



**SCIENTIFIC COMMITTEE
THIRTEENTH REGULAR SESSION**

Rarotonga, Cook Islands

9-17 August 2017

DEVELOPMENTS IN THE MULTIFAN-CL SOFTWARE 2016-17

WCPFC-SC13-2017/SA-IP-05

Nick Davies³, Dave Fournier², Yukio Takeuchi¹, Fabrice Bouyé¹, and John Hampton¹

**Oceanic Fisheries Programme
The Pacific Community, Noumea, New Caledonia**

¹ Oceanic Fisheries Programme, The Pacific Community, Noumea, New Caledonia

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

³ Te Takina Ltd, Whangarei, New Zealand

Table of Contents

Executive summary	4
1 Introduction.....	5
2 Development overview	5
2.1 Team	5
2.2 Calendar.....	6
2.3 Collaboration and versioning	6
2.4 Compilation framework and Source code repository	6
2.4.1 Compilation framework	6
2.4.2 Visual Studio 2017 compilation of Windows executable	6
2.4.3 Upgrade to gcc-5.4.0 compiler	7
2.4.4 Source code repository	7
2.5 Developer’s workshops	8
2.6 Benchmark testing 2016-17	8
2.6.1 Preceding the Sep.2016 developer’s workshop	8
2.6.2 Version 2.0.2.1.....	8
2.6.3 Version 2.0.2.2.....	9
2.6.4 Visual Studio compilation of Windows executable.....	9
2.6.5 Test of the improved feature for stochastic recruitments.....	9
2.7 Postings to website.....	10
2.7.1 24 February 2017 – version 2.0.2.1.....	10
2.7.2 6 April 2017 – version 2.0.2.2.....	10
2.8 Independent Peer Review of the 2011 bigeye tuna stock assessment.....	10
2.9 Tool development	11
2.9.1 R4MFCL.....	11
2.9.2 Testing framework	11
2.9.3 Viewer	12
2.9.4 Condor parallel processing facility	12
2.10 User’s guide	13
3 Training workshop.....	13
4 New features	13
4.1 Self-scaling Multinomial plus random effects (SSMULT_RE)	14
4.1.1 Rationale	14
4.1.2 Methods and Testing.....	14
4.2 Dirichlet-Multinomial Likelihood.....	15
4.2.1 Methods and Testing.....	16
4.3 Tail Compression	16

4.4	Simulation mode	17
4.4.1	Size compositions	17
4.4.2	Relative abundance indices	17
4.4.3	Tagging data	18
4.4.4	Status and future work	19
4.5	Autocorrelation in recruitments	19
4.6	Functional forms for age-specific natural mortality	20
4.7	Likelihood profiling	20
5	Other enhancements and bug fixes	21
5.1	Extension of multi-sex model feature	21
5.1.1	Length-specific selectivity shared among sexes	21
5.1.2	Various combinations of aggregated/disaggregated fisheries data	22
5.1.3	Sex ratio of predicted catch compositions	22
5.1.4	Identifiers for sex in output reports	22
5.2	Improved standard deviation calculation	22
5.3	Improved formulation for logistic selectivity	22
5.4	Bug fixes	23
5.4.1	Fix to option for increasing the number of spline nodes	23
5.4.2	Fix to the age of common selectivity	23
5.4.3	Fix to equilibrium yield curve calculation for multi-sex model	23
5.4.4	Fix to initial population numbers in maximum age class for multi-species/sex models	23
5.4.5	Fix to the input of size composition data	23
5.4.6	Fix to the long int definition for the Windows compilation	24
5.4.7	Fix to the catch calculation for tagged fish	24
5.4.8	Fix to fish_flags used for DM likelihood	24
5.4.9	Fix to parameter labels in xinit.rpt	24
5.4.10	Fix to missing components in report "test_plot_output"	24
6	Application of new features	24
7	Future work	25
8	Discussion	25
9	References	26
10	Tables	28
11	Figures	32
12	Annex	37
12.1	Benchmark testing	37
12.2	Agenda for Training Workshop, March 2017	40

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) from August 2016 to July 2017 and updates the report of Davies et al. (2016).

The progress made on implementing new features included:

- Refining and testing the implementation of an innovative approach for fitting to size composition data, using the self-scaling Multinomial with random effects (SSMULT_RE), and the development of an alternative approach, the Dirichlet-Multinomial likelihood (DM).
- Autocorrelation may be estimated in either the independent log-normal recruitments (relative to the mean), or the BH-SRR estimated deviates (relative to the BH-SRR prediction).
- Estimating functional forms for age-specific natural mortality using either a spline function, or a function consistent with that described by Lorenzen (1996).
- A likelihood profile calculation in respect of the derived model quantities for average absolute biomass or a level of biomass depletion.
- A simulation mode for running MULTIFAN-CL as an operating model for generating pseudo-observations from stochastic projections of size composition data, relative abundance indices, and a working algorithm has been developed for generating tagging data.

Refinements were made to the feature added in 2015-16 for tail compression of size composition data, and a number of improvements and corrections were made to the multi-sex feature. Improvements were also made to the standard deviation calculations and the formulation of the logistic selectivity function.

The new features added, and the improvements made to existing features, were of immediate benefit for the 2017 assessments. The first four features listed above were applied for investigative model runs, sensitivity analysis, or in the final grid of models used to formulate advice for bigeye, yellowfin, or swordfish (McKechnie et al. 2017, Tremblay-Boyer et al. 2017, Takeuchi et al. 2017). The improvements made to the multi-sex feature were pivotal to investigating this model structure as an option for the swordfish assessment (Takeuchi et al. 2017). The fifth feature listed above was used to illustrate its potential for using MULTIFAN-CL within the management strategy evaluation (MSE) framework being developed at OFP (Scott et al. 2017).

As in 2016, a 3-day training workshop for stock assessment analysts was held at Nouméa, during March 2017. This assisted in making the new features accessible to those undertaking the 2017 stock assessments. It is recommended that these workshops be a routine calendar item, because they facilitate the rapid use of new features, and participants can identify improvements to MULTIFAN-CL that make assessment modelling more efficient, e.g. a faster Windows executable.

Ianelli et al. (2012) reported thirteen recommendations from an independent peer review panel specifically relating to MULTIFAN-CL. The rationale for these recommendations is to address the key areas of uncertainty for the tuna assessments reported to the Scientific Committee of the Western and Central Pacific Fisheries Commission (SC), through improving the biological description of population processes, better estimation of observation error, and the improved modelling of fishing mortality. Four of the recommendations remain yet to be implemented in MULTIFAN-CL and this was not addressed during 2016-17. Rather, resources were directed to consolidating and improving the methods for fitting to size composition data. This was because of the importance of these data in WCPO tuna stock assessments. The model uncertainty due to assumptions made for the relative importance amongst the various data types included in the integrated modelling approach has dominated most fisheries assessment models. The self-scaling properties of the SSMULT_RE and DM approaches allow effective sample sizes to be estimated based upon robust statistical methods, and this offers a way to reduce this area of model uncertainty. Consolidating their development has therefore been a priority for 2016-17. Also, a number of tasks arose during 2016-17 that were unforeseen and were a priority for the 2017 assessments, including making improvements to the multi-sex feature, and the developing functional forms for age-specific natural mortality. It is therefore reasonable to maintain some flexibility in the priority of project tasks, and the four recommendations of Ianelli et al. (2012) will be part of the project plan for 2017-18.

The key components recommended for the MULTIFAN-CL project during 2017-18 are itemised below:

Peer review recommendation (Ianelli et al. 2012)	Implementation
b. Non-uniform size bins	Development
c. Long-term tag loss	Development
d. Tags inform movement	Development
k. Maturity-at-length	Development
Other new features	Implementation
Self-scaling multinomial with random effects and Dirichlet-Multinomial	Simulation testing, draft report
Simulation mode – generating pseudo-observations	Development
Initial biomass approximates the equilibrium unexploited biomass	Testing
Exclude reporting rates from tag predictions during mixing period	Testing
Region-specific yield estimation	Testing
Tags inform growth	Development

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 Dave Fournier of Otter Research Ltd and Dr John Hampton (The Pacific Community) for application to south Pacific albacore tuna (Fournier et al. 1998).

MULTIFAN-CL is an integrated age-structured and length-based population model which is described in detail in the User’s Guide (Kleiber et al. 2017). It is typically fitted to total catch, catch rate, size-frequency and tagging data stratified by fishery, region and time period. For example, recent tropical tuna assessments (e.g. Tremblay-Boyer et al. 2017, McKechnie et al. 2017) encompass a time period of 1952 to 2016 in quarterly time steps, and model multiple separate fisheries occurring in 9 spatial regions specified these models. 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 understanding of the fishery, to fix software errors, and to improve model features and usability. This document records changes made since August 2016 to the software and other components of the MULTIFAN-CL project both for the current release version (2.0.2.2), and the current unreleased development version, and updates the report for the previous period, 2015-16, (Davies et al. 2016).

2 DEVELOPMENT OVERVIEW

2.1 Team

The senior developer of MULTIFAN-CL is Dr Dave Fournier, of Otter Software in Canada. Assisting with programming is Nick Davies, with occasional programming carried out by Yukio Takeuchi and John Hampton (SPC, New Caledonia). Other tasks include testing and debugging (ND, JH, and Fabrice Bouye (SPC));

documentation (ND); and planning and coordination (DF, ND, and JH). Related project software are developed or managed by FB (MULTIFAN-CL Viewer, Condor, GitHub, Jenkins), ND, and Robert Scott (R4MFCL).

2.2 Calendar

August – November: Testing, planning and ongoing code development, Developer’s workshop

December – February: Testing and ongoing code development

March: Developer’s workshop

April-July: Testing, ongoing code development and support for stock assessments

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 2.4.4).

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. Some of the tasks identified during the previous reporting period (2015-16) have been addressed in the current period through the model developments completed in 2016-17. A main trunk exists for the MULTIFAN-CL source code from which release versions are posted, and a development branch has been created to hold recent versions of the source currently being developed and tested. 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 has been drafted.

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 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 the Virtual Machine (VM) that is dedicated to MULTIFAN-CL development.

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). So far, our nightly builds have a 99% compilation success rate.

Although it was intended to extend this facility to support automated builds of the Windows release executable in 2016, this was not achieved. 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 Visual Studio 2017 compilation of Windows executable

The MinGW cross-compiler has been used to build the executable for the 64-bit Windows platform but it has performance issues, with, by comparison, around half of the computational performance of the Linux executable (based on rough tests). This difference in performance is evident in recent versions of MULTIFAN-CL

and for the current released version 2.0.2.2. This posed a potential problem for intense stock assessment modelling, and this was identified by users of MULTIFAN-CL attending the March 2016 Training Workshop.

In the past Windows 64-bit compilations had been built using the Visual Studio 2010 cl compiler (used for MULTIFAN-CL compilations up to version 1.1.5.9) which exhibited comparable performance to the Linux 64-bit executable. To ascertain the cause of this difference in performance, a compilation was done using the Visual Studio 2017 (VS2017) cl compiler. This required modifying the development version code, and included cl compilations of the dependent libraries (QD, Openblas, ADMB, pthreads).

It has been a significant milestone to have the MULTIFAN-CL development version compiled using VS2017. Since the previous Visual Studio version, (single species/sex dimensions, and without multi-threading, last compiled in 2015), substantial new code has been developed which required substantial modifications for the VS2017 cl compiler, while maintaining consistency with the Linux gcc compiler. However, it was possible to completely reconcile all differences between linux and VS2017 versions of code, particularly those in respect of passing constant arguments by using the ADUNCONST macro. All source code is now compatible among the gcc and cl compilers. The next step is to replicate this compilation project on a platform at OFP, with a view to including it in the routine automated Jenkins compilations.

It was not possible to achieve a completely static build using VS2017, and tests indicated that 7 dependent .dll files must be ported with the Windows executable for running on either Windows 7 or 10 platforms. This will be the priority for future work on this compilation.

A VS2017 compilation guide will be documented for the full compilations of dependent libraries as well as MULTIFAN-CL.

The performance of the Windows executable was tested using the SKJ2014 model including a complex method for fitting size composition data (SSMULT_RE, section 4.1). The time expended to run 50 model evaluations were compared:

- Linux: 3 mins 32 secs
- MinGW: 8 mins 9 secs
- VS2017_nick: 2 mins 49 secs

Illustrating that the VS2017 compilation is roughly 20% faster than the Linux executable, and confirming that the VS2017 compilation of the Windows 64-bit executable has comparable performance to the Linux compilation. Slight differences in the calculated objective functions were found (identical solutions to within 10 or 12 decimal places), with negligible differences in model derived quantities (Table 1). The Windows executable was therefore included in a benchmark test of the development version (section 2.6.4) that confirmed the integrity of its calculations relative to the Linux executable.

2.4.3 Upgrade to gcc-5.4.0 compiler

Upgrading from the gcc-4.6.3 compiler to gcc-5.4.0 was achieved. This entailed re-compiling all dependent libraries (ADMB, QD, OpenBLAS, pthreads) and then the MULTIFAN-CL source code. The QD project was upgraded using qd-2.3.17.tar.gz (<http://crd-legacy.lbl.gov/~dhbailey/mpdist/>), and the OpenBLAS project using OpenBLAS-0.2.19.tar.gz.

Compilation errors were encountered when using gcc 5.4.0 mostly relating to the compiler's protocol for passing a reference to a constant object to a function. This was remedied by casting the constant reference to a non constant one, using the macro ADUNCONST. Note that the changes required for the gcc 5.4.0 compilations were backwardly compatible with earlier compiler versions, with the modified ADMB and MULTIFAN-CL code able to be compiled using the gcc 4.6.3 compiler.

2.4.4 Source code repository

The MFCL project is now 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>

A total of 27 commits were made to the development branch, including a merge to the master branch preceding the release of version 2.0.2.1.

2.5 Developer's workshops

Developer's workshops were held at Nanaimo, British Columbia, Canada, 1 – 14 September 2016, and at Nouméa, New Caledonia, 8 – 16 March 2017. The participants were the primary developer Dr Dave Fournier and Mr Nick Davies.

The main areas of development during the first workshop were:

- Self-scaling Multinomial M-estimator (**SSMULT_RE**) for composition data, including tail compression and improving the inner minimization stability
- Dirichlet-Multinomial (**DM**) likelihood for composition data
- Drafting a manuscript that introduces these approaches for fitting composition data
- Simulation mode for generating pseudo-observations
- Estimating autocorrelation in recruitments

The main areas of development during the second workshop were:

- Compilation of a Windows executable
- Estimating the autocorrelation of BH-SRR deviates
- Length-specific selectivity shared among sexes
- SSMULT-RE and DM likelihood
- Simulation tagging data

The specific details of these developments are provided in Sections 4 and 5.

2.6 Benchmark testing 2016-17

The benchmark testing framework is described in section 2.9.2, and the series of benchmark tests undertaken in 2016-17 are listed in Annex 12.1 with a brief description of the tests, and the features tested, is provided in this section.

2.6.1 Preceding the Sep.2016 developer's workshop

Prior to the first developer's workshop of 2016-17, the development version code was benchmark tested. This was an "internal" test of the development branch to ensure continuity of the pre- and post-workshop versions. The development version was tested versus the release version 2.0.1.1 using a doital fit with the SKJ2014 test data. Zero evaluation runs for making deterministic comparisons using the respective solution from each version confirmed the identical model and likelihood calculations. The doital fits revealed slightly different minimisation solutions from the two versions (-0.021% difference in the total objective function value), resulting in dynamic and equilibrium model quantities being slightly different (largest difference of about -4% in recent absolute biomass; and up to -1.5% in reference points). This difference was most likely due to code changes made in the development version to improve minimisation stability, which resulted in slightly different minimisation paths taken to achieve the solutions.

2.6.2 Version 2.0.2.1

Following the first developer's workshop, comprehensive benchmark testing was done of the development version versus the release version 2.0.1.1. Substantial developments and bug fixes had been made since the previous testing done in June 2016. The major changes included: an improved algorithm for applying stochastic recruitment deviates to BH-SRR predictions in simulation projections; the Dirichlet-Multinomial likelihood; a simulation mode for generating pseudo-observed data; deriving an autocorrelation coefficient in recruitment estimates; and, an improved variance calculation using the OpenBLAS library routine for the singular value decomposition. A number of bug fixes were identified and made during the testing (detailed in section 5.4). To improve the testing for existing features recently included in the release version, the ALB2015 test data was added to the range of data sets used for benchmark testing. As was found in the Sep.2016 test, differences in the doital solutions among the versions and compilation platforms could be explained by differences in the minimisation "paths" taken to reach a solution by each version, and was attributed to the

same reason (improvements made to the minimisation stability in the development version). Deterministic projections that assume the BH-SRR predictions for future recruitments generated very similar recruitments on average, but the new feature in the development version that includes seasonality results in differences within each quarter (section 2.6.5). A substantial difference among the versions was found in stochastic projections. This was due to the changes in the algorithm for applying stochastic deviates about the BH-SRR predictions. These differences were concluded to be caused by improvements made to the development version, which was consequently advanced as release version 2.0.2.1 and posted on 15 December 2016.

2.6.3 Version 2.0.2.2

A minor revision was made to release version 2.0.2.1 to implement an alternative formulation of the logistic selectivity function that is robust to implausible parameter values. This prevents arithmetic exception errors occurring during the early stages of minimisation. To ensure continuity in the master branch (i.e. the release version), an abbreviated benchmark test was done using single evaluations and doital fits of the YFT2014 test data. The single evaluation test proved positive, (identical model quantities among the versions), while the revised version provided a doital fit solution with only minor differences to version 2.0.2.1. This was attributed to the revised formulation of the logistic selectivity at age resulting in slightly different minimisation “paths” being taken (probably due to the slight differences in the tagging data likelihood term). These differences were minor in respect of model quantities, (up to 2%).

2.6.4 Visual Studio compilation of Windows executable

For release version 2.0.2.2 the Windows executable was compiled using the MinGW cross-compiler, while a compilation had been developed using the Visual Studio 2017 compiler for the development version (section 2.4.2). Source code changes in the development version included: improvements to the length-specific selectivity for the multi-sex case; alterations to the feature to estimate autocorrelation in recruitments; and, corrections to the flag settings used for the DM likelihood. An abbreviated benchmark test was done using single evaluations and doital fits of the YFT2014 test data. Relative to release version 2.0.2.2, the development version provided an identical doital solution for the Linux compilation executable, with only minor differences for the among the Windows executables for the two versions. This was attributed to the compiler differences which results in slightly different minimisation “paths” being taken. However, the differences were minor in respect of both dynamic and equilibrium model quantities, (up to 1.1%).

While the test was positive, the development version was retained within the development branch until comprehensive benchmark testing was possible. The improved Windows executable and new features in the development version were of utility for the 2017 stock assessments, therefore a provisional version number was assigned - 2.0.3.1. Preliminary tests were also performed in respect of the alternative size composition likelihoods (without tail compression) which proved positive, so this development version was distributed internally within OFP.

2.6.5 Test of the improved feature for stochastic recruitments

Following the first developer’s workshop (Sep.2016), two new features in MULTIFAN-CL were tested in detail:

- For models having a quarterly time-step with the BH-SRR being fitted to the annual rather than the seasonal recruitments, (i.e. estimating an “annualised” BH-SRR that is fitted to the sum of the seasonal recruitments in each year), when performing projections using the BH-SRR, the predicted recruitments are apportioned among quarters according to the average of the estimated recruitment proportions in each season.
- For simulation projections, the annualised BH-SRR deviates are stochastically resampled and applied to the predictions in the projection time periods.

While the second feature had been applied in earlier versions of MULTIFAN-CL, it had been improved to avoid slight approximation errors present in the previous implementation. Details of this improvement were described by Davies et al. (2016). The testing entailed four experiments using the ALB2015 model that proved the new features had been correctly implemented as intended. Deterministic projections demonstrated that the BH-SRR predictions in each model year were allocated to each season according to the estimated average

seasonal proportions. Where stochastic deviates were drawn from a theoretical normal distribution, parity was illustrated between deterministic and stochastic projections that use the BH-SRR predictions (Figure 1). However, when stochastic deviates were drawn from those estimated from fitting the BH-SRR, the mean of the stochastic recruitments is 27% lower on average than that of the deterministic recruitments, and consequently the mean stochastic biomass is also lower (Figure 2). This difference between the deterministic and stochastic recruitments relates to the variance of the deviates around the fitted relationship. When the assumption of normality in fitting the BH-SRR has been violated, therefore the correction for log-normal bias provides a poor approximation of the arithmetic mean. Consequently, the bias correction of the BH-SRR predictions may become inaccurate. This violation is probably an outcome of assuming a low penalty upon the BH-SRR fit via `age_flags(145)` that allows “freedom” in the estimation of the independent recruitment estimates. This may be desirable when fitting the dynamic model, however, this will reduce parity between deterministic projections with log-normal bias correction and stochastic projections using BH-SRR deviates that are not normally distributed. This condition should be kept in mind by analysts who apply this feature.

2.7 Postings to website

There have been two postings of the MULTIFAN-CL release version to the website since August 2016.

2.7.1 24 February 2017 – version 2.0.2.1

The main changes relative to the previous version 2.0.1.1 included:

- An improved algorithm for applying stochastic recruitment deviates to BH-SRR predictions in simulation projections
- Predictions of the annualised BH-SRR are allocated by the average seasonal recruitment in the projection periods
- Tail compression was extended for weight composition data with a revised penalty formulation
- Dirichlet-Multinomial likelihood for size composition data
- Improved minimization stability
- The SSMULT_RE implementation completed with a revised phi function formulation
- Simulation mode for generating pseudo-observed data
- Autocorrelation in recruitment estimates can be derived
- Improved the variance calculation to use the OpenBLAS library routine for the singular value decomposition
- Functional forms for natural mortality at age
- Various fixes and minor improvements

2.7.2 6 April 2017 – version 2.0.2.2

A minor revision for an improved formulation of the logistic selectivity that avoids arithmetic exceptions due to implausible parameter values that are within the bounds shared with the spline selectivity formulations. This increased minimization stability in the early phases when estimating logistic selectivities.

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 (Ianneli 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 formed the core of MULTIFAN-CL developments since the review, and an outline of the status in fulfilling these recommendations is provided.

Of the thirteen recommendations, 9 have been implemented and tested, and 4 remain yet to be developed. These remaining recommendations were identified to be undertaken for 2016-17 (Table 2):

- Maturity-at-length (recommendation "k")
- Non-uniform size bins (recommendation "b")
- Long-term and initial tag loss (recommendation "c")
- Tags inform movement only (recommendation "d")

In 2016-17, following internal discussion, high priority was given to: developing approaches for fitting size composition data (the self-scaling multinomial with random effects, SSMULT_RE, section **Error! Reference source not found.**, the Dirichlet-Multinomial likelihood, DM, section 4.2); completing the implementation of tail compression (section 4.3); and, to implementing a simulation mode (section 4.4). These other tasks have therefore taken priority over the implementation of the above four features recommended by Ianelli et al. (2012). Indeed, the implementation of the recommendation for a multinomial distribution for size composition data has extended beyond the scope of that initially intended, with the innovative SSMULT_RE and DM approaches improving upon the simple Multinomial as suggested by the panel. However, it is proposed that work toward these remaining recommendations be resumed for 2017-18 (section 7, and Table 3):

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. Further refinements of these tools were undertaken as part of the 2017 stock assessments that consolidated new features to the utilities package.

2.9.2 Testing framework

The testing framework for MULTIFAN-CL compilations first developed in 2011-12, was applied extensively during 2016-17. 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 and architectures, 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 doital 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 MULTIFAN-CL source code has undergone substantial developments, major ones including: multi-threading of the tagged population calculations, combined tagging observations for multiple species/sexes, multi-sex equilibrium yield calculation, time-variant selectivities for individual fisheries, adding a likelihood term for age-length composition data, approaches for fitting size composition data, and a simulation mode. These developments have been described (Davies et al. 2016) and more recent ones are described in Section 4.

Following the addition of these new features to the development version, regular testing of this versus the release version was undertaken to ensure the integrity of existing operations. Known as “benchmark tests” these are listed in Annex 12.1 and those undertaken in 2016-17 are described in section 2.6. The development version was last tested in April 2017 versus the release version, which defines the development version as the **benchmark** source code. Subsequent development versions were then tested relative to the benchmark to establish their integrity, after which they are 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 single-species data for ALB2012, ALB2015, BET2011, YFT2011, SKJ2011, STM2012, SWO2013, BET2014, YFT2014 and SKJ2014 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, but 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. Noting 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.

The benchmark tests undertaken during 2016-17 are described in section 2.6. All the benchmark tests concluded that the development version conserves the existing features and can either be advanced as the new release version, or accepted for the new benchmark development version.

2. Establishing integrity of new features

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 2016-17 this level of testing was done for the stochastic recruitments feature (section 2.6.5), the SSMULT_RE M-estimator (section 4.1), and the DM likelihood (section 4.2), 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.

While action on these tasks is unlikely in the remaining part of 2017, it is proposed that they be considered in the 2017-18 work plan for the MULTIFAN-CL project.

2.9.3 Viewer

A development version of the MULTIFAN-CL viewer that can display the results of a multi-species or multi-sex application was updated as new output was added to the report files. Development of this version is ongoing since the final output format of the multi-species/sex application is not yet complete. However, substantial improvements were made for displaying results of multi-sex models.

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. Additional Linux Virtual Machines were added to the grid to increase the number of model runs possible using the Linux development version executable. During 2017 the processors on the Condor grid were re-configured for dynamic memory so as to improve the efficient use of available CPU for running jobs.

2.10 User's guide

A revision the MULTIFAN-CL User's Guide (Kleiber et al. 2016) has been completed that documents the developments in version 2.0.2.1 and this has been posted on the website <http://www.multifan-cl.org/>. Proposed future revisions include incorporating the suggestions arising from the Training workshop (see section 3) and the recent features added to release version 2.0.2.2 and also to the development version 2.0.3.1.

3 TRAINING WORKSHOP

The second training workshop for users of MULTIFAN-CL at the Oceanic Fisheries Programme (Pacific Community) was held 6-7 March 2017, at Nouméa, New Caledonia. The aim was to achieve proficiency with the new features in MULTIFAN-CL version 2.0.2.1, particularly with the approaches for fitting to size composition data and generating simulation model pseudo-observations (see agenda in Annex 12.2). The workshop was delivered by Nick Davies and Dr Dave Fournier and was well attended with four core analysts and several observers.

Particular attention was paid to the self-scaling properties of the SSMULT_RE M-estimator and DM likelihood approaches for fitting size composition data. Dave Fournier provided essential clarification on aspects of the M-estimator and the related variance estimation of the Multinomial and the random effects. It was also made clear to the participants as to why including tail compression was important when using the SSMULT_RE M-estimator. This was because zeroes affected estimation of the auto-correlation parameter for the random effects, and tail compression reduces this influence. The question was raised during the workshop if the profile likelihood diagnostic could be produced for models incorporating the SSMULT_RE M-estimator for size composition data. Simply speaking this is not possible, however Dave explained that there may be approaches for approximating the calculation. Also it was not clear how to present the goodness of fit to observations that illustrates the relative estimates of multinomial error and errors in the random effects. It was requested that standardised reports be generated, in order to generate plots of the fits to size composition observations with, and without, the random effects.

Participants were reasonably well-prepared for "hands-on" practical sessions using the examples for implementing the new features and for debugging the examples. Most found the methods easy to follow and were satisfied with their ability to locate simple causes for program interruption.

The feature for generating simulation model pseudo-observations was relatively simple to explain and implement using the example supplied. One question raised was whether the "dummy" size composition frequencies supplied in the .frq fisheries data could be either an integer or real number. All queries arising from the participants have been recorded in the project task list.

During the section on auto-correlation in recruitments, a discrepancy was identified between the coefficient parameter estimated independently, and the moment estimate of the coefficient derived directly from the residuals of the fitted BH-SRR. This had been explored by the developers prior to the workshop, and it had been shown that the coefficient parameter estimated independently was positively biased if the variance was not estimated simultaneous with the coefficient. An alternative method was developed whereby the moment estimate was used among evaluations to "tune" the coefficient to a converged estimate. This method was included in the MULTIFAN-CL development version for further testing.

The key features of the Github repositories for MULTIFAN-CL and the Manual were demonstrated, in particular how to post issues yet to be resolved by members of the project team.

4 NEW FEATURES

All new features that have been implemented in the MULTIFAN-CL source code have been added first to the development version. Once these features have been tested for their integrity, with no impacts on

existing features, then the development version is merged into the release version of the code. The current release version is 2.0.2.2. The developments described below have been made to the development version (currently denoted as version 2.0.3.1) during 2016-17 and will be merged to the release version upon the completion of benchmark testing.

4.1 Self-scaling Multinomial plus random effects (SSMULT_RE)

4.1.1 Rationale

Ianelli et al. (2012) recommendation “j” was to “Add an option which allows the analyst to assume a multinomial likelihood for the compositional data in the first phases and only transition to the robust normal likelihood in the later phases.” This feature has been in development since 2014, it formed a significant component of the developments for 2015-16, and was mostly completed during 2016-17. While the recommendation provided the initial impetus for a development to use the Multinomial likelihood, the three known weaknesses with this method must be considered (Francis 2014):

- It assumes the variance is proportional to the sample size, which is usually violated because variances are often larger,
- It doesn't adequately account for positive correlations, and,
- It doesn't adequately account for process error such that effective sample sizes often underestimate the true error.

Size composition data are pivotal to the tuna stock assessments using MULTIFAN-CL, in particular for estimating growth and recruitments. Improvements that might offset these weaknesses would enable the correct use of these data in the integrated modelling approach, and this became the focus of this development. The lead developer, Dr Dave Fournier, formulated an innovative size composition minimizing function, being the Self-scaling Multinomial plus random effects (SSMULT_RE). Indeed, in developing this feature, the recommendation for a Multinomial distribution for size composition data (recommendation “j”) has been extended well beyond the scope of that initially intended, by researching a substantially better approach.

4.1.2 Methods and Testing

While the SSMULT_RE was described by Davies et al. (2016), a brief overview follows. The SSMULT_RE addresses the above-mentioned three weaknesses by modifying the Multinomial to have three attributes: self-scaling properties (it estimates the effective-sample sizes, ESS), maintaining overdispersion, and to estimate autocorrelated random effects. We refer to this as the self-scaling Multinomial with estimation of random effects (SSMULT_RE), being a form of M-estimator (Huber 2009). In addition to addressing the noted deficiencies in the multinomial, the SSMULT_RE retains the key multinomial property of being able to deal with observed zero proportions in a completely natural way. Therefore, it does not require modification of the data (by adding an arbitrary constant) to remove observed zeros, which is required for the robust-normal or logistic-normal likelihoods.

At both of the Developer's Workshops during 2016-17 the SSMULT_RE was refined, with testing by simulation to assess its performance in estimating the key parameters: ESS and correlation in the residuals; and, testing its implementation using real stock assessment model data.

A problem with the inner minimisation of the random effects parameters was identified. The Φ function used for the random effects was revised to use the formulation being a simple lognormal random effects model implemented as per Hrafnkelsson & Stefansson (2004). This is implemented in the inner minimisation. The key point is that the Laplace approximation is more accurate for this formulation. The performance of the inner minimisation was improved by restoring the Φ formulation to be consistent with the original Hrafnkelsson & Stefansson (2004) version. This improved model minimization stability for the large example data used for testing (SKJ2016). Visually comparing the size composition data fits before and after the correction, for the model+RE predictions, now appear to be less precise. This is an improvement and may reduce the incidence of the SSMULT parameters hitting the bounds.

However, a problem persisted with the correlation parameter tending towards the upper bound. Simulation testing was therefore done using a simple age-structured fish model with simulated autocorrelated composition data. Difficulties in estimating ρ were tested using an alternative estimation method that solves

iteratively using the moment estimate for ρ for the AR(1) penalty function within each model evaluation. While the method was accurate for a simple case where ρ was the single parameter, it was confounded when the ESS was estimated simultaneously. While this had potential for other applications, e.g. autocorrelation in recruitments, it was rejected as being a possible method for the SSMULT_RE.

This development represents a substantial advance in the use of size composition data in population models, and has notable potential benefit. Further testing to improve the method for estimating ρ is proposed for the remainder of 2017.

4.2 Dirichlet-Multinomial Likelihood

The Dirichlet-Multinomial (DM) likelihood is an obvious option for composition data and is implemented in other integrated model software (Methot 2013). It has been re-formulated for introducing the feature in MULTIFAN-CL with a focus on the self-scaling properties. A key parameter is the maximum ESS that must be specified and has been formulated as in the following equation:

$$eff_{-N_i} = \frac{Nmax(1 + \lambda_i)}{(Nmax + \lambda_i)}$$

And $Nmax$ can either be the default value (1000), or specified by a value assigned using a **parest_flags** that applies to both LF and WF data. The observed sample size, N , is used to calculate a relative sample size in respect of the mean,

$$\tilde{n}_i = \frac{N_i}{N}$$

And the DM ESS sample size covariate is,

$$\tilde{m}_i = \tilde{n}_i^c$$

Where c is the sample size covariate exponent, and the ESS multiplier is,

$$\lambda_i = e^d \tilde{m}_i$$

Where d is the sample size multiplier exponent. The parameters estimated are c and d , with the option for grouping their estimation in respect of fisheries. An assumed value is assigned to $Nmax$.

This is the current implementation in the MULTIFAN-CL release version 2.0.2.1. This implementation has also been developed in a simple fish model simulator.

In allowing for grouping of fisheries that share parameter estimates, routines were drafted to ensure the parameters have active flag settings compatible with the grouping, but also checks that the values of the model parameter vector are compatible with the grouping. This detects user-defined values for the parameters inconsistent with the specified grouping.

A useful diagnostic for illustrating the self-scaling properties of the DM, and for evaluating its performance in respect of the ESS, is to compare the ESS with the observed sample sizes. There are two different ESS estimates. The first is the theoretical effective sample size as estimated in the model parameters which is a function of $Nmax$, the input sample size covariate, and the parameters which do overall scaling and include the covariate. The second is a moment estimator calculated from the observed and predicted sample sizes, using the size composition residuals. Two reports are generated, (“wght_dmsizemult” and “dmsizemult”, for weight and length composition data, respectively) containing: the observed sample size; theoretical ESS; and, the moment estimate for the ESS. Direct comparisons are then possible in respect of each fishing incident for which size composition samples are available.

The DM likelihood estimates specific to each fishery were added to the diagnostics report “test_plot_output” used for summarizing all components of the total objective function from the integrated model fit. This report is produced only when the fitted model is complete.

4.2.1 Methods and Testing

The simple fish model simulator previously developed for testing the SSMULT_RE was adapted to generate simulated data from both the Multinomial and the DM distributions, and to include the DM likelihood in the analyzer component. In this case, random effects are not being estimated, but rather simply the ESS. Preliminary simulation experiments were run indicating this likelihood performs well. For example, simulation data incorporating process error (fishing and natural mortality, recruitment) and observation error (total catches, relative biomass, multinomial error with positive autocorrelation) resulted in effective sample size estimates that were negatively biased relative to the “true” values, e.g. around -67% for average true sample sizes of 10 with 40 age classes. At high sample sizes, with increased size composition observation error, the estimated ESS declined. These were the expected results.

Simulation testing of the sensitivity of the DM to the assumed N_{max} value was done for a range from 50 to 2000 where the “true” value was 500. The average relative bias in the DM estimated ESS indicates some level of sensitivity to this assumed parameter (Table 4). If the DM analyzer is supplied with an N_{max} that is larger than the “true” value it will under-estimate the effective sample size, and vice versa. This seems plausible, in that if the DM analyzer “expects” higher precision than the samples exhibit, it will estimate lower sample sizes, and vice versa. Therefore, the DM analyzer is sensitive to the assumed value of the ad hoc parameter N_{max} .

The next steps for the simulation testing using the simple fish model simulator is to design and run experiments that illustrate its performance in respect of a wider range of factors. The design will most likely incorporate the following factors.

Simulator model factors:

- sample size (range 20 to 1000)
- process error
- size composition observation error type - Dirichlet-Multinomial or Multinomial
- size composition observation error – with or without autocorrelation

Estimation model factors:

- DM_noRE (possibly also test SSMULT-RE if considered necessary)
- Assumed maximum sample size ($N_{max} = 1000, 2000, \dots$)

With possible performance indicators being:

- Bias in estimated effective sample size relative to the “true”
- Bias in total population numbers in terminal year

Testing the DM likelihood with an example of real stock assessment model data was done during 2016-17 using the same SKJ2014 data as used for the SSMULT_RE so as to permit relative comparisons among the three approaches. The example was used to explore the effects of the assumed N_{max} value in the DM likelihood. The model runs using DM included tail compression, started from an existing solution obtained using the normal+constants size composition likelihood (robust-normal), and run to convergence for a range of assumed N_{max} : 250, 1000, 1500, and 2000. The results indicated only a moderate effect of N_{max} on model derived quantities such as absolute spawning biomass (Figure 3), and minimal impact on important management-related quantities of relevance for making stock assessment advice (Table 5).

4.3 Tail Compression

Tail compression is an important component in fitting size composition data, in particular in cases where the tail is extended and includes high numbers of zero proportions. Applying tail compression was also considered as a means to avoid positive bias in the SSMULT_RE auto-correlation parameter.

The existing code for performing tail-compression was consolidated so as to place the implementation procedures into discrete routines. The tail compression operation for both length- and weight-frequency data is conditional on specified `parest_flags` values for each, and is done before the call to the `fitting_procedure()` routine.

It is important to note that the predicted proportions outside of the bounds specified by the tail-compression of the observed sample distributions are not included in the SSMULT_RE calculation. To include these in some manner in the objective function, a penalty is applied. This is unique to the SSMULT_RE and DM tail compression cases. When testing the tail compression implementation of the DM likelihood using the SKJ2014 case study, spurious selectivity-at-age estimates were obtained for size intervals outside of the tail-compression bounds. The spurious selectivity estimates were attributable to the penalty formulation and this was revised to account for the variance in the observations. This feature is of significance for the SSMULT_RE because these proportions have impacts upon the random effects estimation.

Currently there are two alternative methods for applying tail compression:

- `parest_flags(311)`: applied for the robust-normal and the logistic normal likelihood (a cumulative proportion in the smallest and largest compressed intervals is specified)
- `parest_flags(320)`: applied for SSMULT_RE M-estimator and DM likelihood (a minimum number of class intervals in the compressed distribution is specified).

While the implementation of these tail compression methods has been completed, it is possible that as part of the testing yet to be done during 2017 for the SSMULT_RE and DM approaches, the second method may be further refined.

4.4 Simulation mode

To implement full simulation capability in MULTIFAN-CL, a simulation mode has been implemented such that an operating model is used for projections to produce pseudo-observed data with specified observation and process error as might be used in management strategy evaluations (MSE). The algorithm for producing predictions (in this case for example for size composition data) for the projection period, is as follows.

- As normal for undertaking a projection, the analyst supplies catch or effort in the fisheries data for the projection period (i.e. in the *.frq* file).
- Analyst populates the length frequency (LF) and/or weight frequency (WF) columns with non-zero frequencies in the fisheries data during the projection period for the specific fisheries for which simulation data are to be generated. The total of the frequencies is the desired effective sample size.
- Analyst sets a `parest_flags` option that activates the feature for generating simulation data from projections using the projection model quantities.

Essentially, the algorithm for generating pseudo-observations is activated by `parest_flags(241)` with the data to be generated specified by entries made in the projection section of the fisheries data in the **.frq* file.

4.4.1 Size compositions

Observation error is applied to the predictions of LF and WF in the projection period, with Multinomial error applied to the predictions. The sample size is taken from the sum of the frequencies specified in the LF and WF data columns of the fishery data in the *.frq* file for the projection periods. As such, the analyst specifies the level of observation error. The size composition pseudo-observations are reported to a file "test_lw_sim" with each record indexed in respect of the parent fishery, true model projection year (e.g. 2019), and month.

4.4.2 Relative abundance indices

The general method for fitting to standardized indices of relative abundance derived from catch-per-unit-effort (CPUE) in MULTIFAN-CL is to express the indices as standardized effort, assume constant catchability, and apply high penalty to the total catch likelihood. The generation of pseudo-observations from simulations for the projection period will depend upon the projected "observed" fishery data supplied by the analyst; i.e. projected catch or effort. As such there are two algorithms for generating the pseudo-observations.

In the first case, the analyst may supply "observed" catches (either in weight or numbers of fish) in the fishery data for the projection period, and "-1" for the effort. As no "observed" effort is supplied, a Newton-Raphson (N-R) solution for the predicted fishing mortality directly provides the predicted catch. The model estimates no predicted effort but solves for the fishing mortality level that achieves the supplied catch. To generate a predicted effort, a new formulation was developed using the Newton-Raphson solution for the

fishing mortality associated with a supplied “observed” catch for a projection fishing incident, the catchability in the final year of the estimation period. To express the predicted effort in the same units as the effort input for the fishery, the average observed effort input for the estimation period (in true units) is applied:

$$\hat{E}_y = \left(\frac{F_y}{q_{y=term}} \right) \times \bar{E}$$

With the term \bar{E} taken from the calculations for normalising fishing effort input to the model. Note that the normalisation is done only for the model “estimation period”, i.e. for the periods for which observations are supplied, and do not include the projection periods. This ensures the catchabilities are estimated for normalised effort over the estimation period only. Using the term \bar{E} in calculating projection effort ensures it is in the same order of magnitude and units as the original effort input to the model.

Lognormal error is applied to the effort predictions with the standard deviation taken from the analyst’s value specified in **age_flags(186)**. If no value is specified, i.e. `age_flags(186)=0`, then no observation error is applied to the prediction.

In the second case the analyst supplies “observed” effort in the fishery data for the projection period, and “-1” for the catch. The implementation in MULTIFAN-CL for calculating the fishing mortality uses the “observed” effort directly with the estimated catchability to derive the mortality level which is applied to the population numbers to derive the predicted catch.

For the projection period, the terminal catchability (ignoring age-specific effects for now) is used with the “observed” effort supplied by the analyst to calculate predicted fishing mortality,

$$F_y = q_{y=term} E_y$$

And the predicted catch is,

$$C_y = (1 - e^{-F_y}) N_y$$

Therefore, the relative abundance variability is inherent in the predicted catch given that effort is “fixed” by the analyst.

Lognormal error is applied to the predictions with the standard deviation taken from the analyst’s value specified in **age_flags(187)**. If no value is specified, i.e. `age_flags(187)=0`, then no observation error is applied to the prediction.

4.4.3 Tagging data

The implementation of the simulation mode was extended to produce pseudo-observed tagging data. The example ALB2015 stochastic projection model was used for the development of this implementation because it offered all possible data types that may be required as pseudo-observations.

This entailed adding to the existing algorithm activated by `parest_flags(241)` for catch, effort, and size data. It was considered reasonable that the analyst would not impose an assumed composition upon a simulation tag release event, because the age or size structure may be incompatible with the simulation population conditions in the projection period. Therefore the only information to be supplied was considered to be the time, location and total number of simulation tag releases.

A new input file called `*.tag_sim` was formatted that contains: tag group number; region of release; year, month, fishery number; and the frequency total number of releases. A new routine called `read_tag_simulation_info()` reads in these data and allocates the simulation variables in structures compatible with the “true” observed tagging data used in the estimation period of the model. A new routine was added to identify the fishing incident associated with the release during the projection period.

Broadly, the algorithm for generating simulation tag recapture pseudo-observations was as follows:

- Assign the release sample to an initial tag release cohort of numbers at age; this is based upon the predicted catch-at-age for the fishing incident associated with the simulation release event.

- Perform catch calculations upon the simulation tagged population. These calculations are based upon the existing code for tag groups within the estimation period to ensure consistency. This generates the predicted tag recaptures for the simulation tag release groups.

A problem is that the predictions are very small real numbers and over a wide distribution of fishing incidents, which do not resemble real tag recapture frequency data. A sub-algorithm was therefore developed for a probabilistic method of generating simulation tag recapture frequency data with multinomial error.

- For each age class, the tag catch calculations are done up to the terminal tag period based upon a hypothetical sample size, and if the probability of recapture > 0 , the fishing incident at recapture is stored relative to the age at release.
- This generates a matrix of probabilities of recaptures in respect of the release sample age class among future fishing incidents; this represents a multinomial probability distribution.
- A routine generates multinomial samples of simulation tag recaptures given:
 - The tag recapture sample size being the total predicted recaptures with respect to tag release group, age and region.
 - The probability of recapture being the matrix calculated from the tag catch calculations in each future fishing incident.
- The non-zero frequencies (trial successes) of simulation recaptures are stored in respect of the age at release, and future fishing incident of recapture, that comprises the simulation tag recapture sample at age.

This sub-algorithm is somewhat analogous to an individual based model, however, it is performed in respect of the individual age classes rather than the individual fish.

The next task for this development is to extend the simulation tagging data in respect of age to be instead in respect of length. A sub-algorithm must therefore be developed to project the simulation tag recapture sample from an age class at release to a range of length classes at release. This entails the estimated growth function and standard deviation for mean length at age over the model estimation period.

4.4.4 Status and future work

In summary, the status of implementing this new simulation mode feature is that: simulation data for size compositions and relative abundance indices can be generated; the algorithm for generating tagging data has been developed and is in progress; and, generating simulation age-length data and simulation data for the estimation period is not yet possible. It is proposed to firstly complete this feature in respect of generating tagging data, and data for the estimation period, as this is most important for validating and undertaking MSE work using MULTIFAN-CL.

4.5 Autocorrelation in recruitments

An implementation was developed for estimating auto-correlation in estimated recruitments. This has utility for providing an algorithm for applying the auto-regression coefficient when generating stochastic recruitments for simulation projections.

A matrix solution was formulated for the autocorrelation using a property of the autocorrelation matrix where the choleski decomposition is used to get the logarithm of the determinant. The AR(1) penalty is then added to the total objective function in solving for the autocorrelation parameter.

The two methods for estimating the autocorrelation are in respect of the residuals for either the:

- independent log-normal recruitments (relative to the mean), or
- the BH-SRR estimated deviates (relative to the BH-SRR prediction).

The default case if ρ is not estimated, is to simply calculate the moment estimate for ρ from the BH-SRR deviates. If the feature is activated, both the moment estimate and the estimated ρ values are reported, otherwise, only the moment estimate is reported. The autocorrelation may be estimated in respect of the full model calculation period, or for a user-defined sub-set of periods. The user may also define the weighting assigned to the AR(1) penalty.

Given that predicted recruitments in simulation projections are often assumed to be a function of projected spawning biomass, it may be preferable that this quantity be considered when estimating a recruitment autocorrelation. Therefore, estimating autocorrelation among the deviates from the BH-SRR relationship may be the better approach. In this case the AR(1) penalty then replaces the lognormal error assumption of the BH-SRR fit.

Besides some minor updates to the method for fitting ρ to the log-normal recruitments residuals, this method is still in development in version 2.0.3.1.

4.6 Functional forms for age-specific natural mortality

A feature for estimating age-specific natural mortality has long been an option in MULTIFAN-CL, however a method that estimates a functional form rather than individual deviates in respect of age has merit in the reduced number of parameters required. This was added to the existing feature using either a spline function, or a function consistent with that described by Lorenzen (1996).

The spline function uses the same formulation as employed with the spline selectivity functions, with a `parest_flags` setting for activating the estimation of the splines with the flag value specifying the number of nodes.

The Lorenzen function was slightly adapted for its implementation in MULTIFAN-CL. Parameterisation of the Lorenzen function is rather based on the scaled lengths from the von Bertalanffy growth estimates

$$M_a = c(l_a)^b$$

Where l_a is the mean length of age class a . Now the mean length is parameterized in the model by the von Bertalanffy, but for numerical stability the function has been “undimensionalised” by the following parameterisation,

$$l_a = d + (1 - d) \left[\frac{1 - e^{-K(a-1)}}{1 - e^{-K(A-1)}} \right]$$

Where L_1 is the mean length of the first age class, L_A is the mean length of the oldest age class, and $d = \frac{L_1}{L_A}$.

The SKJ2016 example was used to illustrate the application of this feature, with a comparison relative to a model where the existing approach for estimating age-specific deviates was employed (Refcase). As might be expected, the spline function (having 5 nodes) produced age-specific indices more closely resembling the Refcase, than the more constrained Lorenzen form (Figure 7). Consequently, the differences in model quantities versus the Refcase for the model using the spline function were less (around 0.3 to 3%) compared to moderate differences for the model using the Lorenzen function (up to 12%), (Table 6). Note that neither of the models using function forms were fully converged solutions, and are presented here only for purposes of comparison.

The preliminary testing of the feature revealed a high potential for instability in the minimisation during the initial phases. This is attributed to the large impact that natural mortality has on model processes, in particular the estimated catches. It was necessary to emulate the sequence used in the control phases for the penalty on the total catch likelihood, specified by `age_flags(144)`. This is to begin with a low penalty of 1, and increase to: 5, 50, 200, 500, 1500, 2500, 5000, 10000, 100000; with at least 50 evals in each stage.

4.7 Likelihood profiling

This was identified as a required improvement to MULTIFAN-CL during the 2016 Training workshop, i.e. to revise the likelihood profile calculation so that this can be applied to particular model dependent variables, such as current biomass. Consequently, penalty terms were formulated for calculating likelihood profiles conditional upon the derived model quantities for average absolute biomass or a level of biomass depletion. The period over which the model derived quantity is calculated may be defined. The new feature is controlled by settings for `parest_flags` that activate the feature, and that specify the weighting applied to the penalty function that ensures the fixed target level is maintained during the minimisation.

An enhancement to this feature was added for using the augmented Lagrangian algorithm for the penalty function calculation between successive stages of the minimisation. The Lagrangian multiplier is written

to the *.par* after a particular set of evaluations, and then read back in to then run with the new fixed multiplier value for the next set of evaluations. This improves the convergence at each point (i.e. scalar value) along the profile.

5 OTHER ENHANCEMENTS AND BUG FIXES

5.1 Extension of multi-sex model feature

During the development of a south-west Pacific swordfish stock assessment model (Takeuchi et al. 2017), a number of enhancement to the feature for a sex-disaggregated model were identified.

5.1.1 Length-specific selectivity shared among sexes

There was a requirement for modelling age-specific fishing mortality such that it approximates a length-specific selectivity function that is shared among sexes having different growth rates. This allows fish of a particular size to have similar relative selectivity while their ages are different. To have a truly length-based selectivity the catch equations must be calculated upon numbers at length, and since MULTIFAN-CL is an age-structured model, this is not possible. Within an age-structured model, ideally, a selectivity in respect of length can be approximated by integrating over a presumed length distribution to produce an age-based selectivity. So, given a “length-based” selectivity, to use it in the catch equations it must be converted into age-based selectivity by assuming you know the length distribution-at-age and integrating the “length-based” selectivity over that to convert it into age-based selectivity.

It is possible to accomplish something very similar with the spline function, where one set of splines is applied for both sexes, but they are evaluated at the different relative mean lengths at age to get different age based selectivities. The spline selectivity at age is therefore estimated that produces a length-specific selectivity that is similar among sexes. The key aspect of this is the scaled mean lengths at age, **tlength**, from 0 to 1 that is shared among the sexes, i.e. only the sex having the largest length has the tlength element equal to 1 (Figure 4). The same set of splines are then applied to the tlength vector but in respect of each sex. The selectivity at age has a maximum index among sexes, such that in the case of asymptotic selectivities, the maximum of 1 applies only one of the sexes; i.e. the one having the maximum length and hence the maximum selectivity at length (Figure 5). When expressed in terms of length, the selectivity is similar among the sexes (Figure 6).

The feature was then extended from the spline function to also apply to the logistic selectivity function, i.e. to use the shared tlength vector in deriving the logistic function selectivity at age. An example is presented in panel “14_Other_2C” in Figure 6. To improve numerical stability during minimisation, it was necessary to extend the feature for the generic selectivity penalty to be fishery specific, so as to enable penalty for logistic selectivity functions only. This allows the relative weight of the penalty to be assigned specific to each fishery, thus enabling fisheries with particular selectivity forms, e.g. logistic, to be penalised differently to other selectivity forms.

It may be desirable to define the ages over which selectivity is estimated to be specific to each sex, especially if they are substantially dimorphic. The existing features for defining the first and last age classes for selectivity were therefore important to the tlength calculation. It was formulated so as to accommodate flexibility in the age subscripts used. Also, the checks for grouping of the fish_flags were relaxed for this special case (multi-sex model with shared length-specific selectivity).

A number of new reports are produced when activating this feature that contain: the re-scaled mean lengths at age in variable “tlength”; the selectivity at length for each fishery (i.e. sex); and, the selectivity at age for each fishery (sex).

While this feature is in the prototype stage of development, preliminary testing indicates it is operating correctly as formulated. It is currently only implemented in the development version 2.0.3.1 and has not yet been released.

5.1.2 Various combinations of aggregated/disaggregated fisheries data

Release version 2.0.2.2 accepts fisheries data in the form where: catches may aggregated or dis-aggregated among sexes; and, size composition data may aggregated or dis-aggregated among sexes. However, size composition data consists of both length and weight frequency data, and observations for either of these data types may be aggregated or dis-aggregated among sexes. The release version only applies these options to both forms of size composition data at once. Therefore, the feature was extended to allow the aggregated or dis-aggregated options for each of the three fisheries data types individually, i.e. for catches, length frequencies, and weight frequencies.

This was achieved by extending the pointers read in from the fisheries data section in the .frq file to include two pointers for the size composition data, one for length frequencies, and another for weight frequencies. This means there will be a total of: $nspp + (3 * nspp)$ columns in each record of the .frq fisheries data section relating to the pointers for species, and aggregated or disaggregated fisheries data types, where *nspp* is the number of sexes or species.

5.1.3 Sex ratio of predicted catch compositions

In the case of fitting to size composition data aggregated among sexes, an assumption must be made on how the predicted size compositions are aggregated. Release version 2.0.2.2 aggregates the predicted proportions in size compositions as being the average among the sexes. Where sex ratios in predicted catches may be disproportionate, assuming equal representation of the sexes in the aggregated size compositions may be incorrect. Therefore, an option was implemented in development version 2.0.3.1 to scale the sex-specific predicted proportions at size for each fishing incident by the sex ratio in the predicted catches for that incident. This option is conditional upon a `parest_flags` setting, and preliminary testing indicates it is operating correctly.

5.1.4 Identifiers for sex in output reports

For various reports generated by MULTIFAN-CL, it was necessary to include indices used internally with the code that identify which of the two sexes is the female, and to which sex the fisheries and regions relate. This is required for the various utilities (R4MFCL, Viewer, etc.) to correctly display the reports graphically or in tables in respect of the two sexes. Consequently, the following indices and model output were added to:

- **.rep**: multi sex pointer; species sex pointer.
- **.fit**: in each fisheries block inserted the species number and the mean length-at-age for that species; pointers for the aggregated or dis-aggregated size compositions.

The reference documents for these report files were updated to reflect these changes and the version numbers at the top of each report were incremented to identify these format changes.

5.2 Improved standard deviation calculation

The standard deviation (`st.dev`) calculations for model-derived variables were improved by applying the singular value decomposition (SVD) method for the matrix calculations. This entailed adding OpenBLAS as a dependent library to the MULTIFAN-CL project to make the SVD routines available. Preliminary testing using the SKJ2016 example showed that after the derivative calculations for the fished and un-fished model evaluations, the `st.dev` calculations were complete in only 2 minutes; substantially faster performance. Comparisons of the `st.dev` estimates before and after the improvement indicated they are either identical or extremely similar. For a sample of 5000 out of 20636 variables the improved calculation estimates are within 3.3% of the original calculation.

During the course of this development, a minor problem was fixed with the `st.dev` report (**.var**) for age-specific natural mortalities. This part of the report was extended to include the estimates obtained from using the functional forms (see section 4.6).

5.3 Improved formulation for logistic selectivity

Preceding version 2.0.2.2, the formulation of the logistic selectivity used a power function that was prone to causing arithmetic exceptions (overflow) in early stages of minimisation due to implausible parameter values. Since the bounds on the parameters are shared among the spline and logistic functions, it is possible for

values that are implausible for the logistic function (depending upon the input data) to be used during the minimisation. This problem has previously been dealt with by applying the generic selectivity penalty term using `parest_flags(74)` to avoid such implausible parameter values. However, this has proved cumbersome and ineffective in some cases. An alternative formulation that produces an equivalent logistic function was developed using an exponential function. It was tested and found to be robust to implausible parameter values and did not require the additional generic selectivity penalty term. It was distributed with the release of version 2.0.2.2 as a minor revision.

5.4 Bug fixes

5.4.1 Fix to option for increasing the number of spline nodes

An existing feature that enables the number of nodes in spline functions for fishery selectivity to be increased was found to fail. This error related to the implementation of time-variant selectivities released in version 2.0.0.1, because increasing the number of nodes was not compatible with the time-variant selectivity data structures. The ability to increase the spline degree is not an integral part of the spline parameterization. The spline models are not nested. Therefore, when increasing the number of nodes, the new node points are not in general a superset of the old nodes, but rather they occur at different points. To avoid repeating previous minimization phases for estimating the selectivities, the values for the new spline coefficients can be selected which produce the closest selectivities to those for the original spline parameterization. This entailed extending the dimensions of the time-variant selectivity data structures, and including them in the algorithm that obtains the selectivities consistent with original parameterisation.

5.4.2 Fix to the age of common selectivity

An option for the estimation of fishery-specific selectivity is to specify an age class at which selectivity indices for all classes at that age, and higher, are common. This option is controlled by `fish_flags(i,3)` having a value as specified by the user. When the value is not set, i.e. the value is 0, the default age class is applied, being one less than the maximum age class, *nage*-1. Although seldom done, it is possible to set a non-zero value to the flag during early phases of the doital fit, and then to re-set the flag value to zero in later phases. Following the implementation of routines that check the integrity of `fish_flags` in versions preceding 2.0.2.1, this practice of restoring `fish_flags(i,3)` to a value of zero in later phases was found to cause an error. The error-checking routines were therefore corrected to restore the minimum age of common selectivity to *nage*-1 in this instance.

5.4.3 Fix to equilibrium yield curve calculation for multi-sex model

Equilibrium yields for the multi-sex model are calculated in respect of the spawning biomass for the female sex only, with the catches at equilibrium being aggregated among the sexes. A minor syntax error resulted in spawning biomass for the males being used when the female was the first sex. This was corrected.

5.4.4 Fix to initial population numbers in maximum age class for multi-species/sex models

Preliminary runs of the multi-sex swordfish model indicated the initial population numbers at equilibrium for the aggregate age class (*nage*) were unexpectedly high. This identified an error in the calculation for the case of multi-species/sex models that differed from the single species case. For the multi-species case the *nage* numbers were being aggregated after each movement between recruitment events, instead of aggregating fish into the maximum age class after all movements between recruitment events. The latter was being correctly applied to single species models. The correction was made for the cases of multi-species/sexes.

5.4.5 Fix to the input of size composition data

During benchmark testing, a change to the size composition likelihood was identified for some test data sets and found attributable to a fix made during the March 2016 developer's workshop. This fix related to the declaration of a long int in earlier versions. The previous declaration resulted in the loss of the input of the first non-zero element of the vector of size composition data input from the *.frq*. Correction of the error was found to have minimal impact on the benchmark testing results due to minimal instances of non-zero values in the first vector element for most test data sets.

5.4.6 Fix to the long int definition for the Windows compilation

During benchmark testing it was found that the Windows executable failed during multiple evaluations of multi-species/sex data sets. The problem was traced to the definition of long int being only 32 bits. The definition was changed conditional upon the macro `__MINGW64__` for the MinGW cross-compiler to be: long long int. In the case of the ADMB compilation the definition was `#define AD_LONG_INT long long int`. This entailed substantial modifications to the ADMB code used for the dependent library of the MinGW compilation for the windows executable.

5.4.7 Fix to the catch calculation for tagged fish

Routine testing following the March 2017 developer's workshop revealed a minor difference in the tagging data likelihood calculation compared to that obtained using the pre-workshop version. This was traced to the tag catch calculations that had been modified for developing the simulation capability of tagging pseudo-observations. A simple syntax error. The correct code was restored in the post-workshop version to produce the identical likelihood as obtained previously.

5.4.8 Fix to fish_flags used for DM likelihood

An existing feature employed `fish_flags(i,70)` in scaling the average annual fishing mortalities by age used for applying fishery-specific penalties in the equilibrium yield calculations. In error, use of `fish_flags(i,70)` was duplicated for activating the estimation of a DM likelihood parameter (length frequency relative sample size covariate exponent). The effect of this bug was to produce spurious estimates for the fishing mortality multiplier (***Fmult***). The use of this `fish_flags` for the DM likelihood was changed to `fish_flags(i,89)`.

5.4.9 Fix to parameter labels in xinit.rpt

The report "xinit.rpt" lists the independent parameters being estimated making up the x-vector. A simple syntax error produced errors in the labels for the DM likelihood parameters for weight frequency data. These were corrected to be the appropriate `fish_pars(24)` and `fish_pars(25)`.

5.4.10 Fix to missing components in report "test_plot_output"

The report "test_plot_output" contains details of the component terms of the total objective function calculated in the integrated model fit. During the objective function calculation, each component term is assigned to members of the class ***ppstf***. Values for the total catch likelihood term in respect of catches in weight, were correctly assigned to `ppstf->tot_catch_like_by_realization`, however this was neglected for catches in numbers of fish. This was correctly assigned. It was also confirmed that the assignments were made correctly for either single or multi-species catches in respect of fish numbers.

A similar error was found in respect of the likelihood component for the BH-SRR fit (***bh_steep***). While for the single species case the ***bh_steep*** contribution was correctly assigned to ***ppstf***; this had not been extended for the cases of multi-species/sexes. This was corrected, and for the multi-sex case, it assigns the single ***bh_steep*** for the female sex only. For the cases of multi-species/sexes the ***bh_steep*** contribution is reported to "test_plot_output" as a vector. In the multi-sex case, the vector element for the male "species" will be zero. For the multi-species case, both elements will be non-zero.

6 APPLICATION OF NEW FEATURES

A number of the new features and improvements described in sections 4 and 5 were employed for the 2017 stock assessments for WCPO bigeye tuna, BET, (McKechnie et al. 2017), yellowfin tuna, YFT, (Tremblay-Boyer et al. 2017), south Pacific swordfish, SWO, (Takeuchi et al. 2017), and for stochastic simulations with potential for Management Strategy Evaluations, MSE, (Scott et al. 2017). Specifically these are listed in respect of the feature and stock assessment:

- Improvements for multi-sex model for the SWO assessment (Takeuchi et al. 2017), where investigative models were developed using:
 - Predicted size compositions scaled by sex ratios
 - Length-specific selectivity shared among sexes

- Mixed combinations of aggregated and dis-aggregated fisheries data
- Improved report files for indexing multi-sex output
- Formulations for age-specific natural mortality (BET and YFT)
- Improved formulation of logistic selectivity (BET, YFT, SWO)
- Improved standard deviation using the OpenBLAS library routine for SVD (BET, YFT, SWO)
- Estimating autocorrelation in recruitments (BET, YFT)
- Likelihood profiling in respect of model derived quantities: average absolute biomass, relative depletion (BET, YFT, SWO)
- Investigative modelling was also done using the DM likelihood (BET)
- Simulation projections under BH-SRR recruitments predictions with stochastic deviates, and generating pseudo-observations of size compositions and relative abundance indices (MSE).

7 FUTURE WORK

A future work plan for the development of new features in MULTIFAN-CL is suggested in Table 3 with those having high priority being undertaken in 2017-18, while others may be addressed in subsequent years.

Of note are to resume work on the four remaining recommendations of Ianelli et al. (2012) yet to be implemented:

- Non-uniform size bins (recommendation "b")
- Long-term tag loss (recommendation "c")
- Tags inform movement (recommendation "d")
- Maturity at length (recommendation "k")

Tasks of immediate priority and relating to the new features recently added (Section 4) are discussed in the following section.

8 DISCUSSION

2016-17 has seen a change of practice in managing the MULTIFAN-CL repository development branch. In previous years the separation of the master and development branches endured for long periods while new features were being implemented. In contrast, in the past year modified code has been committed to the development branch regularly (27 commits), and a merge with the master done in Jan. 2017 preceding the release of version 2.0.2.1. Regular testing of the development branch relative to the master has since followed, especially before and after workshops, with another merge intended upon completion of the next comprehensive set of benchmark tests. This practice is an improvement over previous years, which prevents large deviations occurring between the two branches that creates potential for extensive debugging being required to identify the causes of calculation differences. The current practice is more streamlined and uses the repository as intended.

The list of bug fixes (section 5.4) made during 2016-17 is more than in previous years, but this reflects how important benchmark testing is in revealing coding errors generated when developing new features. Regular benchmark testing and testing immediately following new developments is certainly best practice. Also implementing the multi-sex feature in a real stock assessment context revealed a number of necessary and related fixes. Constructing the multi-sex swordfish model also identified a number of improvements to that feature, particularly for the special case of sharing length-specific selectivity among the sexes which is a better approach than using age-specific selectivity for a sexually dimorphic species such as swordfish (Takeuchi et al. 2017).

As in 2015-16, the core project task has been the development of innovative approaches for fitting to size composition data. The main theme of this task has been consolidating the method formulated by the primary developer Dr Dave Fournier, the SSMULT_RE. A substantial amount of testing has been done using examples of real stock assessment data that have revealed key issues like minimization problems and certain diagnostic requirements. An unsatisfactory result from these trials has been that estimates of the correlation in random effects have tended towards the upper bound. With closer examination through simulation testing, there may be problems in estimating both the ESS and the correlation simultaneously. Consequently, the

obvious alternative of the Dirichlet-Multinomial has been explored with apparent success. Preliminary simulation results indicate its good performance in estimating ESS, although the approach is somewhat sensitive to assumed N_{max} parameter values. This core task is now at an advanced stage and these most recent findings may be confirmed with a final set of simulation experiments proposed for the latter part of 2017. Upon completion of these experiments, some final refinements may be made to the implementation of these methods in MULTIFAN-CL preceding their release.

Directing resources to this core task is considered warranted given the importance of size composition data in WCPO tuna stock assessments. It is worth reiterating that model uncertainty due to assumptions made for the relative importance amongst the various data types included in the integrated modelling approach has dominated most fisheries assessment models. Selecting appropriate “weights” for these data is a “challenging aspect” of fisheries modelling (Punt 2015). The self-scaling properties of the SSMULT_RE and DM approaches allow the ESS to be estimated based upon robust statistical methods. This represents a real advance in reducing this area of model uncertainty. Consolidating their development has therefore been a priority for 2016-17.

The implementation of the remaining four recommendations by Ianelli et al. (2012) was not addressed during 2016-17. Higher priority has been given to: methods for fitting size composition data, a simulation mode for generating pseudo-observations, and several unforeseen tasks, including improvements of the multi-sex feature as needed for the south-west Pacific swordfish model. However, the Peer-review recommendations remain prominent in the project task list for 2017-18.

Increasing the utility of MULTIFAN-CL for MSE is important for its potential role in the simulation framework being developed for this type of management recommendation (Scott et al. 2017). Good progress was made during 2016-17 with the new simulation mode feature, and Scott et al. (2017) have demonstrated the potential of the pseudo-observations generated for undertaking MSE. The innovative algorithm for simulating tag recapture data (section 4.4.3) has utility for exploring the performance of decision rules based upon such data. Completing this feature entails: completing the generation of pseudo-observations of tagging data (transform into recaptures at length, and consider the potential for including overdispersion in the observation error); producing pseudo-observations (all data types) for the estimation period; and, widening the range of process error types to include auto-correlation in the recruitments. Estimating or deriving autocorrelation in the operating model is now possible, making it feasible to include it in generating stochastic recruitments for projections. Completing the development of this feature is the highest priority for the remaining part of 2017.

9 REFERENCES

- Davies, N., Fournier, D., Bouyé, F., and Hampton, J. 2016. Developments in the MULTIFAN-CL software 2015-16. WCPFC-SC12-2016/SA -IP-10
- 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
- Francis, R.I.C.C. 2014. Replacing the multinomial in stock assessment models: a first step. *Fisheries Research* **151**: 70-84
- Huber, P.J. 2009. Robust statistics. 2nd edition/Huber, P.J., Ronchetti, E.M., John Wiley & Sons, Inc., Hoboken, New Jersey
- Hrafinkelsson, B. and Stefansson, G. 2004. A model for categorical length data from groundfish surveys. *Can. J. Fish. Aquat. Sci.* **61**:1135-1142
- 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. 2017. MULTIFAN-CL User's Guide. <http://www.multifan-cl.org/>

Lorenzen, K. 1996. The relationship between body weight and natural mortality in juvenile and adult fish: a comparison of natural ecosystems and aquaculture. *J.Fish Biol.* **49**:627-647

McKechnie, S., Pilling, G. and Hampton, J. 2017. Assessment of bigeye tuna in the WCPO. WCPFC-SC13-2017/SA-WP-3, Rarotonga, Cook Islands 9-17 August 2017.

Methot, R. 2013. User Manual for Stock Synthesis – model version 3.24s. http://nft.nefsc.noaa.gov/Stock_Synthesis_3.htm

Punt, A.E. 2015 in press. Some insights into data weighting in integrated stock assessments. *Fish.Res.* (2015) <http://dx.doi.org/10.1016/j.fishres.2015.12.006>

Scott, R., Davies, N., Pilling, G. and Hampton, J. 2017. Generating pseudo-data in MULTIFAN-CL. WCPFC-SC13-2017/MI-IP-02

Takeuchi, Y. et al. 2017. Assessment of swordfish in the south-western Pacific. WCPFC-SC13-2017/SA-WP-13, Rarotonga, Cook Islands 9-17 August 2017.

Tremblay-Boyer, L., McKechnie, S., Pilling, G. and Hampton, J. 2017. Assessment of yellowfin tuna in the WCPO. WCPFC-SC13-2017/SA-WP-05, Rarotonga, Cook Islands 9-17 August 2017.

10 TABLES

Table 1: Comparison of the total objective function, tagging data likelihood, gradient, and population scalar parameter as calculated using the MULTIFAN-CL Linux compilation (Linux) and the VS2017 compilation of the Windows executable (run on Windows 7 and 10 platforms).

	Linux	Win7 platform	Win10 platform
Obj.fn	-6.17602425864468e+04	-6.17602425864229e+04	-6.17602425864368e+04
Tag.data	2.75971189195884e+04	2.75971189195603e+04	2.75971189195728e+04
Gradient	3.57476468178577e+00	3.57476367998121e+00	3.57476431569964e+00
Totpop	2.10057550179922e+01	2.10057550179928e+01	2.10057550179928e+01

Table 2. New features added to MULTIFAN-CL with respect to their state of completion as of August 2017.

Peer review recommendations		
Task	Description	Status of completion
i. Tail compression	Include a “tail compression” option, which would pool all length- and weight-data for large and small sizes based on a specified percentage.	Development 100%; Testing 100%
j. Multinomial distribution for size composition data	Add an option which allows the analyst to assume a multinomial likelihood for the compositional data in the first phases	Development 100%; Testing 100%
l. Tag likelihood relative weighting	An option to add a likelihood weight to the tagging data component should be added	Development 100%; Testing 100%
k. Maturity-at-length	When maturity data are based on length, converting to ages should be done within the model. Presently, the maturity-at-age is based on a fixed age-length relationship.	Development 0%
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%
c. Long-term tag loss	Allow for long-term and initial tag-loss. Currently initial tag-loss is implemented by reducing the number of animals tagged when inputting data to the model and no account can be taken of long-term tag-loss.	Development 0%
d. Tags inform movement	Include an option which allows the tagging data to inform movement only rather than movement and mortality	Development 0%
Other developments		
Task	Description	Status of completion
Initial biomass approximates the equilibrium unexploited biomass	Abundance in the first model period is equivalent to that estimated using the estimated BH-SRR in an unexploited equilibrium state	Development 90%; Testing 20%
Region-specific yield estimation	Calculate the equilibrium productivity and biomass from each region in a spatially disaggregated model	Development 100%; Testing 20%

Self-scaling multinomial with random effects	Add a multinomial likelihood for composition data that is modified to be self-scaling and estimates auto-correlated random effects.	Development 100%; Testing 90%
Exclude reporting rates from tag predictions during mixing period	Do not to account for the estimated tag reporting rate when solving the Newton-Raphson catch algorithm for tags recaptured during the mixing period	Development 100%; Testing 20%

Table 3. New features to be added to MULTIFAN-CL in 2017-18 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.	2016-17
c. Long-term tag loss	Allow for long-term and initial tag-loss. Currently initial tag-loss is implemented by reducing the number of animals tagged when inputting data to the model and no account can be taken of long-term tag-loss.	2016-17
d. Tags inform movement	Include an option which allows the tagging data to inform movement only rather than movement and mortality	2016-17
k. Maturity at length	When maturity data are based on length, converting to ages should be done within the model.	2016-17
j. Multinomial distribution for size composition data	Add an option which allows the analyst to assume a multinomial likelihood for the compositional data in the first phases	Complete testing 2017-18 (see SSMULT_RE below)
Other developments		
Task	Description	Implementation
Simulation mode	Output of simulation operating model pseudo-observations for use in simulation model testing and as an Operating Model for Management Strategy Evaluations: - implement pseudo-observations for tagging and age-length data; - generate pseudo-observations of all data for the estimation period of the model; - format pseudo-observations consistent with the .frq, .tag and .age_length input file formats; and, - implement process error in future recruitments with application of the derived autocorrelation coefficient in historical recruitment estimates	2017-18
Windows executable	Improve the static build using Visual Studio 2017 to reduce the number of dependent .dll files required to be ported with the executable.	2017-18
Cobb-Douglas scalar	A biomass-related effect on catchability implying hyper-stability in catch rates.	Complete testing 2017-18
Self-scaling multinomial with random effects (SSMULT_RE) and Dirichlet-Multinomial	Add a multinomial likelihood for composition data that is modified to be self-scaling and estimates auto-correlated random effects.	Complete testing 2017-18
Initial biomass approximates the equilibrium unexploited biomass	Calculate the equilibrium productivity and biomass from each region in a spatially disaggregated model	Complete testing 2017-18

Exclude reporting rates from tag predictions during mixing period	Do not to account for the estimated tag reporting rate when solving the Newton-Raphson catch algorithm for tags recaptured during the mixing period	Complete testing 2017-18
Region-specific yield estimation		Complete testing 2017-18
Tags inform growth	Add length increments of tag recaptures to inform growth estimation.	2017-18

Table 4: Simulation average bias in DM estimates of ESS in respect of assumed *Nmax* values.

"True" Nmax in Simulator	DM Nmax in Analyzer	Average bias in ESS
500	50	24.452048
500	250	11.552192
500	400	4.444176
500	500	1.121012
500	750	-4.704076
500	1250	-11.507652
500	2000	-17.153992

Table 5: Comparison among model quantities between RefCase and DM models indicating the effect of using the DM likelihood for size composition data with alternative assumptions for *Nmax*. Percentage difference is between DM_Nmax250 and DM_2000.

Model quantity	RefCase	DM_Nmax250	DM_Nmax1000	DM_Nmax1500	DM_Nmax2000	%diff_nmx2000_250
MSY	1627600	1642800	1670400	1675600	1678000	2.14
Ccurr.MSY	1.011	1.002	0.985	0.982	0.981	-2.09
Fmsy	0.197	0.213	0.208	0.208	0.207	-2.72
Fmult	1.658	1.912	1.983	2.005	2.017	5.49
Fcurr.Fmsy	0.603	0.523	0.504	0.499	0.496	-5.21
B0	6621000	5930000	6211000	6252000	6272000	5.77
Bmsy	2071000	1926000	2005000	2017000	2023000	5.04
Bcurr	3645390	3687621	3921952	3977954	4007873	8.68
SB0	6262000	5574000	5844000	5884000	5905000	5.94
SBmsy	1762000	1617000	1687000	1698000	1704000	5.38
SBcurr	3289576	3315413	3537495	3592042	3621325	9.23
Bcurr.Bmsy	1.760	1.915	1.956	1.972	1.981	3.47
SBcurr.SBmsy	1.867	2.050	2.097	2.115	2.125	3.65
SBcurr.SBcurrF0	0.489	0.527	0.535	0.539	0.540	2.63
SBlatest.SBlatestF0	0.482	0.500	0.510	0.513	0.514	2.72
obj_bhsteep	1.084	1.046	1.037	1.036	1.036	-0.94
obj_effdev	994.179	1018.004	1015.242	1015.416	1015.668	-0.23
obj_catdev	80.205	85.354	83.029	82.320	82.156	-3.75
obj_lencomp	-198131.093	105789.616	147886.245	160515.358	169541.415	60.26
obj_tagdata	25974.935	25784.420	25754.084	25753.700	25753.448	-0.12
obj_ageInгда	0.000	0.000	0.000	0.000	0.000	-
Obj	-170707.071	133068.018	175129.366	187762.167	196789.847	47.89
gradient	0.4128503	0.2302405	0.0706059	0.0654338	0.0178001	-92.27
Lmin	10.000	10.000	10.000	10.000	10.000	0.00
Lmax	88.317	88.317	88.317	88.317	88.317	0.00
K	0.197	0.197	0.197	0.197	0.197	0.00

Table 6: Comparison among quantities for models having age-specific natural mortality estimated using independent age-specific deviate parameters (Refcase), or the functional forms of splines (M_spline) or the Lorenzen function (M_lorenz). Percentage difference is with respect to the Refcase model.

Model quantity	Refcase	M_spline	M_lorenz	%diff_M_spline	%diff_M_lorenz
MSY	472900	474400	485200	0.32	2.60
Ccurr.MSY	3.648	3.636	3.555	-0.316	-2.54
Fmsy	0.244	0.251	0.272	2.875	11.79
Fmult	2.225	2.233	2.471	0.360	11.06
Fcurr.Fmsy	0.449	0.448	0.405	-0.358	-9.96
B0	7098000	6967000	6318000	-1.85	-10.99
Bmsy	1942000	1894000	1782000	-2.47	-8.24
Bcurr	4125554	4011290	3881456	-2.77	-5.92
SBO	6764000	6662000	6021000	-1.51	-10.98
SBmsy	1626000	1605000	1500000	-1.29	-7.75
SBcurr	3749074	3667094	3539762	-2.19	-5.58
Bcurr.Bmsy	2.124	2.118	2.178	-0.306	2.53
SBcurr.SBmsy	2.306	2.285	2.360	-0.907	2.35
obj_bhsteep	0.107	0.107	0.107	0.000	0.00
obj_effdev	1631.811	1631.811	1631.811	0.000	0.00
obj_catdev	58.283	58.283	58.283	0.000	0.00
obj_lencomp	-250887.004	-250887.004	16374.051	0	-106.53
obj_tagdata	24487.383	24735.031	24773.250	1.011	1.17
Obj	-224289.687	-224125.298	-223989.587	-0.073	-0.13
Gradient	0.0097396	62.2693952	4.9839515	639242.25	51072.02
Lmin	10.000	10.000	10.000	0.000	0.00
Lmax	88.317	88.317	88.317	0.000	0.00
K	0.197	0.197	0.197	0.000	0.00

11 FIGURES

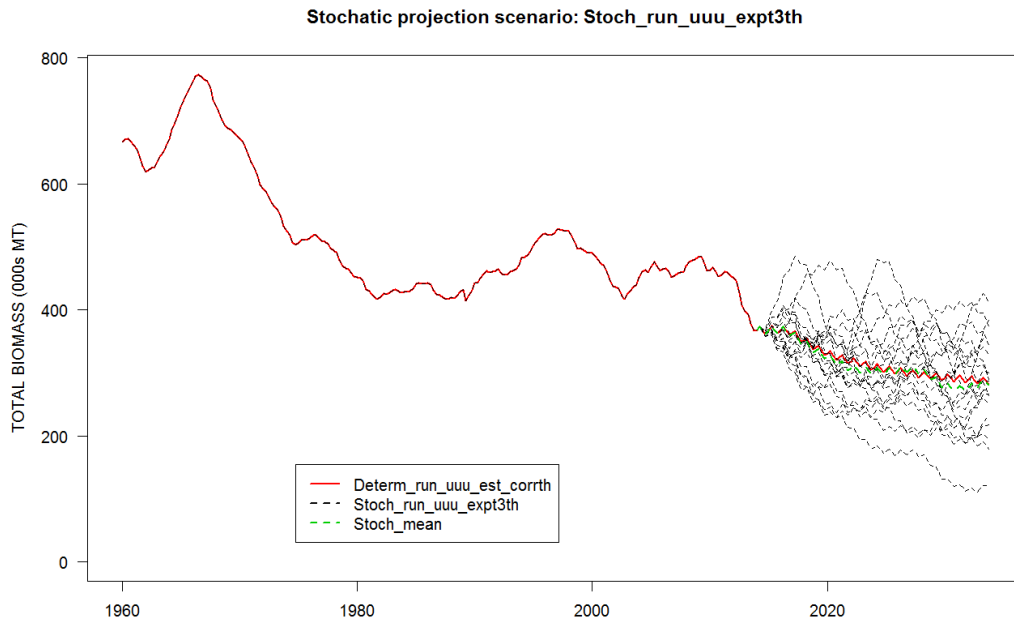


Figure 1: Comparison of total biomass between a deterministic projection having recruitments predicted from the annualised BH-SRR allocated to quarters according to the average seasonal proportions (Determ_run_uuu_est_corrth), and 20 simulation projections using stochastic recruitment deviates sampled from the theoretical distribution (Stoch_run_uuu_expt3th) with the mean level shown (Stoch_mean).

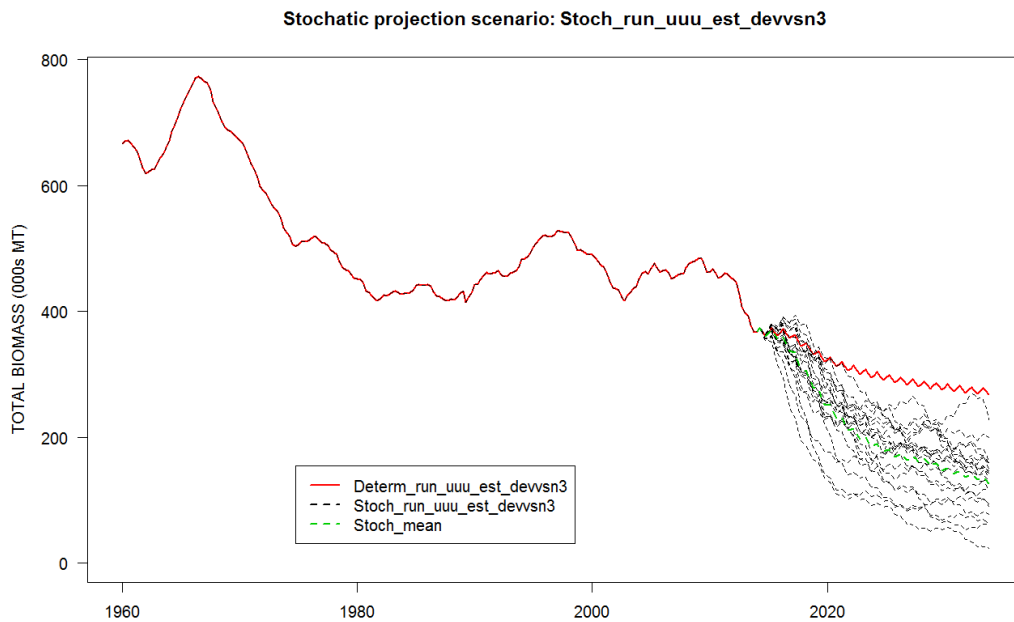


Figure 2: Comparison of total biomass between a deterministic projection having recruitments predicted from the annualised BH-SRR allocated to quarters according to the average seasonal proportions (Determ_run_uuu_est_devvsn3), and 20 simulation projections using stochastic recruitment deviates sampled from the estimated deviates (Stoch_run_uuu_est_devvsn3) with the mean level shown (Stoch_mean).

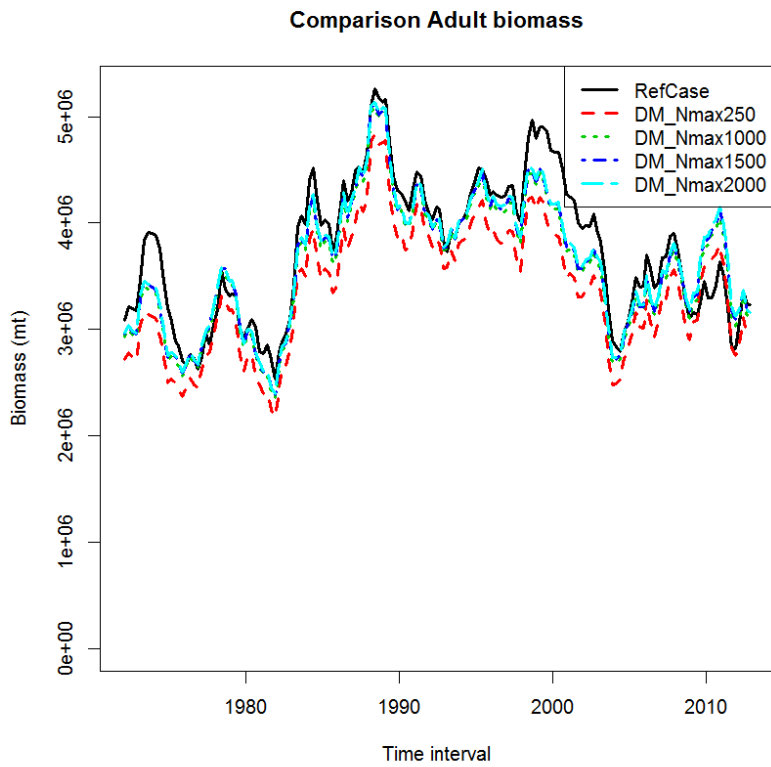


Figure 3: Absolute spawning biomass estimated using the robust-normal size composition likelihood (RefCase) and the DM likelihood under a range of assumed N_{max} values (DM_Nmax250 to DM_Nmax2000).

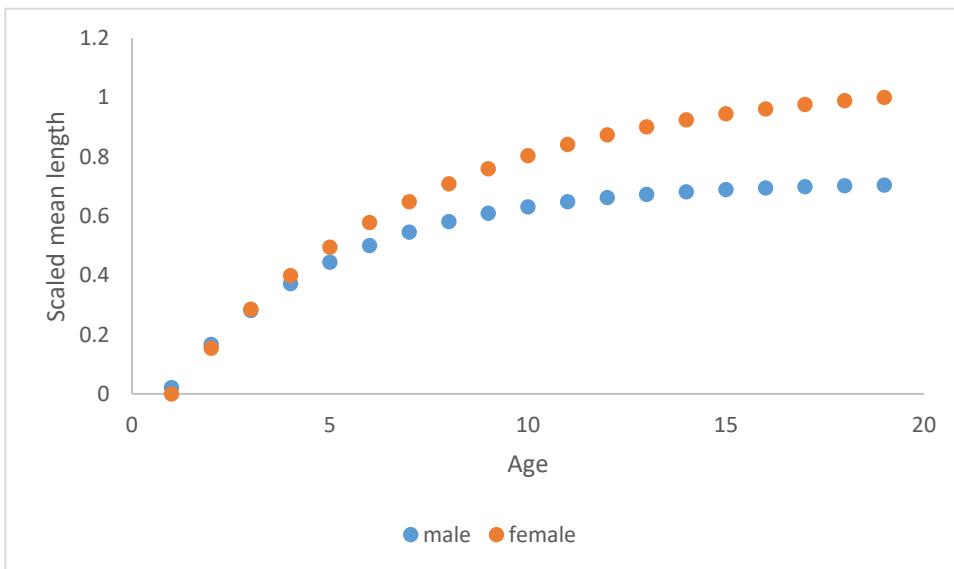


Figure 4: Sex-specific mean lengths at age re-scaled from zero to 1, (tlength).

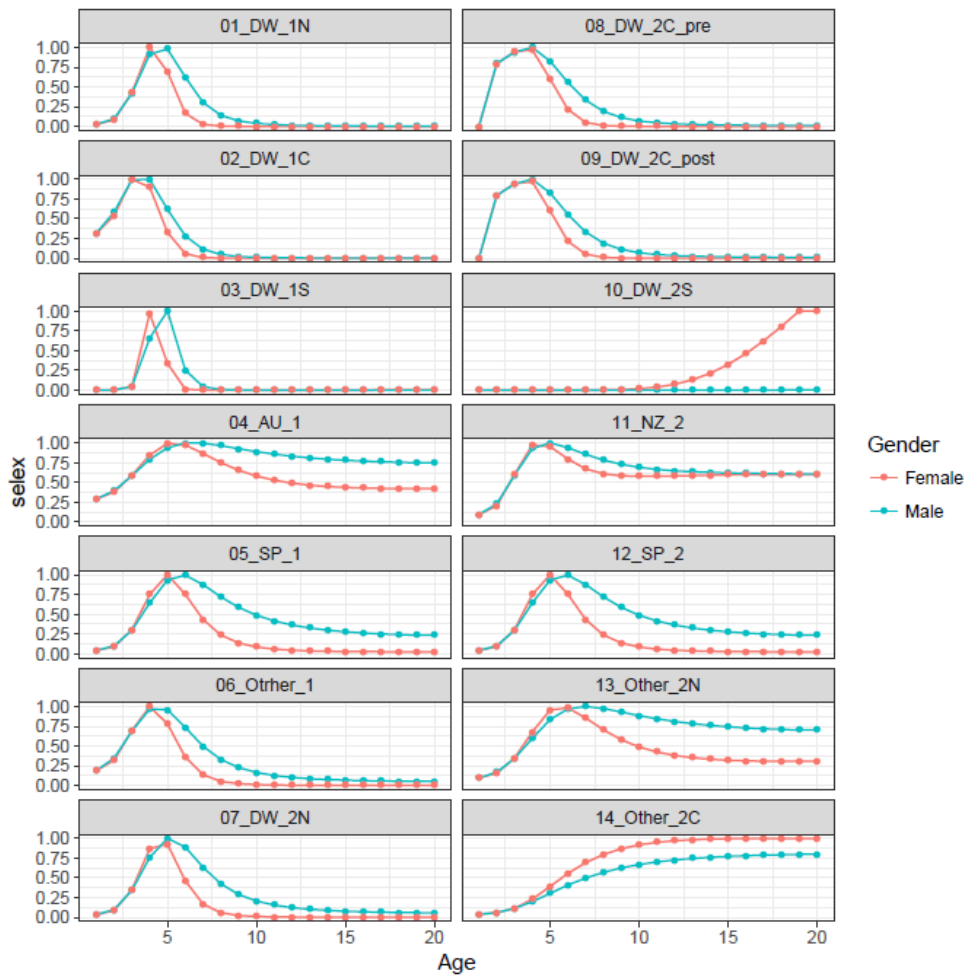


Figure 5: Sex-specific selectivity at age derived from re-scaled mean lengths at age shared among sexes.

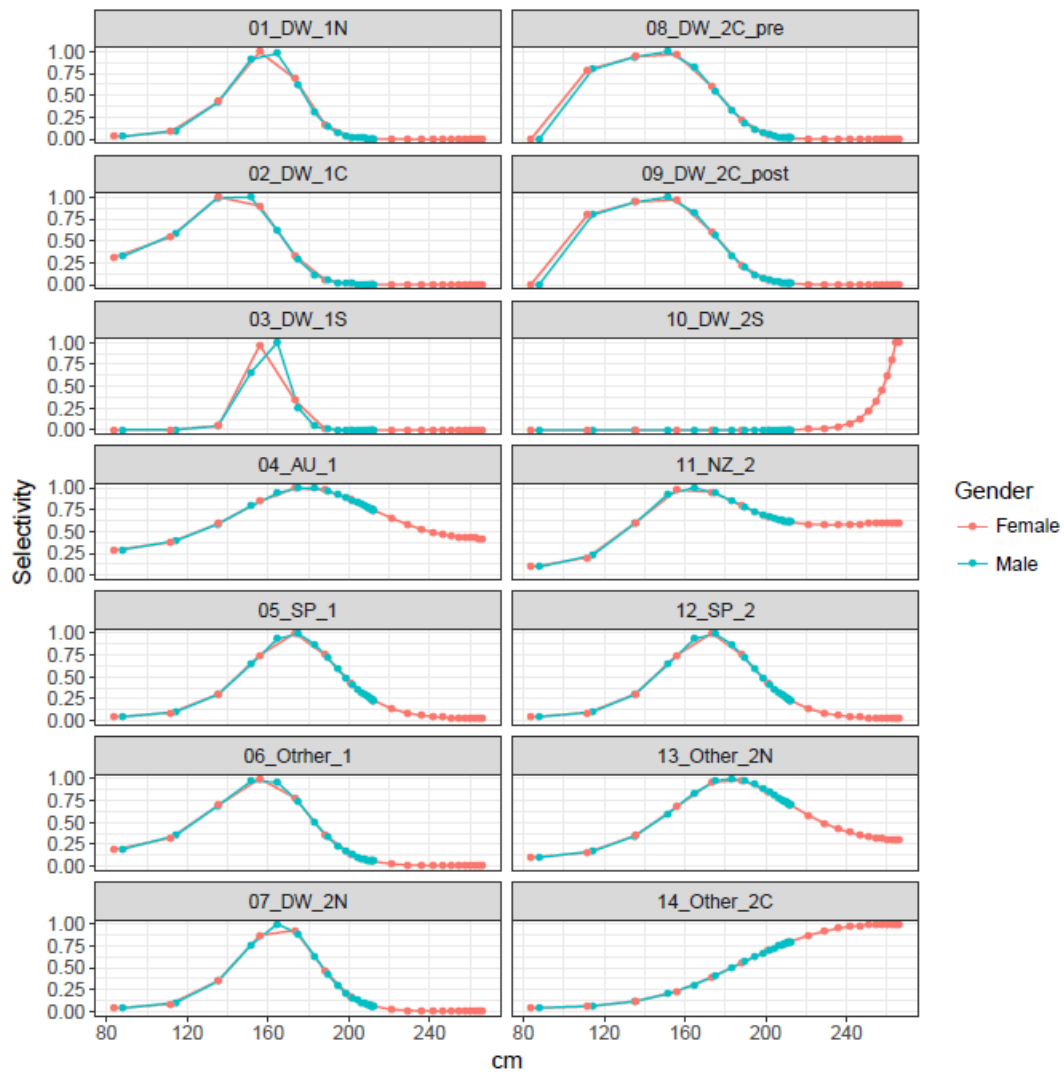


Figure 6: Sex-specific selectivity at length derived from re-scaled mean lengths at age shared among sexes.

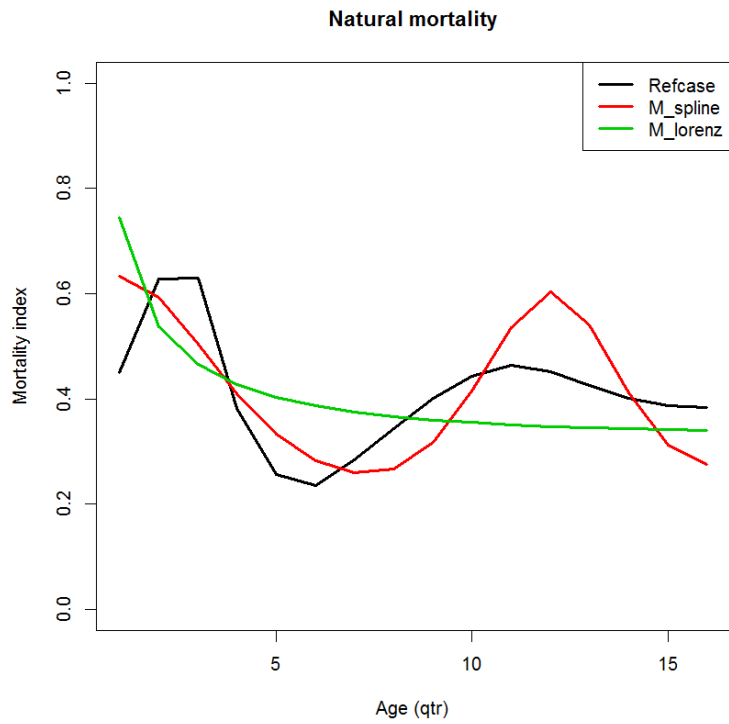


Figure 7: Age-specific natural mortality estimated using independent age-specific deviate parameters (Refcase), or the functional forms of splines (M_spline) or the Lorenzen function (M_lorenz).

12 ANNEX

12.1 Benchmark testing

Summary of benchmark tests of the development version using single species data undertaken August 2013 to July 2017.

Compilation name	Compilation folder & Exec.name	Compilation date	Test folder	Test date	Test configurations	Results folder	Key results
Benchmark testing 16 September 2014							
MTHRD_2014_05_13	multi_spp_2014_05_13; exec_txf/mfclo64_2014_05_13	May.2014	2014-09-12_fit_bet	16/09/2014	af145 = -3; penwt=10.0**af145	2014-09-12_fit_bet	penwt=0.001; Obj.=1068534.244292082731; MSY=3.047e+06; B0=2.217e+08
MTHRD_2014_05_17	multi_spp_2014_05_17; exec_txf/mfclo64_2014_05_17	Sep.2014	2014-09-12_fit_bet	16/09/2014	af145 = -3; penwt=10.0**af145	2014-09-12_fit_bet	penwt=0.001; Obj.=1068556.974073923426; MSY=3.422e+06; B0=2.514e+08
Benchmark and Release version testing 22 September 2014							
MTHRD_2014_05_17	multi_spp_2014_05_17; exec_txf/mfclo64_2014_05_17	Sep.2014	2014-09-22_fit_bet	22/09/2014	af145 = 1; pen_wt=af145	2014-09-22_fit_bet	pen_wt=1; Obj.=1068477.068569175899; MSY=1.894e+04; B0=1.410e+06
Release_vsn.1.1.5.8	C:\Nick\MFCL\2011-10-07\mfcl\ C:\Nick\I_Assessments\MFCL\2014_09_12	Sep.2014	2014-09-22_fit_bet	22/09/2014	af145 = 1; pen_wt=af145	2014-09-22_fit_bet	pen_wt=1; Obj.=1068457.480964803835; MSY=1.913e+04; B0=1.420e+06
Development version and Release version testing 20 October 2014							
MTHRD_2014_09_16	multi_spp_2014_06_19; exec_txf/mfclo64_2014_06_19	8 Oct.2014	2014-09-22_fit_bet	20/10/2014	af145 = 1; pen_wt=af145	2014-09-22_fit_bet	pen_wt=1; Obj.=1068477.068569175899; MSY=1.894e+04; B0=1.410e+06
Release_vsn.1.1.5.8	C:\Nick\MFCL\2011-10-07\mfcl\ C:\Nick\I_Assessments\MFCL\2014_09_12	Sep.2014	2014-09-22_fit_bet	20/10/2014	af145 = 1; pen_wt=af145	2014-09-22_fit_bet	pen_wt=1; Obj.=1068457.480964803835; MSY=1.913e+04; B0=1.420e+06
Benchmark and Development version testing 6 November 2014							
MTHRD_2014_05_17	multi_spp_2014_05_17; exec_txf/mfclo64_2014_05_17	Sep.2014	2014-11-06_fit_bet	6/11/2014	af145 = 1; pen_wt=af145	2014-11-06_fit_bet	pen_wt=1; Obj.=1068474.325769255171; MSY=1.884e+04; B0=1.397e+06
MTHRD_2014_09_16	multi_spp_2014_06_19; exec_txf/mfclo64_2014_06_19	5 Nov.2014	2014-11-06_fit_bet	6/11/2014	af145 = 1; pen_wt=af145	2014-11-06_fit_bet	pen_wt=1; Obj.=1068511.045675329166; MSY=1.899e+04; B0=1.405e+06
Benchmark and Development version testing 19-27 February 2015							
MTHRD_2014_09_16	multi_spp_2014_06_19; exec_txf/mfclo64_2014_06_19	5 Nov.2014	2015-02-27_fit_bet/ bnchmrk_bet2011_fit	27/02/2015	af145 = 1; pen_wt=af145	2015-02-27_fit_bet	pen_wt=1; Obj.=1068511.045675329166; MSY=1.899e+04; B0=1.405e+06
MTHRD_2014_11_07	multi_spp_2014_11_07; exec_txf/mfclo64_2014_11_07	26 Feb.2015	2015-02-27_fit_bet/ devvsn_bet2011_fit	27/02/2015	af145 = 1; pen_wt=af145	2015-02-27_fit_bet	pen_wt=1; Obj.=1068553.274013987277; MSY=1.854e+04; B0=1.386e+06

Benchmark and Development version testing 16 November - 2 December 2015							
MTHRD_2014_11_07	multi_spp_2014_11_07; exec_txf/mfclo64_2014_11_07	26 Feb.2015	2015-11-25_fit_bet/ bnchmrk_bet2011_fit	25/11/2015	af145 = 1; pen_wt=af145	2015-11-25_fit_bet	pen_wt=1; Obj.=1068553.274013987277; MSY=1.854e+04; B0=1.386e+06
MTHRD_2015_09_15	multi_spp_2015_09_15; exec_txf/mfclo64_2015_09_15	3 Nov.2015	2015-11-25_fit_bet/ devvsn_bet2011_fit	25/11/2015	af145 = 1; pen_wt=af145	2015-11-25_fit_bet	pen_wt=1; Obj.=1.06765392664784e+06; MSY=1.775e+04; B0=1.074e+06
Benchmark and Development version testing 17 February 2016							
MTHRD_2015_09_15	multi_spp_2014_11_07; exec_txf/mfclo64_2014_11_07	26 Feb.2015	2016-02-17_fit_skj2014/ release_skj2014_fit	18/02/2016	af145 = -1; pen_wt=10^af145	2016-02-17_fit_skj2014	pen_wt=0.1; Obj.=170692.859; MSY=4.047e+005; B0=6.585e+006
MTHRD_2015_09_15	multi_spp_2015_09_15; exec_txf/mfclo64_2015_09_15	2 Feb.2016	2016-02-17_fit_skj2014/ devvsn_skj2014_fit	18/02/2016	af145 = -1; pen_wt=10^af145	2016-02-17_fit_skj2014	pen_wt=0.1; Obj.=170247.097; MSY=4.056e+005; B0=6.603e+006
Benchmark and Development version testing 20 June 2016							
MTHRD_2015_09_15	multi_spp_2015_09_15; exec_txf/mfclo64_2015_09_15	2 Feb.2016	2016-06-10_fit_yft2014/ bnchmrk_yft2014_fit	17/06/2016	af145 = -2; pen_wt=10^af145	2016-06-10_fit_yft2014	pen_wt=0.01; Obj.=1195760.515; MSY=1.469e+05; B0=4.322e+06
MTHRD_2015_09_15	multi_spp_2015_09_15; exec_txf/mfclo64_2015_09_15	21 Jun.2016	2016-06-10_fit_yft2014/ devvsn_yft2014_fit	17/06/2016	af145 = -2; pen_wt=10^af145	2016-06-10_fit_yft2014	pen_wt=0.01; Obj.=1195760.641; MSY=1.467e+05; B0=4.319e+06
Benchmark and Development version testing 16 August 2016							
MTHRD_2015_09_15	multi_spp_2015_09_15; exec_txf/mfclo64_2015_09_15	21 Jun.2016	2016-08-16_fit_skj2014/2016-08-16_fit_skj2014_bnchmrk	16/08/2016	af145 = -1; pen_wt=10^af145	2016-08-16_fit_skj2014	pen_wt=0.1; Obj.=170249.547; MSY=4.046e+05; B0=6.562e+06
MTHRD_2016_04_15	multi_spp_2016_04_15; exec_txf/mfclo64_2016_04_15	15 Aug.2016	2016-08-16_fit_skj2014/2016-08-16_fit_skj2014_devvsn2	16/08/2016	af145 = -1; pen_wt=10^af145	2016-08-16_fit_skj2014	pen_wt=0.1; Obj.=170214.267; MSY=3.982e+05; B0=6.405e+06
Benchmark and Development version testing 7 November 2016 - devvsn4							
MTHRD_2015_09_15	multi_spp_2015_09_15; exec_txf/mfclo64_2015_09_15	21 Jun.2016	2016-11-07_fit_skj2014/2016-11-07_fit_skj2014_bnchmrk	7/11/2016	af145 = -1; pen_wt=10^af145	2016-11-07_fit_skj2014	pen_wt=0.1; Obj.=170249.547; MSY=4.046e+05; B0=6.562e+06
MTHRD_2016_10_18	multi_spp_2016_10_18; exec_txf/mfclo64_2016_10_18	7 Nov.2016	2016-11-07_fit_skj2014/2016-11-07_fit_skj2014_devvsn4	7/11/2016	af145 = -1; pen_wt=10^af145	2016-11-07_fit_skj2014	pen_wt=0.01; Obj.=170214.267; MSY=3.982e+05; B0=6.405e+06
Benchmark and Development version testing 29 March 2017							
MTHRD_2016_10_18	multi_spp_2016_10_18; exec_txf/mfclo64_2016_10_18	7 Nov.2016	2017-03-29_fit_yft2014/vsn2.0.2.1_yft2014_fit	29/03/2017	af145 = -2; pen_wt=10^af145	2017-03-29_fit_yft2014	pen_wt=0.01; Obj.=1195611.492; MSY=592400; B0=4460000
MTHRD_2016_10_18	multi_spp_2016_10_18; exec_txf/mfclo64_2016_10_18	28 Mar.2017	2017-03-29_fit_yft2014/vsn2.0.2.2_yft2014_fit	29/03/2017	af145 = -2; pen_wt=10^af145	2017-03-29_fit_yft2014	pen_wt=0.01; Obj.=1195578.322; MSY=583200; B0=4374000
Benchmark (vsn.2.0.2.2) and Development version (devvsn7) testing 17 May 2017							
MTHRD_2016_10_18	multi_spp_2016_10_18; exec_txf/mfclo64_2016_10_18	4-May-17	2017-05-17_fit_yft2014/vsn2.0.2.2_yft2014_fit	17/05/2017	af145 = -2; pen_wt=10^af145	2017-05-17_fit_yft2014	pen_wt=0.01; Obj.=1195578.322; MSY=583200; B0=4374000
MTHRD_2017_05_02	..\Assessments\MFCL\2017_05_02_devvsn7	17-May-17	2017-05-17_fit_yft2014/devvsn7_yft2014_fit	17/05/2017	af145 = -2; pen_wt=10^af145	2017-05-17_fit_yft2014	pen_wt=0.01; Obj.=1195578.322; MSY=583200; B0=4374000

Benchmark (vsn.2.0.2.2) and Development version (devvsn7) testing 6 June 2017							
MTHRD_2016_10_18	multi_spp_2016_10_18; exec_txf/mfcl064_2016_10_18	4-May-17	2017-05- 17_fit_yft2014/vsn2.0.2.2_yft20 14_fit	6/06/2017	af145 = -2; pen_wt=10^af145	2017-06- 06_fit_yft2014	pen_wt=0.01; Obj.=1195578.322; MSY= 583200; B0=4374000
MTHRD_2017_05_02	multi_spp_2017_05_02; ..\Assessments\MFCL\2017_05_02_devvs n7	2-Jun-17	2017-05- 17_fit_yft2014/devvsn7_yft2014 _fit	6/06/2017	af145 = -2; pen_wt=10^af145	2017-06- 06_fit_yft2014	pen_wt=0.01; Obj.=1195578.322; MSY= 583200; B0=4374000

12.2 Agenda for Training Workshop, March 2017

MULTIFAN-CL training workshop – 6 to 7 March 2017, OFP

Venue: ICT room, SPC campus

Timetable: 8:30 am to 4:30 pm

Aim: To achieve proficiency with the new features in MULTIFAN-CL version 2.0.2.1, particularly with:

- the size-composition likelihoods
- simulation model operations, and
- using the Linux debugger.

Presenters: Nick Davies; Dr Dave Fournier

Agenda

1. Workshop outline

- Aim
- Scope of material and course structure
- Participant's requirements

2. New features in version 2.0.2.1

An introduction is provided to each feature, followed by an application using an example model.

- Size-composition likelihoods:
 - The self-scaling multinomial M-estimator (SSMULT)
 - Dirichlet-Multinomial likelihood
 - Tail compression for weight composition data with a revised penalty formulation
- Simulation mode for generating pseudo-observed data
 - Description of current implementation
 - Flag settings for activation and outputs generated
 - Further developments proposed for this feature
 - Application using the ALB2015 example
- Autocorrelation in recruitment estimates
- Functional forms for the estimated age-specific natural mortality
- An improved algorithm for applying stochastic recruitment deviates to BH-SRR predictions in simulation projections
- Improved variance calculation that uses the OpenBLAS library routine for the singular value decomposition required for the st.dev report
- Added a "catch" to detect when a vector of zeroes in size frequency data is input

3. Overview of the code structure

- Input of *.frq* and *.par*
- Fitting procedure with implementation of routine `fcomp()`
- Size-composition likelihoods
- The "heart" of the age-structured model - routine `get_numbers_at_age()`

- Equilibrium yield calculations
- Hessian operations (option 7) for generating simulation inputs

4. Debugging

- Using `ddd` with the MULTIFAN-CL debug compilation executable **mfclsdbg64**, with `backtrace` to locate problem using the tools `call mp()`
- Low-level: detection of operation failures due to input errors; e.g. incompatible flags
- High-level: functioning of certain features that collapse with a problem; e.g. arithmetic exception
- Getting assistance with resolving the bug – what to prepare

5. Parked issues