30
firstSeas.....	30
firstSSYr.....	30
firstCalendarYr.....	30
lastCalendarYr.....	30
firstMPYr.....	30
MPDataLag.....	30
catchBridge.....	30
catchBridgeCV.....	30
indexFisheries.....	30
ReccvTin.....	31
ReccvRin.....	31
RecACTin.....	31
NInitCV.....	31
NInitCVdecay.....	31
selExpRange.....	31
selAgeRange.....	31
selWLRRange.....	31
TACEcv.....	31
Ccv.....	32
Icv.....	32
IACin.....	32
Cbcv.....	32
nCALobs.....	32
Ibeta.....	32
ITrendin.....	32
OMss.....	32
Description.....	32



Initializer Arguments .....	32
OMd .....	32
Report .....	33
UseMSYss .....	33
MSE .....	33
Description .....	33
Initializer Arguments .....	33
OM .....	33
MPs .....	33
Interval .....	33
Report .....	33
CppMethod .....	33
UseCluster .....	34
EffortCeiling .....	34
rULim .....	34
TACTime .....	34
5  Operating Model Equations .....	34
5.1  Notation .....	34
5.2  Population Dynamics .....	36
5.3  Management Implementation Errors .....	42
5.4  Observation Errors .....	42
6  Candidate Management Procedures .....	43
7  Performance Measures and Reference Points .....	45
8  Acknowledgements .....	48
9  References .....	49

## 1 Introduction

This user manual provides a roadmap to the initial IOTC yellowfin (YFT) and bigeye (BET) Management Strategy Evaluation (MSE) software, including the mathematical equations for the dynamics, candidate MPs, and R scripts for reproducing the results in the *IOTC Yellowfin and Bigeye Tuna MSE phase 1 project final report*. The initial work was jointly funded by the EU (through IOTC/FAO) and CSIRO. While the user manual focuses on technical implementation details and options, the phase 1 project report describes broader context to the MSE process, the reasoning behind key decisions in the initial software design for YFT and BET, specific demonstration case OM details, and a critique to assist the future development and review process. This user manual is expected to become a "living document", with contributions from various users as long as it remains relevant to the IOTC MSE process.

This project was initiated with an over-arching design objective of keeping the code accessible to IOTC member scientists through the use of freely available software, familiar to the fisheries science community, and ideally re-using and extending established MSE code to the extent possible. Most of the software is R-based (R Core Team 2015), and depends on packages freely available in the CRAN repository. The core of the MSE platform was adapted from the Carruthers et al (2014) application developed for Atlantic Bluefin Tuna (ABT) (and funded by the ICCAT Atlantic-Wide Research Programme on Bluefin Tuna). The general architecture for the IO YFT/BET MSE should be familiar to users of that software, though the implementation details have changed considerably (the main text of the phase 1 project report provides a summary of major changes). The overall package should be relatively simple to install and run for anyone familiar with R, however software modifications might not be easy for novice R users. There are two main components that are not implemented in R: i) Stock Synthesis (SS) assessment software (Methot and Wetzel 2013) is used for Operating Model (OM) conditioning, and ii) an optional C++ projection sub-routine can be called from R (detailed below).

This user manual and the initial release of the BET/YFT MSE software were completed under a finite project contract. It is hoped that the authors will remain engaged with the IOTC MSE process and able to contribute to feature extensions, bug fixes and practical application of the MSE software. However, any future support will be contingent on new funding arrangements.

### 1.1 Acronyms and definitions used in this document

ABT	Atlantic Bluefin Tuna.
BET	bigeye tuna.
CCSBT	Commission for the Conservation of Southern Bluefin Tuna.
CPUE	Catch per Unit Effort; usually assumed to be standardized into a relative abundance index for fish vulnerable to a particular fishery.
HCR	Harvest Control Rule – the numerical algorithm for recommending a management action (e.g. providing a TAC given a biomass estimate). In this document, the term is generally not intended to encompass data collection and analysis or fitting a stock assessment model, as these may be considered to be separate components of a complete MP.

ICCAT	International Commission for the Conservation of Atlantic Tuna
IOTC	Indian Ocean Tuna Commission.
IWC	International Whaling Commission.
OM	Operating Model – this usually refers to the combination of the generic projection software and suite of model specifications used to simulation test the performance of candidate MPs. We often refer to the OM projection software and OM parameterization separately.
MP	Management Procedure – the simulation-tested combination of pre-agreed data collection methods, supporting analysis, and Harvest Control Rule. The term is often used interchangeably with MSE, however the <i>sensu stricto</i> MP definition (as used in the IWC and CCSBT) explicitly requires a very high level of pre-specification (i.e. of the data requirements and supporting analyses), to preclude the inherent risk of assessment groups failing to reach consensus during the application of an HCR. MSE is a broader term that does not necessarily imply the same degree of pre-specification.
MSE	Management Strategy Evaluation – the process (or final product) of simulation testing a fishery management strategy (see MP).
MSY	Maximum Sustainable Yield.
SS	Stock Synthesis assessment model software.
TAC	Total Allowable Catch – the catch quota set by an MP (it could be fishery-specific or the aggregate across fisheries, depending on context).
TAE	Total Allowable Effort – a fishery effort constraint set by an MP. In this document it is manifested as an effort multiplier applied to recent estimates of fishery-specific fishing mortality from an assessment model. For the simulation testing, there is an assumption that effort regulations will translate directly into fishing mortality regulations. In practice, it may be very difficult to define effort in such a way that this can be achieved,
WPM	IOTC Working Party on Methods.
WPTT	IOTC Working Party on Tropical Tunas
YFT	yellowfin tuna.

## 1.2 Documentation conventions

We use the following style conventions in this document:

<b>filename.ext</b>	specific file or directory names or functions using character style FileNames.
Variables	variable and parameter names using character style Variables.
Code	source code using paragraph style Code.
<i>Menu Commands</i>	menu and/or button presses in dialogs and programs shown using Menu character style.

## 2 Obtaining and Installing the Software

The software and demonstration case MSE analyses in the phase 1 final report can be downloaded from Github (<https://github.com/pjumppanen/MSE-IO-BET-YFT>). This software was developed under Windows 7 (Service Pack 1), 64 bit Operating System. The code and documentation is publicly accessible (independent developers should fork their own version of the code, and advise the developers of substantive developments that should be uploaded to the main repository). The following steps are required before an MSE can be set-up and run:

- 1) Install R from <https://cloud.r-project.org/>. We recommend that new R-users should consider an R tutorial, and install an Interactive Development Environment (e.g. Rstudio or TINN-R) to navigate the source files and scripts.
- 2) Install the following R packages from CRAN:
  - *keep*
  - *stringr*
  - *r4ss*
  - *ggplot2*
  - *reshape2*
  - plus any others that are noted to be missing when running the software
- 3) Go to <https://github.com/pjumppanen/MSE-IO-BET-YFT> and click on the *clone or download* button, then press the *download zip* button.
- 4) Extract the **MSE-IO-BET-YFT-master.zip** file to a drive / directory of your choice.

Following a successful installation, the instructions provided in sections 3.1 and 3.2 can be used to reproduce the results of the phase 1 report.

The R code and control files, scripts, etc. are ASCII files and can simply be edited with standard text editors (or within an R IDE), and submitted to an active R session to update the MSE. There are also R callable C++ routines implementing the Baranov catch equation projection code in a DLL. We provide the source code, including the CSIRO-developed ADT software (Automatic Differentiation with Tapenade) and ancillary libraries, and the fully functional DLL. Any change to the C++ code

involving the objective functions will require updated differentials. Although the C++ code is freely modifiable, the step of creating differentials through the use of TAPENADE (INRIA, France) requires a license (as of Sep 2016, an annual license is once again free for academic use).

### 3 Software Organization

Figure 1 illustrates the principle software components and shows how they interact to complete the evaluation of candidate MPs. This section describes the software structures including the main files, objects and functions, including instructions for stepping through some of the high level control scripts and reproducing the results in the phase 1 project report.

The term operating model (OM), generally refers to the whole package of structural relationships and parameters required to simulation test the management performance of an MP. However, from the perspective of the software design, it is useful to consider the OM in different components. OM specification, as used below, involves fitting a suite of Stock Synthesis assessment models to estimate parameters and states that are consistent with the historical data and general understanding of the fishery through the stock assessment process (top section of Figure 1). In this application, the simulation software represents an independent implementation of the model used to estimate the parameters for the OM (lower section of Figure 1), though the assumptions between the two obviously need to be largely the same. The two sections in Figure 1 are discussed separately.

Simple R functions automate the process of setting up an array of stock synthesis analyses using template SS control and data files. Batch files are produced to automate the running of the SS analyses, and additional R functions are provided for exploring and evaluating the suite of assessment results. These tools form a template to help automate parts of the OM conditioning process, but the end-user must decide which models to retain or reject from the ensemble using their own judgement, and modify the template files accordingly. These functions are ad hoc tools that worked for the specific cases described and are included because they may be useful, but they were never intended to be robust all-purpose tools. Users will need to know enough about R to modify them for their own purposes.

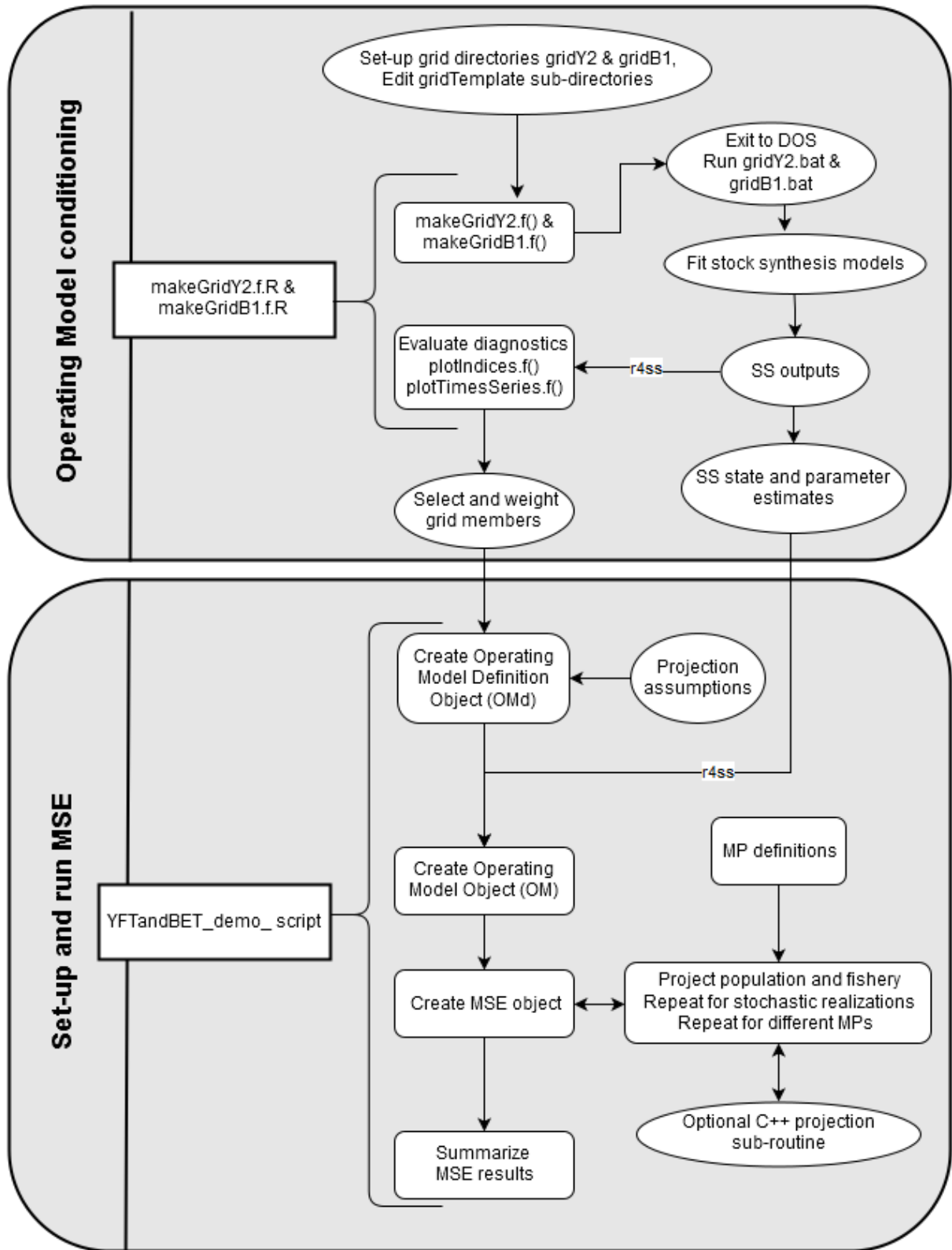


Figure 1. Schematic outline of the main IOTC yellowfin and bigeye MSE software components. Square boxes represent R-based code.

### 3.1 OM Conditioning with Stock Synthesis

Conditioning is the process of estimating OM parameters that are consistent with (conditional on) historical data and assumptions about the dynamics of the fishery and population. The approach to conditioning used here involves fitting a suite of Stock Synthesis (SS) assessment models (Methot and Wetzel 2013) that encompass a range of parameter, data and/or structural uncertainties relative to the most recent stock assessment. Note that this step can be skipped if the desired OM has already been created and saved. In that case, one can jump to the next section and simply `load()` the OM from within R (as indicated in **YFTandBET\_demo\_script.R**)

We refer to the combination of models as a "grid", which is usually a balanced factorial design of all possible assumption interactions (e.g. 3 levels of steepness, 3 levels of  $M$  and 3 alternative catch history scenarios would form a grid of  $3 \times 3 \times 3 = 27$  models). We usually refer to the combination of models comprising an OM as an ensemble, which could be the same as a grid, or include models from multiple grids, or may exclude some models from a grid (e.g. if they are judged to be biologically implausible, or redundant).

Table 1 lists the main scripts and functions used for the conditioning for the demonstration case yellowfin and bigeye OMs.

*Table 1. Main scripts and functions used to conduct the SS conditioning for the demonstration case Indian Ocean yellowfin and bigeye operating models.*

R Script	
<b>make_YFTOMGridScript.R</b>	YFT conditioning
<b>make_BETOMGridScript.R</b>	BET conditioning
<b>seasAsYrToDecYr.f.R</b>	Functions for converting date formats
R Function	
<code>makeGridY2.f()</code>	Make YFT grid
<code>makeGridB1.f()</code>	Make BET grid
<code>importGrid.f()</code>	Import grid of SS output files into R
<code>seasAsYrToMSEYrSeas.f()</code>	Function for converting between SS seasons-defined-as-years and calendar years (in file <b>seasAsYrToDecYr.f.R</b> )
<code>plotIndices.f()</code>	Plot some summary SS output statistics partitioned by grid assumptions
<code>timeSeriesPlots.f()</code>	Plot time series of some SS output statistics as summary distribution percentiles

For the demonstration cases, we used SS version 3.24Y (included in the bundled files as **ss3.24Y.exe**). Earlier versions of SS may fail because of a change to the growth parameterization, or provide incorrect results, due to a bug in the spawning biomass calculations (discussed in the phase 1 project report). The file **make\_YFTOMGridScript.R** reproduces the yellowfin results from the final report which is the example case discussed below (**make\_BETOMGridScript.R** is the analogous file for bigeye). Key steps (following software installation described in section 2):

- 1) Prior to running **make\_YFTOM\_script.R**, it is necessary to create the root dir **gridY3**, which must contain a **gridTemplate** folder and SS executable (see: ...\**MSE-IO-BET-YFT\OMconditioning\YFT\gridY3\gridTemplate**). **gridTemplate** includes all the files required to conduct an SS model fitting (e.g, control, data and forecast files). The files **templateYFT.dat** and/or **templateYFT.ct1** need to be modified to include all assessment model options required for the grid. This is accomplished with character string option flags. For example, to represent three levels of stock recruit steepness, **templateYFT.ct1** contains the following lines:

```
# steepness switches
# xxx h70 0.2 1 0.7 0.7 0 0.2 -1 # SR_steep
# xxx h80 0.2 1 0.8 0.8 0 0.2 -1 # SR_steep
# xxx h90 0.2 1 0.9 0.9 0 0.2 -1 # SR_steep
```

Each line starts with a character string identifier e.g. # xxx h70, where the first character is the Stock Synthesis comment character # that causes SS to ignore the remainder of the line.

- 2) The function **makeGridY3.f()** then creates a file structure in which a separate SS model is assigned to its own directory, which contains all of the required SS inputs. A model sub-folder is created for every combination of model options and the options are captured in the folder name. That is, folder **GridY3** contains 54 model configurations, across 5 dimensions, each given a sub-folder with a name resembling **R4MvEst\_h70\_M06\_t00\_q0**. In this case, the first dimension **R4MvEst** refers to spatial and population structure (4 regions, movement estimated) and is actually identical for all models in this grid. The other 4 dimensions have 2-3 levels each (see the phase 1 project report for details), where **h70** refers to the stock recruit steepness option ( $h = 0.7$ ), while the other options include **h80** and **h90** ( $h = 0.8$  and  $0.9$  respectively).

**makeGridY3.f()** copies over the requisite template files, and modifies them by stripping out the relevant option flags (removing the # xxx h70 comment above activates the first line of the **.ct1** file above, while the other options remain as inactive comments).

The function also creates a DOS batch file within each new directory to run the individual SS analysis. There are two batch file options, with or without the inverse Hessian calculation. Including the calculation can greatly increase the SS run time, but may provide useful information about parameter estimation uncertainty or convergence problems. A master batch file at the **GridY3** root, **gridy3.bat** is also created that sequentially calls all of the individual batch files. By default, 4 smaller non-overlapping batch files are also created that can be run instead of the master batch file (in parallel to take advantage of multiple CPUs).

- 3) After running **makeGridY3.f()**, pause in the **make\_YFTOM\_script.R** script, open a DOS command windows and call **gridy3.bat**. This runs a SS analysis (fits an assessment model) for each element of the grid, and could take many hours, depending on the size of the grid and complexity of the models.
- 4) Function **importGrid.f()** imports each SS model output as an individual R object.



- 5) Function `plotIndices.f()` plots some simple summary distribution statistics across the grid to rapidly inspect for outlier behaviour in terms of convergence, gross fit between predictions and observations, recruitment trends, general stock status inferences, etc. Each index is disaggregated according to assessment option, and the distribution (boxplot) is plotted with the other options marginalized, (e.g. see the phase 1 final report).  
Function `plotTimeSeries.f()` can be used to plot the distribution of some of the standard time series, e.g.  $B_t/B_{MSY}$  (see phase 1 final report).

The procedure outlined in this R-script does not constitute a comprehensive stock assessment model evaluation, but provides a useful way to set-up and review a substantial number of models. The demonstration grids for both bigeye and yellowfin are very well-behaved, and reasonably consistent with the inferences from the recent stock assessments. However, a large grid with interesting contrast in a large number of dimensions can be expected to produce implausible dynamics, and subjective decisions about what options to include in the grid, and how to weight (or reject) individual models are always required, and may involve several iterations. Detailed results from some individual models can and should be examined using the **r4ss** package (version 1.24.0, from CRAN) functions `SS_outputs()` and `SS_plots()`, as in a traditional SS stock assessment.

In this MSE software, differential weighting of SS models within the OM ensemble is achieved by the number of stochastic simulations for each set of SS parameters (i.e. number of simulations is proportional to the weight). Most OM model specifications are adopted directly from the SS outputs, including the initial numbers-at-age for the first year of the projections, key biological parameters (M, stock recruit steepness, size-at-age, etc.) and fishery selectivity. However, a number of additional parameters are required to fully specify an OM as detailed in the OM definition and OM objects below.

The full suite of SS output files required to populate the demonstration case OMs in the phase 1 report are not provided on GitHub (because of the large number and size of files). They can be recreated using the scripts above, however, the full demonstration case OMs are supplied as R objects that can be loaded to run the MSE. The SS output files are only provided for a couple of models to demonstrate the OM loading.

## 3.2 MSE Control and Projection software

Once the conditioned assessment models have run and the grid of YFT and BET model results created, the demo script **YFTandBET\_demo\_script.R** can be run. To do so, run **Rgui (64 bit)**, then change the working directory to the folder **YFT-MSE** (which is a subfolder of the **MSE-IO-BET-YFT-master** project that you downloaded and extracted from GitHub). You can do this by selecting the *File/Change Dir...* menu item and then navigating to and selecting the **YFT-MSE** folder and pressing *OK*. Now the demonstration cases can be run by selecting the *File/Source R Code...* menu, then navigating to the **Rscripts** subfolder to select the **YFTandBET\_demo\_script.R** and finally press the *OK* button. It may be preferable to step through the R script by submitting the code line by line or in small blocks. Using an IDE for R is a preferable way to work with the R script files.

Table 2 lists the object classes, methods and function classes of the MSE software. This core architecture remains similar to the ABT platform of Carruthers et al (2014). Depending on the level at which one wishes to engage with the MP process, the software implementation may be largely irrelevant.

*Table 2. The object classes, methods and function classes of the YFT/BET MSE software*

Object Classes	
OMd	Operating Model definition - includes list of conditioned SS models and projection assumptions including process, observation and implementation error assumptions, and projection sub-routine options (R/C+, single-threaded, multi-threaded)
OMss	Operating Model - an ensemble of models including the parameters and initial states imported from the SS output files, and other specifications defined by the corresponding OMd
MSE	Management Strategy Evaluation - summary of MSE results (generated from the internal MSE projection function code or C++ projection subroutine) by MP
Methods (core)	
<code>new(OMss)</code>	Creates an instance of an OMss object -
<code>new(MSE)</code>	Creates an instance of an MSE object
Methods (ancillary)	
<code>tableMSE.f(MSE)</code>	Tabulates the performance measures for the selected MSE object(s)
<code>tableMSEbyRF.f(MSE)</code>	Tabulates the selected performance measures by region/fishery for the selected MSE object(s)
<code>plotTO.f(MSE)</code>	WPM standard bi-variate trade-off plots.
<code>plotUS.f(MSE)</code>	WPM standard confidence interval plots.
<code>plotTS.f(MSE)</code>	WPM standard time series plots.
Function classes	
IO_MP	Management Procedure - an algorithm that imports observed data and produces an aggregate TAC and fishery-specific effort multiplier.

The R script **YFTandBET\_demoScript.R** demonstrates several MSE applications (lower portion of Figure 1) and assumes that the grid of SS models for the YFT and BET cases have been created. Internally, it calls scripts to create operating model definition objects, and from those a corresponding Operating Model. For instance,

```
source("Rscripts/Build OMyft2r108.R")
```

```
print(system.time(OMyft2r108<-new("OMss",OMd, Report=F)))
```

creates the operating model OMyft1r108 from the definition in the **Build OMyft1r108.R** script file. It can take many minutes up to a couple hours to create an OMss object, however, this should only need to be done once, as these objects can be saved and subsequently loaded as shown in **YFTandBET\_demoScript.R**. Note that `print(system.time(...))` is only included to record how long a function call takes and is not required. With this operating model it then performs a management strategy evaluation by creating an instance of the MSE class with:

```
MPL1 <- c("CC050", "CC400", "PT41.100.2", "IT2.50")

print(system.time(mseOMyft1r216.r<-new("MSE",
                                       OMyft1r216,
                                       MPs=MPL1,
                                       interval=3,
                                       Report=F,
                                       UseCluster=0)))
```

In this case the management procedures to be used in the evaluation are named in the list `MPL1` argument. The `interval=3` argument indicates that the MP is to be applied every 3 years, `Report=F` indicates that various runtime diagnostic outputs will be suppressed, and `UseCluster=0` indicates that multi-threading will not be used. Results can then be plotted with one of the plot functions provided in the **mseGraphics.R** file (e.g. Table 2), which includes: confidence interval plots, bivariate trade-off plots or time series plots (see examples in the phase 1 final report). For example,

```
plotTS.f(mseObj=mseOMyft2r108.MPL1)
```

makes time series plots for the MSE object `mseOMyft2r108.MPL1`.

The `OMd` object instance defines the characteristics of the OM. This includes a list of SS models from the grid, a vector of weighting factors (of length equal to the number of models in the list), and various parameters that control the projection characteristics (e.g. including the number of years to project, and observation, process and implementation errors). The full list of `OMd` parameters is listed in Table 3. Some of these parameters support additional options which are not required in the examples shown here. Some have not been tested in the new code, but were retained from the original ABT application, in recognition that it may be desirable for future expansion (e.g. to allow non-stationary or region-specific variability in parameters).

The `OMss` object instance creates an operating model from the `OMd` named in the instantiation call. This process involves reading and interpreting the SS model data and massaging it into an appropriate format for the model, creating and initialising state variables, creating placeholders for the model simulation results and calculating MSY estimates for each SS model (but not for each replicate). This can take some time (in the order of 30 seconds per SS specification for the YFT models), although the OM can be saved and reloaded quickly for future re-use. OM parameters are listed in Table 4. In a regular MSE there should be no need for the user to access the OM contents directly, as they are populated by the `OMd` object and the SS output files.

MP objects need to be defined before the MSE simulations can be run (with a template of options defined in `MPs.R`). The MP imports key historical data (i.e. actual data used in conditioning, plus new data from the projections, with simulated observation error), and outputs an aggregate annual catch quota, plus a vector of fleet-specific effort multipliers (for those fleets not managed as part of the quota system). This is detailed in the following section on MPs.

The MSE object is the computational workhorse of the MSE software, projecting the population and fishery forward in time, subject to management interventions, and repeating the process for the specified number of simulations and for each MP requested. The option exists here to either use the R-based projection code, or a C++ based sub-routine (detailed below). The completed MSE object is primarily an archive of historical realizations and observations.

The original specification of the MSE code maintained the full array structure of the Carruthers et al. (2014) ABT application, and proved problematic for the demonstration case YFT application described here. While it was functional on a powerful cluster, it was unnecessarily slow (despite the provision for parallel computing) and impractical in terms of memory management for standard laptop or desktop computers. Two approaches were taken to improve the efficiency of the code. The first involved removing the year dimension of the main OM data structures. The second involved a re-implementation of the R projection code using more efficient C++ code to implement a complete Baranov solution, as discussed in the following section. A practical comparison and critique of the two approaches is provided in the phase 1 final report.

### 3.3 C++ Projection Code

In addition to the computational efficiency gains to be expected by using C++ based projection code for the main projection calculations, there were two other perceived benefits: i) parallel implementation by a second programmer provided an independent check on the integrity of the R-based projection code, and ii) the C++ code was implemented with the standard Baranov catch equations. The main problem with the Baranov equation is that there is no closed form solution for fishing mortality, given a known catch (e.g. when simulating a TAC extraction). An iterative, computationally intensive algorithm is required to reach a solution (to a user-defined degree of precision).

Because of computational efficiency limits, the R-based MSE projection code uses an approximation to the catch equation (described in section 5). While there is some debate about whether the Baranov catch equation is the most appropriate to describe the natural and fishing mortality processes (e.g. Liu and Heino 2014), it is the most commonly used approach in fisheries assessments, including the Stock Synthesis models used for recent yellowfin and bigeye assessments and OM conditioning. The phase 1 final report provides a comparison of the two approaches for YFT and BET demonstration case OMs using several MPs with a range of constant catch objectives, from which it is evident that:

- C++ allows the computationally-demanding Baranov equations to be implemented with about the same run time as the R approximation, such that there is no obvious speed advantage to either.
- When fishing mortality rates were low enough that the prescribed quotas could be attained, the numerical results were essentially equivalent and should not make a noticeable difference to the MP selection process.

- The BET results were also equivalent when quotas were too high to be attained. However, the YFT results can potentially deviate substantially when fishing mortality is very high. The problem appears to arise in YFT because of the spatial structure, when exploitation rates in some areas can be extremely high, while other areas remain rather less exploited. In these cases, the deviation between Baranov and the approximation was lowest when the catch was extracted at the beginning of the time-step (rather than the mid-point), and when the proportional harvest rate constraint was activated at 0.3 rather than 0.5 (see section 5). However, this observation is not the result of a systematic analysis, and may not be true in all circumstances.
- The choice between R and C++ may have implications for choosing between some MPs, but we expect that neither approach would be very accurate under the relevant circumstances, because it is likely that fishery behaviour would change if depletion was that high. And presumably those MPs would not be consistent with the Commission objectives.

The C++ based projection code represents the solution that is the most consistent with the SS conditioning (and most fisheries applications), but it comes with a cost in terms of programmer skill requirements. CSIRO developed an adjunct tool, ADT (Automatic Differentiation with Tapenade), that when combined with TAPENADE (*TRansformations et Outils Informatiques Pour le Calcul Scientifique*), allows the development of C++ code that can make use of forward and reverse mode differentials created through code translation (as opposed to operator overloading) and higher order differentials. TAPENADE (<http://www-sop.inria.fr/tropics/tapenade.html>) was developed by INRIA, National Institute for Research in Computer Science and Control, France). We developed the C++ portion of the MSE tool with the understanding that TAPENADE would be freely available to provide high quality differential codes. Part way through the phase 1 project, INRIA changed their licensing arrangement, in a manner that undermined the intended distribution options for the software. As a consequence, we maintained fully functional R and C++ implementations of the MSE projection code, and made it simple to use either interchangeably. As of Sep 2016, INRIA have reverted to their original license agreement, such that TAPENADE annual licenses are again available free of charge for academic use.

### 3.3.1 General Coding Details Regarding the Mseom C++ Library

The **Mseom** library provides functional support to do MSY projections and managed catch / effort projections. It is implemented in C++ and relies upon ADT to create R callable interface code and differentiated versions of the respective objective functions used in minimisation. The latter part requires the TAPENADE tool. Here we give a brief overview of how the library is structured in an ADT based project but for an in depth discussion refer to the **users-manual.pdf** in the ADT software distribution, which is available from GitHub (<https://github.com/pjumpnanen/ADT>). Note that the **Mseom** library source code is bundled with ADT as a sample project.

ADT based libraries are built by deriving a class from `AdtArrays` and adding the functionality required. The `AdtArrays` class is provided by the **ADlib** library and provides support for multi-dimensional C style arrays with arbitrary indexing and contiguous memory block allocation. Any code that needs to be interfaced with R, or needs to be differentiable, must be coded in a subset of C++ and use only intrinsic data types and the array types provide by **ADlib**. This is required because ADT parses and translates the source code, and the complete language definition is hideously complex. The code generally also requires special comment fields that tell ADT about the sizes of arrays and other details regarding the generation of interface code. An ADT make file tells ADT which files to parse and process. The processing involves parsing the code, performing the prescribed

operations on the code and producing new methods and possibly interface code added to a class derived from the parsed one.

In this case, the base class is `OperatingModelBase` and is declared in the `Mseom.hpp` header file. A subset of the class definition is shown below.

```
class OperatingModelBase : public AdtArrays
{
protected:
    /* AD_LIBNAME Mseom */
    /* AD_ALIAS OmB=D_OperatingModelBase */
    /* AUTOINIT */
    int      nsim;
    int      npop;
    int      nages;
    int      nsubyears;
    int      nareas;
    int      nfleets;

    ARRAY_1I Recsubyr /* nsubyears */;

    /* AUTODEC */
    int      SpawnPerYr;

    ARRAY_4D EforYear /* nfleets, nareas, nsubyears, nsim */;
```

The comment `AD_LIBNAME` tells ADT the name of the library generated in compilation. It is needed so that ADT can generate the correctly named registration function to register R callable C functions with R.

The `AD_ALIAS` comment tells ADT the name of the class it will be generating interface code on and its alias. The long name is the actual class name whereas the alias is used to generate functional prefixes for the R callable interface.

The `AUTOINIT` comment instructs ADT that any class attributes appearing after this comment should be initialised via the constructor. This remains in force until the `AUTODEC` comment, which instructs ADT to only generate code to instantiate those attributes but not initialise them from R via constructor arguments. Therefore, in the sample shown the attributes `nsim`, `npop`, `nages`, `nsubyears`, `nareas`, `nfleets` and `Recsubyr` will be initialised via constructor arguments whereas `SpawnPerYear` and `EforYear` will not.

Note the two array declarations and the corresponding comments. Arrays are multi-dimensional and currently Adlib provides support for up to 10 dimension arrays. The type name for arrays always begins with an `ARRAY_` prefix followed by the number of dimensions followed by a type specifier. `I` refers to a signed integer type whereas `D` refers to a double type. The comment appearing after the attribute names the dimensions for the array, the general format of which is,

```
/* {from index:}to index */
```

with the `from index` being optional. When not specified the `from index` defaults to 1 in deference to the array indexing in R. The `EforYear` array therefore has the dimensions `nfleets`, `nareas`, `nsubyears` and `nsim`. Note that in a class declaration context all dimension variables must be declared as class attributes in an `AUTOINIT` context and in a class method context any dimensions not appearing as class attributes must be passed in the method argument list. It should be noted that arrays in R are organised in column major format whereas the C/C++ language is row major. Therefore, all the array dimensions in the C++ code appear in reverse order to that of R to avoid the need to re-map the memory when using R arrays in C++.

Another feature of ADlib made use of in this library is parellisation, which is handled transparently with the `parallelFor()` function, which, as the name suggests, runs a for loop in parallel. A typical example in the code is,

```
ParallelForContext Context(this,
                           nReport,
                           ECurrent,
                           qY,
                           R0,
                           M,
                           mat,
                           Idist,
                           Len_age,
                           Wt_age,
                           sel,
                           mov,
                           h,
                           Recdist,
                           SRrel,
                           N,
                           NBefore,
                           SSN,
                           C,
                           SSBA,
                           ntargets,
                           targpop,
                           run_years,
                           MSY,
                           BMSY,
                           SSBMSY,
                           SSBMSY_B0,
                           MinPar,
                           maxit);
```

```

allocateShallowCopies(this);

parallelFor(&Context, OperatingModelMin::findMSYrefsTask, 1,
nsim);

freeShallowCopies();

```

The function `OperatingModelMin::findMSYrefsTask` is called from multiple threads with a unique simulation number. As the for loop will generally be acting on the one object there is potential for shared memory access conflicts so to avoid this we need to create unique copies of the object, although not everything requires a full copy. The calls to `allocateShallowCopies()` and `freeShallowCopies()` provide this support. Typically, the points where parallel execution are used in the library revolve around performing minimisation. Unfortunately, the stock minimizers that R provides C callable interfaces to are not thread safe so **ADlib** provides a thread safe implementation. These have the same calling semantics and names as their parent R implementations but with a name prefix of `ts_` to indicate thread safety.

With this knowledge on basic aspects of an ADT built library it should be possible for someone familiar with C/C++ programming to be able to understand the **Mseom** library source code.

### 3.3.2 R DLL and Interface Files

The R DLL and interface files consist of; **Mseom.dll**, **Om\_R\_interface.r**, and **OmB\_R\_interface.r**. The two R files provide human friendly R functions that wrap corresponding `.Call` invocations. Both must be included in the R code to use the library via `source()` commands.

### 3.3.3 Loading the Library

The dynamic library should be loaded into R via the `dyn.load()` command. Furthermore, the code should check for previous loading of the DLL and only load it once. Failure to do so usually results in R crashing, although the exact cause of this behaviour is unclear. The supplied code assumes the R dll resides in the **lib** sub-folder off the **YFT-MSE** parent folder and is illustrated below. It is generalised to support Linux builds as well as Windows, and includes the corresponding R interface code. It should be noted that we only provide a 64 bit build of the DLL so it will be necessary to run under a 64 bit version of R. For runs with many simulations, this is essential as the 32 bit version of R will typically fail to run owing to array allocations exceeding the imposed minimum of 32 bit R.

```

#load Mseom module and R interface code
# Load the library
if (version$os == "mingw32")
{
  # Running in Windows
  LibName      <- "Mseom"
  LibExtension <- ".dll"
}

```



```

} else
{
  # Running in Linux
  LibName      <- "libMseom"
  LibExtension <- ".so"
}

# Only load library if not already loaded.
# Loading more than once results in R mis-behaving
# and crashing
if (is.na(match(LibName, attr(getLoadedDLLs(), "names"))))
{
  LibPath <- paste("./lib/", LibName, LibExtension, sep="")
  dyn.load(LibPath)
}

source("./lib/OmB_R_interface.r")
source("./lib/Om_R_interface.r")

```

### 3.3.4 Creating and Destroying an Object Instance

The callable code within the mseom library comprise a number of methods in a given class. Therefore, to be able to make use of the code we must first create an object instance in R. We do so with, for example,

```

Obj <- Om.create(nsim,
                npop,
                nages,
                nsubyears,
                nareas,
                nfleets,
                OM@Recsubyr)

```

Internally the class constructor creates a number of working arrays needed for the model code whose dimensions are based on `nsim`, `npop`, `nages`, `nsubyears`, `nareas` and `nfleets`. When we no longer need the object instance it should be discarded by calling,

```
Om.destroy(Obj)
```

### 3.3.5 The MSY projection

The **Mseom** library provides methods to return both the MSY projection and the managed catch / effort forward projection. To ensure consistency it implements the population dynamics in common code shared between both these tasks. The population dynamics implementation is functionally

identical to R implementation although in the case of the R code there is duplication between the different tasks.

The C++ implementation of the MSY projection is functionally identical to the R based implementation as testing has demonstrated. The essential difference in returned results is attributable to the different behaviour of the minimizer used. In the case of the R code the `optimize()` function is used for minimisation whereas the C++ code uses the equivalent of the `lbfgsb()` C callable minimizer which is used internally by the `optim()` function in R when using the "L-BFGS-B" method.

The MSY projection works by maximising the yield of a forward projection in time using a scaled effort based on the reference catch years. The projection is taken forward a fixed number of years specified in the call and needs to be sufficient in length to ensure steady state behaviour is reached. With a valid object instance the MSY level is found by calling the `Om.nt.findMSYrefs()` function as with,

```
Om.nt.findMSYrefs(Obj,
                  Report,
                  .Object@ECurrent[SimNums,,,],
                  .Object@q[SimNums,],
                  .Object@R0[SimNums,],
                  .Object@M[SimNums,,,1],
                  .Object@mat[SimNums,,,1],
                  .Object@Idist[SimNums,,,],
                  .Object@Len_age[SimNums,,,1],
                  .Object@Wt_age[SimNums,,,1],
                  .Object@sel[SimNums,,],
                  .Object@mov[SimNums,,,,],
                  .Object@h[SimNums,],
                  .Object@Recdist[SimNums,,],
                  .Object@SRrel,
                  N,
                  NBefore,
                  SSN,
                  C,
                  SSBA,
                  ntargets,
                  as.integer(.Object@targpop),
                  as.integer(nyears),
                  MinPar,
                  MSY,
                  BMSY,
                  SSBMSY,
                  SSBMSY_B0,
                  maxit)
```

Note that some arguments are output arguments (SSN, C, SSBA, MinPar, MSY, BMSY, SSBMSY, SSBMSY\_B0), some are state variables (N, NBefore) and others input parameters (ECurrent, q, R0, M, mat, Idist, Len\_age, Wt\_age, sel, mov, h, Recdist, SRrel). The output arguments and the state variables are owned by R and must have the correct type and size for the code to function as expected.

### 3.3.6 The Managed Catch / Effort Forward Projection

The managed catch / effort forward projection is driven by an overall TAC level and fishery specific effort levels (TAE's). All fisheries with zero effort levels are assumed to be managed under an overall TAC. This overall TAC is broken down to fleet specific TACs calculated on the basis of the reference catch mass case in CMCCurrent (plus implementation error). The projection is carried out over a full year with the fleet specific TACs corresponding to annual figures. Forward projection is solved by minimizing,

$$\text{objective} = \sum_f \left( \ln\left(\frac{TAC_f}{C_f}\right) \right)^2$$

where the summation is carried out over all fleets under TAC management. There is no attempt to impose a penalty on extremely high effort in order to obtain a given TAC, other than a maximum effort boundary constraint controlled by the `EffortCeiling` parameter in the calling R code. This parameter represents the maximum absolute effort for any given controlled fishery. The objective parameters being optimized are simply scalar multipliers applied to the historic effort in the `ECurrent` array. As such, fisheries with low initial effort will have a higher boundary constraint than heavily fished ones as the parameter multipliers are relative (to `ECurrent`) rather than absolute parameters. The forward projection takes place in the call to the `Om.nt.projection()` function call, as in,

```
Om.nt.projection(Obj,
                 Y,
                 as.integer(if (Report) 1 else 0),
                 EffortCeiling,
                 TAC,
                 TAEbyF,
                 TACEError,
                 ECurrent,
                 CMCCurrent,
                 q,
                 R0,
                 M_Y,
                 mat_Y,
                 Idist,
                 Len_age_Y,
                 Wt_age_Y,
                 Wt_age_mid_Y,
```

```

selTS,
mov,
h,
Recdist,
Recdevs_Y,
recSpatialDevs,
OM@SRrel,
N_Y,
NBefore_Y,
SSN_Y,
C_Y,
SSBA_Y,
as.integer(100));

```

As with the MSY projection some of the arguments represent output and state parameters which need to be setup within the calling R session. Furthermore, the `Om.nt.projection()` function assumes the `N` and `NBefore` state variables are initialised to a known state consistent with the point in time it is projecting from. For this reason it is necessary to run a historic projection for the first year to ensure `N` and `NBefore` are in the correct known state. This is handled by the call to `OmB.nt.runHistoric()`. Prior to running either, it is necessary to call `OmB.nt.initialiseParameters()` to initialise the initial population and to calculate survivorship. It is also necessary to call `Om.nt.beginProjection()` prior to forward projection. This function call sets up the initial value of the optimisation parameters, which are saved in between annual projections with subsequent year starting point carrying on from the previous year fit. The rationale behind this behaviour is to (hopefully) reduce the number of iterations required to solve the projection step by starting at a point close to the anticipated solution. The assumption is that the effort multiplier series is likely to be highly correlated with previous years.

### 3.4 Some issues to be aware of:

- R, R packages and Stock Synthesis are all continuously evolving, and newer or older versions than those described here can be expected to cause software instability, e.g. the phase 1 final report discusses problems with Stock Synthesis version 3.24F, which was used in the Langley (2015) assessment; switching to 3.24Y required some model re-parameterization.
- Year-season configuration. Because of a limitation in the way that SS assigns seasonal tag age-classes, at this time it is preferable for the tropical tuna assessment models to define calendar seasons as model years (e.g. Langley 2015). Some functions are provided for converting time units back and forth in **seasAsYrToDecYr.f.R**, for plotting and importing to the MSE software below. In the OM software, rate processes are defined in annual units (standard IOTC reporting units). Many SS parameters in a calendar-seasons-as-SS-model-years configuration must be re-scaled (e.g.  $M_{OM} = 4M_{SS}$ ,  $MSY_{OM} = 4MSY_{SS}$ ). The conversion is not necessarily intuitive, e.g.  $F_{OM, quarter} = 1/4 F_{SS}$ , because the calendar-seasons-as-SS-model-years configuration includes a fishery duration definition of 0.25, which is internally converted to an "annual" equivalent for reporting, and which must in turn be externally converted back to a true quarterly  $F$  because the OM operates on a quarterly time-

step. Use of a true year-season SS configuration was not explicitly supported in the OM importing software, and will (probably) cause errors if attempted.

- In spatial models, SS reports spatially-aggregated fishing mortality, and not the region-specific fishing mortality, which is required to estimate the local effective effort for effort-based management. In principle, it should be possible to back-calculate fishing mortality by region from the SS numbers-at-age outputs, but initial attempts to do so suggested that the relevant outputs are not intuitively defined. This is important for calculating MSY-based reference points and using effort-based management, but does not affect the quota-based management projections which have been favoured by IOTC MSE initiatives to date.
- The OM software has flexibility to accept SS inputs with different structural assumptions (e.g. number of spatial units, fisheries, age-classes, etc.). However, these structures must be constant for an individual OM definition, i.e. At this time, different spatial structures or numbers of fisheries, etc. must be defined as different OMs. The MSE software could be run on these different OMs independently, and the results subsequently merged (though the need has not yet arisen to develop such functions).
- The software was originally intended to support multiple independent spawning populations, with independent biology (i.e. retaining this feature from the Carruthers et al 2014 ABT application). However, this feature remains untested in the projection code, and the code for importing multi-stock OMs would need modification.

*Table 3. OMD (Operating Model definition) object class attributes defined in file Objects-OMss.R (additional attributes may be defined for potential future expansion).*

Slot/attribute	
<b>Identifiers</b>	
Name	= "character"
Label	= "character"
Date	= "character"
Author	= "character"
Notes	= "character"
PrimarySource	= "character"
OMList	= "list"
SSRootDir	= "character"
SBlim	= "numeric"
Flim	= "numeric"
CppMethod	= "numeric"
UseCluster	= "numeric"
seed	= "integer"
recentPerFirst	= "integer",
recentPerLast	= "integer",
seasonCEDist	= "integer",
catchBridge	= "karray",
catchBridgeCV	= "numeric",
NInitCV	= "numeric",
NInitCVdecay	= "numeric",
selExpRange	= "numeric",

---

selAgeRange	= "numeric" ,
selWLRRange	= "karray" ,

---

---

#### Dimensions

---

npop	= "integer" ,
proyears	= "integer" ,
targpop	= "integer" ,
nsimPerOMFile	= "karray"
nfleets	= "integer" ,
nsubyears	= "integer" ,
firstCalendarYr	= "integer" ,
lastCalendarYr	= "integer" ,
lastSeas	= "integer" ,
firstSeas	= "integer" ,
firstSSYr	= "integer" ,

---

---

#### MP details

---

firstMPYr	= "integer" ,
MPDataLag	= "integer" ,

---

---

#### Fishery

---

---

#### Biology

---

ReccvTin	= "karray" ,
RecACTin	= "karray" ,
ReccvRin	= "numeric" ,

---

---

#### Observation and Implementation errors

---

indexFisheries	= "integer" ,
----------------	---------------

---

TACEcv	= "karray" ,
Ccv	= "numeric" ,
Cbcv	= "numeric" ,
nCAAobs	= "numeric" ,
nCALobs	= "numeric" ,
Lcv	= "numeric" ,
Ibeta	= "numeric" ,
Icv	= "numeric" ,
IACin	= "numeric" ,
ITrendin	= "numeric" ,
Mbcv	= "numeric" ,
Kbcv	= "numeric" ,
t0bcv	= "numeric" ,
Linfbcv	= "numeric" ,
LFCbcv	= "numeric" ,
LFSbcv	= "numeric" ,
FMSYbcv	= "numeric" ,
FMSY_Mbcv	= "numeric" ,
BMSY_B0bcv	= "numeric" ,

---

ageMbcv	= "numeric",
Dbcv	= "numeric",
Dcv	= "numeric",
Btbcv	= "numeric",
Btcv	= "numeric",
Ftbcv	= "numeric",
Ftcv	= "numeric",
hbcv	= "numeric",
Recbcv	= "numeric",
IMSYbcv	= "numeric",
MSYbcv	= "numeric",
BMSYbcv	= "numeric"

---

*Table 4. OMss (Operating Model) object class attributes in file Objects-OMss.R (additional attributes may be defined for potential future expansion)*

Slot/attribute	
<b>Identifiers and Reporting specifications</b>	
Name	= "character",
Label	= "character",
Date	= "character",
Author	= "character",
Notes	= "character",
PrimarySource	= "character",
targpop	= "integer",
seed	= "integer",
recentPerFirst	= "integer",
recentPerLast	= "integer",
seasonCEDist	= "integer",
SSRootDir	= "character",
SBlim	= "numeric",
Flim	= "numeric",
OMList	= "list",
OMid	= "karray",
CppMethod	= "numeric",
UseCluster	= "numeric",
<b>Dimensions</b>	
nsim	= "integer",
nages	= "integer",
nyears	= "integer",
nsubyears	= "integer",
nareas	= "integer",
npop	= "integer",
proyears	= "integer",
nsimPerOMFile	= "karray",
nCPUE	= "integer",

---

---

---

**MP details**

---

lastCalendarYr	= "integer",
yrLabels	= "karray",
seasonLabels	= "karray",
yrSeasLabels	= "karray",
firstMPYr	= "numeric",
catchBridge	= "karray",
catchBridgeCV	= "numeric",
MPDataLag	= "numeric",
NInitCV	= "numeric",
NInitCVdecay	= "numeric",
selExpRange	= "numeric",
selAgeRange	= "numeric",
selWLRRange	= "karray",
selTSSign	= "karray",
selTSWaveLen	= "karray",

---

---

**Fishery**

---

---

---

**Biology**

---

SRrel	= "integer",
mat	= "karray",
h	= "karray",
recgrad	= "karray",
ReccvT	= "karray",
RecAC	= "karray",
ReccvR	= "karray",
Linfmu	= "karray",
Kmu	= "karray",
t0	= "numeric",
Ksd	= "karray",
Kgrad	= "karray",
Linfsd	= "karray",
Linfgrad	= "karray",
a	= "numeric",
b	= "numeric",
ageMmu	= "karray",
ageM	= "karray",
ageMsd	= "karray",
ageMgrad	= "karray",
mov	= "karray",
Mmov	= "karray",
movvar	= "karray",
movsd	= "karray",
movgrad	= "karray",
Mmovvar	= "karray",
Mmovsd	= "karray",
Mmovgrad	= "karray",

---

---

**Observation and Implementation errors**

---



---

```

indexFisheries = "integer",

TACEcv         = "karray",
Ccv            = "numeric",
Cbcv          = "numeric",
nCAAobs       = "numeric",
nCALobs       = "numeric",
Lcv           = "numeric",
Ibeta         = "numeric",
Icv           = "numeric",
IAC           = "karray",
ITrend        = "karray",
Mbcv          = "numeric",
Kbcv          = "numeric",
t0bcv         = "numeric",
Linfbcv       = "numeric",
LFCbcv        = "numeric",
LFSbcv        = "numeric",
FMSYbcv       = "numeric",
FMSY_Mbcv     = "numeric",
BMSY_B0bcv    = "numeric",
ageMbcv       = "numeric",
Dbcv          = "numeric",
Dcv           = "numeric",
Btbcv         = "numeric",
Btcv          = "numeric",
Ftbcv         = "numeric",
Ftcv          = "numeric",
hbcv          = "numeric",
Recbcv        = "numeric",
IMSYbcv       = "numeric",
MSYbcv        = "numeric",
BMSYbcv       = "numeric"

```

---

Simulation data

---

```

Recdevs       = "karray",
R0            = "karray",
M             = "karray",
Z             = "karray",
B             = "karray",
SSB           = "karray",
NBeforeInit   = "karray",
C             = "karray",
CAA           = "karray",
MSY           = "numeric"
BMSY          = "numeric"
VBMSY         = "numeric"
SSBMSY        = "numeric"
UMSY          = "numeric"
FMSY1         = "numeric"
SSB0          = "numeric"

```

---

B0	= "numeric"
SSBMSY_SSB0	= "numeric"
IMSY	= "numeric"
Idist	= "karray",
ECurrent	= "karray",
CMCurrent	= "karray",
EByQtrLastYr	= "karray",
Recsubyr	= "integer"
Recdist	= "karray",
initIDev	= "karray",
SSB0ss	= "numeric"
B0ss	= "numeric"
MSYss	= "numeric"
BMSYss	= "numeric"
SSBMSYss	= "numeric"
FMSYss	= "numeric"
F_FMSYss	= "karray",
Frepss	= "karray",
Len_age	= "karray",
Len_age_mid	= "karray",
Wt_age	= "karray",
Wt_age_SB	= "karray",
Wt_age_mid	= "karray",
Css	= "karray",
CAAFss	= "karray",
SSBAss	= "karray",
CBss	= "karray",
Bss	= "karray",
Recss	= "karray",
RecYrQtrss	= "karray",
NLLss	= "karray",
NLLIss	= "karray",
CPUEobsMR	= "karray",
CPUEobsY	= "karray",
CPUEFleetNums	= "karray",
CPUEFleetAreas	= "karray",
Edist	= "karray",

---

Table 5. MSE object class attributes in file MSE\_Source-OMss.R (additional attributes may be defined for potential future expansion).

Slot/attribute	
<b>Identifiers</b>	
OMName	= "character",
Label	= "character",
Date	= "character",
Author	= "character",
Notes	= "character",
PrimarySource	= "character",
seed	= "integer",
CppMethod	= "numeric",
UseCluster	= "numeric",
OMid	= "karray",
SBlim	= "numeric",
Flim	= "numeric",
yrLabels	= "karray",
seasonLabels	= "karray",
yrSeasLabels	= "karray",
firstMPYr	= "numeric",
lastCalendarYr	= "numeric",
catchBridge	= "karray",
catchBridgeCV	= "numeric",
MPDataLag	= "numeric",
<b>Dimensions</b>	
nsim	= "integer",
npop	= "integer",
nages	= "integer",
nyears	= "integer",
nsubyears	= "integer",
nareas	= "integer",
nfleets	= "integer",
proyears	= "integer",
targpop	= "integer",
<b>MP details</b>	
nMPs	= "integer",
MPs	= "character",
<b>Fishery</b>	
CM	= "karray",
CMbyF	= "karray",
C_MSY	= "karray",
B_B0	= "karray",
SSB_SSB0	= "karray",
B_BMSY	= "karray",
SSB_SSBMSY	= "karray",
F_FMSY	= "karray",
B	= "karray",
SSB	= "karray",

---

TAC	= "karray" ,
-----	--------------

---

Management quantities

---

CM	= "karray" ,
CMbyF	= "karray" ,
C_MSY	= "karray" ,
B_B0	= "karray" ,
SSB_SSB0	= "karray" ,
B_BMSY	= "karray" ,
SSB_SSBMSY	= "karray" ,
F_FMSY	= "karray" ,
B	= "karray" ,
SSB	= "karray" ,
TAC	= "karray" ,

---

Observation model

---

Cimp	= "numeric" ,
Cb	= "numeric" ,
Cerr	= "karray" ,
Iimp	= "numeric" ,
Ibeta	= "numeric" ,
Ierr	= "karray" ,
IAC	= "karray" ,
ITrend	= "karray" ,
CPUEerr	= "karray" ,
nCAAobs	= "numeric" ,
nCALobs	= "numeric" ,
Lcv	= "numeric" ,
Mb	= "numeric" ,
Kb	= "numeric" ,
t0b	= "numeric" ,
Linfb	= "numeric" ,
LFCb	= "numeric" ,
LFSb	= "numeric" ,
FMSYb	= "numeric" ,
FMSY_Mb	= "numeric" ,
BMSY_B0b	= "numeric" ,
ageMb	= "numeric" ,
Dimp	= "numeric" ,
Db	= "numeric" ,
Derr	= "karray" ,
Btimp	= "numeric" ,
Btb	= "numeric" ,
Bterr	= "karray" ,
Ftimp	= "numeric" ,
Ftb	= "numeric" ,
Fterr	= "karray" ,
hb	= "numeric" ,
Recbcv	= "numeric" ,
IMSYb	= "numeric" ,
MSYb	= "numeric" ,

---

```
BMSYb          = "numeric",
```

---

## 4 R Classes

### OMd

#### Description

This class forms a container for all the parameters required to uniquely define an operating model within the MSE. It has no explicit initializer. Normal usage involves using the new operator to create a new but empty object instance and then initializing the attributes explicitly. Additional attributes are defined in the OMd file, but not actively used at this time (and may not be completely implemented).

#### Attributes

##### Name

A character string naming the operating model definition. This parameter is for future reference purposes only.

##### Label

A character string labelling the operating model definition. This parameter is used for result presentation purposes.

##### Date

A character string naming the date of authorship of the operating model definition. This parameter is for future reference purposes only.

##### Author

A character string naming the person responsible for the model definition preparation. This parameter is for future reference purposes only.

##### Notes

A character string giving any additional notes pertinent to the model definition. This parameter is for future reference purposes only.

##### PrimarySource

A character string parameter describing the source of the conditioned model data this definition is based on. This parameter is for future reference purposes only.

##### CppMethod

A numeric parameter governing whether to use C++ code for model MSY and management forward projection. 0 = use R code, 1 = use C++ code.

##### UseCluster

A numeric parameter governing whether to use R cluster processing for model MSY and management forward projection. 0 = no cluster processing, 1 = use cluster processing.

### **Npop**

A numeric parameter setting the number of populations in the model definition.

### **Nfleets**

A numeric parameter setting the number of fleets in the model definition.

### **SSRootDir**

A character string parameter containing the path to the folder containing the Stock Synthesis output files that the model will be using.

### **SBlim**

A numeric parameter setting the reference line value for SSB / SSBMSY plots.

### **Flim**

A numeric parameter setting the reference line value for F / FMSY plots.

### **OMList**

A list of Stock Synthesis models to include in the definition. Typically, this will be the list returned from a call to one of the grid creation functions such as `makeGridY3.f()`, or a subset of that list.

### **nsimPerOMFile**

A numeric parameter setting the number of simulations to perform per model in the `OMList`.

### **proyears**

A numeric parameter setting the number of projection years to carry out in the MSE.

### **targpop**

A numeric parameter governing which population/s to collect the MSE performance data for. This can be a vector of more than one population (assuming the model has more than one), in which case performance measures are summed over the named populations (not tested).

### **seed**

A numeric parameter used to seed the R random number generator.

### **recentPerFirst**

A numeric parameter setting the first season to use in calculating the seasonal catch effort pattern to use in forward projection. This is a value in sub-years relative to the final sub-year of assessment data and counted backwards. A value of 0 means the final sub-year.

### **recentPerLast**

A numeric parameter setting the last season to use in calculating the seasonal catch effort pattern to use in forward projection. This is a value in sub-years relative to the final sub-year of assessment data and counted backwards. A value of 0 means the final sub-year. Combined with `recentPerFirst`, it provides a window of sub-years to base an average seasonality on.

### **seasonCEDist**

A numeric parameter governing whether to use seasonal catch effort distribution. 0 = no seasonality (all sub-year catch and effort are equal), 1 = seasonal catch and effort based on the average obtained over the `recentPerFirst` and `recentPerLast` sub-years (must be a multiple of whole years).

### **nsubyears**

A numeric parameter setting the number of subyears in the model definition. For quarterly models this will have a value of 4.

### **lastSeas**

A numeric value representing the last season (subyear) of historic assessment data. This allows assessment data to end mid-year.

### **firstSeas**

A numeric value representing the first season (subyear) of historic assessment data. This allows assessment data to start mid-year.

### **firstSSYr**

A numeric value governing the first year of stock synthesis results to include in the MSE. This value is in *Stock Synthesis* years, which represent quarters in the context of the SS assessment of YFT and BET.

### **firstCalendarYr**

A numeric value representing the first calendar year of historic assessment data.

### **lastCalendarYr**

A numeric value representing the first calendar year of historic assessment data.

### **firstMPYr**

A numeric value representing the calendar year in which MP management begins. The supplied management procedures will be utilised from this year onward.

### **MPDataLag**

A numeric value setting the delay in years between the current year of operation and the last set of assessment data supplied to the management procedure. This parameter accounts for the fact that data collation takes time and assessment results always typically lag the year in which management decisions are to be implemented.

### **catchBridge**

A numeric vector indicating the known (annualised) catch history between the final year of assessment and the beginning of management under MPs. The vector is of length between 0 and  $firstMPYr - lastCalendarYr - 1$ . Years not covered by the catch bridge vector are considered unknown and are assigned catch based on the previous catch, combined with a lognormal random deviate with CV specified by `catchBridgeCV`.

### **catchBridgeCV**

A numeric value setting the CV for the lognormal random deviate used to arrive at the catch for unknown years in the catch bridge. Unknown catch is arrived at by applying the deviate to the last known year up to but not including the year that the management projection starts.

### **indexFisheries**

A numeric vector of the fishery numbers from which (fishery-selected) CPUE-based relative abundance indices are to be calculated. In the initial application, there must be exactly one index per region, and these are combined into an aggregate index for the MP calculations.

### **ReccvTin**

A numeric vector, one entry per population, representing the temporal CV for recruitment aggregated over regions. If a value is negative then the recruitment CV is calculated from the RMSE of the SS outputs.

### **ReccvRin**

A numeric value setting the CV for the lognormal random deviate used to provide variability in recruitment among areas.

### **RecACTin**

A numeric value setting the recruitment temporal auto-correlation for the recruitment process (applied annually, even though recruitment occurs quarterly).

### **NInitCV**

A numeric value setting the CV for the lognormal random deviate applied to the initial age 1 population numbers. The other ages have this CV reduced by an exponential decay factor, `NInitCVdecay`.

### **NInitCVdecay**

A numeric value governing the relationship between the initial population CV at age 1 (`NInitCV`) and the corresponding value at age  $a$ . The relationship is an exponential one described by  $\exp(\text{NInitCV} * (a-1))$ .

### **selExpRange**

A numeric value governing the exponent range of temporal variability in selectivity. The exponent variation is parameterised as a sinusoid with angular frequency governed by `selWLRRange` and amplitude `selExpRange`. The angular frequency is randomly chosen from a uniform distribution whose range is `selWLRRange`.

### **selAgeRange**

A numeric value governing the age range shift of selectivity as a function of time. The age shift in selectivity is parameterised as a sinusoid with angular frequency governed by `selWLRRange` and age shift amplitude `selAgeRange`. The angular frequency is randomly chosen from a uniform distribution whose range is `selWLRRange`.

### **selWLRRange**

A pair of numeric values governing the range of angular frequencies used to parameterise temporal selectivity change. The angular frequency is randomly chosen from a uniform distribution whose range is `selWLRRange`. The angular frequency is related to the period of variation in years by the relationship,

$$\lambda = \frac{2\pi}{T}$$

Where  $\lambda$  is the angular frequency, and  $T$  is the wave period in years (either time or ages depending on context of use).

### **TACEcv**

A numeric vector of length `nfleets` specifying the CV for a lognormal random deviate applied to fleet specific TAC and TAE.



### **Ccv**

A numeric vector of length 2 specifying the range of CV for a lognormal random deviate to be applied to fleet specific catch as observation error. The actual CV for each fleet is determined by sampling a uniform distribution whose range is specified by `Ccv`.

### **Icv**

A numeric vector of length 2 specifying the range of CV for a lognormal random deviate to be applied to fleet specific relative abundance indices. The actual CV for each fleet is determined by sampling a uniform distribution whose range is specified by `Icv`.

### **IACin**

A numeric value setting the autocorrelation in relative abundance lognormal error deviates.

### **Cbcv**

A numeric value specifying the catch bias CV. The fleet specific catch bias is obtained by sampling a lognormal distribution with mean 1, whose CV is `Cbcv`.

### **nCALobs**

A numeric vector of length 2 specifying the range of the number of annual catch at length (CAL) observations. The actual number of observations is determined by sampling a uniform distribution whose range is specified by `nCALobs`.

### **Ibeta**

A numeric vector of length 2 specifying the log range of the hyper-stability parameter. The actual value of the hyper-stability parameter is determined by sampling a uniform distribution whose range is specified by the log of `nCAAobs` and then taking the inverse log (not tested to date).

### **ITrendin**

A numeric value controlling trending in abundance index estimates. A negative value means the trend is extracted from the Stock Synthesis assessment model filename (file-specific index trends assume the single digit following a single `q` in the filename defines the trend). A positive value is the multiplier applied to a 1% per-annum compounded trend. For example if we use 3 for `ITrendin` then the trend in abundance indices is 3% per annum.

## **OMss**

### **Description**

This class creates a suite of operating models according to the model definition found in the `OMd` class instance passed to the initialiser for this class. The initializer creates placeholders for state and result information and initialises the models from data found in the SS assessment output. The `OMss` object instance is then supplied to the initializer of the `MSE` class to implement the management strategy evaluation.

### **Initializer Arguments**

#### **OMd**

Instance of `OMd` object containing the attributes describing the operating models to be implemented. This argument must be supplied.

## Report

A boolean argument specifying whether to output detailed run information on the progress of model initialisation. If not specified it defaults to F (false).

## UseMSYss

A boolean argument specifying whether to use MSY projection statistics from Stock Synthesis. If T (true) then MSY reference points are obtained from Stock Synthesis output. If F (false) the MSY reference points are calculated by performing an MSY projection. If not specified it defaults to F (false).

## MSE

### Description

This class performs an MSE on the supplied suite of operating models using the supplied suite of management procedures. The initializer creates placeholders for the summary statistics and performance results of the MSE projection and fills them by running the projection for all supplied MPs.

### Initializer Arguments

#### OM

Instance of `OMSS` object containing the suite of operating models to be tested. This argument must be supplied.

#### MPs

A string vector naming the MP classes to be used for the MSE. This argument must be specified. At present the following MPs are defined in the `MPs.R` source file:

PT41.100.9, PT41.100.5, PT41.100.2, PT41.100.1, PT42.125.2, PT42.125.5, PT61.75.2, PT61.75.5, PT62.125.2, PT62.125.5, PT82.150.2, PT82.150.5, PT82.100.2, PT82.100.5, IT1.00, IT1.50, IT2.00, IT2.50, IT3.00, IT3.50, CC001, CC050, CC100, CC150, CC200, CC250, CC300, CC350, CC400, CC413, CC450, CE1.0, CE0.01, CE0.1, CE0.25, CE0.5, CE0.75, CE1.5, CE2.0, CE5.0, CC100CE1.0yft, CC50CE1.0bet

#### Interval

A numeric value setting the number of years between MP updates. If not specified this argument defaults to 3.

#### Report

A boolean argument specifying whether to output detailed run information on the progress of MSE projection. If not specified it defaults to F (false).

#### CppMethod

A numeric value specifying whether to force use of the `CppMethod`. Normally this is specified in the model definition but can be over-ridden with this argument. If not specified it defaults to NA in which case it uses the value from the OM definition. If non-zero the C++ method is used and if zero the R method is used.

## UseCluster

A numeric value specifying whether to R cluster processing for MSE forward projection. Normally this is specified in the model definition but can be over-ridden with this argument. If not specified it defaults to NA in which case it uses the value in the OM definition. If non-zero then the R cluster is used and if zero the R cluster is not used.

## EffortCeiling

A numeric value specifying the maximum absolute effort for any given controlled fishery in the MSE projection when running the C++ projection code. In the case of the C++ code there is no penalty applied to high effort level in the Baranov solution objective function other than an upper boundary constraint. This upper limit is controlled by `EffortCeiling`.

## rULim

A numeric value indicating the maximum proportional harvest rate that can be taken in the R catch equation approximation. If the harvest rate required to reach the TAC exceeds this threshold, then the actual harvest rate is derived from a function that returns a value higher than `rULim`, and monotonically increasing toward an asymptote less than 1.0.

## TACTime

A numeric value between 0 and 1 specifying where in the course of a full year the instantaneous catch extraction in the Baranov approximation will occur. This parameter defaults to 0.5 meaning the extraction occurs in the middle of the time-step (Pope's approximation). This parameter has no effect on the C++ method which implements a full solution to the Baranov catch equation.

# 5 Operating Model Equations

## 5.1 Notation

We have attempted to maintain consistency with the Carruthers et al. (2014) Atlantic Bluefin MSE naming conventions and presentation style, to facilitate comparison for users involved with both IOTC and ICCAT.

States, parameters and subscripts are summarized in Table 6. For readability, we often omit subscripts where the context should be self-evident (e.g. maturity is not scripted by year, season and region, because it is usually assumed to be invariant, though that could change in future versions). Subscripts denoting multiple stocks have also been omitted as they are not relevant for any of the demonstration cases to date. When seasonality is not explicitly important, we usually substitute the quarterly time subscript  $t$  to represent the combination of year ( $y$ ) and season ( $s$ ):

$$\left\{ \begin{array}{ll} X_{y,s+1} = f(X_{y,s}) & s < S \\ X_{y+1,1} = f(X_{y,s}) & s = S \end{array} \right\} \rightarrow X_{t+1} = f(X_t).$$

There is also some redundancy in the specification of fisheries and regions. Because each fishery operates in exactly one region, the subscript  $f$  implies a unique region  $r$  (though the reverse is not true, as multiple fisheries can operate in the same region).

We recycle the greek characters  $\tau$  and  $\omega$  to indicate auto-correlated and independent random normal deviates, respectively, and  $\sigma$  for variance-related parameters:

- 1)  $\tau_t = \rho\tau_{t-1} + \omega_t\sqrt{1 - \rho^2}$
- 2)  $\omega \sim \text{Normal}(\mu = 0, \sigma)$ ,

where  $\rho$  is the lag(1) correlation co-efficient (seasonal or annual). The OM was only set up for some processes to use auto-correlated errors (and  $\rho = 0$  can be specified to remove it).

Table 6. Operating model states, parameters, scripts and superscripts used in this document (but not necessarily in the code).

Variables / Parameter	
$N$	Number in population
$SB$	Spawning Biomass
$C$	Catch
$M$	Natural mortality
$M$	Maturity (proportion)
$E$	Fishing effort
$S$	Fishery selectivity
$F$	Fishing mortality
$h$	Beverton-Holt stock-recruit steepness
$P, \phi$	Proportion of a distribution
$T$	A lag(1) auto-correlated random normal deviate
$\Omega$	An independent random normal deviate
$\Sigma$	Variance-related parameter
$W$	Mass-at-age
$L$	Length-at-age
$\alpha, \beta$	Mass-length parameters
$\Psi$	Movement probability
$TAC$	Related to quota-based management
$TAE$	Related to effort-based management
$R$	Recruitment
$R_0$	Virgin Recruitment
$\rho$	Auto-correlation co-efficient
Subscripts / Superscripts (capitals denote the total/final, i.e. $A$ = oldest age class)	
$a, A$	Age-class (quarters)
$l, L$	length-class
$y, Y$	Year
$s, S$	Season (quarters)
$t, T$	Time (simplified representation of combined year and season)
$r, R$	Region
$TAC$	Related to quota-based management
$TAE$	Related to effort-based management
$beforeX$	Before event $X$
$afterX$	After event $X$
$rec$	Recruitment-related
$recent$	Relates to a user-defined recent period
$imp$	Relates to implementation error
$init$	Relates to initial conditions
$obs$	Relates to observation error
$SS$	Relates to Stock Synthesis state/parameter estimates
$vul$	Vulnerable (fishery-selected)
$mass, numbers$	Catch units

## 5.2 Population Dynamics

The following equations describe the dynamics for an individual stochastic realization, conditional on a specific SS scenario within an ensemble of models.

The model follows fairly standard fisheries assumptions for the most part: age-structured with years and quarterly seasons, spatially-structured (optional) with multiple fisheries. The option of describing multiple stocks (independent biology including spawning) has been retained from the ABT MSE for potential future implementation, but this option is not fully tested or supported in the Indian Ocean OM conditioning process at this time. The Indian Ocean MSE has simpler dynamics than the ABT application, in that each fishery is assigned to exactly one area and assumed not to move among areas. Accordingly, there is no attempt to model the fishery effort distribution as a function of the fish population. However, the Indian Ocean model is more complicated in the sense that there is potentially simultaneous catch-based and effort-based management. It is up to each MP to define which fisheries are managed by a catch quota, and which fisheries are managed by effort multipliers. The default assumption is that the annual aggregate catch quota is allocated among the quota-managed fisheries in proportion to the recent observed distribution (defined in the OMs). The fishery-specific quotas are in turn extracted quarterly, in proportion to the recently observed quarterly catch distributions. At this time the OM does not have the capacity to simulate IUU fishing, or effort-creep scenarios, but these scenarios could be explored by imposing effort multiplier time series within MPs.

The initial numbers-at-age,  $N$ , of the population to be projected forward are extracted from the last year of the SS outputs, with some additional user-defined error:

$$3) \quad N_{t=1,a} = N_{t=1,a}^{SS} \exp(\sigma_{init} \exp(d(-a + 2))); a > 1.$$

The parameter  $\sigma_{init}$  controls the magnitude of the error at age 2 (the youngest age imported from SS), and  $d$  describes how fast the error declines with age (i.e. the estimated precision of partially recruited cohorts is usually artificially low because i) most tuna assessments assume stationary selectivity, and ii) recruitment deviations for recent cohorts are often set to zero to avoid dubious (and often inconsequential) estimates.

Recruitment,  $R_t$  (to age 1, corresponding to age 0 in SS notation), is calculated for each year and season, assuming a spatially-aggregated spawning population (at the beginning of the time-step) according to a Beverton-Holt stock-recruitment relationship with auto-correlated log-normal recruitment deviates:

$$4) \quad R_t = \exp\left(\tau_t - \frac{\sigma_{rec}^2}{2}\right) \frac{0.8R_0 h SB_t}{0.2SB_0(1-h) + (h-0.2)SB_t}.$$

The spawning stock biomass,  $SB$ , is the aggregate across all regions at the beginning of the time-step:

$$5) \quad SB_t = \sum_a \sum_r m_{t,a} W_{t,a} N_{t,a,r}$$

where  $m$  is the maturity schedule and  $W_a^{beg}$  is the mass-at-age at the beginning of the time-step, related to length with the standard relationship:

$$6) \quad W_a^{beg} = \beta(L_a^{beg})^\alpha.$$

Note that SS distinguishes between size-at-age at the beginning of the time-step (for spawning purposes), and at the mid-point of the time-step (for catch calculations). This is an unnecessary complication for populations which are assumed to have continuous spawning, however, it was added to minimize inconsistencies between the conditioning and projection code. The aggregate recruitment is partitioned among regions according to a stationary distribution:

$$7) \quad N_{t,a=1,r} = \phi_r R_t$$

Age-specific, season-specific movement occurs instantaneously, after recruitment, but before mortality:

$$8) \quad N_{y,s,a,k}^{afterMove} = \sum_r N_{y,s,a,r}^{beforeMove} \psi_{a,s,r,k}$$

where  $\psi$  is the probability of an individual of age  $a$  moving from region  $r$ , to region  $k$  in season  $s$ . (there was no spatial structure or migration in the BET demonstration case, while the YFT demonstration case had constant migration among seasons).

There are two options for the catch equations. The C++ based projection sub-routine uses the standard Baranov equations (9, 10) to apply continuous natural and fishing mortality ( $M$  and  $F$ ):

$$9) \quad N_{t,a,r}^{afterZ} = N_{t,a,r}^{afterMove} \exp\left(-\left(M_a + \sum_f F_{t,a,r,f}\right)\right),$$

and

$$10) \quad C_{t,a,f}^{Numbers} = \frac{F_{t,a,f}}{M_a + \sum_f F_{t,a,f}} N_{t,a,r}^{afterMove} \left(1 - \exp\left(-\left(M_a + \sum_f F_{t,a,f}\right)\right)\right).$$

An iterative algorithm is required to simultaneously solve for the catch of the effort-managed fleets and the fishing mortality of the quota-managed fleets.

The R-based approach is a hybrid of Baranov and something similar to Pope's approximation, which does not require an iterative solution (described in equations 11-13). Natural mortality and effort-based fishery extraction are first applied up to a point  $\Delta$  within the time-step ( $0 \leq \Delta \leq 1$ ; 0.5 by default):

$$11) \quad N_{t,a,r}^{afterE1} = N_{t,a,r}^{afterMove} \exp\left(-\Delta\left(M_a + \sum_{f \in TAE} F_{t,a,r,f}\right)\right)$$

where  $f \in TAE$  refers to all effort-managed fisheries. This is followed by instantaneous catch extraction for the quota-managed fisheries:

$$12) \quad N_{t,a,r}^{afterC} = N_{t,a,r}^{afterE1} - \sum_{f \in TAC} C_{t,a,r,f}^{Numbers}$$

where  $f \in TAC$  refers to all fisheries that are managed by catch quotas. Finally there is a second continuous mortality event covering the remainder of the time-step:

$$13) \quad N_{t,a,r}^{afterZ} = N_{t,a,r}^{afterC} \exp\left(- (1-\Delta)(M_a + \sum_{f \in TAE} F_{t,a,r,f})\right).$$

Total catch for this latter approach is the sum of the TAC extraction (from 12) and the catch corresponding to each of 11 and 13 which is equivalent to the closed form of equation 9 (i.e. in 11 and 13,  $F$  is a known function of effort for the effort-managed fleets). Note that the dynamics of equations 9-10 and 11-13 are identical if all fisheries are effort-managed.

In equations 9, 11, and 13, fishing mortality is a function of purely age-based selectivity and effective effort, i.e.

$$14) \quad F_{t,a,f} = E_{t,f} S_{t,a,f}$$

where effort is known for the effort-managed fleets. Each fishery is managed by either quota or effort. The MP is required to provide a non-zero TAE (in the form of an effective effort multiplier) for each effort-managed fishery, and an aggregate TAC for the remaining fisheries. The effective effort multiplier assumes that the fishing mortality will increase or decrease relative to a recently observed level  $F^{recent}$ , (defined in the main text for the demonstration cases). This implicitly assumes that fishery catchability is stationary relative to the effort units defined in the MP (note that in recent YFT and BET assessments, this is only assumed to be true for standardized longline effort).

Selectivity for each fishery is represented by a vector of length  $A$ , scaled to a maximum of 1.0 for the most vulnerable age(s). There are two options for introducing temporal variability to the selectivity in projections. The first shifts the baseline selectivity function (imported from SS) toward younger or older ages by an integer amount,  $\eta_f$ , as a function of time:

$$15) \quad \eta_f = \text{round}(\Omega_f \sin(\lambda_f (y-1)))$$

$$16) \quad S_{t,a,f} = \begin{cases} S_{t,a+\eta_f,f}^{SS}; & 1 \leq a + \eta_f \leq A \\ S_{t,A,f}^{SS}; & a + \eta_f > A \\ S_{t,1,f}^{SS}; & a + \eta_f < 1 \end{cases}$$

Where  $\Omega_f$  and  $\lambda_f$  fishery-specific amplitude and angular frequency parameters respectively, and  $y$  is the projection year starting from 1 (i.e.  $y=1$  sets the phase angle so that selectivity is unchanged in the first projection year). An analogous option increases or decreases the degree of targeting in the baseline selectivity:

$$17) \quad \gamma_f = \exp(\Omega_f \sin(\lambda_f (y-1)))$$

$$18) \quad S_{t,a,f} = (S_{t,a,f}^{SS})^{\gamma_f}$$

A value of  $\gamma > 1$  disproportionately increases ( $\gamma < 1$  decreases) the relative vulnerability of the more highly selected age-classes relative to the other age-classes. Separate  $\lambda_f$  and  $\Omega_f$  values are used for each type of selectivity process error. The amplitude  $\Omega_f$  for the temporal variability is user-defined and identical among fisheries, but the sign of the amplitude is random for each fishery (i.e the initial

trend can be in either direction). The angular frequency  $\lambda_f$  is a fishery-specific random sample from a user-defined uniform distribution. The demonstration case parameters in operating models OMyft2 and OMBet1 were selected such that:

- The age-shift amplitude,  $\Omega_f = 1$  results in a maximum selectivity shift of +/- 1 (quarterly) age-class
- The selectivity targeting amplitude exponent,  $\Omega_f = 0.6$ , was arbitrarily selected (i.e. from a subjective judgement that the value is big enough to have an effect, but not too big to render the fishery unidentifiable).
- The uniform distribution for the angular frequency parameters was [0.0625,0.5], such that the lower bound results in a monotonic directional change in the selectivity for a projection period of 26 years or less, while the upper bound results in 2 full oscillations of the selectivity vector within a 26 year projection period.

Figure 2 illustrates the temporal variability in selectivity observed for a purse seine and longline fishery from OMBet1. Note that temporal variability in selectivity was added primarily as a means of ensuring that the information content of unbiased size composition data was not unrealistically informative, however, it also introduces time series error structure to the CPUE series.

The aggregate TAC is distributed among seasons and TAC-managed fisheries according to the "recent" distribution of effort or catch. In the case of the C++ based Baranov equations (9, 10), this is achieved by removing the total annual TAC assuming the recent effort distribution remains constant:

$$19) \quad F_{y,s,f \in TAC} = x F_{f \in TAC,s}^{recent},$$

while  $F_{f \in TAE}$  is provided by the MP. The function minimizer solves for the effort multiplier,  $x$ , that satisfies the catch equations for all 4 quarters simultaneously.

In contrast, the R-based Baranov approximation (equation 12) has a closed form solution. For those fisheries that are managed as part of the quota system, it is assumed that the TAC is apportioned according to the recent (user-defined) catch-in-mass distribution by fishery and season:

$$20) \quad C_{y,s,f}^{mass} = TAC_y \left( \frac{C_{f,s}^{mass,recent}}{\sum_s \sum_{f \in TAC} C_{f,s}^{mass,recent}} \right).$$

The age-specific, region-specific, catch extraction is calculated each season:

$$21) \quad C_{y,s,a,f}^{mass} = \left( \frac{B_{y,s,a,r,f}^{vul}}{\sum_a B_{y,s,a,r,f}^{vul}} \right) C_{y,s,f}^{mass},$$

where  $B^{vul}$ , represents the relative biomass that is vulnerable to each fishery via the selectivity function,  $S$ :

$$22) \quad B_{y,s,a,r,f}^{vul} = N_{t,a,r} S_{a,f} W_a^{mid},$$



The catch in numbers extracted from equation 12 is then:

$$23) \quad C_{t,a,r,f}^{numbers} = \frac{C_{t,a,r,f}^{mass}}{W_a^{mid}}.$$

Lastly, the age-classes and time-steps of both approaches are incremented from the end of each time-step to the beginning of the next time-step in the standard way (including a plus-group accumulator, in which the characteristics of all fish  $\geq$  age  $A$  are assumed to be identical):

$$24) \quad N_{t+1,a+1,r}^{beforeMove} = \begin{cases} N_{t,a,r}^{afterZ} & a < A-1 \\ N_{t,A-1,r}^{afterZ} + N_{t,A,r}^{afterZ} & a = A-1 \end{cases}.$$

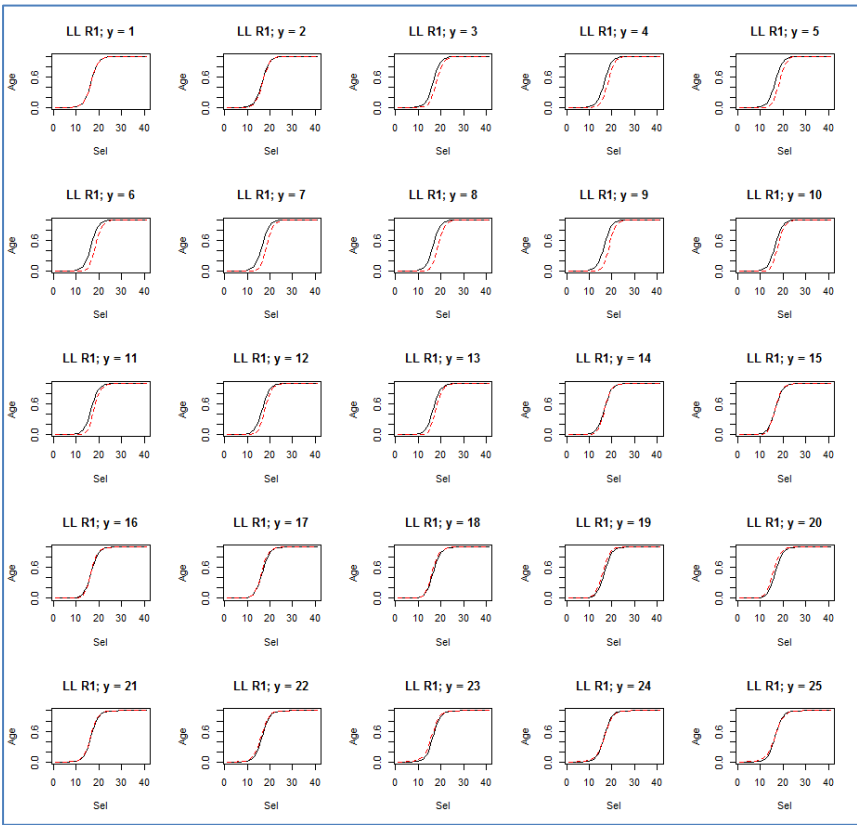


Figure 2. Examples of annual changes in longline selectivity (top panel) and purse seine associated selectivity (bottom panel) over a 25 year projection period for OMBet1. Black line is the stationary selectivity estimated in the SS model, coloured line is the temporally variable selectivity.

### 5.3 Management Implementation Errors

There is independent implementation error on each fishery-specific quota or effort multiplier (applied equally to all seasons within years):

$$25) \quad C_{y,f}^{mass} = TAC_{y,f} \exp\left(\omega_{y,f} - \frac{\sigma_{imp,f}^2}{2}\right)$$

$$26) \quad F_{y,f} = F_{y,f}^{TAE} \exp\left(\omega_{y,f} - \frac{\sigma_{imp,f}^2}{2}\right)$$

To prevent numerical problems when harvest rates are very high, there is an additional constraint imposed on the approximate catch extraction of equation 12, manifested as a modification to the normal harvest rate:

$$27) \quad U_{t,a,r} = \frac{C_{t,a,r}^{Numbers}}{N_{t,a,r}}$$

$$28) \quad U'_{t,a,r} = \begin{cases} U_{t,a,r}, & U_{t,a,r} \leq U_{lim} \\ U_{lim} + (1 - U_{lim} - \kappa)(1 - \exp(-U_{t,a,r} + U_{lim})), & U_{t,a,r} > U_{lim} \end{cases}$$

This constraint is required to prevent the catch equation approximation from extracting more than 100% of the population. Harvest rates  $> U_{lim}$  continue to increase monotonically toward an asymptote of  $1 - \kappa$ . It is under these high harvest rate situations, when the TAC cannot be attained, that the projections from the catch approximation can deviate appreciably from the Baranov equations. The final report was completed with values ( $U_{lim} = 0.5$ ,  $\kappa = 0.12$ , and  $TACTime = 0.5$ ), while it appears that the approximation yields results closer the Baranov equation with ( $U_{lim} = 0.3$ ,  $\kappa = 0.3$ , and  $TACTime = 0.01$ ).

### 5.4 Observation Errors

Currently, the MPs only have access to a subset of relatively aggregated data. Further disaggregation (e.g. all or different data by season and fishery) may eventually prove desirable, but providing these data to the MP without proper consideration of the error structures risks providing unrealistically informative data. e.g. If a relatively high CPUE observation error  $CV = 0.6$  is applied independently each quarter to each of 4 fisheries, the aggregate annual  $CV \sim 0.15$ , which is probably unrealistically optimistic for commercial CPUE.

Annual observed catch-in-mass is reported as the product of catch-in-numbers and mass-at-age, summed over ages, seasons and fisheries (fisheries and regions are redundant), with an aggregate log-normal error:

$$29) \quad C_y^{mass,obs} = \exp\left(\omega_y - \frac{\sigma_{Cobs}^2}{2}\right) \sum_a \sum_s \sum_r \sum_f C_{a,s,r,f} W_a^{mid}$$

Observed catch-at-length frequency distributions,  $X$ , are provided as a multinomial sample from the true catch distributions:

$$30) \quad X_{l,y,f} = \text{Mult}(n, P_{l,y,f}),$$

where  $P$  represents the true proportion of fish caught in length class  $l$ . The sample is unbiased, however, the sample size,  $n$ , should be set much lower than most real catch-at-length sampling programmes to recognize that samples are not truly random. Furthermore, temporally-structured selectivity can be used (eqn. xxx) to ensure that size data are not unrealistically informative.

An annual aggregate relative abundance index is calculated:

$$31) \quad I_y = q \sum_a \sum_s \sum_{rf \in LL} S_{y,a,f} N_{y,a,s,r}$$

where  $rf \in LL$ , indicates the region or fishery (as appropriate) corresponding to the (informative) longline fisheries,  $S$  is the fishery selectivity, and  $q$  is the constant of proportionality calculated from the whole of the conditioning assessment model CPUE time period. The CPUE observation consists of:

$$32) \quad I_y^{obs} = I_y^{\beta \delta y} \exp\left(\tau_y - \frac{1}{2}\sigma_I^2\right)$$

where  $\beta$  is a hyper-stability / hyper-depletion parameter, and  $\delta$  is a temporal catchability trend, and the stochastic error and auto-correlation are applied on an annual basis.

## 6 Candidate Management Procedures

The phase 1 final report includes a qualitative description (cartoon schematic) of the IT and PT MPs described below. The parameter values from the main report refer to the MP equations defined in Table 7.

For the PT MPs, the Pella-Tomlinson model dynamics consist of:

$$33) \quad B_{y+1} = B_y + \left(\frac{p+1}{p}\right) r B_y \left(1 - \left(\frac{B_y}{K}\right)^p\right) - C_y,$$

where  $B$  is aggregate biomass,  $C$  is the total catch in mass,  $r$  and  $K$  are the estimated population growth rate and carrying capacity, and  $p$  is the equilibrium yield "shape" parameter ( $p = B_{MSY}/B_0$ , fixed at 0.33 in the demonstration MPs). The model is fit using the least-squares (observation error) objective function:

$$34) \quad \sum_y (\hat{I}_y - I_y)^2$$

where  $I$  is the aggregate annual CPUE index, and

$$35) \quad \hat{I}_y = \frac{\sum_y I_y}{\sum_y B_y} B_y$$

The HCR requires estimates of  $B_y/B_0$  and

$$36) \quad MSY = \frac{rK}{(p+1)^{p-1}}.$$

Using the R-based fitting algorithm, the model appeared to fit some complicated dynamics very poorly (i.e. presumably due to the inability to describe recruitment variability), but the MP generally provided respectable management performance. We did not explore the frequency with which the base R fitting algorithm, `optim()`, was actually finding the global minimum.

*Table 7. MPs included with the initial IOTC MSE release. Additional control parameters are omitted for the IT and PT MPs, including: 1) MP start year, ii) frequency of MP application, 3) data lag (assumed 0 below).*

---

**CCE (constant catch and/or effort)**

---

**TAC/TAE calculation:**

$$TAE_f = k_f^{TAE}$$

$$TAC = k^{TAC}$$

**Control parameters:**

$k_f^{TAE}$  = constant fishery-specific effort-multiplier; constant for all years,

$k^{TAC}$  = aggregate TAC for fisheries with  $k_f^{TAE} = 0$ ; constant for all years.

---

**IT (aim for CPUE target)**

---

**TAC/TAE calculation:**

$$TAE_f = 0$$

$$TAC_{y+1}^{init} = \begin{cases} TAC_y(1 + k_1m + k_3(\bar{I}_y - I^T)); & m > 0, \bar{I}_y > I^T \\ TAC_y(1 + k_1m + k_4(\bar{I}_y - I^T)); & m > 0, \bar{I}_y < I^T \\ TAC_y(1 + k_2m + k_3(\bar{I}_y - I^T)); & m < 0, \bar{I}_y > I^T \\ TAC_y(1 + k_2m + k_4(\bar{I}_y - I^T)); & m < 0, \bar{I}_y < I^T \end{cases}$$


---

$$TAC_{y+1} = \begin{cases} TAC_{y+1}^{init}; & TAC_{y+1}^{init}/TAC_y - 1 < \Delta^{\max} \text{ and } 1 - TAC_{y+1}^{init}/TAC_y > \Delta^{\min} \\ (1 + \Delta^{\max})TAC_y & TAC_{y+1}^{init}/TAC_y - 1 > \Delta^{\max} \\ (1 - \Delta^{\min})TAC_y & 1 - TAC_{y+1}^{init}/TAC_y < \Delta^{\min} \end{cases}$$

**Data/analysis:**

$I_y$  = stanardized longline CPUE in year  $y$  (aggregated over areas and seasons)

$\bar{I}_y = 0.5I_y + 0.33I_{y-1} + 0.17I_{y-2}$  (weighted mean CPUE of last 3 years to reduce volatility)

$m$  = CPUE trend (linear regression slope from  $I_{y-4}$  to  $I_y$ )

**Control parameters:**

$I^T$  = target CPUE

$k_1...k_4$  = responsiveness (gain) parameters

$\Delta^{\min}, \Delta^{\max}$  = maximum change constraints

PT (Pella Tomlinson with 40:10 type HCR)

**TAC/TAE calculation:**

$$TAE_f = 0$$

$$TAC_{y+1}^{init} = \begin{cases} 0 & , D_{PT} \leq k_1 \\ \frac{k_3 MSY_{PT}}{k_2 - k_1} (D_{PT} - k_1), & k_1 < D_{PT} < k_2 \\ k_3 MSY_{PT} & , D_{PT} \geq k_2 \end{cases}$$

$$TAC_{y+1} = \begin{cases} TAC_{y+1}^{init}; & TAC_{y+1}^{init}/TAC_y - 1 < \Delta^{\max} \text{ and } 1 - TAC_{y+1}^{init}/TAC_y > \Delta^{\min} \\ (1 + \Delta^{\max})TAC_y & TAC_{y+1}^{init}/TAC_y - 1 > \Delta^{\max} \\ (1 - \Delta^{\min})TAC_y & 1 - TAC_{y+1}^{init}/TAC_y < \Delta^{\min} \end{cases}$$

**Data/analysis:**

$D_{PT}$  = Current depletion estimated from fitting Pella-Tomlinson (Observation error) model to catch and CPUE

$MSY_{PT}$  = MSY estimate from production model

**Control parameters:**

$k_1...k_3$  = HCR function modifiers

$\Delta^{\min}, \Delta^{\max}$  = maximum change constraints

## 7 Performance Measures and Reference Points

The software reports a standard set of management performance indicators as proposed by the IOTC Working Party on Methods, endorsed by the Scientific Committee (SC 2015), and subsequently refined by the WPM informal sub-group on MSE. Example plots are shown in the phase 1 report.

The IOTC WPM informal MSE working group (April 2016) proposes:

*Time series will be calculated over projection windows of 1, 5, 10 and 20 years, where year 1 is the first year that a TAC or TAE is applied (i.e. as opposed to the first year of projections which might be based on a known or assumed catch because of data and decision process time lags). While it is recognized that MSE is intended to look at medium to longer term performance, one year is included because it is inevitable that industry stakeholders will want to know what the implications of adopting an MP will be in the first year (and they may be particularly disruptive in a rebuilding situation).*

*The performance indicators described in Table [A1] are calculated for each stochastic realization, and then presented as percentiles (10, 25, 50, 75 and 90%) from the distribution of all realizations*

*It was noted that there is currently an inconsistency in the [Commission] identified management objectives, in that achieving target reference points of  $F_{MSY}$  and  $B_{MSY}$  with near perfect precision would correspond to roughly equal probabilities of being in the green and red Kobe quadrants, while a high probability of being in the green quadrant implies  $F < F_{MSY}$  and  $B > B_{MSY}$ .*

Standard graphical outputs are described below (and illustrated elsewhere in the user manual and phase 1 project report):

*Time series (quantiles plus some individual iterations) plots will be used to describe key MSE outputs, including  $SB/SB_{MSY}$  ( $SB/SB_0$ ,  $B/B_{MSY}$ ,  $B/B_0$ ), Catch,  $F/F_{MSY}$ , Recruitment and/or CPUE:*

- *Interim reference point reference lines should be included (green target, red limit) when appropriate.*
- *Plots are to indicate the median with a line, and the 25-75th and 10-90th percentiles with shaded ribbons.*
- *When appropriate, 3 individual realizations should be plotted on top, corresponding to the 25, 50, 75th percentiles of  $SB/SB_{MSY}$  (or  $SB/SB_0$ ) over the 20 year projection period. These same three individual realizations should then be plotted in all relevant time series plots (i.e. irrespective of which percentile the realization corresponds to in the other plots)*

*Four core trade-off plots, computed for each of 10 and 20 years of projection (i.e. year 1 = first TAC/TAE implementation)*

1.  *$SB/SB_{MSY}$  (or  $SB/SB_0$  for skipjack) vs. Yield*
2. *Pr(Green Kobe) vs. Yield*
3. *Pr( $SB > B_{Lim}$ ) vs. Yield*
4. *mean( $1 - C_y/C_{y-1}$ ) vs. Yield*

*Confidence interval plots (double whisker aka udon-soba plots). The summary statistics from [Table xxx] will be summarized graphically by their median, thick confidence interval whiskers for the 25-75th percentiles, and thin whiskers indicating 10-90th percentiles. These plots can compare several MPs for a single performance statistic*

*within a single panel and can pack a lot of information into a small space, but they are less convenient for identifying broad patterns than the other plot types.*

MSY-based reference points are calculated assuming the same "recent" effort distribution (by fishery and season) used for TAC partitioning.

Annual aggregate fishing mortality for reporting purposes is calculated using the SS approach, in which  $Z$  is first calculated on a spatially aggregated basis, and  $F = Z - M$  :

$$37) \quad \bar{F}_y = \frac{1}{AS} \sum_a \sum_s \left( -\log_e \left( \frac{\sum_r N_{y,s+1,a+1,r}}{\sum_r N_{y,s,a,r}} \right) - M_a \right)$$

where  $s$  consists of the 4 seasons within year  $y$  (and strictly speaking the numerator in the final element of the summation above would be  $N_{y+1,s=1,a+1,r}$ ).



Table 8. Performance measures (the interim target and limit reference points from Resolution 15/10 are shown for yellowfin and bigeye only; they differ for other IOTC species;  $F_{limit} = 1.4F_{MSY}$ ,  $F_{target} = F_{MSY}$ ,  $SB_{limit} = 0.4SB_{MSY}$ ,  $SB_{target} = SB_{MSY}$ )

<b>Status : maximize stock status</b>		
1. Mean spawner biomass relative to pristine	$SB/SB_0$	Arithmetic mean over years
2. Minimum spawner biomass relative to pristine	$SB/SB_0$	Minimum over years
3. Mean spawner biomass relative to $SB_{MSY}$	$SB/SB_{MSY}$	Arithmetic mean over years
4. Mean fishing mortality relative to target	$F/F_{tar}$	Arithmetic mean over years
5. Mean fishing mortality relative to $F_{msy}$	$F/F_{MSY}$	Arithmetic mean over years
6. Probability of being in Kobe green quadrant	$SB, F$	Proportion of years that $SB \geq SB_{tar}$ and $F \leq F_{tar}$
7. Probability of being in Kobe red quadrant	$SB, F$	Proportion of years that $SB < SB_{tar}$ and $F > F_{tar}$
<b>Safety : maximize the probability of remaining above low stock status (i.e. minimize risk)</b>		
8. Probability of spawner biomass being above 20% of $SB_0$	$SB$	Proportion of years that $SB > 0.2B_0$
9. Probability of spawner biomass being above $B_{Lim} = 0.4SB_{MSY}$	$SB$	Proportion of years that $SB > 0.4SB_{MSY}$
<b>Yield : maximize catches across regions and gears</b>		
10. Mean catch	$C$	Arithmetic mean over years
11. Mean catch by region and/or gear	$C$	Arithmetic mean over years
12. Mean catch relative to MSY	$C/MSY$	Arithmetic mean over years
<b>Abundance: maximize catch rates to enhance fishery profitability</b>		
13. Mean catch rates by region and gear (for fisheries with meaningful catch-effort relationship)	$I$	Arithmetic mean over years
<b>Stability: maximize stability in catches to reduce commercial uncertainty</b>		
14. Mean absolute proportional change in catch	$C$	Arithmetic mean over years of $\text{abs}(1 - C_t/C_{t-1})$
15. Variance in catch	$C$	Variance over years
16. Probability of shutdown	$C$	Proportion of years that $C < 0.1MSY$

## 8 Acknowledgements

The first iteration of this work (completed June 2016) was jointly funded by the IOTC (through the European Union and Indian Ocean Commission) and CSIRO. The R-based MSE code was adapted from the Atlantic Bluefin MSE work developed by Tom Carruthers (and funded by the ICCAT GBYP project). Technical oversight and advice was provided by the IOTC Working Party on Methods, notably Iago Mosqueira, Toshihide Kitakado, Nokome Bentley, Gerald Scott, Gorka Merino, Hilario Murua and Rishi Sharma. Adam Langley and Ian Taylor provided helpful insight on the inner workings of Stock Synthesis. Some of the summary result graphics were adopted from the Fisheries Library in R (FLR) code provided by Iago Mosqueira.

## 9 References

- Carruthers, T, Kell, L, Davies, C. 2014. Evaluating management strategies for Atlantic bluefin tuna. Report 1 – contract report to support to BFT assessment (GBYP 02/2014) of the Atlantic-wide research programme on bluefin tuna (ICCAT GBYP– Phase 4).
- Liu, X, Heino, M. 2014. Overlooked biological and economic implications of within-season fishery dynamics. *Can. J. Fish. Aquat. Sci.* **71**: 181-188 (2014) [dx.doi.org/10.1139/cjfas-2013-0029](https://doi.org/10.1139/cjfas-2013-0029).
- Methot, R.D., Wetzel, C.R., 2013. Stock synthesis: A biological and statistical framework for fish stock assessment and fishery management. *Fish. Res.* 142: 86– 99.
- R Core Team (2015). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL <https://www.R-project.org/>.