SCIENTIFIC COMMITTEE

NINETEENTH REGULAR SESSION

Koror, Palau

16-24 August 2023

DEVELOPMENTS IN THE MULTIFAN-CL SOFTWARE 2022-23

WCPFC-SC19-2023/SA-IP-02

Nick Davies[3], David Fournier[2], Fabrice Bouyé[1], John Hampton[1] and Arni Magnusson[1]

Oceanic Fisheries Programme

The Pacific Community, Nouméa, New Caledonia

---

[1] Oceanic Fisheries Programme, The Pacific Community, Nouméa, New Caledonia

[2] Otter Research Ltd, North Saanich, British Columbia, Canada

[3] Te Takina Ltd, Whangarei, New Zealand

Table of Contents

EXECUTIVE SUMMARY

This paper summarises the developments made within the MULTIFAN-CL software project as carried out by the team at the Oceanic Fisheries Programme (OFP, The Pacific Community, Noumea, New Caledonia) from August 2022 to July 2023, and updates the report of Davies et al. (2022).

The current version 2.1.0.0 has been benchmark-tested and is available for wider distribution. The development version 2.2.3.0 has been tested for the specific developments to existing features, and has been employed within the OFP, Pacific Community, for undertaking the 2023 tuna stock assessments.

An important enhancement of an existing feature that was essential for providing stock assessment advice was that to extend the catch-conditioned feature to perform stochastic projections; having future fisheries conditioned in respect of either catch or effort. Significant in this development was the assignment of stochastic recruitments in respect of: random deviates of the Beverton-Holt stock recruitment relationship; or, random samples of the absolute historical estimates. This development with subsequent testing was essential for undertaking the risk analysis of future managements strategies based upon the 2022 skipjack tuna stock assessment, and as presented to the fisheries review at the WCPFC19 meeting.

Cumulatively, enhancements to existing features have been notably advantageous to the 2023 assessment modelling, and include:

- Allowing for grouped index fisheries that share the same stationary catchability in the CPUE likelihood, but to have non-shared penalty weights;
- Employing the default parameter scaling settings for key parameters as they were in version 2.0.7.3;
- Allowing for fixed Lorenzen parameter values, assumed initial values with which to start estimation, and improved parameter scaling; and,
- An optional abbreviation of the derivatives calculation of the dependent variables, being most convenient for application to models comprising the uncertainty grid.

An improvement to MULTIFAN-CL was the correction made to the calculation of the standard deviations of von Bertalanffy mean lengths-at-age, as used for deriving mean weight-at-age and applied to predicted weight-frequencies.

The primary existing feature that was consolidated in 2022-23 was the estimation of length-based selectivities. Developed originally in 2008-09, the method was thoroughly reviewed and re-instated. It holds promise in that fitted size compositions are comparable with that of the “standard” age-based selectivities, however, certain anomalies identified with the method must first be resolved before it can be considered a real improvement over the age-based estimates.

Setbacks and obstacles encountered in completing the tasks set out in the 2022-23 workplan, included introducing to the workplan in October 2022 the recommendations of an independent peer review panel (Punt et al. 2023). Consequently, immediate priority was assigned to re-instating the length-based selectivity feature and other tasks were duly set aside. An operational setback that stopped production of the macOS executable, was the Mini Mac PC becoming unavailable to the MULTIFAN-CL project due to cybersecurity concerns in the Pacific Community computing network. A task has been added to the 2023-24 workplan to explore an arrangement that will restore this facet of the project.

Substantially fewer project resources (due to Dr Fournier’s retirement in Dec. 2021), and lower assigned priority to maintaining the project’s support structures (because of the importance of preparing features for production assessments), means this area continues to be neglected. Strategic planning is urgently needed to address this shortage of resources in the project.

Neither on-site, nor on-line, training workshops were held for stock assessment analysts. Rather, more resources were directed towards user support in 2022-23 compared to previous years; with a reasonably large commitment to assisting with the bigeye and yellowfin 2023 stock assessments.

The focus of the 2023-24 workplan, similar to last year, is to consolidate recent new features, enhance existing features, improve processes and reporting, make corrections, and update support documentation. No substantial new developments to the project are proposed. Key elements of the plan, (order of priority yet to be determined), include:

- Consolidate the catch-conditioned method for:
    - fishery-specific penalties on the fml_effort_rltnshp,
    - add the fml_effort_rltnshp penalty to the likelihood components report,
    - testing of the control phases, and
    - applying the method to multi-species, multi-sexes.
- CPUE log-normal likelihood
    - Complete the concentrated formulation with normalised λ_i
- Recommendations of the Yellowfin Tuna Independent Peer Review Panel 2022
    - specify CPUE overdispersion as an additive rather than multiplicative factor.
    - Take into account variability in weight-at-length,
    - Specify the number of spline knots when defining selectivity and where they are located with respect to age (length),
    - Account for age-reading error when fitting to conditional age-at-length data, and
    - Integrate the calculation of M-at-age from the sex-ratio data.
- Update the User's Guide for the catch-conditioned method, and other recently added features not yet documented.
- Tidy the code, complete a backlog of bug fixes, and any "small-scale" requests in the tasks list.
- Provide support for MSE requirements and improvements.
- Training and stock assessment support for OFP.

Some of the proposed items from the 2022-23 plan will be retained in the MULTIFAN-CL development list, and carried over into the 2023-24 workplan.

1 INTRODUCTION

MULTIFAN-CL is a statistical, age-structured, length-based model routinely used for stock assessments of tuna and other pelagic species. The model was originally developed by Dr David Fournier (Otter Research Ltd) and Dr John Hampton (The Pacific Community) for its initial application to South Pacific albacore tuna (Fournier et al. 1998). It has since provided the basis for undertaking stock assessments in the Western and Central Pacific Ocean.

The MULTIFAN-CL model is described in detail in the User's Guide (Kleiber et al. 2018). It is typically fitted to total catch, catch rate, size-frequency and tagging data stratified by fishery, region and time period. For example, recent tuna and billfish assessments (e.g., Day et al. 2023) encompass long time periods, e.g., 1952 to 2019 in quarterly time steps, and model multiple separate fisheries occurring in up to 9 spatial regions. The main parameters estimated by the model include: initial numbers-at-age in each region (usually constrained by an equilibrium age-structure assumption), the number in age class 1 for each quarter in each region (the recruitment), growth parameters, natural mortality-at-age (if estimated), movement, selectivity-at-age by fishery (constrained by smoothing penalties or splines), catch (unless using the catch-conditioned catch equation), effort deviations (random variations in the effort-fishing mortality relationship) for each fishery, initial catchability, and catchability deviations (cumulative changes in catchability with time) for each fishery (if estimated). Parameters are estimated by fitting to a composite (integrated) likelihood comprised of the fits to the various data types, and penalized likelihood distributions for various parameters.

Each year the MULTIFAN-CL development team works to improve the model to accommodate changes in our understanding of the fishery, to fix software errors, and to improve model features and usability. This document records changes made since August 2022 to the software and other components of the MULTIFAN-CL project both for the current release version (2.0.8.7), and the current unreleased development version, and updates the report for the previous period, 2021-22 (Davies et al. 2022).

2 DEVELOPMENT OVERVIEW

2.1 Team

The senior developer of MULTIFAN-CL until December 2021 was Dr David Fournier, of Otter Research Ltd, (Canada), who has since retired. Development and testing are now undertaken by Mr Nick Davies. Other tasks include testing and debugging (ND, John Hampton, and Fabrice Bouyé (SPC)); documentation (ND); and planning and coordination (ND, Paul Hamer and JH). Related support project software is developed or managed by FB (MULTIFAN-CL Viewer, Condor, GitHub, Jenkins), Arni Magnusson, ND, and Robert Scott (R4MFCL, FLR4MFCL).

2.2 Calendar

In the absence of a second developer, developer's workshops are no longer held, and the calendar year is less structured. However, broad periods may be identified with those for which more support is given to OFP stock assessment modelling.

August – November: Consolidating recent developments, benchmark testing, developments required for risk analyses

December – February: Code development, and testing

March – April: Training, stock assessment support, code development, and testing

May – July: Code development, and testing

2.3 Collaboration and versioning

The repository and overall development are coordinated via the GitHub website on GitHub.com at https://github.com/PacificCommunity/ofp-sam-mfcl which is administered by Fabrice Bouye (fabriceb@spc.int) (section 2.4.7).

Problems with MULTIFAN-CL operation or compilation have been reported to the project management website so as to maintain a list of desired enhancements, and to allocate tasks among the project team. A "master" branch exists for the MULTIFAN-CL source code from which release versions are posted, and development branches ("ongoing-dev", "mac-dev") have been created for holding development versions of the source undergoing development and testing. A formal testing procedure has been designed before source code is merged from the branch to the trunk, and a manual for the testing of new compilations, standardizing the source code compilation procedure, and posting of executables is maintained.

2.4 Compilation framework and Source code repository

2.4.1 Compilation framework

A continuous integration facility allows for automatic nightly compilations of the MULTIFAN-CL source on the GitHub repository "master" branch. This automation is done using the software called Jenkins (https://jenkins-ci.org/), an Open-Source continuous integration tool that comes bundled with a web server used for administration. This software is now installed on a Linux Virtual Machine (VM) that is dedicated to MULTIFAN-CL development, and administers the compilations over the OFP network.

In this tool, we've added a custom scheduled task that automatically retrieves the MULTIFAN-CL source code out of the GitHub code repository (master branch); it also retrieves required libraries for the compilation. When done, our task compiles both debug and optimized versions of the software. We've also configured this task to produce code documentation out of the source code and to run some C++ code quality checking.

Doing a nightly compilation allows us to find out more quickly whether issues have been included in the source code repository without being solved by the developer. It also helps us identify issues in the makefile configurations that may prevent the compilation of MULTIFAN-CL on some more neutral environment (i.e.: on a machine that is different from the one of the developer's).

During 2018-19 this facility was extended to support automated builds of the Windows (Visual Studio 2019) and the macOS release executables. The Windows10 VM used for undertaking the benchmark testing framework (see section 2.6) and the Mac Mini provides the platform for undertaking the routine compilation administered by Jenkins (see section 2.4.4). These automated builds were maintained throughout 2022-23, apart from a problem encountered with accessing the Mac Mini since October 2022 (see section 2.4.4).

It is also intended to add to the Jenkins tool the running of automated tests using example fish model data, and, in the future, unit tests for the software.

A directory structure on the dedicated VM was used that is mirrored on all the developer's platforms in respect of source code **Projects/**, associated libraries **libs/**, and **Testing/** directories. This ensures portability of source and makefiles among the developers and the automated build software.

2.4.2 Compilation of dependent libraries

For compilation of the dependent **OpenBLAS** library, the "dynamic architecture feature" is included to the routine compilations that builds several kernels for various processor types, and allows selection of the appropriate kernel at run-time. This may avoid the case where a MULTIFAN-CL executable that was compiled with OpenBLAS on a platform having a very recent processor, fails upon execution because function calls to the OpenBLAS library are attempted on platforms having relatively older processors. This compilation method results in a substantial increase (22 MB) in the executable size. However, it was noted that OpenBLAS libraries are important for the calculations used for the eigenvalues and eigenvectors of the Hessian and also aspects of the self-scaling size composition likelihood. This trade-off is therefore considered acceptable for the increased utility achieved.

In order for the MULTIFAN-CL project to be completely portable, three shell scripts automate the compilation of all the dependent libraries, before compiling MULTIFAN-CL. These scripts apply different options for OpenBLAS, QD and compilation flags for MULTIFAN-CL. The script "build_openblas4mfcl.sh" builds 3 options of this library: "default", "generic", and "dynamic", where the "dynamic architecture feature" builds several kernels for various processor types, and allows them to be selected at run-time. Similarly, the script "build_qd4mfcl.sh" builds 4 options of the QD library: "default", "O3", "O3fma", and "native". Given the various combinations of compilation options among the dependent libraries, ADMB and MULTIFAN-CL, compilations of 25 different executables may be produced. For a single option, it compiles in total: 49 minutes 5 seconds. This facilitates the portability of the entire MULTIFAN-CL compilation project including the dependent libraries, such that the complete project may be constructed and compiled with one step.

It is now possible to include in the automated compilation administered by Jenkins, compilation of the dependent libraries QD and OpenBLAS. With the exception of a couple of manual steps required to configure particular options, the integrated compilation of the entire project is now undertaken within the Jenkins routine compilation procedure.

2.4.3 Compilation of Linux executable

In September 2022 the Linux version used for compilations was upgraded from Ubuntu 18.04 to 20.04, with the gcc compiler version upgraded from 7.3.0. to 9.4.0. No changes were required to the source code for the new compiler.

2.4.4 Compilation of Mac OS executable

During 2018-19, routine macOS compilations of the "master" and "development" branches were added to the compilation framework of the MULTIFAN-CL project. Compilations are done on the MULTIFAN-CL Testing PC (Mac Mini) that has the macOS Mojave installed ("macOS 10.14.6 Mojave").

The most recent version compiled on the Mac Mini was version 2.0.8.5 in May 2022. The PC has two compilation directories:

- Local compilation - stand-alone directory for testing development versions
- Jenkins compilation - for routine automated Jenkins compilations of the master branch (checked out from the repository, see section 2.4.1)

A Software ID certificate was assigned to the macOS compilation using an Apple Developer ID certificate for SPC. The macOS version is signed "Developer ID Application: The Pacific Community" issued by Apple. No differences were detected in the computations or performance among the signed and un-signed compilations.

Apple has now officially announced that they are ceasing production with Intel's CPUs in favour of their own RISC ARM-based CPUs. The macOS compilation of MULTIFAN-CL will ultimately need to accommodate this change. A draft strategy for changing to the ARM-mac compilation for MULTIFAN-CL may entail:

- Following the 2021 Apple conference decide on a machine purchase to be made during 2022
- Explore the potential for compiling on the existing Intel-mac with output target set to the new ARM-CPU using flags in the most up-to-date or the next version of XCode (the Apple dev tools/compiler)
- Maintain a careful watch of the capability of the Rosetta 2 emulator for running the MULTIFAN-CL executable (compiled for the Intel CPU) on an ARM-mac; this could offer a "breathing space" for our switch to the ARM-mac compilation
- Consider the merits of upgrading the mac Mini from Mojave to Catalina or Big Sur
- Potentially consider the lead developers purchasing an ARM-mac (external of SPC) for developing the compilation
- Aim for making the switch to the ARM-mac compilation for MULTIFAN-CL in mid-late 2022

In October 2022, a major setback occurred in respect of the macOS compilations. The MULTIFAN-CL Testing PC (Mac Mini) was removed from the Pacific Community computing network for security reasons. A secure arrangement suitable for the network's standards to reinstate this PC has not yet been developed. Consequently, no compilations for the macOS executable can currently be produced. Finding a solution for this is proposed in the workplan for 2023-24 (section 7).

2.4.5 Visual Studio 2019 Windows compilation

It was determined that the lead developer was compiling the Windows executable using Visual Studio 2017, whereas the preferred compilation would use Visual Studio 2019 (VS2019). The upgrade was completed in March 2023 on the developer's workstation successfully.

Two important differences were identified in compiling the MULTIFAN-CL project with the VS2019 compiler.

1. Casting as char*

Whereas the Linux gcc compiler tolerates casting a character formal argument as, for example: int save_identifier_string(char* str), this produced a compilation error from the VS2019 cl compiler. It was necessary to specifically cast the character as a constant, as in: void verify_identifier_string(const char * s). The necessary changes affected 32 source code files, and impacted on 7 source code files in the ADMB dependent library. Upgraded compilations of the ADMB and QD dependent libraries were completed using VS2019. These changes were compatible with the Linux gcc compiler.

2. Compilation option for floating point precision

Testing of the Windows executable compiled with VS2019 produced a run-time error such that the double-double Newton-Raphson calculations for the catch-conditioned fishing mortality estimation failed to achieve convergence to the required threshold. This was diagnosed to be due to normal double precision being applied in these calculations rather than the double-double precision implemented for these routines that use the QD library. The cause of this was found to be due to a change in the compiler options, such that **fp:fast** compiler option used previous for VS2017 compilations that optimise floating point calculations, now the optimisation allows for the transforming formula that undermines the QD methodology for double-double floating point calculations. Restoring the VS2019 compiler option for floating point maths calculations to **fp:precise** reinstated the double-double precision in the Newton-Raphson routines.

MULTIFAN-CL versions 2.1.1.0 and greater include the above two changes to the VS2019 compilation of the Windows executable.

2.4.6 Development version

Upon completing benchmark testing of a development version, the source code in the repository development branch is merged to the master branch and tagged with a release version number. At this point the development branch is created afresh for implementing any subsequent code developments, and a new compilation directory created. Other points where a new development version number is assigned is immediately following changes that may impact upon a minimised model solution, or alterations to the format of output files. These are then added to the development branch following preliminary testing, and tagged with the new version number. During 2022-23 a new development version was created following the benchmark testing of version 2.0.8.7.

The main developments currently implemented in the development version since version 2.1.0.0 (April 2023) that have not yet been benchmark tested include:

- Survey Fishery CPUE likelihood Grouped catchabilities with varying assumed sigma, and the revised concentrated likelihood – with assumed sigma and normalised lambdas
- Abbreviated derivatives calculation of the dependent variables, and new MSY-related quantities
- Lorenzen initial parameter values and parameter scaling
- Growth – standard deviation of von Bertalanffy mean length-at-age
- Likelihood components report - correction to Dirichlet-Multinomial for size compositions

While not benchmark tested, detailed testing of the effects specific to each development has been completed to confirm it has not impacted on other existing features employed by the examples used in the 2023 stock assessments.

2.4.7 Source code repository

The MFCL project is hosted on GitHub.com at:

- https://github.com/PacificCommunity/ofp-sam-mfcl

This site is only accessible to registered members of the OFP-SAM team. In order to better coordinate developments within components of the project, separate repositories were created for the:

- User's Guide: https://github.com/PacificCommunity/ofp-sam-mfcl-manual
- ADMB dependent library: https://github.com/PacificCommunity/ofp-sam-admb

The branches of the repository are managed such that following benchmark testing, the development version that has tested positive and held in either of the "mac-dev" or "ongoing-dev" branches, is then merged to the "master" branch. This creates a clear node in the "master" branch tagged as being the next release version. At that point a new development version is created in one of the "mac-dev" or "ongoing-dev" branches for undertaking the next phase of developments. This approach was followed for each of the versions during 2022-23, with the current development version being maintained in the "ongoing-dev" branch (version 2.2.3.0).

Between 29 Jul. 2022 and 2 Aug. 2023, a total of 31 source code commits were made to the master and development branches (63 source files have been modified, see section 12), including a merge to the master branch on 25 Nov. 2022 for the distribution of version 2.0.8.7 to the Pacific Community. The current version in the "ongoing-dev" development branch is version 2.2.3.0.

2.5 Developer's workshops

In the absence of a second developer, no workshops were held during 2022-23. Since January 2022, Mr. Nick Davies has continued alone with the consolidation, enhancement and corrections to the existing features.

2.6 Benchmark testing during 2022-23

The benchmark testing framework is described in section 2.9.2, and two sets of benchmark tests were undertaken in 2022-23. A brief description of the tests, and the features tested, is provided in this section.

2.6.1 Version 2.0.8.7

In October-November 2023, comprehensive benchmark testing was done between the MULTIFAN-CL development version, and the benchmark release version 2.0.8.5 previously tested in May 2022. A complete set of tests were undertaken, to examine the cumulative effect of a number of enhancements and corrections made to the development version source code since May 2022, as described in section 5 and by Davies et al. (2022), primarily including:

- **Catch-conditioned method –** Corrections to the initial equilibrium population as obtained using the catch-conditioned method for fishing mortality estimation, and to that calculated under the zero-fishing mortality model evaluation. Implementation for deterministic projections with effort-conditioned fisheries and stochastic projections with both the random: BH-SRR deviates; or, absolute recruitment options. Correction to the assignments to the BH-SRR deviates variable.
- **Zero-fishing mortality model evaluation (F0eval)** – For the case of N_init derived using natural mortality only, the multiplier is disabled by default for the F0eval. Implementation of the recruitment compensation option with respect to the unfished adult biomass for the catch-conditioned method. Correction to zeroing out of log-effort for the initial survival rate calculation.
- **Variance and correlation calculation of dependent variables** – Correction to the F0eval dependent variables being equal to those of the normal evaluation in the *.var report. Correlation report includes the F0eval dependent variables. Correction to the depletion ratio dependent variables in the correlation report. Added the SB_{recent} depletion ratio dependent variable in the variance and correlation reports, (*.var and *.cor), and added log-scale MSY-related dependent variables in the correlation report.
- **Long-term tag loss** - Implementation for the catch-conditioned and catch-errors methods, with its application during the mixing periods.
- **CPUE likelihood** – Observed and predicted variables included in the plot.rep report. Corrections to the ungrouped CPUE likelihood penalty weight, and to the formulation of the concentrated CPUE likelihood.
- **Catch-errors projections** – Corrections to the stochastic projections using the random absolute recruitments option.
- **Orthogonal-polynomial recruitments** – Correction to make robust to projection scenario, and to exclude the first year-effect coefficient from the normal prior penalty calculation.

Note: all the benchmark testing was done using the compilation of the development version for the "standard" 64-bit precision, as this ensures comparability with the benchmark version 2.0.8.5. The range of testing data sets was the same as for the previous benchmark test, comprising: 7 single species sets; a multi-species set; a multi-sex set; a single-species deterministic projection set; and, a single-species stochastic projection set. The single species test data was updated to include the **YFT2020** and **SKJ2022** (a catch-conditioned model) examples, and the older STM2012 and ALB2015 examples were discarded.

Single evaluation tests for single-species, multi-species, multi-sex data, and deterministic single species projection, with or without gradient calculations and a minimisation step – produced identical model quantities among versions.

Doitall fits of single species data – negligible differences (<1%) in key model quantities, viz. SB_{curr}/SB_{currF0}, for six of the test data sets, with minor differences (<4%) for the YFT2020 set.

Doitall fits of multi species data – minor differences (<2.5%) for the Linux executable, but identical model quantities for the Windows executable. Doitall fits of multi-sex data – for the Linux executable multiple minima for this test data was indicated, that produced different solutions among versions, resulting in substantial differences in model quantities (up to 30%) among versions. Cross-validation using the respective solutions among the versions demonstrated this to be a minimisation issue due to floating point differences on the host machines. Identical model quantities were obtained for the Windows executable.

Testing of stochastic projections revealed a correction required to the recruitment assignments that is described in section 5.1.4. The effect of this correction is evident for simulations subsequent to the first simulation, and will produce different stochastic projection recruitments and hence, absolute dependent variables for both the normal and F0eval projection quantities.

Tests of the development version concluded that the results were consistent with respect to the benchmark version 2.0.8.5 as all existing features remain intact, therefore the development version was advanced to the new MULTIFAN-CL release version, **2.0.8.7**.

2.6.2 Version 2.1.0.0

In April 2023, comparative testing was done between the MULTIFAN-CL development version, and the benchmark release version 2.0.8.7 previously tested in October-November 2022. An abbreviated set of tests were undertaken, simply to examine the effect of a several enhancements and corrections made to the development version source code described in section 5, primarily including:

- enhancements and consolidation of the length-based selectivity feature;
- a review of the cohort-strength covariate for growth, and a correction for the recruitment estimates to be available for making the assignments to relative year class strengths;
- reversing the sign of the total log-likelihood reported to the *.par report (to be the negative log-likelihood); and,
- code changes as were required for the Visual Studio 2019 compilation.

The aim of the tests was to confirm the integrity of existing features in respect of changes made in the development version in advance of this version being used in the 2023 stock assessment modelling.

Doitall fits of single species data were undertaken for four examples (SKJ2022 – catch-conditioned; ALB2018, YFT2020, SWO2017 – effort-conditioned), indicating negligible differences (<1%) in key model quantities for all species except SWO2017 for which small differences (<5%) were found.

The tests concluded that the development version results were consistent with respect to the benchmark version 2.0.8.7 as all existing features remain intact, therefore the development version was advanced to the new MULTIFAN-CL release version, **2.1.0.0**.

2.7 Postings to website

There have been no postings of the MULTIFAN-CL release versions to the website since July 2020.

2.8 Independent Peer Review of the 2011 bigeye tuna stock assessment

An outcome of an independent peer review of the 2011 bigeye tuna stock assessment (Ianelli et al. 2012) was a set of recommendations for improvements and developments to the MULTIFAN-CL software. These aim not only to improve the software's application in the context of the bigeye assessment specifically, but also its stock assessment application more generally. These recommendations have been the basis of MULTIFAN-CL developments since the review, and an outline of the status in fulfilling these recommendations is provided.

At the beginning of 2022-23, of the thirteen recommendations, 12 had been implemented and tested, and 1 remained yet to be developed:

- Non-uniform size bins (recommendation "b")

No further progress was made on recommendation ("b") during 2022-23, and remains as an incomplete task on the work plan.

2.9 Tool development

2.9.1 R4MFCL

The R scripts for working with MULTIFAN-CL, developed by OFP are maintained on a GitHub repository and have been partially updated to adapt to the recent MULTIFAN-CL release version file formats. These scripts are used to manipulate the input files, so that submitting model runs can be automated from R. Other scripts can be used to read in the output files, analyze the results, and generate plots and tables. A total of 55 commits were made to the repository during 2022-23, for improvements to this aspect of the project by Arni Magnusson (OFP, Pacific Community).

2.9.2 Testing framework

The testing framework for MULTIFAN-CL compilations first developed in 2011-12, was applied during 2022-23 for the benchmark testing of versions 2.0.8.7 and 2.1.0.0 (section 2.6). This framework ensures the repeatability and traceability of testing by streamlining the process for new source code developments through a system of model testing procedures and directories. The testing criterion is based upon pair-wise comparisons of model run results obtained using an existing MULTIFAN-CL compilation (usually the current release version) versus those from a development version compilation. Tests are undertaken over multiple processor platforms (64-bit architecture only), with application to multiple input testing data sets, and with various options for the MULTIFAN-CL operation, viz. single or multiple model evaluations, or full doitall model fits to convergence. This ensures a thorough integrity-check of model quantities and components of the objective function prior to the distribution of new versions.

Since March 2013, the MULTIFAN-CL source code has undergone substantial developments, and those have been described in earlier reports (e.g., Davies et al. 2022), and the recent developments in 2022-23 are described in Sections 4 and 5.

Following the addition of these new features to the development version, regular testing of this versus the release version aims to ensure the integrity of existing operations. Known as “benchmark tests”, those undertaken in 2022-23 are described in section 2.6. The development version was last tested in April 2023 versus the release version 2.0.8.7. The positive result then defines the development version as being the **benchmark** source code, then to be posted as version 2.1.0.0. Subsequent development versions will then be tested relative to the benchmark to establish their integrity, after which they may be defined as the new benchmark development version. The testing framework entails two levels of tests.

1. Establish the accepted development version

The first level of testing ensures the integrity of existing model features by undertaking tests using a range of single-species data including: ALB2012, ALB2015, BET2011, BET2014, BET2017, YFT2011, SKJ2011, STM2012, SWO2013, SWO2017, YFT2014, YFT2017, SKJ2014, SKJ2016, YFT2020 and SKJ2022; to conclude that single model evaluations and the fitted solutions are sufficiently close to regard the development version estimates as being essentially similar to the benchmark version. This indicates integrity of the development version for undertaking single-species model evaluations. Results are compared among the versions and operating systems, to confirm that the development and release versions produced identical solutions. When differences are found, which can be attributable to improvements in the development version, these are accepted.

Tests using multi-species data disaggregated among species are done which entails comparing the fitted solutions of the development version code versus those solutions obtained using the corresponding data for each species fitted individually. These tests concluded that the operations applying to each population in the disaggregated model have integrity and effectively emulate the solutions obtained when each population is modelled individually. Note that species-specific fisheries data were supplied to the models in the test data examples used. Testing was not conducted using test data for which all fisheries data were aggregated among species (or sexes).

Similarly, tests are done for deterministic and stochastic projections with the pair-wise comparisons among versions and operating systems being made.

A positive test result is when the benchmark tests conclude that the development version conserves the existing features, and so can either be advanced as the new release version, or accepted for the new benchmark development version.

2. Establishing integrity of new features, enhancements, and corrections

This second level of testing entails a detailed examination of new features. The inputs and model configuration are customized for the new features and the operation of the new algorithms are evaluated in respect of the original formulations. During 2022-23 this level of testing was done for the enhancements and corrections (see section 5), to ensure the correct calculations and the expected results.

Review of Testing Framework

In January 2016 the testing framework was reviewed by project members with the following agreed tasks for improvements:

a) Tidy up the testing framework functions and utilities so as to be as automated as possible and more user-friendly with a view to including other team members in running the tests.
b) Upgrade testing framework functions and utilities for applicability to both single-sex and multi-sex file formats, with portability over condor.
c) Integrate the testing framework functions and utilities into the R4MFCL package and ensure compatibility with all assessment modelling applications.
d) Create a GitHub repository for the testing framework functions, utilities, and testing data.
e) Consolidate the R4MFCL GitHub repository with Rob Scott as the lead developer, and add access levels to Nick Davies as a support developer.
f) Construct a suite of routine tests for the R4MFCL package to be run following each revision to the repository, and load the updated R4MFCL package to the testing framework.
g) Construct a single routine MULTIFAN-CL test operation (e.g., single-evaluation of a fitted test model solution) to be conducted daily and directly from the Jenkins compilation utility that returns an exit status value, with an email report sent to the project developers.

Little action has been taken on these tasks and is also unlikely in the remaining part of 2023. It has been identified as a concern, and that they be included in the 2023-24 work plan for the MULTIFAN-CL project.

With the routine compilation and development of a macOS executable now firmly implemented in the project, the testing framework includes the MacMini host, and the macOS executable within tests among platforms and versions. The framework therefore has capacity for conducting tests upon all 3 platforms simultaneously over the Condor network. The test analyses perform pair-wise comparisons among versions and over three platforms: Linux, Windows, and macOS. In respect of the macOS, a recent setback in Oct. 2022 (see 2.4.4) has prevented it being included in the testing framework.

2.9.3 Viewer

The main notable changes during 2022-23 included:

- Fixed a bug that prevented the chart being exported into an image
- This bug also impacted:
  - Chart export into PDF
  - Printing the chart
- Fixed a bug that prevented chart export into Excel outputs
- Inverted colors used to render each series in order to match other pre-existing Observed vs. Predicted plots
  - Observed CPUE is now rendered using blue dots.
  - Predicted CPUE is now rendering using solid crimson line.

2.9.4 Condor parallel processing facility

The Condor (www.condor.wisc.edu) facility has been used routinely for managing multiple MULTIFAN-CL model runs on a grid currently numbering more than 40 computers; being windows or Linux platforms. Support for 32-bit architecture has been discontinued since MULTIFAN-CL executables in this architecture have not been produced since version 1.1.5.9. This grid enables intensive model runs for: benchmark testing MULTIFAN-CL development versions; undertaking stock assessments that entail multiple model runs (e.g. sensitivity analyses and structural uncertainty analyses), and for management strategy evaluation. During 2022-23, additional Linux Virtual Machines were added to the grid to increase the number of model runs possible using the Linux development version executable.

2.10 User's guide

A revision to the MULTIFAN-CL User's Guide (Kleiber et al. 2018) has not yet been completed to include the developments since version 2.0.5.1. Proposed future revisions include: incorporating the suggestions arising from the earlier Training workshops; and the recent features and enhancement added to up to and including version 2.2.3.0. The revised version will be posted on the website http://www.multifan-cl.org/.

3 TRAINING WORKSHOP

No training tutorials were required during 2022-23 with the last training undertaken in June 2022.

4 NEW FEATURES

No innovative or new features were implemented into the MULTIFAN-CL source code during 2022-23. The current development version is **2.2.3.0** which holds all the developments described in section 5 that relate to enhancements and corrections made to existing features since July 2022. These will be merged to the next release version upon the completion of the forthcoming benchmark testing.

5 ENHANCEMENTS AND BUG FIXES

An overview of the enhancements and corrections made to existing features in MULTIFAN-CL during 2022-23 was provided to the pre-assessment workshop (Hamer, 2023). Those, and other developments made subsequent to that meeting, are described in more detail in this section.

5.1 Catch-conditioned method – stochastic projections

A substantial enhancement made to the catch-conditioned feature in the previous year (2021-22) was to prepare this feature for implementing projections of the catch-conditioned method, having fisheries conditioned upon either catch, or effort, during the projection time periods. This enhancement enabled deterministic model projections, including zero-fishing mortality model evaluations for fishing impact analyses of projected population states.

During 2022-23 the catch-conditioned feature was further enhanced for implementing stochastic projections. This was required for undertaking management strategy simulations that underpinned advice to the 19th Regular Session of the WCPFC Commission in respect of the 2022 skipjack stock assessment target reference point (Pacific Community, 2022).

The integrity of model variables among stochastic simulations was ensured, including:

- dependent variables for the estimation model time periods;
- the Newton-Raphson fishing mortality calculations;
- the Beverton-Holt stock-recruitment relationship (BH-SRR) predictions, and the assignment of random absolute recruitments; and,
- the effects of zero-fishing mortality evaluations (F0eval) upon model variables within each simulation.

The development for stochastic projections entailed:

- Implementing deterministic projections using average absolute recruitments;
- The assignment of stochastic recruitments, either as absolute values, or as deviates from the annualized BH-SRR predictions, with apportionment by the average regional and seasonal distributions;
- Ageing the simulation population numbers at age matrix; and,
- Ensuring model variables remained intact for subsequent simulations following the zero-fishing mortality projection evaluation within each simulation.

With respect to the assignment of deterministic and stochastic recruitments as random samples of the absolute estimates, modifications were required to both the catch-conditioned and catch-errors model projections, such that the absolute values are apportioned by the average regional and seasonal distributions. In respect of the catch-errors model projections, the method applied and results are presented in section 5.1.3.

The testing of the implementation indicated it is operating correctly for both the normal fishing mortality (Fnorm) and zero-fishing mortality (F0eval) model evaluations in each simulation. The key criterion for testing the development was satisfied, such that the mean of the stochastic projection recruitments and biomass were unbiased with respect to the deterministic values.

5.1.1 Deterministic projections – absolute recruitments

In order to evaluate the development for stochastic projections with random absolute recruitments, it was necessary to implement this recruitment configuration in deterministic projections of the catch-conditioned model. This is because the deterministic projection is the basis of the testing criterion (section 5.1.2.1). The assignments of absolute average annual or seasonal recruitments in the deterministic projection time periods were implemented and tested. These assignments took account of the annualised BH-SRR relationship if activated.

5.1.2 Stochastic projections – BH-SRR and absolute recruitments

The catch-conditioned version of the population numbers-at-age matrix calculations lacked all the developments since 2008 for stochastic recruitments in projection time periods as applied to the catch-errors model, and took no account of the annualised BH-SRR relationship. The code was therefore consolidated, and the routines for assigning the average annual recruitment, and average recruitment by season were added as required. Stochastic recruitment variables were created and assigned from the input files, with new options added with respect to the projection time periods. Schemas of the logic in respect of recruitments were constructed under both options (absolute or BH-SRR deviates), and source code was developed accordingly.

5.1.2.1 Method

A SKJ2022 catch-conditioned model (the maximum likelihood estimator, MLE) was used for testing. Projection fisheries data was generated based upon the status quo catch and effort. The stochastic projection feature was tested in the following two configurations for projection recruitments:

- As stochastic deviates applied to the annualized BH-SRR predictions (**bh_devs**)
- As random samples of the absolute recruitment estimates (**abs_smp**)

In both configurations, the projection recruitments were distributed in respect of both the average regional and seasonal distributions.

1. **bh_devs**

Annualised BH-SRR predicted recruitments were applied in each projection calendar year, adjusted by the stochastic deviate, and with allocation to each quarterly time period according to the average season-region distribution estimated during the MLE model time periods. The time period specified for the BH-SRR fit was based upon that assumed for the MLE regression. Consequently, the random deviates were sampled from this same subset. A total of 3 simulations were performed.

2. **abs_smp**

Randomly resampled absolute recruitments from the MLE estimation time periods were applied in each projection calendar year with allocation to each quarterly time period according to the average season-region distribution as estimated during the MLE model time periods. The random samples were taken from the same subset time period as for the bh_devs example.

For both configurations, the allocation of recruitments to each quarterly time period was made according to the average season-region distribution estimated during the MLE model time periods.

The key testing criterion of the development entailed comparisons of the projected stochastic recruitments and spawning biomass with respect to that obtained from the deterministic projection. Given the assumptions of the fitted BH-SRR, the deviates would be normally-distributed in log-space, and with the log-normal bias correction being applied, the projection recruitments (and therefore dependent variables such as adult biomass), would also be normally distributed with respect to the deterministic projection values as obtained using the BH-SRR predictions without deviates. The mean of the stochastic projection values would therefore be expected to be unbiased with respect to the deterministic values.

5.1.2.2 Testing

There was no visible indication that the random re-sampling with replacement of the absolute recruitments or the BH-SRR deviates used in the stochastic simulations resulted in a distribution that was not normal (Figure 1). As expected, the simulation recruitments during the projection period were on average very similar to that of the deterministic projection under BH-SRR predicted recruitments (bh_devs), with the individual simulation values exhibiting the effects of large positive or negative deviates (Figure 2). The percentage difference in the mean of the deterministic and stochastic recruitments over the projection period was -0.74%. This similarity was also reflected in the stochastic absolute recruitment configuration (abs_smp), where the percentage difference in the mean of the deterministic and stochastic recruitments over the projection period was 5.8% (Figure 2). Note that the quarterly fluctuation in projected recruitments is due to the apportionment of the annual recruitment in respect of the average regional and seasonal distribution (age_flags(183) =1).

The adult biomass predicted from the stochastic simulations during the projection periods were broadly similar to that of the deterministic projection model, with the percentage difference in mean deterministic and stochastic adult biomass over the projection period being -0.32% and -0.58%, for the bh_devs and abs_smp configurations, respectively (Figure 3).

As may be expected, the consistencies in deterministic and stochastic adult biomass were also evident in the F0eval simulations for both model configurations (Figure 4). Consequently, the stochastic simulation depletion levels were also consistent with the deterministic values (Figure 5).

5.1.3 Catch-errors model stochastic projections with absolute recruitments

The corrections and modifications presented above for the catch-conditioned model in respect of the assignment of deterministic and stochastic recruitments from the randomly resampled absolute estimates, and apportioned by the average regional and seasonal distributions, were also required for the catch-errors model projections (deterministic and stochastic). The example used for testing this development was a catch-errors version of the same SKJ2022 population model, that while having a different solution and consequently different dependent variables, was sufficiently similar to explore the effects of the developments on a similar magnitude to that of the catch-conditioned model example.

Stochastic recruitments and adult biomass were compared for stochastic projections over 5 simulations (Figure 6), and indicated close consistency in the simulation values to that of the deterministic projection model, with the percentage difference in the mean deterministic and stochastic adult biomass over the projection period being 1.14%.

5.1.4 Assignments to BH-SRR deviates and variance

During this development, an assignment error was found in the code that affects simulations > 1 in respect of the BH-SRR deviates used for the stochastic recruitments. This had a marked effect on the simulation dependent variables. The error was manifest only when doing multiple simulations, i.e., age_flags(20) > 1, as for a risk analysis under a single harvest strategy. It would not occur during Management Strategy Evaluations (MSE), where a single sample is drawn from the random recruitments, and single projections are performed with Fnorm and F0eval model evaluations in each; i.e., age_flags(20) =1. The necessary correction was made to ensure no assignment was made during simulations or F0evals. An evaluation of the effect of the error for the example was made. The average of the mean depletion over all projection time periods for the corrected version was 23% higher than that of the previous MULTIFAN-CL version that included the error (i.e., the population is predicted to be less depleted in the corrected version). Therefore, the correction has had a substantial impact on the simulation estimates.

In light of this correction, the integrity of the assignments to estimated BH-SRR variance (used during stochastic simulations) was also checked. The storage of the input solution variance estimate, before the implementation of the simulations, was found to be intact and ensured the correct log-normal bias correction was applied in each simulation. The assignment includes the condition that parest_flags(1) = 1, and is satisfied only upon the very first function evaluation using the input variance value, during the initial minimisation step. Subsequent simulation evaluations do not satisfy this condition, and the variance is therefore not re-assigned. This ensures the original solution value is used in all simulations. The condition that parest_flags(1) = 1 is specified clearly in the manual (Section 4.4.1 - Stochastic projections), however this point was emphasised more strongly in the updated Manual.

5.1.5 Summary

The implementation of stochastic projection model evaluations using the catch-conditioned method is operating correctly for both the normal fishing mortality (Fnorm) and zero-fishing mortality (F0eval) model evaluations in each simulation. Existing flag settings for projections applicable to the catch-errors method model also apply unchanged for the catch-conditioned model, and are already documented in the Manual. The stochastic recruitment configuration was tested where random deviates of the annualised BH-SRR are applied to the predictions in each projection year, and are distributed in respect of the average regional and seasonal distribution in each quarter. The key criterion for testing the development was satisfied, such that the mean of the stochastic projection recruitments and biomass were unbiased with respect to the deterministic values.

A correction was identified for the stochastic recruitment configuration for projections entailing randomly resampled absolute recruitments. The recruitments input in the “simulated_numbers_at_age” file are the annual total when the annualized BH-SRR is configured in the MLE model (age_flags(182) = 1). This resulted in the stochastic absolute quarterly recruitments being approximately 4-fold higher than the deterministic values. A correction was made to the assignments of the stochastic absolute recruitments during the projection time periods that rectified this instance. This correction was made in respect of both the catch-conditioned and catch-errors models.

5.2 Length-based selectivity feature

5.2.1 Rationale

MULTIFAN-CL has historically included a feature that approximates a length-based selectivity within the age-structured model. This is an approximation because the population model is age-structured, and fishing mortalities are applied in respect of ages and not lengths. The feature estimates a length-based selectivity function, but to use it in the catch equations it is converted into age-based selectivity by assuming the length distribution-at-age, and integrating the length-based selectivity over these distributions to derive the age-based selectivity. Cubic splines are used for the length-based selectivity functions, with the nodes being equally spaced. With the growth function, the mean lengths-at-age increase more rapidly for young fish than for old fish, and hence more of the nodes are available for the young fish and allow selectivity to increase more quickly when growth is rapid.

This feature was first developed in 2008-09 but was not fully completed or tested. A review is presented of the feature that: describes the method’s theory; reviews the code to correctly implement it; and, tests its implementation using a real tuna model example (YFT2020). The performance of the feature is assessed and recommendations made for its improvement.

The major impetus for this review was a key recommendation from an independent review panel of the YFT 2020 stock assessment model (Punt et al. 2023) to:

“Implement length-based selectivity. Length-based selectivity seems more natural for the fisheries concerned and the fact that two of the primary data sources are length- and weight-composition data.”

To have a truly length-based selectivity applied to fishing mortality-at-length, the catch equations must be applied to the population numbers-at-length, i.e., within a length-based or an integrated age-length-based model. The length-based selectivity feature in MULTIFAN-CL provides an approximation to this by integrating over presumed length distributions to produce an age-based selectivity.

5.2.2 Method

Broadly the method can be separated into two main components: those that are time-invariant; and, those that are time-variant.

5.2.2.1 Time-invariant components

The distribution at length within each age class, r_{ji}, is derived from the estimated growth function in terms of the mean length-at-age and estimated variance, in respect of age-class *j* and size-class *i*. The size class intervals may relate to either length- or weight-class intervals. For the weight frequency (WF) application, it is firstly necessary to translate the size class intervals assumed for the WF data w_k, into the corresponding mean lengths in each k^{th} interval using the length-weight relationship parameters. Given

$$w_i = a * {l_i}^b$$

is the weight of a fish in the i^{th} length interval, then the mean length of a fish in the k^{th} weight interval is

$$\bar{l}_{w_k} = \left(\frac{w_k}{a}\right)^{\frac{1}{b}}$$

where *a* and *b* are the length weight parameters and $\boldsymbol{w_k}$ is the weight in the mid-interval of weight class ***k***. The $\bar{l}_{w_k}$ over all weight classes *k* are therefore the corresponding lengths in each class, and comprise the length classes for the application of the feature to WF data, and for calculating the corresponding $\boldsymbol{r_{ji}}$ matrix.

The probability distribution of lengths in each age class subsequently assigned to r_{ji} is derived given the mean length-at-age and estimated variance.

$$prob_{ji} = e^{\frac{-t*\left(\frac{t}{2}\right)}{\sigma_j}}$$

where

$$t = \frac{i - \bar{l}_j}{\sigma_j}$$

and $\bar{l}_j$is the mean length at age *j* and σ_j the standard deviation.

In the case of the r_{ji} for the WF application, the probability is derived in respect of $\bar{l}_{w_k}$ and is subsequently transformed

$$r_{ji} = \frac{prob_{ji}}{\omega_i}$$

where

$$\omega_i = \left(\bar{l}_{w_k}\right)^{b-1}$$

and *b* is the exponent parameter of the length-weight relationship. Ther_{ji} for the WF application is then normalized in respect of the sum in each age class. The variable ω_i is denoted as **wm2** in the code.

A spline function is used for estimating the selectivity-at-length, $\hat{S}_i$, with the independent variables (parameters) being the values at each node. A single selectivity is applied in generating length-frequency (LF) and WF predictions, in that the estimated spline is mapped onto the corresponding length intervals of each.

Given the estimated selectivity-at-length $\hat{S}_i$, and the distribution at length within each age class r_{ji}, the selectivity at age *j*, within each length class *i*, is calculated

$$\hat{S}_{ji} = r_{ji} * \hat{S}_i$$

Effectively, this produces the length distribution in each age class "scaled" by the relative selection pattern at length, i.e., the probability of a fish at length being selected for within an age class.

Given the matrix of selectivity-at-age within each length class $\hat{S}_{ji}$, the selectivity-at-age $\hat{S}_j$, is derived by integrating over all the length classes

$$\hat{S}_j = \sum_i \hat{S}_{ji}$$

This can be calculated from both the applications to LF and WF, and is expected to be identical because it is only the size class interval values that differ among the two data types, and the components of the calculation are equivalent. The $\hat{S}_j$ is subsequently applied in the fishing mortality calculations of the dynamic model.

Given the estimated fishery-specific selectivity-at-length, and the derived selectivity-at-age within each length class $\hat{S}_{ji}$, the normalized distribution in each age class may be calculated to produce the fishery-specific mean age-length size distribution.

$$\hat{P}_{ji} = \frac{\hat{S}_{ji}}{\sum_i \hat{S}_{ji}}$$

Effectively this represents the pattern of size-specific selectivity within each age class.

5.2.2.2 *Time-variant components*

Given the selectivity-at-age $\hat{S}_j$, as derived above, the fishery-specific mortalities are applied in each fishing incident to produce predicted proportions caught-at-age, $\hat{p}_j$. This is a time-variant variable as it depends on the dynamic population numbers-at-age in each fishing incident. Given the fishery-specific mean age-length size distribution $\hat{P}_{ji}$, the predicted catch size composition is calculated.

$$\hat{p}_i = \hat{p}_j * \hat{P}_{ji}$$

where * is the dot-product of $\hat{p}_j$ and $\hat{P}_{ji}$, and ***i*** is the size class interval for either the LF or WF data depending upon which $\hat{P}_{ji}$ is employed (length or weight) in the calculation. This is the point in the method that interfaces between the length-specific elements that are time-invariant and dependent upon selectivity and growth, and the age-specific element that is time-variant and dependent upon the population dynamics.

5.2.3 Code revisions

Code for this feature was developed in 2008-09 in prototype form, but was not tested fully using a real tuna model example. The data structures in the feature were upgraded to be consistent with the time-variant selectivity parameterization applied to MULTIFAN-CL in 2015, which allows for a time-block and/or seasonal selectivity parameterisation. The essential elements of feature were documented with flow charts and a code description, and the calculations were found to be operating correctly.

Various code corrections were required in order to fully evaluate the feature, including:

- The reports to *.fit files were made consistent with those of the predicted LFs and WFs
- Added the assignment of the cubic spline function to the selectivity-at-length for the WF application
- Added the assignment to the time-variant selectivity-at-age for the WF application
- Reporting to the plot.rep uses the correct selectivity-at-age from the length-based selectivity feature
- Using the global wm2 variable was implemented as the default case

5.2.4 Testing

The standard age-based selectivity model was used as a basis for comparison with four alternative models using the length-based selectivity feature. The YFT2020 model was used as an example for testing the alternatives.

Definitions of the five models tested are:

- **selage**: age-based selectivity estimation
- **sellen_wm2**: length-based selectivity estimation; with global wm2
- **sellen_wm2_newrji**: sellen_wm2 model; with r_{ji} matrix excluding added constants (zeroes assigned)
- **selwt_wm2**: sellen_wm2 model; with translation to weight frequency size classes and assignment to Fage for fisheries with WF data
- **selwt_wm2_newrji**: selwt_wm2 model; with r_{ji} matrix excluding added constants (zeroes assigned)

Use of the global wm2 variable is to be accepted as an improvement, and therefore the sellen_wm2 and selwt_wm2 models are often referred to here as simply the: **sellen** and **selwt** models. The two models excluding the added constant to the r_{ji} matrix are often referred to as the newrji models, or models using the newrji method.

The models estimating length-based selectivity were minimised starting from an input of an existing solution *.par file, and run for multiple (~10-40,000 evals) until reasonable convergence was achieved, or until exceeding the maximum number of evaluations. The minimisation characteristics were examined for the initial sellen model employing length-based selectivity estimation, and found to be well-behaved and stable with steady declines in both the total objective function and the gradient.

The five models were compared in respect of: the integrated total likelihood and its component terms; selectivity and growth estimates; fits to the length-frequency (LF) and weight-frequency (WF) composition data; and, the model dependent variables.

Both the sellen_wm2 (sellen) and selwt_wm2 (selwt) models produced a moderately worse total likelihood compared to the selage model, by 1946 (0.1%) and 2428 points (0.12%), respectively. Mostly, this is due to worse fits to the LF and WF data, but this is offset substantially by better fits to the tagging (1753 points) and effort data (~490 points). Calculation of the r_{ji} matrix without the added constant for zero cells (the newrji method) substantially worsens the fit of both the sellen and selwt models; altering the total likelihood by around 9000 and 12000 points, respectively. The main components of this difference are the WF and effort deviate terms. These models appeared unstable in the estimation of absolute abundance, and performed worse.

Estimated selectivity-at-length and growth of the sellen and selwt models are almost identical, and these models produced similar dependent variables, within around 3% or less, for the main quantities of interest. The growth estimates for the selage model differ from those of all other models, having a higher $L_{infinity}$ and a relatively slower rate. The length-based selectivity models have more rapid growth rates and a relatively lower $L_{infinity}$. (Figure 7).

Fits to the LF data for most fisheries are visually broadly similar among the sellen and selwt models to that of the selage model (Figure 8). For some fisheries, the fits are improved (PL-ALL-7, PS PHID 7), whereas for many they are visually worse (PS UNA 7, PS UNS 8, HL PHID 7). The two length-based selectivity models without the newrji method (sellen_wm2 and selwt_wm2) exhibit highly modal predictions for the smaller size intervals. This is due to the effect of the increasing selectivity index for young age classes having low variance in mean length-at-age.

Fits to the WF data clearly show the effect of an anomaly due to the small constant added to the r_{ji} matrix that produces spuriously high length-based selectivities as applied to the WF data for the young age classes. For some fisheries the sellen_wm2, sellen_wm2_newrji and selwt_wm2 models produce an anomalously high predictions for the first few size class intervals in all fisheries (Figure 9). This anomaly appears to be mostly resolved for the models that remove this added constant, i.e., the newrji models. Generally, the fits to the WF data appear visibly worse for the length-based selectivity models.

The dichotomy among the length-based selectivity model estimates in respect of the added constant to the r_{ji} matrix is transmitted to the dependent variables, as may be expected; with those for the two groups (either: including; or, excluding the added constant) being substantially different. However, the dependent variables of models in the group that exclude the newrji method (i.e., sellen and selwt models), are more similar to the selage model (Figure 10).

5.2.5 Summary

All the elements of the calculation of predicted size compositions based upon the estimated spline functions of selectivity-at-length are being computed correctly and are consistent, as expected, among the LF and WF applications.

Three anomalies were found in the method as applied to the YFT2020 example:

1. This is caused by the differences in the length intervals over which the calculations are performed in respect of the LF and WF applications. For the WF application, having a much larger first size interval, the entire distribution of lengths for the youngest age class is apportioned to the first length interval, which creates a discrepancy in the calculated selectivity-at-age among the LF and WF applications, where these should correspond exactly.
2. This is caused by the tails of the distribution of lengths in each age class (the r_{ji} matrix) abruptly being assigned zero values when values are < e-03, rather than a "smooth" decline towards near-zero values. Combined with an ascending selectivity-at-length function, this results in truncating the modal size distributions for the young age classes such that the right-hand tail is effectively "removed", and this is visible as discrete "steps" in predicted catches-at-length.
3. This caused by the normalising of the selectivities-at-age in each length class for those values that are near-zero but have increased very slightly due to the increasing selectivity-at-length function. This produces higher selectivities-at-age values in the larger size classes, which results in an inconsistency in the modal distributions for the young age classes of the fishery-specific age-length distribution as applied to the WF data. Consequently, for these age classes higher selectivity occurs at the larger lengths than at the shorter lengths, despite the selectivity being near-zero over the entire size range of these age classes.

Anomaly 1 is clearly a consequence of the stratification assumed for the LF and WF data, and will therefore be case-specific. Anomalies 2 and 3 may be ameliorated by adding smoothing functions to the r_{ji} matrix. Having long tails would allow for predictions in size classes far from that of the mean length-at-age, which would improve the model fit. These tails may be approximated by a suitable exponential decline function or by adjusting the probability-at-length calculation.

Using the YFT2020 example model, the method has been demonstrated to produce reasonable estimates, has stable minimization, and converges to a solution comparable to that using the standard age-specific selectivity estimation. Model options were explored that:

- Attempt to address anomalies 2 and 3 by assigning zero values to the r_{ji} matrix for the WF application (denoted **newrji**); and,
- Apply the selectivities-at-age derived from the WF application for assignment to fishing mortalities-at-age for those fisheries having WF data (denoted **selwt**).

The attempt to offset anomalies 2 and 3 appear to resolve the latter, however, these models (newrji) encountered minimisation problems in estimating the total population scalar and produced unreasonable growth estimates, resulting in implausible differences with respect to the selage model. However, models without the offset exhibited anomaly 3, with spuriously high length-based selectivities as applied to the WF data for the young age classes, resulting in unreasonably high WF predictions in these age classes. These anomalies remain a problem for this feature and must be addressed.

Assigning the selectivities-at-age derived from the WF application did not significantly improve the feature's performance, and given that anomaly 1 is inherent in this approach, it is not recommended for cases having WF size class stratification similar to that of the YFT2020 example.

The length-based selectivity models produced selectivity-at-age estimates different to the standard selectivity-at-age (**selage**) model, and this is attributable to the different growth estimates obtained. Growth was estimated to be faster and to a lower maximum size for the length-based selectivity models. This was the major effect of implementing the length-based selectivity estimation with estimated growth having: a lower $L_{infinity}$; a faster rate; and, in the case of the newrji models, a substantially higher variance. The models that exclude the added constant (newrji) may be producing higher estimated variance in growth as a consequence of the lack of long tails in the r_{ji} distributions in each age class. Given the direct parametric interaction between the growth and selectivity parameters in producing the predictions, this effect upon growth may indicate confounding among growth and selectivity estimation, however, not to the degree where convergence was not possible.

Overall, using the YFT2020 example, the length-based selectivity feature did not improve upon fitting to the observed size compositions over that of the selage model, and indeed performed worse. It is most likely that this is due to the effects of the three anomalies identified with: the example used (its assumed WF size stratification); and the method.

Recommendations for improving and employing the feature:

- Address anomalies 2 and 3 by adding smoothing functions to the r_{ji} matrix that apply long tails having near-zero values and would allow for predictions in size classes distant from that of the mean length-at-age.
- Do not assign the selectivities-at-age derived from the WF application when anomaly 1 is exhibited.

Further work to investigate the feature may include:

- Fitting size compositions with a multinomial-type likelihood (SSMULT or Dirichlet-Multinomial) or implement tail-compression. This may avoid the effect of the truncated tails in the r_{ji} distributions.
- Exploring further to integrate the cohort-strength covariate feature with length-based selectivity to incorporate time-variant growth in the calculations.

5.3 Survey Fishery CPUE likelihood

Davies et al. (2022) described the completed implementation of a log-likelihood term for CPUE indices derived from survey fisheries to be applied in fitting models that use the catch-conditioned method for estimating fishing mortalities. Two enhancements were made during 2022-23 to:

- Enable grouped catchabilities with varying assumed sigma
- Implement the concentrated likelihood with an assumed sigma

5.3.1 Grouped catchabilities with varying assumed sigma

The likelihood term can be applied for the case of more than one survey fishery having the same catchability, e.g., where an identical fishery (either commercial or research) operates in more than one region. The assumption of stationary catchability in such cases is reasonable. Consequently, the absolute index values among the fisheries impact upon the model predictions for each component fishery within the group. This enables the assumption that the fishery-specific indices may reflect relative abundance among the population in the respective regions where each fishery operates; often termed as “regional weighting”.

The purpose of this enhancement was in case of grouped survey fisheries that share the same stationary catchability, to allow the capability for the component fishery to have non-shared penalty weights, i.e., unequal assumed σ.

Taking the general form for the negative log-normal likelihood

$$0.5\sum_{i}\log(\lambda_i\sigma^2)+0.5\sum_{i}\frac{(P_i-O_i)^2}{\lambda_i\sigma^2}$$

where P_i and O_i are the normalised predictions and observations, respectively, on the log-scale. The normalised predictions will therefore be apportioned to fishery-specific sub-vectors, *k*:

$$P_{ik}=\log(\hat{I}_{ik})-\text{mean}(\log(\hat{I}_i))$$

yet this is derived in respect of the $\mathrm{mean}(\log(\hat{I}_i))$ of the indices for all fisheries within the group, which maintains the assumption of stationary catchability in each sub-vector.

While the indices are grouped into single vectors for deriving the normalised predictions and observations, P_i and O_i, respectively, fishery-specific penalty weights, σ_k are assigned to the corresponding sub-vectors within the likelihood formulation in respect of fishery *k*, i.e., they are assigned via the distributive property. When no index-specific precision, λ_i, is available the general likelihood form will then simplify and include the fishery-specific penalty weights σ_k , and may be called the **non-concentrated likelihood** form:

$$0.5\sum_{k}\log\left(\sigma_k^2\right) + 0.5\sum_{k}\sum_{i}\frac{(P_{ik}-O_{ik})^2}{\sigma_k^2}$$

For the case where index-specific precision λ_i is available, the relative weighting is applied in respect of the fishery-specific sub-vectors, *k* and included in the general formulation, and may be called the **concentrated likelihood** form:

$$0.5\sum_{k}\sum_{i}\log\left(\hat{\lambda}_{ki}\sigma_k^2\right) + 0.5\sum_{k}\sum_{i}\frac{(P_{ik}-O_{ik})^2}{\hat{\lambda}_{ki}\sigma_k^2}$$

where *i* are the observations for fishery *k*.

In the instance of grouping the survey fisheries, all the indices are grouped into single vectors for deriving the normalised predictions and observations, P_i and O_i, respectively. The assumption can be made that the **term *σ* is constant among all fisheries within the group**. In other words, a single assumed “penalty weight” is assigned to the likelihood term, and this was the existing code in MULTIFAN-CL.

The purpose of this enhancement was to group the survey fisheries, so as to share the same stationary catchability, but **to have the capability for non-shared penalty weights**. This will allow flexibility in the assumption, and that the σ is not constant among all fisheries within the group.

For both likelihood forms, where the value of σ is assumed, the estimated $\hat{\sigma}$ of the fitted solution may be calculated using:

$$\hat{\sigma}^2 = \sum_{i}\frac{(P_i-O_i)^2}{n\lambda_i}$$

where for the non-concentrated form there is no λ_i term.

5.3.1.1 Testing and results

The feature was tested using a simple three-region tuna model (ALB2021-3region), using a design that explored a small range of penalty weight variability for each of the non- concentrated and concentrated forms of the CPUE negative log-likelihood.

Non-concentrated likelihood scenarios

A fishery-specific multiplicative penalty weight was assigned that is analogous to σ_k. The following table specifies the respective values for the penalty weights assigned (using **fish_flags(*k*,92)/10**) for fisheries *k* = 18, 19, and 20, respectively.

	pen_eq	**pen_varmin**	**pen_var**	**pen_varmax**
Penalty weight	10:10:10	10:7:10	10:2:10	10:50:10

Such that:

- **pen_eq** – maintains the assumption of equal penalty weights
- **pen_varmin** – applies minimal variation in penalty weights (lower)
- **pen_var** – applies moderate variation in penalty weights (much lower)
- **pen_varmax** – applies large variation in penalty weights (substantially higher)

Fishery 19 was selected to employ variable weighting because of the higher absolute CPUE indices compared to the other fisheries, and it therefore may exhibit greater effects of this feature.

Concentrated likelihood scenarios

The fishery-specific σ_k in Eqn.3 is assigned using **fish_flags(*k*,92)/100**, such that e.g., fish_flags(*k*,92) = 25 assigns σ_k = 0.25. The following table specifies the respective values for the σ_k for fisheries *k* = 18, 19, and 20, respectively within the range.

	conc_sig_eq	**conc_sig_varmin**	**conc_sig_var**	**conc_sig_varmax**
Penalty weight σ_k	0.25:0.25:0.25	0.25:0.30:0.25	0.25:0.50:0.25	0.25:0.05:0.25

This range explores relative weighting very similar to that of the non-concentrated model scenarios, from constant to high variability.

The scenarios within the range for the two likelihood forms were therefore denoted:

Likelihood form	**Penalty scenario**			
Non-concentrated	**pen_eq**	**pen_varmin**	**pen_var**	**pen_varmax**
Concentrated	**conc_sig_eq**	**conc_sig_varmin**	**conc_sig_var**	**conc_sig_varmax**

The example tuna model included one survey fishery occurring in each of the three regions: fishery 18 (region 1), fishery 19 (region 2) and fishery 20 (region 3). The observed CPUE trends were somewhat different among the fisheries (Figure 11), and the time-variant precision of the observed indices also differed, with those for fishery 18 having highest precision, medium precision for fishery 19, and lowest precision for fishery 20. For all fisheries, precision is lowest for the initial period of about 10 years.

The effect of the change in penalty weights is illustrated in comparisons among the scenarios relative to the **pen_eq** and **conc_pen_eq** scenario. The effect of the likelihood form is illustrated in a comparison of only the **pen_eq** and **conc_pen_eq** scenarios of each likelihood form.

Results – non-concentrated form

The quality of fit to the fishery-specific CPUE indices changed visibly with altered penalty weights among the fisheries (Figure 12). Reducing the weight for fishery 19, pen_var, worsened the fit to that fishery but improved the fit for fishery 20. Conversely, increasing the weight for fishery 19, pen_varmax, substantially improved the fit, but worsened the fit for fishery 20.

For all models the assumption of stationary catchability resulted in the non-uniform distribution of absolute biomass among the regions, with the region of fishery 19 having highest biomass, and the region of fishery 18 the lowest (Figure 13). The effect of substantially increasing the penalty weight for fishery 19 (pen_varmax) is evident in a small increase in the biomass for that region, but also a marked increase for the region of fishery 20 (Figure 13), for which the fit to the observed CPUE was worse (Figure 12). This indicates the indirect effects of variable penalty weights that may influence not only the quality of fitting the observed CPUE, but also the stationary catchability assumption's effect in regional biomass distributions.

Only for the case of high variability in penalty weight (pen_varmax), having a 5-fold increase in the weight of fishery 19, was a marked difference produced in total absolute biomass (Figure 14), being generally higher.

Results – concentrated form

As for the non-concentrated likelihood models, the quality of fit to the fishery-specific CPUE time series was affected by the assumed penalty weight for the concentrated likelihood models (Figure 15). Using the pen_eq model as a reference for comparison (Figure 12), the predicted indices for the concentrated likelihood model, conc_sig_eq, for fisheries 18 and 19 are less variable, and visibly better fits are obtained compared to the non-concentrated model, with a steeper and consistent decline for fishery 19 to the 60th time period. In contrast, the predicted indices for the concentrated likelihood model, conc_sig_eq, for fishery 20 are more variable, with a visibly worse fit in the early period, exhibiting a steeper decline, compared to the non-concentrated model. This exhibits the effect of the time-variant precision in the concentrated likelihood form.

As with the non-concentrated likelihood models, for all the concentrated likelihood models the assumption of stationary catchability resulted in the non-uniform distribution of absolute biomass among the regions, with the region of fishery 19 having highest biomass, and the region of fishery 18 the lowest.

Moderate and slight decreases in penalty weight for fishery 19 produced little impact on the absolute biomass for the corresponding region 2, but produced small or moderate decreases in the regional biomass corresponding to fishery 20 (region 3), (Figure 16). The effect of substantially increasing (5-fold higher) the penalty weight for fishery 19 (conc_sig_varmax) is evident in a moderate increase in the biomass for that region, but also a marked increase for the region of fishery 20, for which the fit to the observed CPUE was worse (Figure 15). The direction and degree of this effect was similar to that of the non-concentrated likelihood models. This indicates the indirect effects in cases of extremely variable penalty weights that may influence, not only the quality of fitting the observed CPUE, but also the stationary catchability assumption's effect in regional biomass distributions.

As with the non-concentrated likelihood models, the scenario of high variability in penalty weight (conc_sig_varmax), (5-fold increase in the weight of fishery 19), produced a marked difference in total absolute biomass (Figure 17), being generally higher. However, the scenario of 100% decrease in the penalty weight (conc_sig_var) resulted in lower total absolute biomass.

Results – non-concentrated versus concentrated form

A comparison among the models fitted with either the non-concentrated or concentrated likelihood formulation was made using the respective scenarios of constant penalty weight among the grouped fisheries (pen_eq and conc_sig_eq, respectively). This was to illustrate simply the effect of time-variant precision of the observed indices within the CPUE time series of each fishery.

Using the pen_eq model as a reference for comparison, the total adult biomass estimates of the concentrated model (conc_sig_eq) are substantially higher (35%) in the early period, and then around 24% higher in recent years, compared to the non-concentrated model (Figure 18). This difference is consistent with the poorer fit to the early CPUE decline for fishery 20, and the improved fit to the steeper and consistent decline for the fishery 19, obtained for the concentrated model compared to that of the non-concentrated model. This difference in fit for fishery 20 is then reflected in the substantially higher estimated biomass in the corresponding region 2 obtained for the concentrated model compared to that of the non-concentrated model (Figure 19), which largely accounts for the higher total estimated biomass.

5.3.2 Revised concentrated likelihood - assumed sigma, with normalised lambdas

The concentrated form of the negative log-likelihood where **index-specific precision** λ_i is available, is presented in the previous section. The capability for non-shared penalty weights for the grouped fisheries having shared catchability has been implemented. Various assumptions were explored for the units of λ_i, either as a coefficient of variation or as a normalised coefficient. Given that the value of σ is assumed, it is reasonable for λ_i to be expressed as a normalised coefficient, such that the mean of the denominator variance term is equivalent to the assumed σ. This adjustment has been implemented in MULTIFAN-CL, and testing is in progress.

A further enhancement to the concentrated likelihood formulation has commenced that is to implement the YFT2020 review panel's recommendation (Punt et al. 2023):

5) Add the ability to specify overdispersion in CPUE as an additive rather than multiplicative factor.

The formulation of this is being drafted and implemented in the code, and is a task in the proposed workplan for 2023-24.

5.3.3 Summary

The effects of the assumed stationary catchability produced an expected pattern in the estimated regional biomass distribution, that were reasonably consistent among the non-concentrated and concentrated likelihood models.

Varying the penalty weights in the grouped CPUE likelihood alters the relative goodness of fit to the observed indices according to the assumed weights, as might be expected. However, it may also influence the stationary catchability assumption's effect on the distribution of regional biomass in cases of extreme differences in the penalty weights. This may have a moderate effect on estimates of total biomass and depletion levels.

Substantial differences in the assumed precision (e.g., 5-fold) among the fisheries being grouped and assumed to have stationary catchability, can impact upon the estimated biomass distribution among the regions. Differences of < 100% were found to have less impact on these estimates.

Differences in quality of fit among the non-concentrated and concentrated likelihood models are consistent with the expected effects of the time-variant precision in the observations being accounted for within the concentrated likelihood.

5.4 Negative log-likelihood output to *.par

The total integrated objective function reported to the *.par output file was the log-likelihood term. In order to achieve consistency with the component terms and the total reported to the run-time screen output, this function value was changed to the negative log-likelihood. This also achieved consistency with the component terms reported to the output report "test_plot_output" that is used for diagnosing the total integrated objective function in respect of its component terms.

This change makes no difference to the function evaluation or minimization procedure, and only alters the reported total integrated value reported the *.par file.

5.5 Adding first model year to *.par

The MULTIFAN-CL support package: FLR4MFCL, is now fundamental in undertaking stock assessments and management strategy evaluations. To facilitate streamlined processing of the *.par report, a new section added to the .par file format: # First year in model, with the next line being the first calendar year of the model time periods. For example:

```
# First year in model
1960
```

This new section has been located close to the end of the .par file so as not to interfere with the order of key flags and parameter inputs/outputs.

The version of the .par file (in parest_flags(200)) has been increased from 1065 to 1066. This is to maintain backward compatibility for the input of older .par versions.

5.6 Abbreviated derivatives calculation of the dependent variables

For large and complex models, such as the bigeye and yellowfin tuna stock assessments, it is usual for the calculation of the dependent variable derivatives to be computationally very intensive. These variables can number up to 10,000 and the calculation time is of the order of 20 hours. For large stock assessments entailing structural uncertainty grids having 100s of cells, this can be prohibitive.

The key management quantities of interest for models in the grid include: $SB_{recent}/SB_{F=0}$, F_{recent}/F_{MSY}, and SB_{recent}/SB_{MSY}, the latter two quantities were added to the default range of dependent variables for which the derivatives and standard deviations are calculated. Given that the dependent variables required for these quantities represent a subset of the default range, it was possible to assign a parest_flags that "leap-frogs" the vector and calculates the derivatives for only those required.

For the bigeye tuna example, the default number of dependent variables was 9776, and with the abbreviation this reduced to 1402 (around 14.3%). On a stand-alone Linux machine, the default calculation took 19h21m. With the abbreviated calculation required only around 1 minute. This greatly improves the efficiency for summarizing the structural uncertainty grid for the key management quantities of interest.

Testing of the default and abbreviated variances confirmed they were identical for the corresponding dependent variables of interest.

5.7 Bug fixes

5.7.1 Growth – standard deviation of von Bertalanffy mean length-at-age

The von Bertalanffy mean length-at-age ***a*** is formulated in section A.1.4 Growth, of the MULTIFAN-CL manual (Kleiber et al. 2018) as:

$$\mu_a = L_1 + (L_A - L_1)\left[\frac{1 - e^{-\kappa(a-1)}}{1 - e^{-\kappa(A-1)}}\right]$$

Eq.n 1

Where $L_1$is the mean length of the first age class, L_A is the mean length of the oldest age class, and κ is the von Bertalanffy growth coefficient. For example, the bigeye tuna mean length-at-age (Figure 20, left panel).

A more convenient form of the growth coefficient exponent term is:

$$\rho = e^{-\kappa}$$

Which simplifies the growth equation to:

$$\mu_a = L_1 + (L_A - L_1)\left[\frac{1 - \rho^{(a-1)}}{1 - \rho^{(A-1)}}\right]$$

Eq.n 2

The length-dependent variance (standard deviation σ) as formulated in section A.1.4 Growth, of the MULTIFAN-CL manual (Kleiber et al. 2018, equation A.3) is:

$$\sigma_a = \lambda_1 e^{-\lambda_2(1-2\left(\frac{\mu_a - L_1}{L_A - L_1}\right))}$$

Eq.n 3

Such that the variance is assumed to be a function of length, where λ_1 is the standard deviation at an intermediate age, and λ_2 is the length-dependent trend. Reversing the sign of the exponent term, makes this expression consistent with the equations.

$$\sigma_a = \lambda_1 e^{\lambda_2(-1+2\left(\frac{\mu_a - L_1}{L_A - L_1}\right))}$$

Or simply

$$\sigma_a = \lambda_1 e^{\lambda_2 \tilde{\mu}_a}$$

The exponent term $\tilde{\mu}_a$ is the re-scaled mean length-at-age from -1 to 1 as:

$$\tilde{\mu}_a = -1 + 2\left(\frac{\mu_a - L_1}{L_A - L_1}\right)$$

Eq.n 4

Eq.n 2 may be modified slightly to:

$$\frac{\mu_a - L_1}{L_A - L_1} = \left[\frac{1 - \rho^{(a-1)}}{1 - \rho^{(A-1)}}\right]$$

Therefore, substituting the right-hand term into Eq.n 4 expresses the re-scaled mean length-at-age in terms of ρ:

$$\tilde{\mu}_a = -1 + 2\left(\left[\frac{1 - \rho^{(a-1)}}{1 - \rho^{(A-1)}}\right]\right)$$

Eq.n 5

And this is the form of the re-scaled von Bertalanffy mean length-at-age employed in almost all cases in MULTIFAN-CL (Figure 20, right panel). However, it was found in once instance, an alternative formulation was used:

$$\tilde{\mu}_a = \frac{\big(2(a-1)\big) - (A-1)}{(A-1)}$$

Eq.n 6

which produces the linear re-scaled mean lengths at age in Figure 20 (right panel). The implications of this inconsistency on the variance of mean length-at-age are shown in Figure 21, where the standard deviations derived using Eq.n 5 are consistent with the length-dependence of the increasing mean length-at-age, while those derived using Eq.n 6 have a concave increasing trend.

The effect of implementing the corrected formulation (Eq.n 5) was to remove the potential for an estimated increase in variance with respect to age compared as is obtained when implementing Eq.n 6 (Figure 22). Given that the instance where Eq.n 6 was being implemented impacted upon predicted weight-at-age, the effect was also evident in a moderate decrease in the goodness of fit to observed weight frequencies.

5.7.2 Mean weight-at-size interval global variable

The global variable for the length-weight conversion of the length of a weight size interval *k* is **wm2**:

$$wm2 = (l_k)^{b-1}$$

Where *b* is the length-weight parameter. This variable is used globally for the model weight-frequency predictions. However, in a small number of instances, local calculations of wm2 were performed using an exponent = 2, instead of *b*-1. While the difference is minimal, these local calculations were removed and replaced with the correct global variable.

The effect of the correction related to the calculations of predicted weight frequencies, slightly changing the weight-frequency likelihood term, e.g., a value of -935892.345 may change by 6.285 points. Testing with the bigeye 2020 PDH model indicated this does not significantly change the solution, nor the PDH result.

5.7.3 Default parameter scaling

In version 2.0.8.2 parest_flags(387) was implemented to enable an option that restored the parameter scaling settings as were the default in version 2.0.7.3. The affected independent variables were significant including the: total population scalar; mean regional recruitment proportions; von Bertalanffy growth parameters; selectivity; and, tag reporting rates. Testing done relative to the 2.0.7.3 scaling settings indicated minimisation performance in versions subsequent to version 2.0.7.3 was found to be markedly worse.

The implementation of parest_flags(387) assigned for the parameter scaling was reversed, such that the default scaling of version 2.0.7.3 would be the current default, with the revised scaling being an option when parest_flags(387) = 1.

5.7.4 Lorenzen start parameter values and parameter scaling

In the case where the Lorenzen functional form is being estimated without an existing set of parameter estimates, a plausible initial value was assigned within MULTIFAN-CL; facilitating a suitable starting point for the minimization. However, in the case where existing parameter values are available, or have been fixed at an assumed value, and are being input, the facility for assigning plausible initial values was overwriting the input values. This was corrected to apply the initialization only in the case of input parameter values equal to zero.

Testing with alternative scaling of the Lorenzen parameters during minimization was undertaken, and a more suitable scalar was established, and applied in the minimizing routine.

5.7.5 Lower bound of fml_effort_rltnshp parameters

Catch-conditioned models usually include the estimation of the fml_effort_rltnshp regression as it may be required for predicting catches when: unavailable historically; or, during population projections including effort-conditioned fisheries in the future. In unique cases where the historical observed effort values are extremely small, the regression parameters are also estimated to have a very low magnitude, and for some examples were on the lower bounds assigned in the minimization. The lower bounds were therefore decreased to -500 to accommodate these rare instances.

5.7.6 Revised orthogonal-polynomial penalty function

The orthogonal-polynomial recruitments parameterization includes the calculation of normal prior penalties on the coefficients at each polynomial level (year, season, region, season-region), that implements the assumption that they are normally distributed subject to the user-specified relative penalty weight. A tuna model example that applied this parameterization indicated that a primary contribution of the likelihood profile was the orthogonal-polynomial penalty term; that steadily increased over the profile range of average absolute biomass.

The penalty relating to the year-effect coefficients was found to include the 1st coefficient that defines the total population scaling, and is not actually related to the polynomial for temporal recruitment variability. This was an error, and explained the behaviour of the orthogonal-polynomial recruitments penalty observed that steadily increased as the likelihood profile explores higher average biomass values, and hence larger total population scalars, resulting in higher penalties.

The correction was made to the year-effect penalty calculation to exclude the 1st year-effect coefficient from the penalty term. This was warranted because it is natural not to include the scaling coefficient in the normal prior, but only the year-effect coefficients.

The effect of this correction for the example tested, was for higher year-effect variability resulting in higher initial biomass, but relatively little effect on estimated adult biomass depletion. It reduced the overall negative log-likelihood and improved the MLE fit to the observations, which is attributable to removing the confounding among the total population scaling coefficient, and the assumed normal prior on the year-effect coefficients.

During the course of making this correction, the orthogonal-polynomial year-effect penalty term was added to the “test_plot_output” report of the likelihood components.

5.7.7 Likelihood components report - correction to Dirichlet-Multinomial

A large discrepancy was evident among the components of the Dirichlet-Multinomial (DM) likelihood for size composition data as reported to the “test_plot_output” report containing the likelihood components, and the corresponding term in the run-time screen output.

Two problems were identified that caused this discrepancy. Following the normal model evaluation and report production, the zero-fishing mortality evaluation undertaken for the fishing impact analysis was incrementing the DM likelihood term a second time before reporting it to the output file. Also, unstable minimisations can produce model re-starts, which results in incrementing of the DM likelihood multiple times before producing the report.

The corrections made included: making assignments to the DM likelihood report variables only when not doing a zero-fishing mortality evaluation; and, initializing these variables at the beginning of the model evaluation immediately preceding the report production.

5.7.8 Maturity at length vector

In the rare case where numerous sequential model developments are done starting from an initial .par file, with subsequent model developments made beginning with the .par of each preceding step, it was found that the maturity-at-length vector reported in the final .par was substantially different to that of the .par input at the initial model development. Given that maturity-at-length is a constant, this was an error.

The cause was found to be a copy constructor that was being incremented by a small value to avoid arithmetic errors, but which caused compounding during sequential input/output of .par files. The copy constructor was removed, and replaced with a local variable.

5.7.9 Growth cohort strength covariate

The code for implementing the cohort-strength covariate was reviewed, described, and a correction was made such that the recruitment estimates were available for making the assignments to relative year class strengths. This was done for both possible recruitment parameterisations (mean+deviate and orthogonal-polynomial). Preliminary testing indicated this feature requires further review and testing for it to become fully operational.

6 APPLICATION OF FEATURES

6.1 Developing advice for WCPFC19

The enhancement of the catch-conditioned feature for implementing stochastic projections was applied for undertaking management strategy simulations that underpinned advice to the WCPFC19 Commission meeting in respect of the 2022 skipjack stock assessment target reference point (Pacific Community, 2022).

6.2 Stock assessments for SC19

The 2023 stock assessment models (Day et al. 2023, Magnusson et al, 2023) were undertaken with implementation of the following features in the updated version of MULTIFAN-CL:

- The correction to the calculation of the standard deviation of von Bertalanffy mean length-at-age as applied to weight-frequencies;
- Grouped index fisheries that share the same stationary catchability in the CPUE likelihood, to allow the capability for the component fishery to have non-shared penalty weights, i.e., unequal assumed σ;
- The option that restored the default parameter scaling settings as they were in version 2.0.7.3;
- Allowing for input Lorenzen start parameter values with improved parameter scaling; and,
- Abbreviation of the derivatives calculation of the dependent variables as required for the models of the uncertainty grid, reducing the computational load to being negligible

7 FUTURE WORK

A listing of the status of the work: originally proposed, that is pending, in progress, or completed during 2022-23 is provided in Table 2. The proposed future work plan for the development of MULTIFAN-CL in 2023-24 is presented in **Error! Reference source not found.**. As was the case in 2022-23, compared to earlier work plans, no substantial new developments are proposed in the future workplan that require innovative mathematical elements. Rather, resources are being directed to: consolidating those aspects that are incomplete for recent new features; enhancements of existing features; and, documentation. The general approach for the future workplan includes:

- Testing the implementation of examples that employ all the new features and refine the I/O and diagnostic reports.

- The code for existing features will be reviewed and refined; a backlog of bug fixes will be completed; outstanding tasks from the bigeye and yellowfin tuna independent review panel recommendations will be addressed; and any "small-scale" requests in the tasks list.
- Provide training and support for OFP stock assessment scientists (including participation in an independent review of the yellowfin tuna stock assessment model).
- Providing support for MSE requirements and improvements.
- Catching up on the remaining documentation required for updating the MULTIFAN-CL User's Guide.

Some of the items in Table 2 (Tasks rolled over from 2022-23) have been retained, but will be fit within the context of the above workplan for 2023-24, and others have been set aside for the years that follow.

8 DISCUSSION

The achievements made in updating MULTIFAN-CL for 2022-23 can be ordered in terms of those: essential for providing stock assessment advice; advantageous enhancements that improve assessment modelling; beneficial corrections; and, that consolidate existing features.

Extending the catch-conditioned feature to perform stochastic projections having future fisheries conditioned in respect of either catch or effort was essential to providing risk estimates for the 19th Commission meeting in 2022. This task consumed most of the second part of 2022, and entailed substantial code development in the numbers-at-age calculation routine specific to the catch-conditioned method; particularly in respect of the projection recruitment assignments. Testing of the feature confirmed the projection calculations were correct, and the subsequent benchmark testing done preceding the Commission meeting, confirmed the version used for the advice presented to the meeting had maintained the integrity of all existing features. With these assurances in respect the updated version, it underpinned the advice arising from the 2022 skipjack tuna risk assessments.

A number of advantageous enhancements were made that improved the process of stock assessment modelling in 2023. Allowing for grouped index fisheries that share the same stationary catchability in the CPUE likelihood, to have non-shared penalty weights, i.e., unequal assumed σ for each component fishery, enabled better statistical assumptions to be made. This being reasonable given the differences in precision among the standardized CPUE of the index fisheries. Testing carried out by the stock assessment team clearly indicated better performance when employing the default parameter scaling settings for key parameters as they were in version 2.0.7.3. Employing these settings by default in the current version improved minimisation performance and stability. The corrections that allowed for fixed Lorenzen parameter values, assumed initial values with which to start estimation, and improved parameter scaling, were essential and substantially improved the implementation of this feature in the stock assessment models. A significant and convenient improvement to the logistics of computing the model statistical uncertainty from the uncertainty grid, was the abbreviation of the derivatives calculation of the dependent variables. This reduced the computational load to being almost negligible (around 1 minute). The dependent variables required for the grid models was only a handful, and the abbreviation from nearly 10,000 variables allowed for this improved efficiency. Cumulatively, these enhancements have been notably advantageous to the 2023 assessment modelling, and also in the future. Some of the enhancements were made at the suggestion and request of those actively engaged in the 2023 stock assessment modelling (pers. comm. John Hampton, Jemery Day, and Arni Magnusson, OFP Pacific Community).

A substantial improvement to MULTIFAN-CL was the correction made to the calculation of the standard deviations of von Bertalanffy mean lengths-at-age, as used for deriving mean weight-at-age and applied to predicted weight-frequencies. This achieved internal consistency within the model in respect of growth variance, and aside from producing more biologically realistic variances at higher ages, it appeared to improve minimization stability, performance, and achieved better-determined (PDH) model solutions. The latter was beneficial to the development of stock assessment models in 2023. Identification of this inconsistency arose from a useful and insightful quality assessment performed by the stock assessment team during model developments in 2023 that focused attention on the variance calculations of both the von Bertalanffy and Richards growth formulations (pers. comm. Jemery Day).

The primary existing feature consolidated in 2022-23 was the estimation of length-based selectivities. This was a substantial task during the year that required full documentation of the method and a thorough description of the code implementation. After reconciling this feature with the new structures associated with the time-variant selectivity parameterization, it was found to be largely intact and operating as originally intended from the early development done in 2008-09. This feature holds promise in that fitted size compositions are comparable with that of the “standard” age-based selectivities, however, the anomalies identified with the method must first be resolved. These produced undesirable discrepancies in predicted size compositions and worse likelihood estimates compared to the age-based method. Resolving these anomalies has been added to the long-term tasks of the MULTIFAN-CL workplan.

A number of setbacks and obstacles were encountered in completing the tasks set out in the 2022-23 workplan (Davies et al. 2022). Many tasks were postponed by higher priority tasks introduced during the year as were required or considered important for the 2023 stock assessments. For example, recommendations of an independent peer review panel were added to the tasks for 2022-23, of which reinstating length-based selectivity estimation was assigned a high priority, and was undertaken immediately. Others that assumed higher priority were the enhancements including: the abbreviated derivatives calculation of dependent variables, improvements to the *.par format, and the correction to the orthogonal-polynomial recruitment penalty. These changes resulted in a number of the smaller tasks for 2022-23 being postponed to the following year. An unexpected setback was a halt in producing the macOS executable caused the Mini Mac PC becoming unavailable to the MULTIFAN-CL project due to cybersecurity concerns. Logistically, it is impossible to develop and test this compilation without the project having network access to this PC. A task has been added to the 2023-24 workplan to explore an arrangement that will restore this facet of the project, to ensure the delivery of the macOS executable to MULTIFAN-CL users based on that platform.

The recommendations of an independent peer review panel were added to the workplan for 2023-24, some of which will draw heavily upon the project’s resources for development, implementation and testing. An important first step in the coming year, will be to prioritise the list of tasks to identify those of immediate priority for the 2024 stock assessments.

A significant feature in 2022-23 was the lack of innovative new features being added to the project, due to the retirement in Dec. 2021 of the lead developer, and original architect of MULTIFAN-CL, Dr David Fournier. This was anticipated (Davies et al. 2022), and rather, developments have been restricted to consolidating on recent features (e.g., catch-conditioned projections), enhancing existing features, improvements to processes and reporting, and making corrections. The workplan for 2023-24 continues with this theme, but also addresses the much-needed tasks of documentation and project administration. In the years preceding 2022, new developments progressed at a rapid pace, and therefore, the User’s Guide for MULTIFAN-CL requires substantial updates. This is considered critically important as many new flag settings vital to stock assessment model development, remain undocumented, and so the risk of errors is increasingly high.

The continuation of limited improvements to the support structures of the MULTIFAN-CL project made during 2022-23, is an ongoing concern. Substantially fewer project resources (due to Dr Fournier’s retirement), and lower assigned priority to this area of the project (because of the importance of preparing features for production assessments), means the support structures continue to be neglected. An obvious consequence of this, is that versions since (and including) 2.0.7.0 have not yet been posted on the website. Also, testing of some of the new features (e.g., new movement parameterization) is incomplete, which makes their utility relative to existing methods unclear. The proposed workplan listed in section 7, seeks to address this situation, but this will be subject to: the priority-setting exercise required in the immediate future; and, a consideration of the resources available to the project in the future.

9 REFERENCES

Bull, B., Francis, R.I.C.C., Dunn, A., McKenzie, A., Gilbert, D.J., Smith, M.H., Bian, R., and Fu, D. 2012. CASAL (C++ algorithmic stock assessment laboratory): CASAL User Manual v2.30-2012/03/21. NIWA Technical Report 135. 280 p.

Davies, N., Fournier, D., Bouyé, F., and Hampton, J. 2022. Developments in the MULTIFAN-CL software 2021-22. WCPFC-SC18-2022/SA-IP-03. Electronic Meeting.

Day, J., Magnusson, A., Teears, T., Hampton, J., Davies, N., Castillo Jordan, C., Peatman, T., Scott, R., Scutt Phillips, J., MacKenzie, S., Scott, F., Yao, N., Pilling, G., Williams, P., and Hamer, P. 2023. Stock assessment of bigeye tuna in the Western and Central Pacific Ocean: 2023. WCPFC-SC19-2023/SA-WP-05. Koror, Palau, 16 – 24 August 2023

Fournier, D.A., Hampton, J., and Sibert, J.R. 1998. MULTIFAN-CL: a length-based, age-structured model for fisheries stock assessment, with application to South Pacific albacore, *Thunnus alalunga*. *Can. J. Fish. Aquat. Sci.* **55**:2105-2116

Hamer, P. 2023. Report from the SPC pre-assessment Workshop – April 2023. WCPFC-SC19-2023/SA-IP-01. Koror, Palau, 16 – 24 August 2023

Ianelli, J., Maunder, M., and Punt, A. 2012. Independent review of 2011 WCPO bigeye tuna assessment. WCPFC-SC8-SA-WP-01

Kleiber, P., Fournier, D., Hampton, J., Davies, N., Bouyé, F., and Hoyle, S. 2018. MULTIFAN-CL User's Guide. http://www.multifan-cl.org/

Magnusson, A., Day, J., Teears, T., Hampton, J., Davies, N., Castillo Jordan, C., Peatman, T., Scott, R., Scutt Phillips, J., MacKenzie, S., Scott, F., Yao, N., Pilling, G., Williams, P., and Hamer, P. 2023. Stock assessment of yellowfin tuna in the Western and Central Pacific Ocean: 2023. WCPFC-SC19-2023/SA-WP-04. Koror, Palau, 16 – 24 August 2023

Pacific Community. 2022. Evaluations to support decisions on the WCPO skipjack tuna target reference point based upon the 2022 stock assessment. WCPFC19-2022-10

Punt, A.E., Maunder, M.N., and Ianelli, J.N. 2023. Independent review of recent WCPO yellowfin tuna assessment. WCPFC-SC19-2023/SA-WP-01. Koror, Palau, 16 – 24 August 2023

10 TABLES

Table 1. Modifications to MULTIFAN-CL with respect to their state of completion as of July 2023.

2011 Bigeye Tuna Peer review recommendations		
Task	Description	Status of completion
b. Non-uniform size bins	Allow the length bins to be of different widths. One might, for example, want many narrow length bins for the smaller lengths, but fewer but wider length bins for the larger lengths.	Development 0%
d. Tags inform movement	Include an option which allows the tagging data to inform movement only rather than movement and mortality.	Development 100%; Testing 90%
Other developments (* those added to the workplan after July 2022)		
Task	Description	Status of completion
Catch-conditioned	Testing of catch-conditioned deterministic and stochastic projections, both catch- and effort-conditioned fisheries.	Development 100%; Testing 100%
CPUE likelihood	Survey fisheries CPUE indices likelihood, options for grouped (stationary) catchability and concentrated formulation, with varying penalty weights.	Development 100%; Testing 100%
Length-based selectivity *****	Re-instate the pre-existing feature (fish_flags(26)=3) to estimate selectivities as function of length rather than age.	Development 100%; Testing 100%
Dependent variables *****	Add abbreviated derivatives calculation of selected dependent variables to the *.var report.	Development 100%; Testing 100%
Enhancements to .par *****	Add a new section for the first model year; modify the total objective function term to the negative log-likelihood	Development 100%; Testing 100%
Orthogonal-polynomial penalty *****	Implementation and testing the revised orthogonal-polynomial recruitment penalty function – trial the corrected version in likelihood profiles, and single- or multi-region models	Development 100%; Testing 100%
Recruitment random effects	Estimate the autocorrelation and variance of recruitment deviates as random effects rather than as independent parameters. Attempt to extend this functionality also to effort deviates.	Development 60%; Feasibility has been assessed
Constrain deviates	Apply constraints on the recruitment deviates such that the $\bar{x}$ = 0. Apply also to the effort deviates for fisheries with missing effort data for the complete time series.	Development 10%; Testing 0%
Tags inform growth	Development of a feature that incorporates size data from tag recaptures to inform growth estimation.	Development 0%; Testing 0%
EM fit only projection pars	For the “assessment” estimation model (EM) embedded in a management procedure, only fit parameters relating to the “new” data provided for the projection time periods, i.e. for effort devs, catchability devs, recruitment devs, while holding all other parameters fixed at the initial values.	Development 80%; Testing 0%
OM size comps	Generate a report of the OM size compositions for projection period without error at the end	Development 0%; Testing 0%

	of the projection period as required for deriving economics-based indicators.	
Turing test	Ensure the quality of pseudo-observations to be made more realistic by: • Including the sel_dev_coffs and eff_devs estimates in applying process error in projection size compositions and effort • Including over-dispersion error in tagging data	Development 0%; Testing 0%
Stochastic projection functionality	• Implement process error in future recruitments with application of the derived autocorrelation coefficient in historical recruitment estimates • Fix a bug in generating inputs for stochasticity in $N_{terminal}$ (more stable method is to use terminal year less 5 as the period for obtaining variance) and eff_devs	Development 0%; Testing 0%

Table 2. Proposed workplan for MULTIFAN-CL in 2023-24 and subsequent years, and those for which implementation and testing is to be completed.

Peer review recommendations		
Task	Description	Implementation
b. Non-uniform size bins	Allow the length bins to be of different widths. One might, for example, want many narrow length bins for the smaller lengths, but fewer but wider length bins for the larger lengths.	2023-24
d. Tags inform movement	Testing for the case of multi-species/stocks/sexes.	2023-24; Development 100%; Testing 90%
Other developments		
Task	Description	Implementation
Catch-conditioned method	Add the fml_effort_rltnshp penalty term to the members of the class used for creating test_plot_output.	2023-24
	Allow the fml_effort_rltnshp penalty calculation to be conditional on a fish_flags(fi) that facilitates it being disabled, or specified, only for particular fisheries.	2023-24
	Review the operation of existing control phase routines undertaken in Phase 1 in respect of their suitability for a catch-conditioned model, and draft a new control phase specific to the catch-conditioned model as needed.	2023-24
	Document the catch-conditioned method, and flags for Manual	2023-24
	Fishing mortality estimation for multi-species	2023-24
CPUE likelihood	Review the formulation of the concentrated log-normal likelihood formulation as used in CASAL (Bull et al. 2012) for its suitability for implementation in MULTIFAN-CL.	2023-24
Hessian operations	Debug the running of the Hessian calculations that interact with the test_plot_output report; replaces 0s in for all of the length comp likelihood entries	2023-24
User’s Guide	Documentation of the catch-conditioned method and flags	2023-24
Tasks from the Yellowfin Tuna Independent Peer Review Panel 2022		
Length-weight variance	Extend MULTIFAN-CL so that variability in weight-at-length can be taken into account	2023-24
Selectivity splines	Extend MULTIFAN-CL so that it is possible to specify the number of spline knots when defining selectivity and where they are located	2023-24

	with respect to age (length) as the current approach means that the selectivity for some knots is constrained to zero.	
Age-reading error	Extend MULTIFAN-CL so that account can be taken of age-reading error when fitting to conditional age-at-length data.	2023-24
CPUE overdispersion	Add the ability to specify overdispersion in CPUE as an additive rather than multiplicative factor.	2023-24
Natural mortality at age	Integrate the calculation of M-at-age from the sex-ratio data into MULTIFAN-CL unless a sex-specific assessment is used.	2023-24
Tasks rolled over from 2022-23 and newly added		
Outstanding testing of existing features	Multi-sex model projections	2023-24
Simulation CPUE data	Implement the generation of simulation observed CPUE indices for the index fisheries of a catch-conditioned operating model.	2023-24
PDH diagnostics	Refine and describe the algorithm of positive definite Hessian (PDH) diagnostics to be done for an assessment; add a unique identifier to the Hessian file, therefore its parallelised components, and possibly to the *.par, to ensure continuity among them when stitching parallel *.hes files.	2023-24
Evaluations report	Report the number of function evaluations completed during the minimisation in an output file.	2023-24
macOS compilation	Develop an arrangement for the Mac Mini PC to be connected to the Pacific Community computing network, to facilitate compilations for macOS and testing of the executable versions.	2023-24
Recruitment random effects	Report on the feasibility of its implementation in MULTIFAN-CL.	post-2023-24
Constrain deviates	Apply constraints on the recruitment deviates such that $\bar{x} = 0$. Apply also to the effort deviates for fisheries with missing effort data for the complete time series.	post-2023-24
Tags inform growth	Development of a feature that incorporates size data from tag recaptures to inform growth estimation.	post-2023-24
Length-based selectivity	Resolve the anomalies that produce undesirable discrepancies in predicted size compositions.	post-2023-24
Self-scaling multinomial with random effects	Complete draft paper for peer review.	post-2023-24
Recruitment correlates	Region-specific environmental recruitment correlates estimated within the orthogonal polynomial parameterisation for recruitments	post-2023-24
Movement correlates	Add a time-series structure (e.g., random walk, time blocks or using environmental correlates) to movement coefficients	post-2023-24
Recruitment deviate penalties	Allow for time-variant penalties on recruitment deviate estimates	post-2023-24
Tagging multi-sex	Account for instances of differences between size composition the tag releases and the sex-specific populations	post-2023-24
Region specific SRR	Allow that region-specific spawning biomass is responsible for recruitment within that region. This is consistent with the assumption that stocks may not be truly panmictic. This would estimate region-specific SRRs.	post-2023-24
Report comments	Add comment descriptions of the selectivity parameter configurations in the output .par and .rep reports	post-2023-24
EM fits only projection parameters	For the “assessment” estimation model (EM) embedded in a management procedure, only fit parameters relating to the “new” data provided for the projection time periods, i.e., for effort devs, catchability devs, recruitment devs, while holding all other parameters fixed at the initial values.	post-2023-24

OM size comps	Generate a report of the OM size compositions for projection period without error at the end of the projection period as required for deriving economics-based indicators.	post-2023-24
Stochastic projection functionality	- implement process error in future recruitments with application of the derived autocorrelation coefficient in historical recruitment estimates - stochastic variability in terminal numbers at age - consolidate the generation of stochasticity in effort_dev_coffs	post-2023-24
Turing test	Ensure the quality of pseudo-observations to be made more realistic by: - Including the sel_dev_coffs and effort_dev_coffs estimates in applying process error in projection size compositions and effort - Including over-dispersion error in tagging data	post-2023-24

11 FIGURES

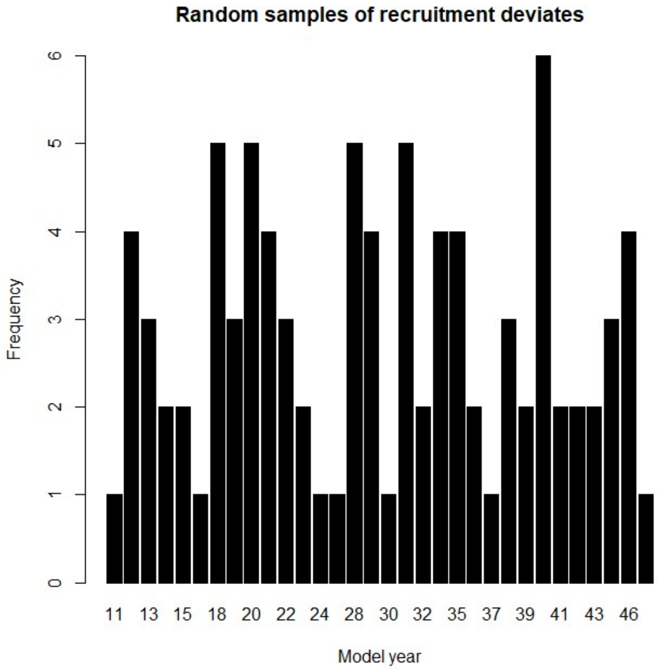


Figure 1. The frequency distribution of the randomly sampled annual BH-SRR deviates, or absolute recruitments, as applied in the stochastic simulation projections, nsim=3, (note the BH-SRR was fitted to a sub-set of annual recruitments, year 11 to 47).

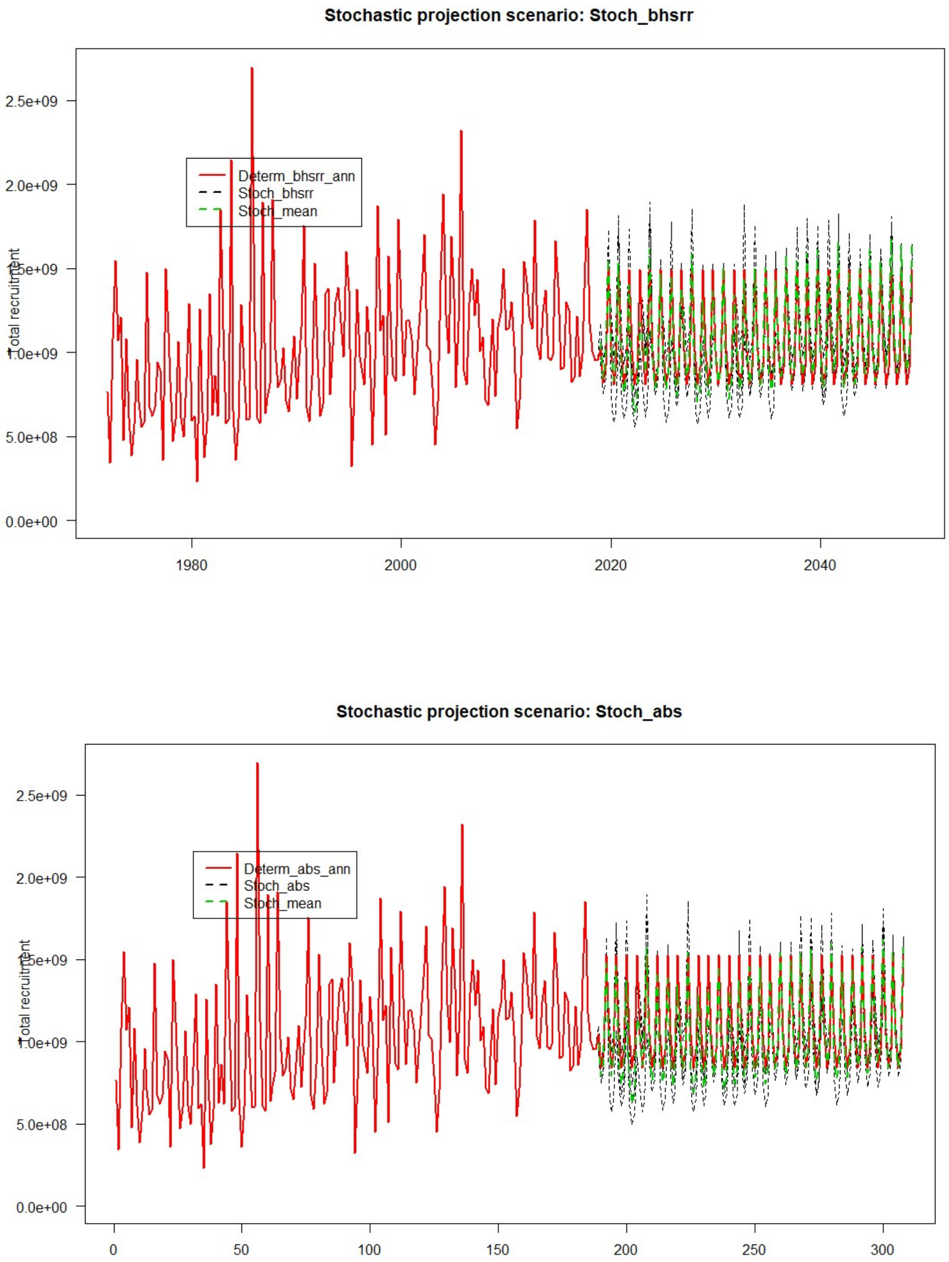


Figure 2. Absolute recruitments in the estimation and projection (>2018) time periods for the: deterministic (Determ_bhsrr_ann, Determ_abs_ann), and stochastic (Stoch_bhsrr, Stoch_abs) projection models using the bh_devs and abs_smp configurations, respectively, with the mean of the annual stochastic recruitments plotted (Stoch_mean).

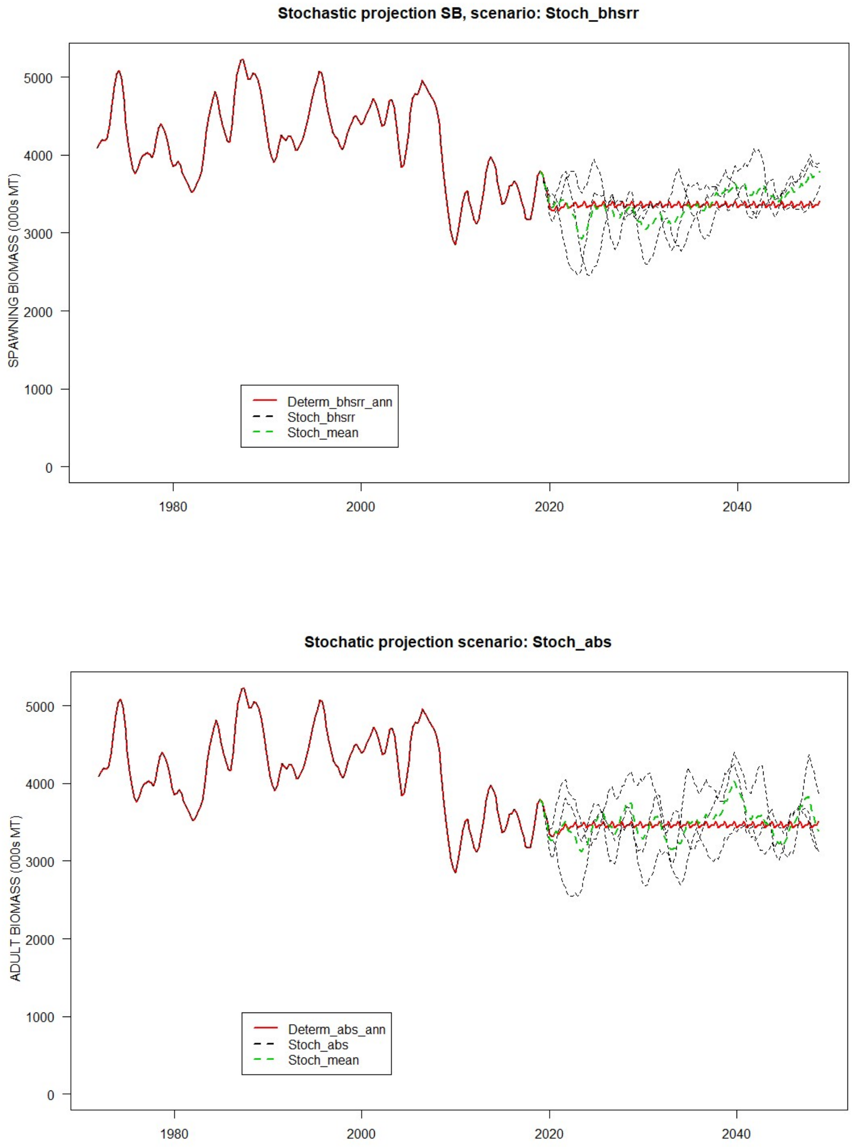
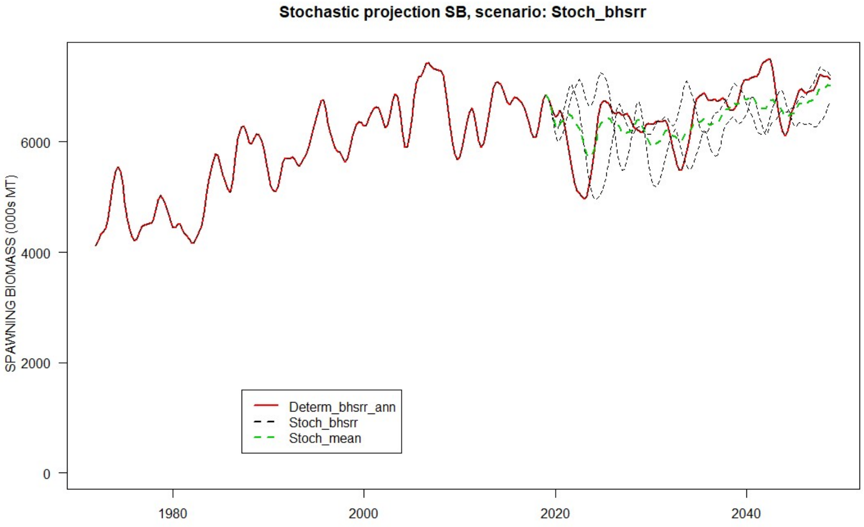


Figure 3. Absolute adult biomass in the estimation and projection (>2018) time periods for the: deterministic (Determ_bhsrr_ann, Determ_abs_ann), and stochastic (Stoch_bhsrr, Stoch_abs) projection models, for the bh_devs (top panel) and abs_smp (bottom panel) configurations, respectively; with the mean of the annual stochastic recruitments plotted for each configuration (Stoch_mean).



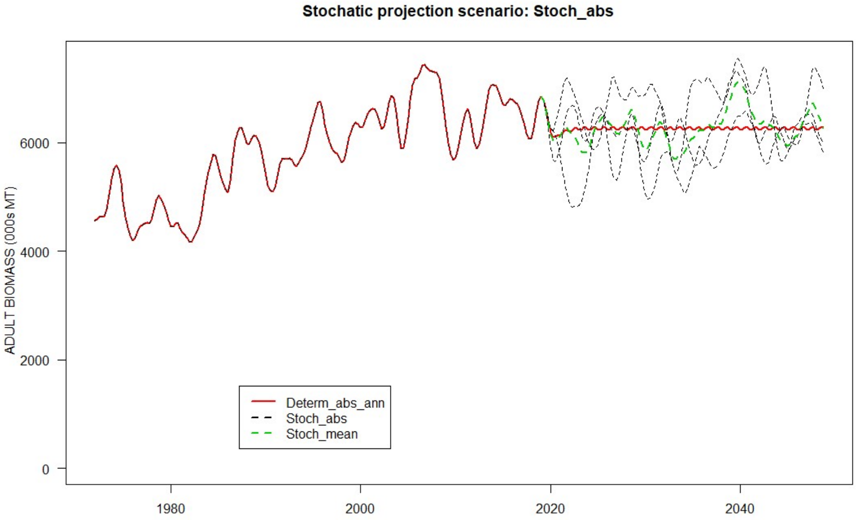
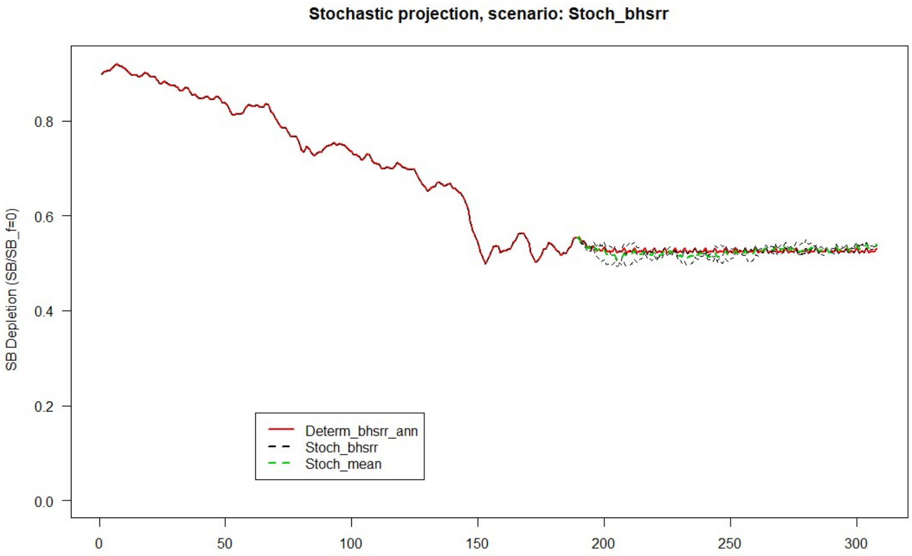


Figure 4. Absolute adult biomass in the estimation and projection (>2018) time periods for the: deterministic (Determ_bhsrr_ann, Determ_abs_ann), and stochastic (Stoch_bhsrr, Stoch_abs) projection models for the F0eval model evaluations, for the bh_devs (top panel) and abs_samp (bottom panel) configurations, with the mean of the annual stochastic adult biomass plotted (Stoch_mean).



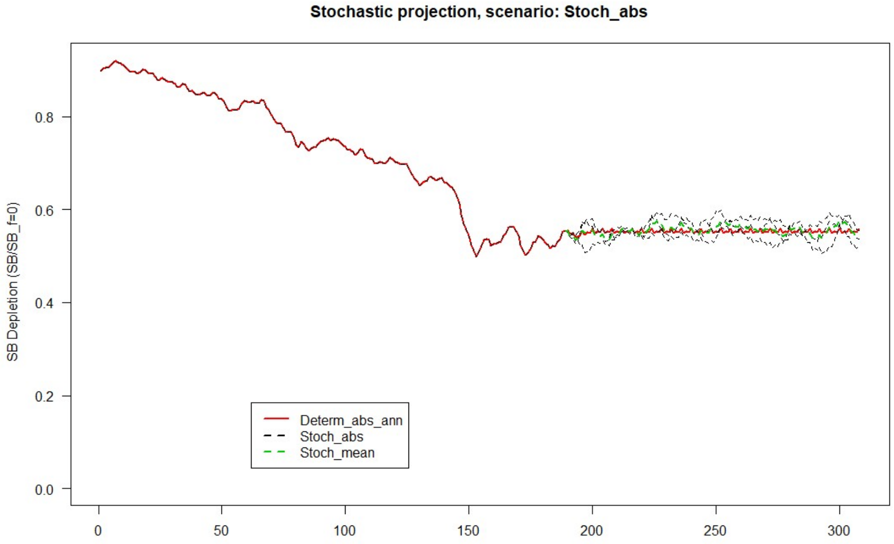
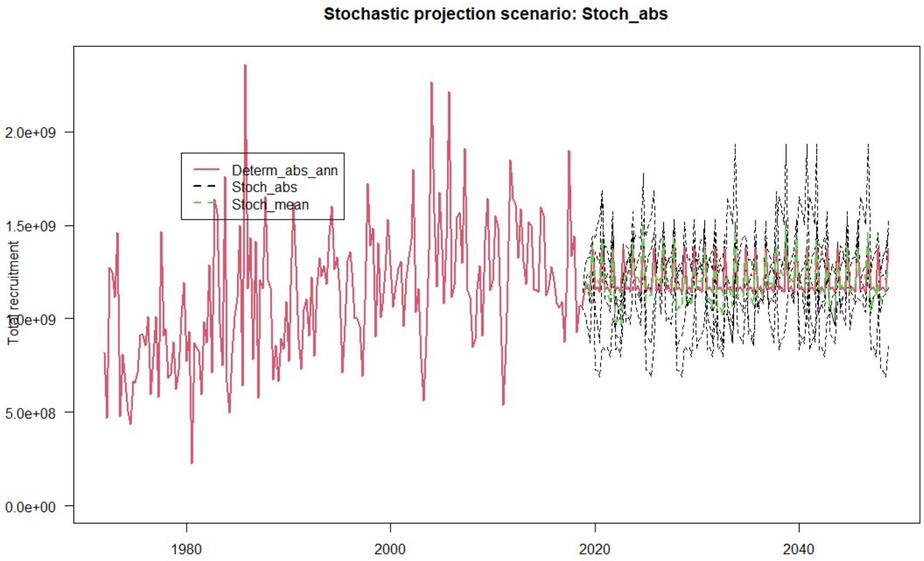


Figure 5. The adult biomass depletion level (SB/SBF=0) in the estimation and projection (>2018) time periods for the: deterministic (Determ_bhsrr_ann, Determ_abs_ann), and stochastic (Stoch_bhsrr, Stoch_abs) projection models, for the bh_devs (top panel) and abs_samp (bottom panel) configurations, with the mean of the annual stochastic depletion plotted (Stoch_mean).



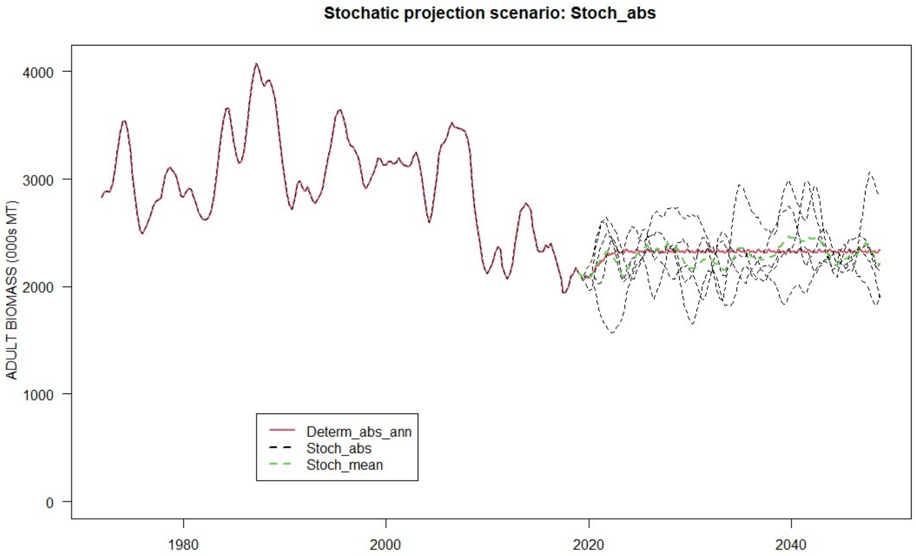


Figure 6. Comparison of the catch-errors model absolute recruitments and adult biomass (top and bottom panels, respectively) in the estimation and projection (>2018) time periods for the: deterministic (Determ_abs_ann), and stochastic (Stoch_abs) projection models, with the mean of the annual stochastic recruitments and adult biomass plotted (Stoch_mean).

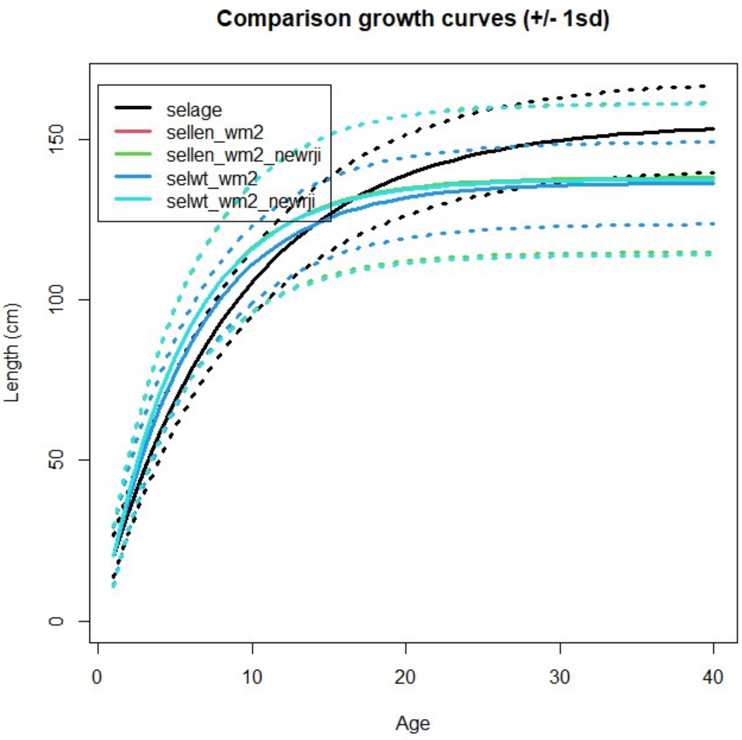


Figure 7. Estimated growth functions in terms of mean length-at-age (solid lines) and standard deviation (dashed lines). Black line – selage. Red line – sellen_wm2; green line – sellen_wm2_newrji; dark blue line - selwt_wm2; light blue line – selwt_wm2_newrji.

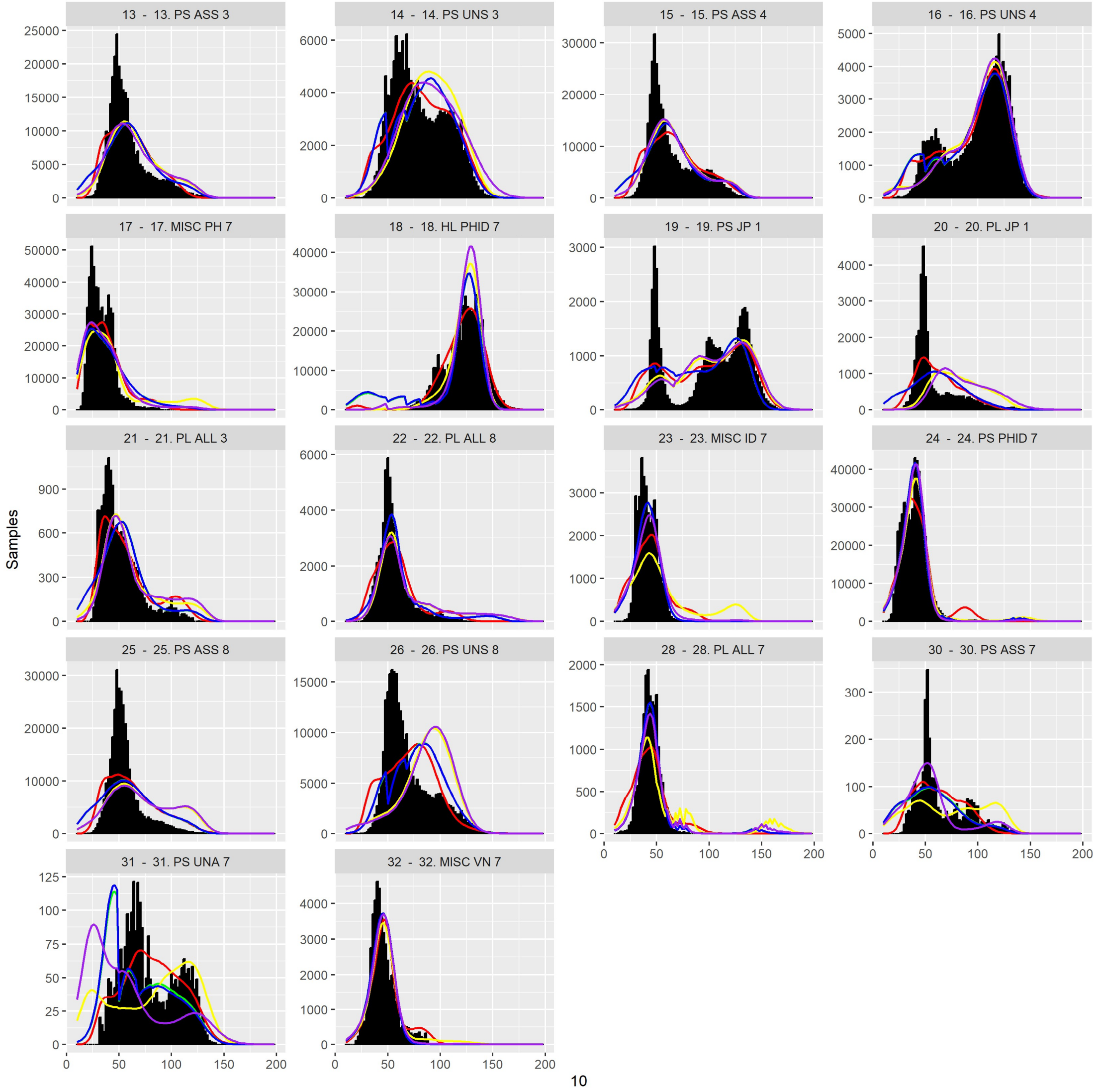


Figure 8. Summary fits of fishery-specific length compositions (aggregated over all samples) to the observations (histograms) for the models: selage (red line); sellen_wm2 (green line); sellen_wm2_newrji (yellow line); selwt_wm2 (blue line); and, selwt_wm2_newrji (purple line).

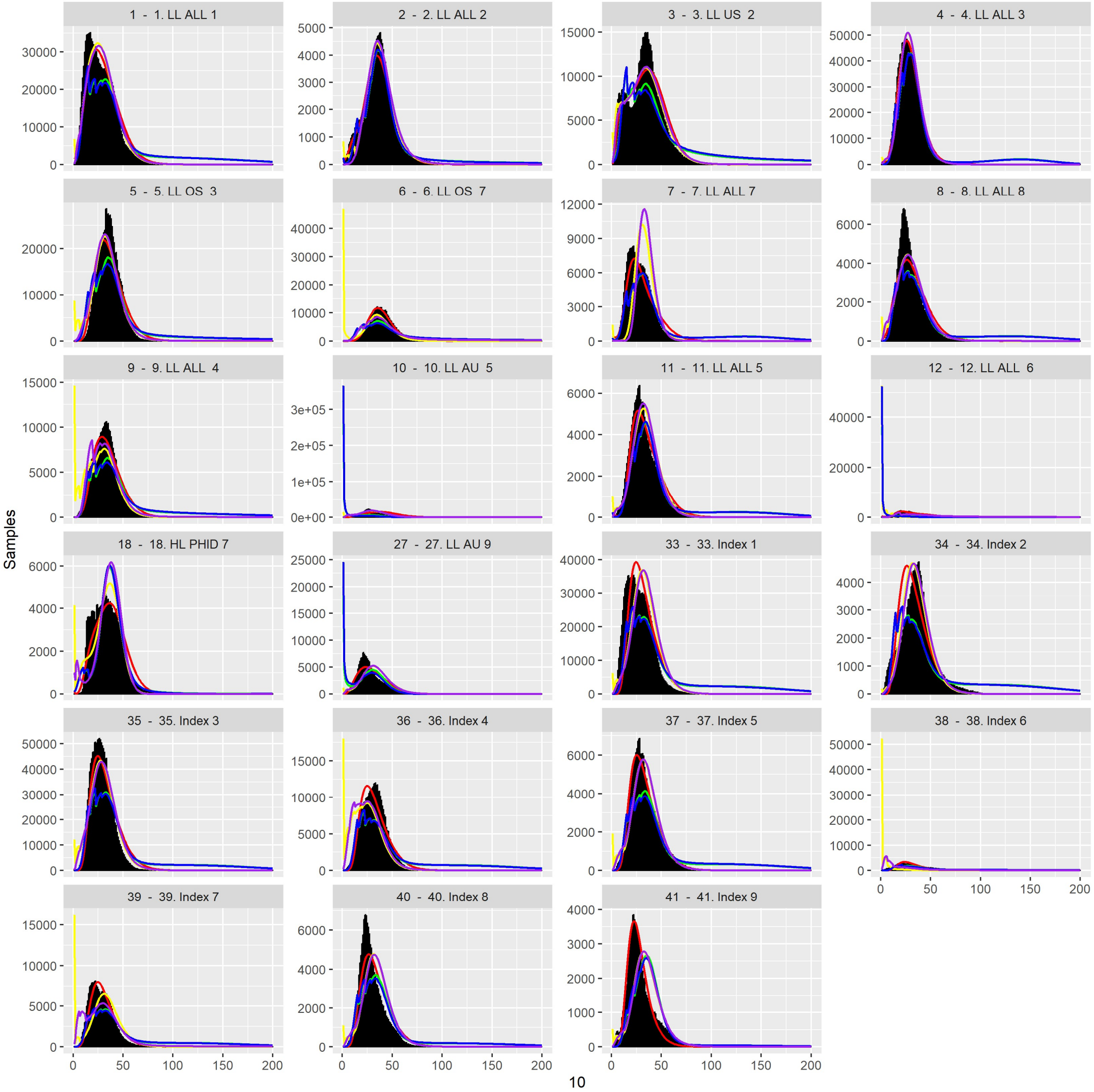


Figure 9. Summary fits of fishery-specific weight compositions (aggregated over all samples) to the observations (histograms) for the models: selage (red line); sellen_wm2 (green line); sellen_wm2_newrji (yellow line); selwt_wm2 (blue line); and, selwt_wm2_newrji (purple line).

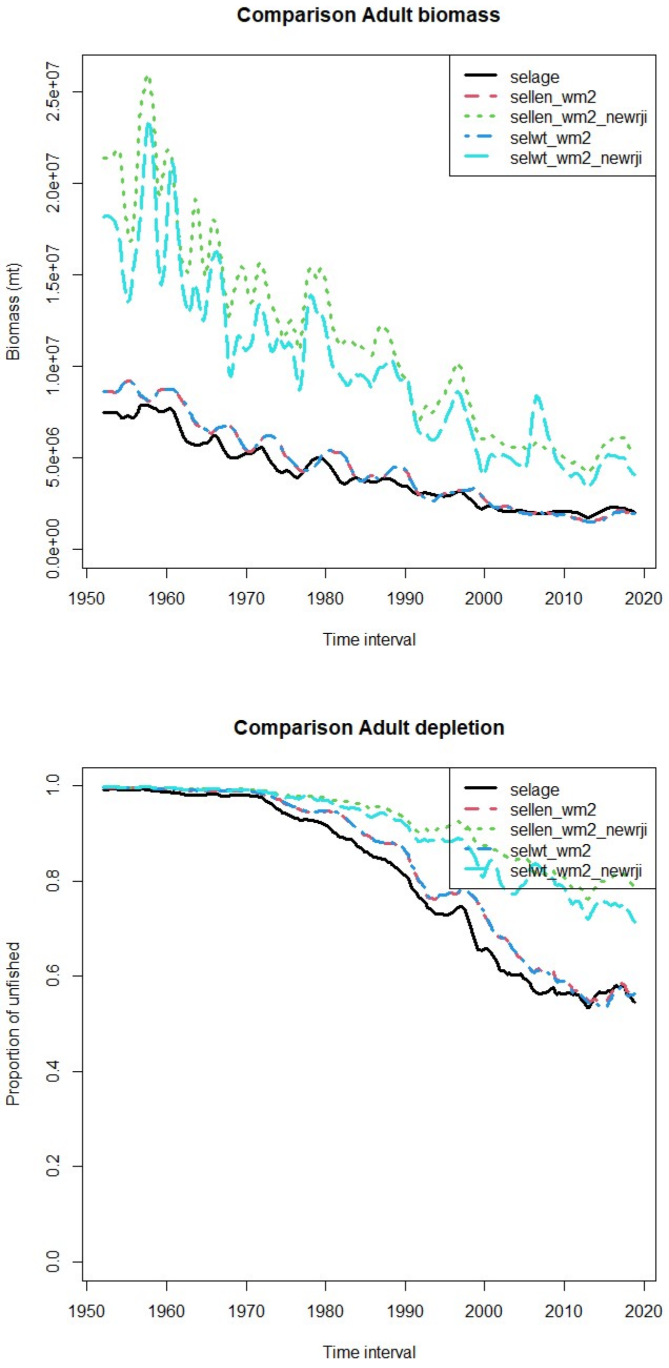


Figure 10. Comparison of the adult biomass (top panel) and adult biomass depletion (bottom panel) trajectories among the models employing either: age-based selectivity (selage), or length-based selectivity (sellen_wm2, sellen_wm2_newrji, selwt_wm2, and selwt_wm2_newrji).

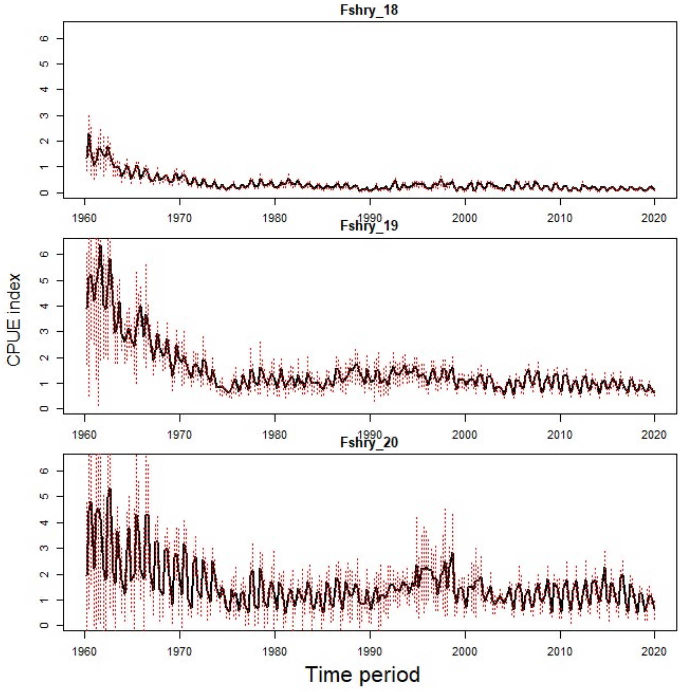


Figure 11. Observed CPUE indices of the grouped survey fisheries 18 to 20 in the three respective regions.

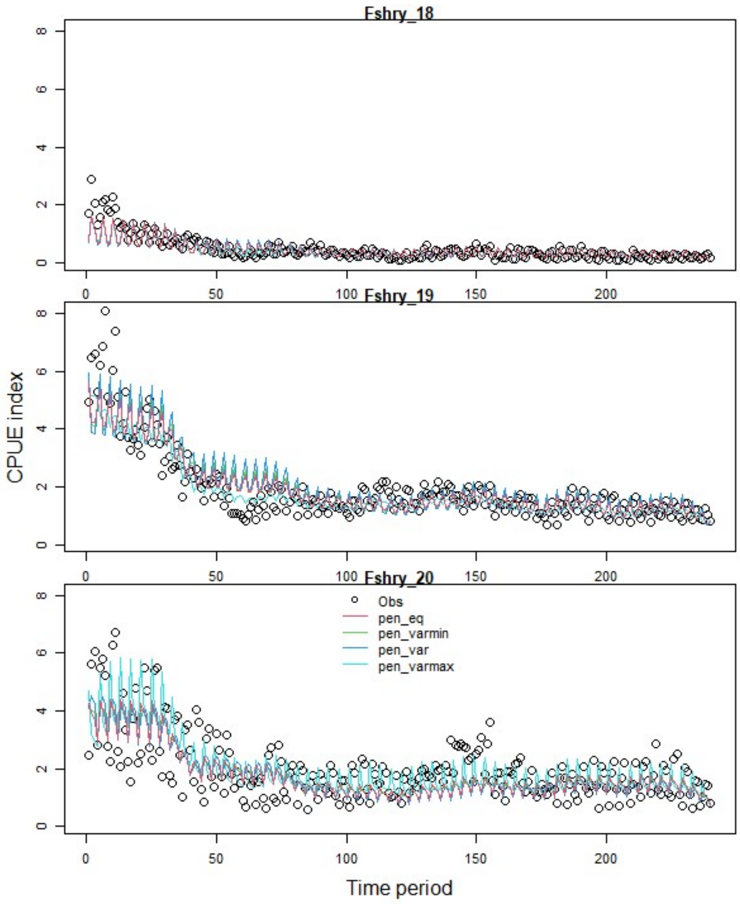


Figure 12. Fits of the non-concentrated likelihood models to the observed CPUE time series.

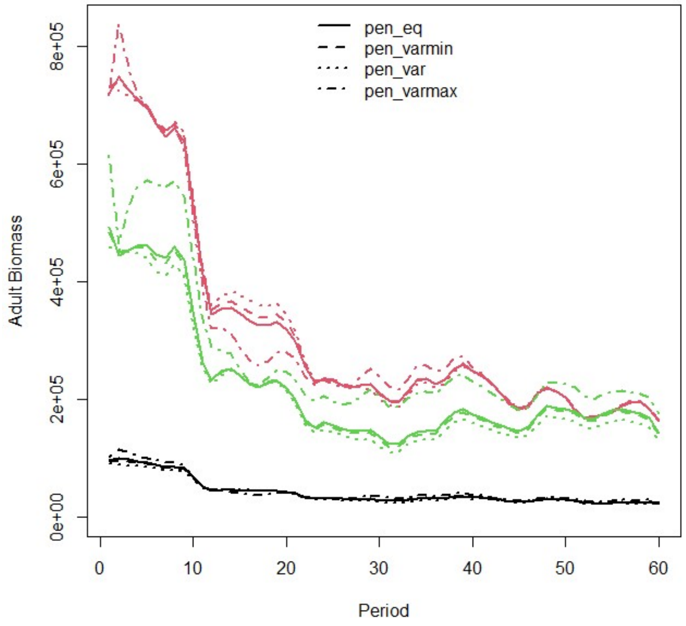


Figure 13. Model derived variable: adult biomass in each region corresponding to: fishery 18 (black lines), fishery 19 (red lines), and fishery 20 (green lines), for the non-concentrated likelihood models.

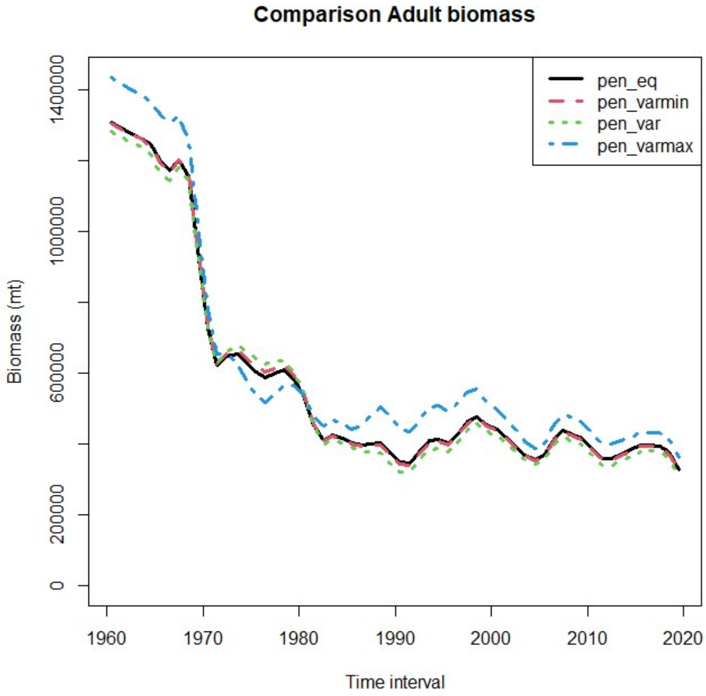


Figure 14. Derived variable: total adult biomass among all regions as estimated by the range of non-concentrated likelihood models.

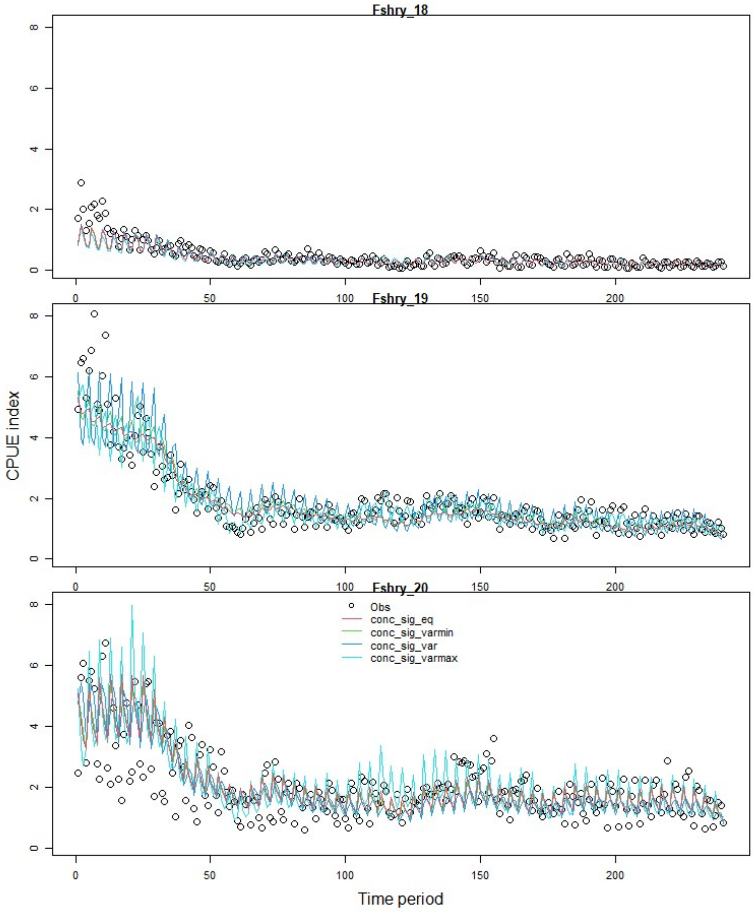


Figure 15. Fits of the concentrated likelihood models to the observed CPUE time series.

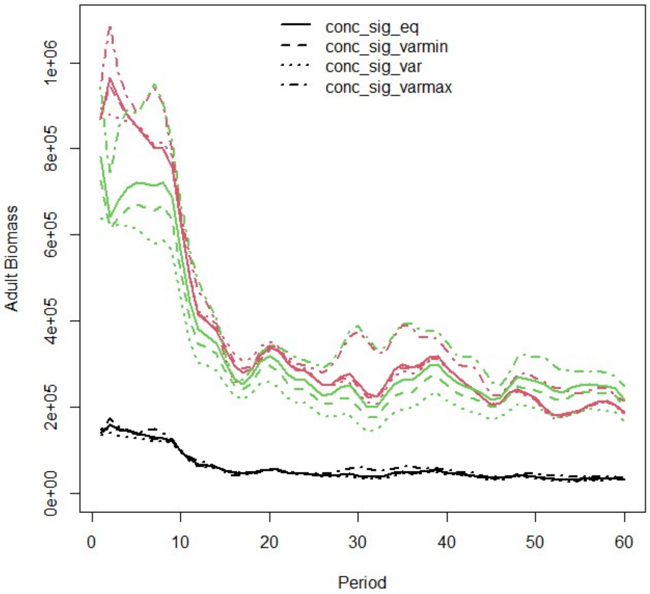


Figure 16. Model derived variable: adult biomass in each region corresponding to: fishery 18 (black lines), fishery 19 (red lines), and fishery 20 (green lines), for the concentrated likelihood models.

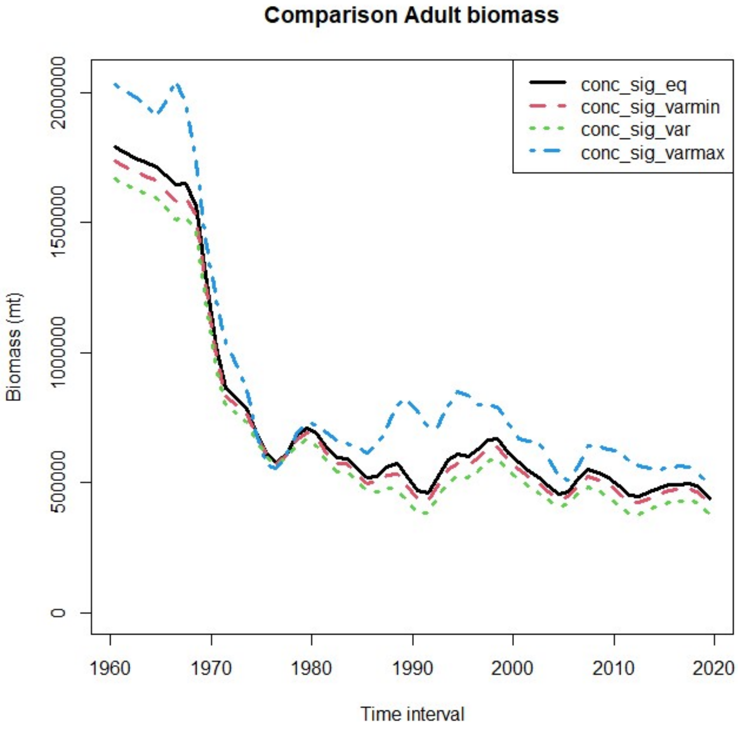


Figure 17. Derived variable: total adult biomass among all regions as estimated by the range of the concentrated likelihood models.

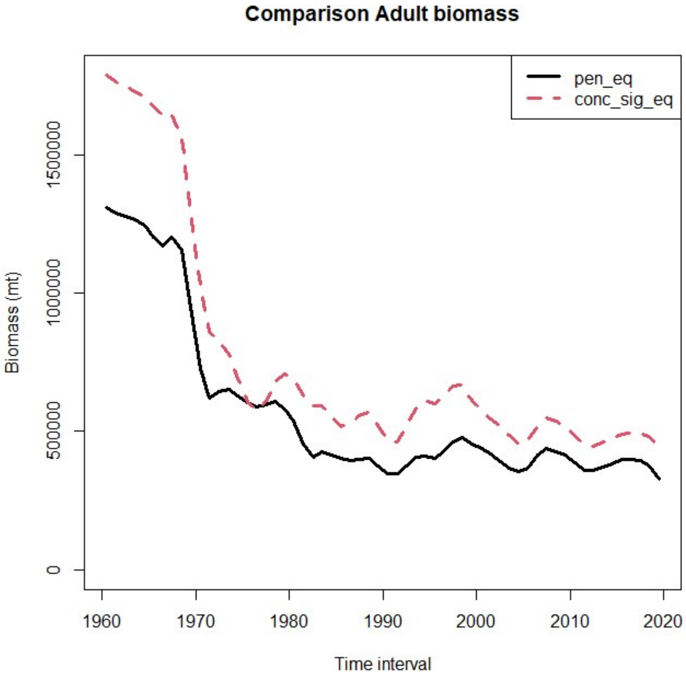


Figure 18. Derived variable: total adult biomass among all regions as estimated by the non-concentrated and concentrated likelihood models assuming constant penalty weight among the fisheries.

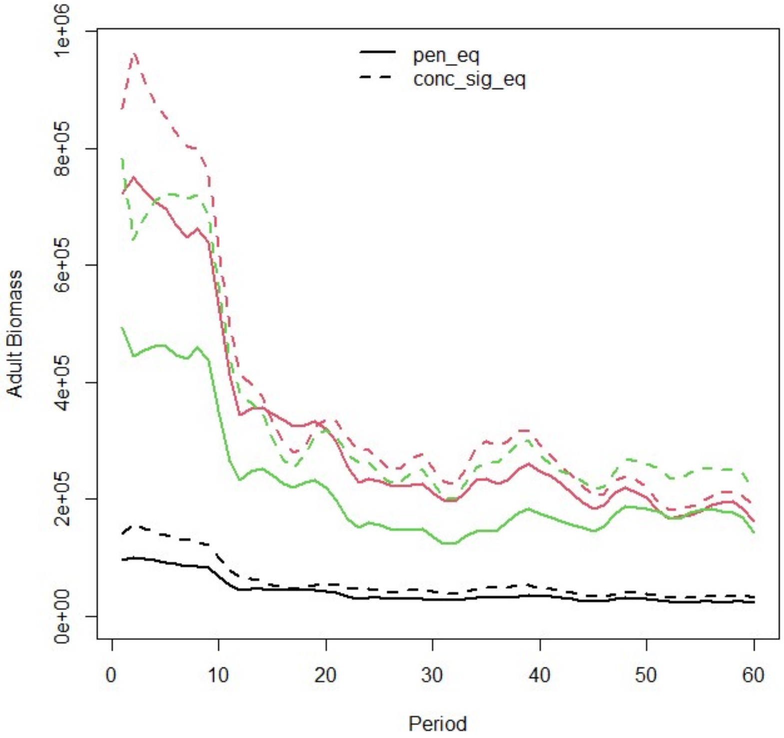


Figure 19. Model derived variable: adult biomass in each region corresponding to: fishery 18 (black lines), fishery 19 (red lines), and fishery 20 (green lines), for the non-concentrated and concentrated likelihood models assuming constant penalty weight among the fisheries.

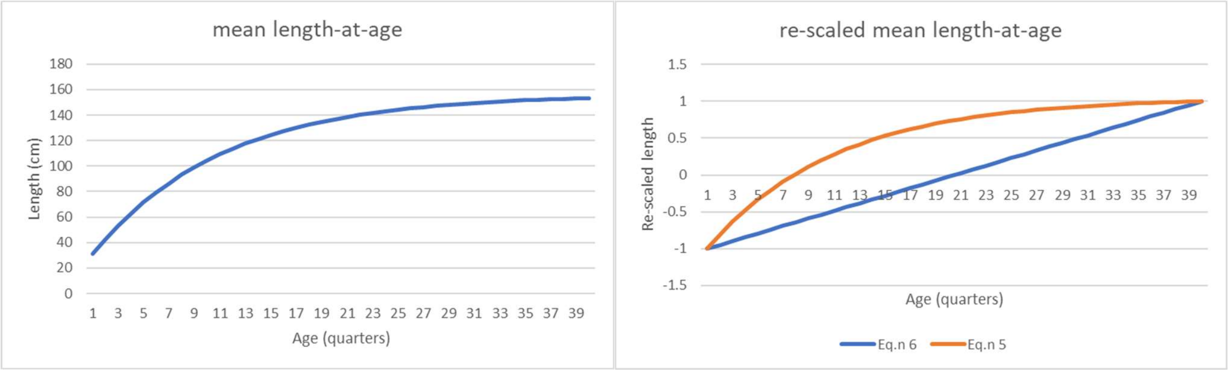


Figure 20. Mean length-at-age derived using von Bertalanffy parameters (left panel), and the re-scaled values on a scale of -1 to 1 on the right panel derived using equation orig_code_u and Eq.n 5 of this review (right panel).

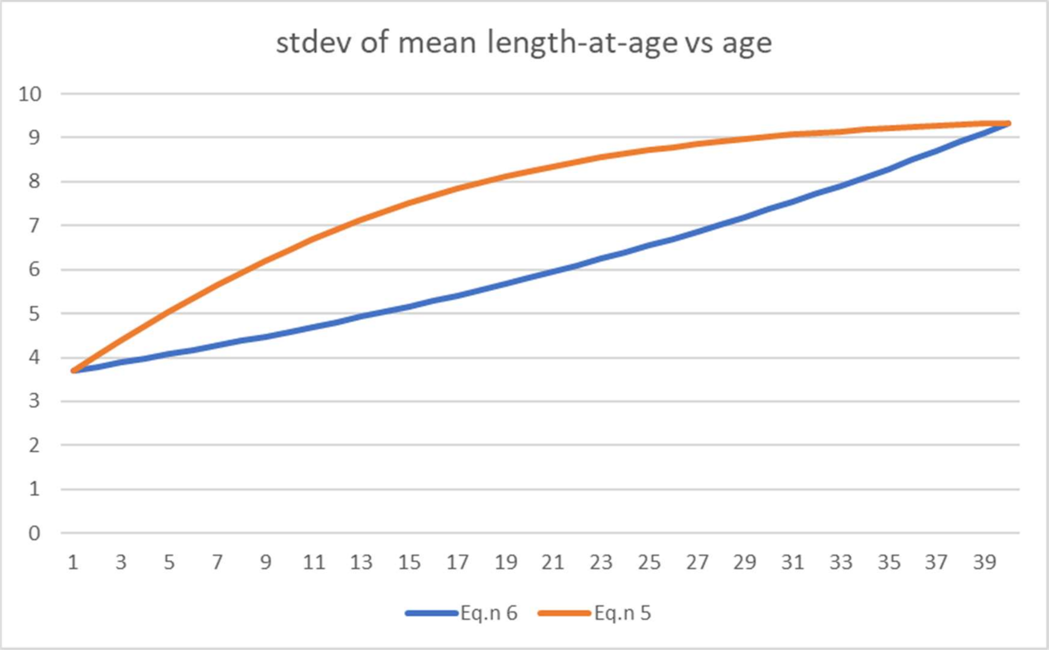


Figure 21. Standard deviation of mean length-at-age derived using the re-scaled mean length-at-age from equations: Eq.n 5 and Eq.n 6.

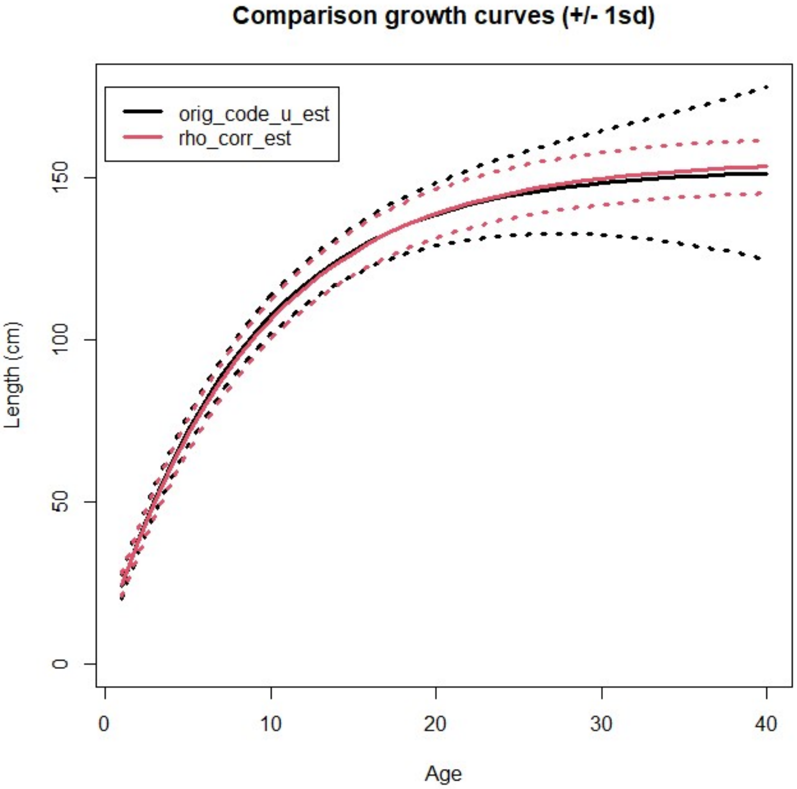


Figure 22. Comparison of mean length-at-age and standard deviation for model estimates of growth when implementing equations: Eq.n 5 (rho_corr_est) and Eq.n 6 (orig_code_u_est).

12 ANNEX 1

Modified source code files (63) during 2022-23.

lesmatrix.cpp
newrshimp.cpp
recinpop_orth.cpp
newmult.hpp
plot.cpp
vrbioclc.cpp
do_all_for_empirical_autocorrelated_bh.cpp
testnewl3.cpp
gradrout.cpp
testeq1.cpp
alldevpn.cpp
alllengthsel.cpp
fvar_a29.cpp
fvar_m24.cpp
fvar_m58.cpp
len_dm_nore.cpp
lmul_io2.cpp
lmul_io4.cpp
lognormal_mult_ll_cs.cpp
lognormal_multinomial4.cpp
new_cross_derivs.cpp
new_incident_calc.cpp
newl2.cpp
newl5.cpp
newm_io3.cpp
newmaturity.cpp
newmau5a.cpp
newmaux5.cpp
newmovement2.cpp
newmult.cpp
nfast96.cpp
nfast96a.cpp
nnewlan.cpp
nrcatch3.cpp
nrcatch4.cpp
old_new_cross_derivs.cpp
onevar.cpp
plotstuff.cpp
print_survey.cpp
printfrq.cpp
selbreaks.cpp
short2.cpp
sim_realtag_pd.cpp
sim_tag_pd.cpp
spline.cpp
tag3.cpp
temppred.cpp
thread_stuff.cpp
thread_stuff2.cpp
thread_stuff3.cpp
thread_stuff4.cpp
thread_stuff5.cpp
thread_stuff6.cpp
thread_stuff7.cpp
thread_stuff8.cpp
threaded_tag3.cpp
variable.hpp
VERSION
version.h
wfast96.cpp
wght_dm_nore.cpp
xthread_stuff.cpp
ythread_stuff.cpp

No new source code files were added during 2022-23.