**The data-limited methods toolkit (DLMtool): software for informing management of data-limited fisheries.**

Thomas R. Carruthers and Adrian R. Hordyk.

Institute for the Oceans and Fisheries, AERL, 2202 Main Mall, University of British Columbia, Vancouver, B.C., Canada, V6T 1Z4

Corresponding Author

Thomas R. Carruthers

Tel: (+1) 604 822-6903

Fax: (+1) 604 822-8934

Email: t.carruthers@oceans.ubc.ca.

**Abstract**

To do last

**Keywords**

Data-limited, data-moderate, data-poor, stock assessment, management strategy evaluation, management procedure

**Introduction**

In 2006, the U.S. Magnuson-Stevens Fishery Conservation and Management Act (MSA) was amended to require annual catch limits (ACLs) to prevent overfishing for most federally managed fish stocks. More than two-thirds of these stocks lacked conventional stock assessments at the time the law was being implemented, meaning that ACLs were adopted despite uncertainty overcurrent stock size, stock depletion, and exploitation rate at maximum sustainable yield (Newman et al. 2015). Given the MSA’s requirements to prevent overfishing and to rebuild overexploited stocks, the path to defensible ACLs for these so-called data-limited stocks was unclear. In response, various Management Procedures (MPs) were proposed to provide management advice where fishery data were limited. For example, setting the ACL according to median historical catches (MAFMC 2010), third highest catch (SAFMC 2011), Depletion Corrected Average Catch (DCAC, MacCall 2009) and Depletion-Based Stock Reduction Analysis (DB-SRA, Dick and MacCall 2011). In some cases, these were compared with stock assessments (e.g. Dick and MacCall 2011). However, for most data-limited MPs, critical questions remained unanswered, including: whether the MPs would work to achieve management objectives, whether they were appropriate for the fishery in question, whether they would perform well across different depletion levels, and how performance might be affected by data quality (e.g., sparsity, precision, bias).

There were also practical problems:

1. The MPs were disparate, and it was difficult to simultaneously apply them to the same data: they were coded separately and required custom formatting of data or, in some cases, coding from scratch.
2. The relative performance of the approaches was unknown: it was not clear which MPs to select, if any, given specific stock dynamics, fishery dynamics, and data.
3. The most effective path forward for data-collection was unclear.

In 2014, a meeting was convened between the U.S. National Marine Fisheries Service, the Natural Resources Defense Council and the University of British Columbia to address these questions and problems (Newman et al. 2014). The group proposed Management Strategy Evaluation (MSE) as the most theoretically coherent solution. MSE is a simulation approach which generates plausible population, fishery, and observation dynamics to test MPs over a projected time period accounting for feedbacks among MPs and the operating model (Butterworth and Punt 1999; Punt et al. 2016). The rationale for MSE in the data-limited context is clear: if the performance of an MP cannot be established explicitly through a stock assessment, then performance must be evaluated implicitly by simulation. For example, a size-limit or spatial closure that does not itself inform stock status may consistently achieve management objectives over a wide range of simulated conditions.

The MSE approach was used to evaluate a range of data-limited MPs (Carruthers et al. 2014) and this modelling framework was formalised in an initial version of the Data-Limited Methods Toolkit (DLMtool), which was released publicly as open-source software on the Comprehensive R Archive Network on [May 2014] (Carruthers and Hordyk 2017).

The first practical applications of DLMtool were in U.S. Federal fisheries, where it was used to evaluate MP performance for six stocks in the Caribbean (SEDAR 2016a), eight stocks in the Gulf of Mexico (SEDAR 2016b) and three stocks in the U.S. Mid-Atlantic (McNamee 2015, Miller 2015, Weidenmann 2015). Since then, DLMtool has also been applied to four California State fisheries as part of a major amendment to the State’s Master Plan for Fisheries (Hordyk et al. 2017). For a complete list of fisheries that DLMtool has been applied to, see Table 1.

This paper describes the DLMtool software, its design, features, limitations, and future extensions.

**Methods**

***Aims and Objectives***

The overarching aim of DLMtool was to use the MSE approach to support transparent and rigorous decision making in data-limited fisheries. Toward this end, the software was designed to be: (1) open-source with straightforward distribution and installation; (2) well documented with an intuitive system for accessing documentation; (3) as concise as possible; (4) easy to use, yet hard to misuse; (5) extensible, allowing for users to specify operating models and easily develop and test new MPs; and (6) sufficiently flexible in design to evolve with the emerging field of data-limited fisheries management. These objectives determined the software development model, environment, programming language and programming paradigm, all of which are described in more detail below.

***Software Development Model***

The DLMtool software was developed following an ‘Evolutionary Prototyping’ (EP) model (Nielsen 1993), whereby a robust prototype with limited functionality was released and subsequently expanded and refined to address the needs of users engaged in fishery management problems. To facilitate the EP development model, DLMtool is freely available online from a dedicated GitHub code repository (Carruthers and Hordyk 2017). The GitHub repository allows users to request software features, report bugs, and track software updates. GitHub also supports continuous integration of code from multiple developers (Duvall 2007) and features branching (Berczuk and Appleton 2003), a form of version control where additions can be coded in parallel and merged with the master branch, both of which facilitate the EP model. The principal challenge of the EP development model was adopting a language and a programming paradigm that was sufficiently powerful and flexible to allow for increasingly complex additions.

***Environment, Programming Language and Software Dependencies***

DLMtool was developed in the statistical environment R (R Core Team 2017) and is primarily coded in the R language. There are important reasons for using R in this context. It is arguably the most widely used and most flexible software for scientific and statistical analysis, and is commonly applied in quantitative fisheries science for the processing and analysis of data (Muenchen 2013). R has proven popular with the scientific community due to a large and diverse range of freely available packages and the ability to produce well-designed publication-quality figures. It follows that R provides an ideal environment for specifying operating models and then interpreting MSE outputs.

R is open-source and includes a centralized repository (CRAN) allowing for online distribution of packages such as DLMtool and its software dependencies. With an internet connection, installation of DLMtool is achieved in a single command from the R console. R packages provide users with a certain degree of confidence since they must meet CRAN policy (CRAN 2017), which includes error checking, installation size limits, compatibility among operating systems and documentation requirements. DLMtool makes full use of integrated R help and includes detailed documentation including worked examples for all objects and functions. The DLMtool software manuals and training materials are automatically rebuilt from the latest software release using R markdown (Allaire et al. 2017). DLMtool also benefits from automated unit-testing (Huizinga and Kolawa 2007) via the package ‘testthat’ (Wickham 2011) which provides quality control for every software release.

A principal limitation of computing in R is that calculations can be slow compared with languages such as C++, Python and Java. This is particularly important in this context because MSE has relatively high computational requirements. To address this, DLMtool uses compiled C++ code for computationally intensive processes (using the package Rcpp: Eddelbuettel and Francois 2011) and where necessary Template Model Builder (Kristensen et al. 2016) for non-linear estimation tasks. Additionally, DLMtool defaults to parallel processing using the package ‘parallel’ (R Core Team 2017) to distribute calculations over either multiple cores of a workstation or a larger cluster of many hundreds of virtual machines –DLMtool is compatible with online computing resources such as Amazon Web Services (2017) and Google Cloud (2017).

***Programming Paradigm and Design***

An important feature of R is that it supports Object-Oriented Programming (OOP) (Jacobsen et al. 1992; Mitchell 2003) in which data, models and results are organized in standardized objects (classes) on which standardized functions (methods) may be applied. In general OOP is desirable because it allows for the reuse of code, can impose stricter requirements of object attributes (e.g. annual catches must be a vector of non-negative real numbers), it demands careful software planning from the outset and is generally more extensible and easy to maintain that non-OOP code. DLMtool adopts the OOP paradigm, and includes seven principal object classes that describe:

(1) stock dynamics (*Stock*), such as somatic growth and natural mortality;

(2) fleet dynamics (*Fleet*), such as historical trend in exploitation rate and length vulnerability;

(3) observation processes (*Obs*), such as bias and imprecision in annual catch data;

(4) management implementation error (*Imp*), such as overages of the catch limit;

(5) complete operating models (*OM*) that combine *Stock*, *Fleet*, *Obs* and *Imp* objects

(6) MSE outputs (*MSE*), such as projections of biomass and exploitation rate for each MP and

(7) fishery data (*Data*), such as observations of annual catches and size composition data (see Figure 1 for a diagram of the relationships among these objects, Tables 2 and 3 for a summary of these objects and related methods respectively, and Appendix A for a comprehensive guide to DLMtool object classes).

The primary objective of adopting the OOP paradigm was to standardize the formatting of data and support rapid building of operating models by combining prebuilt objects. For example, evaluating the robustness of management procedures for a particular population dynamics model (*Stock* object) with respect two hypotheses regarding historical fishing (two *Fleet* objects). With this goal in mind, where possible, *Fleet*, *Stock*, *Obs* and *Imp* objects are parameterized to be compatible and easily combined in an operating model (*OM*). For example, by default the size vulnerability of fishing fleets is described in units of length at maturity, allowing the same fleet object to be paired with stock objects of varying size-at-maturity.

There are also important secondary benefits to the OOP approach in the context of MSE and the specification of operating models. By establishing classes and methods, it is easier to link the DLMtool MSE framework to other modelling software. For example, DLMtool includes functionality to automatically specify components of operating models from annotated MS Excel files and various data-rich assessments such as Stock Synthesis (Methot and Wetzel 2013), iSCAM (Martell 2017) and Stochastic Stock Reduction Analysis (Walters et al. 2006). This allows for MPs to be tested on the dynamics inferred for data-rich, peer-reviewed assessments and subsequently implemented for comparable data-limited fisheries for which assessment are not available (a form of ‘Robin-Hood’ analysis: Punt et al. 2011). Another advantage is that once a user has developed a prospective MP, it can be tested rapidly over many combinations of operating model components that are pre-specified from previous DLMtool applications (Table 1).

***Operating Model Dynamics***

At the heart of DLMtool is a spatial, age-structured operating model that simulates the interaction between a population and fishing. The model is intended to be an annual model, but the temporal resolution can be reduced providing there are commensurate adjustments to parameters controlling rates such as somatic growth and natural mortality. Appendices B-E provide a full description of DLMtool equations; all code is open-source and available online (Carruthers and Hordyk 2017).

There are two phases to the MSE simulation modelling. Phase 1 creates multiple simulations that are plausible representations of the historical fishery. This is a ‘spool—up’ phase where historical population (*Stock*) and fishing (*Fleet*) dynamics are generated that create simulated fishery conditions that are representative of today and also provide the basis for calculating historical data series (e.g. an average catch MP applied in the first projection year requires multiple years of historical catch data).

Phase 2 involves forward projection, in which each historical simulation branches into a projection of the stock and fishery for each MP, applied iteratively into the future. In each iteration of this second phase, observed data (e.g. catches, relative abundance indices) are generated that are used by MPs to make management recommendations (e.g., a catch limit), which are then implemented. In addition to the parameters of the *Stock* and *Fleet* objects, the second MSE phase also requires an observation model (*Obs*) to generate imperfect data and an implementation error model (*Imp*) to simulate the degree of adherence to the management recommendations. All future simulated conditions, such as recruitment deviations, growth and natural mortality, are identical among the candidate MPs. By ensuring a level playing field, many fewer simulations are required to reveal genuine disparities in MP performance (Carruthers et al. 2014).

***The Stock object***

The *Stock* object contains various inputs that specify an age-structured population model with time-varying parameters. The principal stock attributes are described in Table 2, a full description of the stock object is included in Table App.A.1 and all population dynamics equations are available in Appendix B. Several *Stock* attributes can have strong impacts on the absolute and relative performance of MPs such as stock depletion (current spawning biomass relative to unfished), natural mortality rate, degree of recruitment compensation (‘steepness’ that controls resilience to exploitation), size at maturity and recruitment variability (Carruthers et al. 2014; 2015, Harford and Carruthers 2017).

It can be difficult to quantify temporal changes in stock parameters even in a data-rich setting However in cases where additional robustness testing is required to further discriminate among candidate MPs, the *Stock* object can also specify the magnitude and duration of simulated changes in recruitment regimes in addition to gradients in mean recruitment, natural mortality rate, maximum length and somatic growth rate.

For most attributes of the *Stock* object, the default approach to specifying uncertainty is the simplest: a uniform random variable where the user specifies upper and lower bounds from which a value is drawn for each simulation (see ‘The OM object’ below for details on custom parameter distributions).

***The Fleet object***

The *Fleet* object contains attributes that specify both historical exploitation in the MSE ‘spool-up’ phase and future exploitation in the MSE projection phase. Among the most important attributes are trend and pattern in historical exploitation rate, the size vulnerability to fishing, and retention rate of fish at size. The principal *Fleet* attributes are described in Table 2, a full description of the *Fleet* object is included in Table App.A.2 and all fishing dynamics equations are available in Appendix C.

***The Obs object***

DLMtool currently features more than 90 data-limited MPs that require a wide range of inputs including time-series data (e.g., annual catches, relative abundance indices), point estimates (e.g., current absolute abundance) and derived quantities (e.g., natural mortality rate, growth rate, target catch level). To test these MPs using MSE, the observation model must simulate these inputs subject to typical observation processes. A full description of the *Obs* object is included in Table App.A.3 and all observation model equations are available in Appendix D.

For derived quantities such as natural mortality rate and point estimates such as absolute current abundance, by default DLMtool samples biases for each simulation from a log-normal distribution of mean 1 (unbiased on average) with a CV that is user specified (i.e. Carruthers et al. 2014). Again users can specify custom parameter values that, for example make all observations of catches negatively biased. For time series data such as annual catches, DLMtool includes independent parameters for mean bias and observation error (e.g., allowing for time series of catches that are biased and precise or unbiased and imprecise).

***The Imp object***

In many fisheries there can be large discrepancies between management recommendations and fishing actions (‘implementation error’, for example, TACs may be exceeded) which can strongly determine the performance of MPs under MSE testing (Dichmont et al. 2008; Fulton et al. 2011). The *Imp* object contains attributes that control implementation error for the three principal types of management recommendation: TACs (output control MPs of class *Output*), TAEs (input control MPs of class *Input*) and size-limits (MPs also of class *Input*). Similarly to bias and imprecision of time-series data, fishing actions can be consistently above or below recommendations with a degree of annual variability around this error (for example taking an average of 90% of the TAC but with annual catches in the range of 70% and 110% of those recommended). A full description of the *Imp* object is included in Table App.A.4 and all implementation model equations are available in Appendix E.

***The OM object***

The operating model object (class *OM*) combines all of the *Stock*, *Fleet*, *Obs* and *Imp* attributes in addition to several others that control the MSE projections (Table App.A.5). The central motivation behind the combined *OM* object is that it should include all specifications required to reproduce the MSE and its results. It therefore includes additional attributes controlling the duration of the MSE projection phase, the length of the interval between recalculation of management recommendations by the MP and the number of simulations. An important additional attribute is the random seed which ensures exact reproducibility of results thereby allowing users to test new MPs on the same simulations.

During the development of the DLMtool software its user base and fishery applications have expanded. Throughout there have been requests for greater control of operating model specification such as new distributions for the sampling of OM parameters and functionality to specify cross-correlation among parameters. To address these requests, the OM object contains a custom parameter attribute that allows users to provide samples of any operating model parameter originating from any distribution or analysis they choose which can include cross correlations among sampled values. For example these might be posterior samples of the maximum growth rate and maximum length parameters of a fitted growth curve. The custom parameters attribute also allows for sampling of time series such as historical effort patterns by year and year-specific size vulnerability. Essentially users have the option to ‘hijack’ the inner variables of the operating model. It is using this functionality that DLMtool can be specified to match exactly the stock and fleet dynamics estimated by an age-structured stock assessment model such as Stock Synthesis. By allowing for user-sampled parameters and variables, flexibility is maximized without requiring numerous additional object attributes to control the type of sampling distribution and parameter correlations.

***The MSE object***

For any *OM* object an MSE may be run in DLMtool using the method *runMSE* to produce an object of class *MSE*. This contains information about what occurred in the historical and projection phases of the MSE for every simulation, including catches taken, stock biomass trajectories, fishing mortality rate trajectories, calculated reference points and sampled parameters (Table App.A.6). The *MSE* object can be used by a large range of methods to, for example, quantify MP performance, characterize trade-offs, select MPs by satisficing, check for simulation convergence (whether there were enough simulations to have confidence over results) and quantify the value of information (which observation processes most affected MP performance). A key requirement of the *MSE* object is that it stores all of the MSE data and variables required to calculate performance for a wide range of management performance metrics.

***The Data object***

At the heart of DLMtool is a common standard for organizing fishery data: the *Data* object. This object class addresses several of the fundamental questions and problems outlined in the Introduction. For example, if data-limited methods can be coded to operate on a common *Data* object then once real data are organized in this object, all MPs can be applied simultaneously. Similarly, simulated data can also be organized in the *Data* object allowing for simultaneous simulation testing of multiple MPs (e.g. MSE). Consistent with OOP, DLMtool treats MPs as methods that are applied to objects of class *Data*.

The *Data* object contains various slots that contain the common fishery data types and parameters including time-series of catches, relative abundance indices, fishery independent surveys, age composition data, size composition data, mean length data, natural mortality rate estimates, current estimates of stock depletion and growth parameters (Table App.A.7). The *Data* object allows for a sufficiently broad range of data types to accommodate full range of contemporary data-limited MPs and simple data-rich MPs (for performance comparison).

***Recommended Process for Using DLMtool***

DLMtool was designed to be applied according to the following guidelines for MSE best practice identified by Punt et al. (2016, Table 1 of that paper) (Figure 1):

1. Through a consultative process, including decision makers and stakeholders, management objectives are identified conceptually (e.g., avoid overfishing) and then formalized into quantitative performance statistics (e.g., less than a 20% probability of annual exploitation rates greater than those associated with maximum sustainable yield over a 20-year projection).
2. The available data and information are synthesized to identify a broad range of uncertainties over which an MP should be robust (e.g. stock depletion, over reporting of catches, increasing fishing pressure, dead discarding, increasing natural mortality rate). Based on these uncertainties operating models are specified representing biological (*Stock*) and fishery dynamics (*Fleet*), observation processes (*Obs*) and adherence to management recommendations (*Imp*)that include suitable imprecision in parameter values and structural uncertainties.
3. Candidate MPs are identified that are feasible given management options available (e.g. the management system requires catch recommendations) and the types of data available now and in the future.
4. Candidate management procedures are applied to the simulated fishery as represented by the operating model as part of the MSE (the *runMSE* method).
5. The results of the MSE analysis are presented, including trade-offs and robustness testing (plotting methods for the *MSE* object). Select a management procedure for example by satisficing (acceptance of MPs that meet minimum performance requirements) (Miller and Shelton 2010). Apply the management procedure (*Output* or *Input* method) to real fishery data in an object of class *Data* to calculate a management recommendation.
6. Formal protocols for MP review are adopted, including the circumstances where the OM (and MP) should be revisited (e.g. observed data that are atypical of simulated data).

***Management procedures***

The primary purpose of DLMtool is to provide an extensible software package for testing user-designed management procedures. The package includes a number of pre-specified data-limited MPs that use various types of data to make various types of management recommendations including Total Annual Catches (TACs), Total Annual Effort (TAE), size limits and time-area closures. The details of management procedures are not the subject of this paper, for a description and performance evaluation of DLMtool MPs see Carruthers et al. (2014; 2015) and Harford and Carruthers (2017). Additionally Appendix E includes an overview of the various DLMtool MPs that are currently available.

***Performance metrics***

Similarly to management procedures the purpose of DLMtool is to provide a flexible basis for calculating any number of performance metrics that are appropriate to each fishery management setting. The purpose of this paper is not to discuss the diversity of management objectives (e.g. rebuild a stock), performance measures (e.g. biomass relative to MSY levels) and statistics (e.g. probability of biomass above BMSY levels in 50 years). However to demonstrate the breath of MSE outputs that are available in DLMtool we include examples from previous DLMtool applications (Figure 2). <some brief detail on the nature of these figures>

***Features***

A summary of DLMtool features are described in Table 4.

**Discussion**

Through various applications of DLMtool and wider experiences applying MSE, it is clear that the behaviour of MPs within the feedback-control loop of an MSE simulation is complex, non-linear, and unpredictable. MP selection can be strongly dependent on (or not at all dependent on) a number of factors, including:(1) management objectives, (2) the status of the stock, (3) population life-history characteristics, (4) fishery dynamics, (5) data quality, (6) implementation error and (7) feasibility of various types of management control (data availability or management constraints, e.g., input and output controls). For example, fisheries subject to rebuilding objectives generally require much more conservative MPs than stocks at healthy biomass levels (Benson et al. 2017, Carruthers and Agnew 2016). Similarly obviously, in most cases, stocks can be more heavily exploited by less conservative MPs if they have higher recruitment compensation (Carruthers et al. 2015). For short-lived stocks, where biomass trends are driven by variable recruitment, MPs that rely more heavily on short-term data may be preferable to stock assessment approaches that rely on long-term time series data (e.g., Butterfish; Carruthers et al. 2014). A fishery that only exploits mature individuals with minimal discard mortality is likely to favour a wide range of more aggressive MPs despite large uncertainty over all other operating model factors (e.g., the dive fishery for California red sea urchin; Hordyk et al. 2017).

The complex and unpredictable nature of MP selection calls into question subjective data-limited approaches that simply consider equally weighted additive or multiplicative scoring systems as an approach for informing management (e.g., Productivity-Susceptibility Analysis Hobday et al. 2011, Patrick et al. 2010). These systems may be convenient, but their theoretical basis is unclear. Where possible, data-limited management schemes should be objective and testable, and be able to include wide ranges of uncertainty where quantities (e.g., stock status) are poorly known. These findings confirm that the recommendations of the initial DLMtool meeting to pursue MSE for testing data-limited MPs was prescient (Newman et al. 2014). While traditionally a data-rich concept, MSE may be the most coherent approach for selecting MPs in data-limited settings where uncertainty is high regarding many stock and fishery attributes, including stock status.

An interesting finding arising from DLMtool development is the large potential benefit of establishing a standard for organizing fishery data. DLMtool requires a *Data* object class so that multiple MPs can be applied simultaneously to the same simulated data or real fishery data. Clearly this is easier to achieve in data-limited fisheries where fewer types of data are generally available at a coarser spatial-temporal resolution. Nonetheless, establishing a fishery data standard (or database model) should be a priority for global fisheries management. If methods of fishery data processing and analysis were compatible with such a standard, scientists and managers could easily benefit from the endeavours of the wider fisheries science community.

DLMtool has a number of important limitations. The software was designed in the context of the US ACL requirement and is focused on testing MPs that provide annual advice. Generally, the annual operating model equations are a poor approximation of stock dynamics for species that live less than four years (instantaneous natural mortality rates higher than around 0.7). In many cases these fisheries have important seasonal dynamics and rely on MPs that provide advice in-season, MPs that are not currently supported by the annual operating model of DLMtool.

With the exception of the-stock recruitment relationship, the DLMtool operating models do not account for density-dependent population dynamics (for example relating to growth, maturity and natural mortality rate) or fishing dynamics (e.g. fishing efficiency). This is an important limitation, and a key area for future development because density dependent phenomena have been attributed to the decline of various stocks (Kuparinen et al. 2014). Another principal limitation of DLMtool is that it does not currently account for spatial life-history dynamics such as ontogenetic habitat shifts (change in spatial distribution with age). This may be an important omission since this is a common characteristic of reef fishes such as groupers and snappers which may represent a large fraction of global data-limited fish species.

Other software packages exist for the MSE testing of management procedures, the most notable is Fisheries Library in R (FLR; Kell et al. 2007) which is an extensive open resource for a large range of stock assessment and MSE tasks. The first DLMtool concept was a ‘wrapper’ that would use FLR MSE functions as the ‘engine’. However, at that time the decision was made to write custom operating model code from scratch for a dedicated data-limited MSE package. The most important reason for this was a lack of extensibility: it was not straightforward to add MPs in FLR and even more difficult add new observation models to allow for new types of MPs. An additional concern was that FLR is not available on CRAN and would leave DLMtool software dependant on other software that did not adhere to a rigid standard (and would also require an additional installation step). Nonetheless, a priority for future development is establishing links with other R software such as FLR to share code, for example, by making data-rich MPs from FLR compatible with DLMtool.

The DLMtool software is subject to ongoing development. In partnership with the Canadian Department of Fisheries and Oceans (DFO), a principal area for investigation is the use of operating models to classify stocks for management. For example, the National Marine Fishery Service (NMFS) tiers stocks according on the availability of data to support various types of stock assessment or TAC setting (REF). However data availability is just one of the 7 factors determining MP selection that were described above. Instead, MSE may be used to distinguish between data-limited and *information*-limited fisheries. In this case *information* may be defined as the ability to meet management objectives considerate of the 7 factors listed above. For example, the B.C. dungeness crab fishery may meet DFO management objectives relating to sustainability given that current management measures only allow for retention of male crabs that have reproduced at least once (DFO 2017). Such a fishery may be considered data-limited by NMFS standards, but information-sufficient according to a wide range of uncertainties and the performance objectives of DFO. To this end, DLMtool is being rapidly expanded to include a greater number of data-rich MPs to better evaluate whether gathering data to support an assessment approach is likely to be cost-effective. Also under development is a multispecies extension that allows for consideration of more complex spatial distribution of stocks in addition to parsimonious ecosystem dynamics models (e.g., models of intermediate complexity; Plagányi et al. 2012).

**Acknowledgements**

Many thanks for the ongoing support of the Natural Resources Defense Council, and in particular DLMtool team members David Newman and Lisa Suatoni who have provided input at every stage. Special thanks to the original members of the 2014 NMFS-NRDC-UBC data-limited methods meeting who provided the inspiration and impetus for this research: Jerry Ault, Meaghan Bryan, Jon Brodziak, Jason Cope, E.J. Dick, Nick Farmer, John Hoenig, Alec MacCall, Clay Porch, Jim Thorson and Carl Walters. DLMtool’s development has been funded by the Gordon and Betty Moore Foundation, the Packard Foundation, U.S. National Oceanic and Atmospheric Administration, Fisheries and Oceans Canada, the Walton Foundation, Resources Legacy Fund, the Natural Resources Defense Council, the United Nations Food and Agricultural Organization and the California Department of Fish and Wildlife. Special thanks are due to the ongoing support of DLMtool users that provide critical feedback that continues to improve the software, in particular: Helena Geromont, Bill Harford, Skyler Sagarese, John Wiedenmann, Sean Anderson, Robyn Forrest, Chris Legault, and Quang Huynh.

**References**

Allaire, J.J., Horner, J., Marti, V., Porte, N. 2017. Markdown: 'Markdown' Rendering for R. R package version 0.8. Available at: <https://CRAN.R-project.org/package=markdown> [accessed September 2017].

Amazon Web Services. 2017. Available at: <https://aws.amazon.com/> [accessed September 2017].

Berczuk, S., Appleton, B. 2003. Software Configuration Management Patterns: Effective Teamwork, Practical Integration. Addison-Wesley. ISBN 0-20174117-2.

Butterworth, D.S., Punt, A.E., 1999. Experiences in the evaluation and implementation of management procedures. ICES J. Mar. Sci. J. Cons. 56,985–998, <http://dx.doi.org/10.1006/jmsc.1999.0532>.

Carruthers, T.R. 2017a. Evaluation of BC Rougheye Rockfish (*Sebastes aleutianus*) and Canary Rockfish (*Sebastes pinniger*) in reference to COSEWIC criteria. Technical report for Fisheries and Oceans Canada. Available at: <http://www.datalimitedtoolkit.org/wp-content/uploads/2017/06/COSEWIC_Evaluation_Canary_and_Rougheye_Rockfish_Carruthers.pdf> [accessed September 2017]

Carruthers, T.R. 2017b. Customizing DLMtool for Canadian Fisheries. Technical report for Fisheries and Oceans Canada. Available at: <http://www.datalimitedtoolkit.org/wp-content/uploads/2017/09/Customizing-DLMtool-for-Canadian-Fisheries-WEB.pdf> [accessed September 2017]

Carruthers, T.R., Hordyk, A.R., 2017. DLMtool: data-limited methods toolkit. Available at: <https://cran.r-project.org/web/packages/DLMtool/index.html>.

Carruthers, T.R., Punt, A.E., Walters, C.J., MacCall, A., McAllister, M.K., Dick, E.J.,Cope, J., 2014. Evaluating methods for setting catch limits in data-limited fisheries. Fish. Res. 153, 48–68, <http://dx.doi.org/10.1016/j.fishres.2013.12.014>.

Carruthers, T.R., Kell, L.T., Butterworth, D.D.S., Maunder, M.N., Geromont, H.F.,Walters, C., McAllister, M.K., Hillary, R., Levontin, P., Kitakado, T., Davies, C.R., 2015. Performance review of simple management procedures. ICES J. Mar. Sci.J. Cons., <http://dx.doi.org/10.1093/icesjms/fsv212>.

CRAN. 2017. Comprehensive R Archive Network Repository Policy. Available at: <https://cran.r-project.org/web/packages/policies.html> [accessed September 2017]

DFO. 2017. Dungeness crab: habitat, species distribution, stock assessment and life-cycle. Canadian Department of Fisheries and Oceans. Available at: <http://www.dfo-mpo.gc.ca/species-especes/profiles-profils/dungeness-crab-crabe-dormeur-eng.html> [accessed September 2017]

Dichmont, C.M., Deng, A., Punt, A.E., Ellis, N., Venables, W. N., Kompas, T., Ye, Y., Zhou, S., Bishop, J. 2008. Beyond biological performance measures in Management Strategy Evaluation: bringing in economics and the effects of trawling on the benthos. Fish. Res. 94: 238–250.

Dick, E.J., MacCall, A.D., 2011. Depletion-based stock reduction analysis: a catch-based method for determining sustainable yields for data-poor fish stocks. Fish. Res. 110, 331–341.

Duvall, P. M. 2007. Continuous Integration. Improving Software Quality and Reducing Risk. Addison-Wesley. ISBN 0-321-33638-0.

Eddelbuettel, D., Francois, R. 2011. Rcpp: Seamless R and C++ Integration. Journal of Statistical Software, 40(8), 1-18.

Edwards, A.M., Haigh, R., Tallman, R., Swain, D., Carruthers, T.R., Clearly, J.S., Stenson, G., Valcroze, T. 2017. Proceedings of the Technical Expertise in Stock Assessment (TESA) National Workshop on ‘Incorporating an ecosystem approach into single-species stock assessments’. Nanaimo, British Columbia. Available at: <http://publications.gc.ca/collections/collection_2017/mpo-dfo/Fs97-6-3213-eng.pdf> [accessed August 2017]

FAO. 2017. Training on data-limited stock assessment methods for Tuna species. Common Oceans - A partnership for sustainability. The Areas Beyond National Jurisdiction (ABNJ) project. United Nations Food and Agricultural Organization. Available at: <http://www.fao.org/in-action/commonoceans/news/detail-events/en/c/887806/> [accessed September 2017]

Fulton, E.A., Smith, A.D.M., Smith, D.C., van Putten, I.E. 2011. Human behavior: the key source of uncertainty in fisheries management. Fish. Fish. 12: 2–17.

Google Cloud. 2017. Available at: <https://cloud.google.com/products/compute/> [accessed September 2017]

Harford, W.J., Carruthers, T.R. 2017. Interim and long-term performance of static and adaptive management procedures. Fish. Res. 190: 84-94.

Hobday, A.J., Smith, A.D.M., Stobutzki, I.C., et al. 2011. Ecological risk assessment for the effects of fishing. Fish. Res. 108, 372–384.

Hordyk, A., Newman, D., Carruthers, T., Suatoni, L. 2017. Applying Management Strategy Evaluation to California Fisheries: Case Studies and Recommendations. Available at: <http://www.datalimitedtoolkit.org/wp-content/uploads/2017/07/Applying-MSE-to-CA-Fisheries-Case-Studies-Recommendations.pdf> [accessed September 2017]

Huizinga, D., Kolawa, A. 2007. Automated defect prevention: best practices in software management. Wiley.

Jacobsen, I., Christerson, M., Jonsson, P., Overgaard, G. 1992. Object Oriented Software Engineering. Addison-Wesley ACM Press. pp. 43–69. ISBN 0-201-54435-0.

Kell, L.T., Mosqueira, I., Grosjean, P., Fromentin, J.M., Garcia, D., Hillary, R., Jardim, E., Mardle, S., Pastoors, M.A., Poos, J.J., Scott, F. 2007. FLR: an open-source framework for the evaluation and development of management strategies. ICES Journal of Marine Science, 64(4): 640-646.

Kristensen, K., Nielsen, A., Berg, C.W., Skaug, H., Bell, B.M. 2016. TMB: Automatic Differentiation and Laplace Approximation. Journal of Statistical Software, 70(5), 1-21. doi:10.18637/jss.v070.i05

Kuparinen, A., Keith, D.M., Hutchings, J.A. 2014. Allee effect and the uncertainty of population recovery. Cons. Biol. 28(3): 790-798. doi:10.1111/cobi.12216

MacCall, A.D., 2009. Depletion-corrected average catch: a simple formula for estimating sustainable yields in data-poor situations. ICES J. Mar. Sci. 66, 2267–2271.

MAFMC, 2010. Amendment 11 to the Atlantic mackerel, squid and butterfish fishery management plan. Mid Atlantic Fishery Management Council. US National Marine Fisheries Service. http://www.mafmc.org/meeting materials/2010/February/Tab05\_Squid, Mackerel, Butterfish Amendment 11 Public HearingDocument.pdf.

Martell, S. 2017. Integrated Statistical Catch-at-Age Model. The iSCAM project. Available at <https://github.com/smartell/iSCAM> [accessed August 2017]

McNamee, J., Fay, G., Cadrin, S. 2015. Data Limited Techniques for Tier 4 Stocks: An Alternative Approach to Setting Harvest Control Rules Using Closed Loop Simulations for Management Strategy Evaluation. U.S. Mid-Atlantic Fishery Management Council. <http://sedarweb.org/docs/wsupp/S46_RD_08_Black%20Sea%20Bass_Mid-AtlSSC.pdf>.

Methot, R.D. and Wetzel, C.R. 2013. Stock synthesis: a biological and statistical framework for fish stock assessment and fishery management. Fish. Res. 142: 86-99.

Miller, D.C., Shelton, P.A. 2010 “Satisficing” and trade-offs: evaluating rebuilding strategies for Greenland halibut off the east coast of Canada. ICES J. Mar. Sci. 67: 1896–1902.

Miller, T. 2015. Blueline Tilefish Working Group Report. U.S. Mid-Atlantic Fishery Management Council. http://static1.squarespace.com/static/511cdc7fe4b00307a2628ac6/t/56e046a37c65e4cd0ba8f8ce/1457538725993/BLT+Subcommittee+Report+20160309.pdf.

Mitchell, J.C. 2003. Concepts in programming languages, Cambridge University Press. ISBN 0-521-78098-5, p.278.

Muenchen, R.A. 2013. The Popularity of Data Analysis Software, Technical report, http://r4stats.com/articles/popularity/

Newman, D., Berkson, J., Suatoni, L., 2015. Current methods for setting catch limits for data-limited fish stocks in the United States. Fish. Res. 164, 86–93, <http://dx.doi.org/10.1016/j.fishres.2014.10.018>.

Newman, D., Carruthers, T.R., MacCall, A., Porch, C.E., Suatoni, L., 2014. Improving the science and management of data-limited fisheries: An evaluation ofcurrent methods and recommended approaches. Natural Resources Defense Council, NRDC Report R:14-09-B, New York, NY. <https://www.nrdc.org/sites/default/files/improving-data-limited-fisheries-report.pdf>.

Nielsen, J. 1993. Usability Engineering (2nd ed.). Boston: AP Professional. ISBN 0-12-518405-0.

Patrick, W.S., Spencer, P., Link, J., et al. 2010. Using productivity and susceptibility indices to assess the vulnerability of United States fish stocks to overfishing. Fish. Bull. 108, 305–322.

Plagányi, E.E., Punt, A.E., Hillary, R., Morello, E.B., Thebaud, O., Hutton, T., Pillans, R.D., Thorson, J.T., Fulton, E.A., Smith, A.D.M., Smith, F., Bayliss, P., Haywood, M., Lyne, V., Rothlisberg, P.C. 2012. Multispecies fisheries management and conservation: tactical applications using models of intermediate complexity. Fish. Fish. 2–22. 10.1111/j.1467-2979.2012.00488.x.

Punt, A.E., Butterworth, D.S., de Moor, C.L., De Oliveira, J.A.A., Haddon, M., 2016. Management strategy evaluation: best practices. Fish Fish. 17, 303–334, <http://dx.doi.org/10.1111/faf.12104>.

Punt, A.E., Smith, D.C, Smith, A.D.M. 2011. Among-stock comparisons for improving stock assessments of data-poor stocks: the "Robin Hood" approach. ICES. J. Mar. Sci. 68 (5): 972-981.

R Core Team 2017. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

SAFMC, 2011. Comprehensive Annual Catch Limit (ACL) Amendment forthe South Atlantic region. South Atlantic Fishery Management Council, North Charleston, South Carolina, <http://sero.nmfs.noaa.gov/sf/pdfs/Comp%20ACL%20Am%20101411%20FINAL.pdf>

SEDAR, 2016a. Stock Assessment Report: U.S. Caribbean Data-Limited Species. SouthEast Data, Assessment, and Review (SEDAR) 46. <http://sedarweb.org/sedar-46>.

SEDAR, 2016b. Stock Assessment Report: Gulf of Mexico Data-Limited Species. SouthEast Data, Assessment, and Review (SEDAR) 46. http://sedarweb.org/sedar-49.

Walters, C.J., Martell, S.J.D., Korman, J. 2006. A stochastic approach to stock reduction analysis. Can. J. Fish. Aqua. Sci. 63: 212-223.

Wickham, H. 2011. testthat: Get Started with Testing. The R Journal, vol. 3, no. 1, pp. 5—10.

Wiedenmann, J. 2015. Application of Data-Poor Harvest Control Rules to Atlantic Mackerel. U.S. Mid-Atlantic Fishery Management Council. http://sedarweb.org/docs/wsupp/S46\_RD\_09\_Mackerel\_ABC\_reportOpt.pdf.

**Tables**

Table 1. DLMtool applications.



Table 2. DLMtool object classes



Table 3. DLMtool methods



Table 4. DLMtool features

|  |
| --- |
| **Primary Features** |
| * Specify operating models for a wide range of fishery types * Quantify the performance of a large range of MPs * Reveal management performance trade-offs among MPs * Select robust MPs for data limited fisheries * Prioritize data collection * Identify critical operating model uncertainties * Provide management recommendations from a wide range of MPs * Determine what MPs can be applied for a given set of data * Identify what additional data are required to use a particular MP * Rapidly develop and test new MPs * Construct operating models from existing Stock, Fleet, Obs and Imp objects * Describe operating models and visualize results * Establish a suitable update interval for management advice * Evaluate the relative efficacy of models of management such as catch-limits, effort control, size limits and time-area closures * Allow for custom control of parameter distributions and parameter cross correlation for all operating model parameters * Use parallel processing and cluster computing to speed up computationally intensive tasks |
| **Secondary Features** |
| * Automatically specify operating models from existing assessments * Diagnose MSE convergence (have sufficient simulations been carried out to have confidence over conclusions?) * Investigate the impact of time-varying stock parameters on MP performance * Evaluate the relative additional value of data-rich MPs * Determine sensitivity of real management recommendations to uncertainties in real data and MP inputs * Identify suitable indicators of stock status * Prioritize stocks for management based on status-quo risks * Investigate the impacts of discarding and catch over- and under-ages on MP efficacy * Frame performance in terms of plausible outcomes (what are realistic best and worst case scenarios?) * Evaluate feasibility of management objectives (can they be achieved given zero exploitation or ‘optimal’ exploitation) * Calculates common reference points for calculation of performance metrics such as those relating to maximum sustainable yield and minimum viable population size. * Characterize the theoretical performance of MPs subject to perfect information. |
|  |

**Figures**

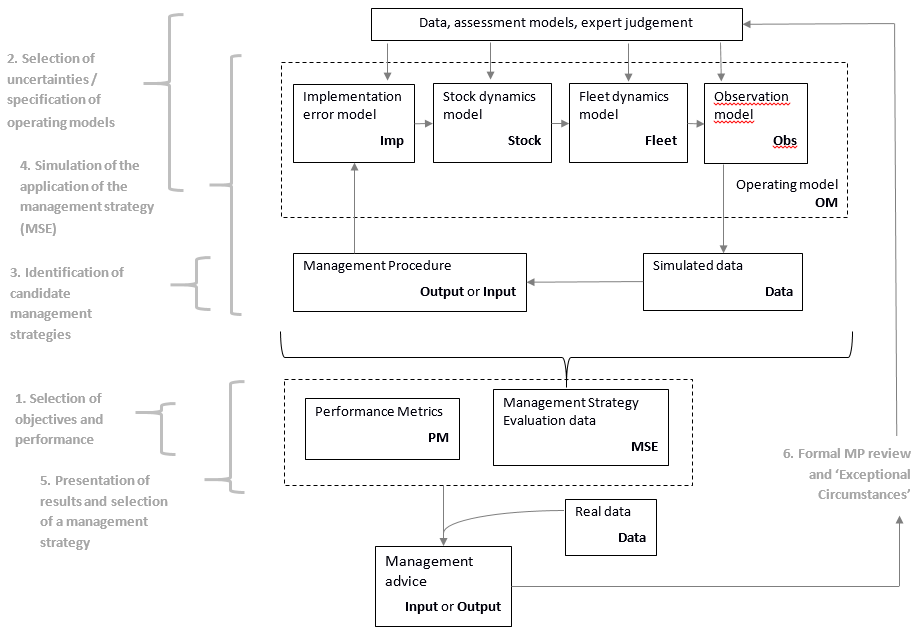
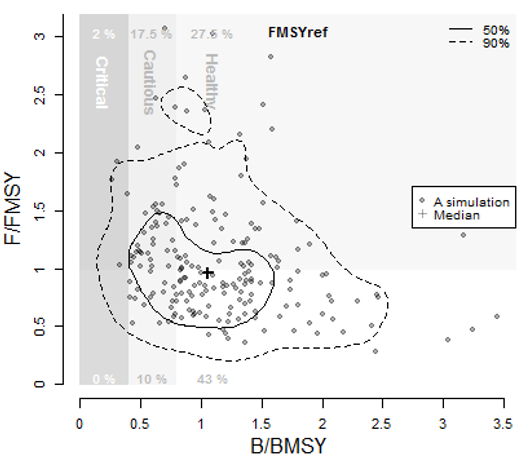
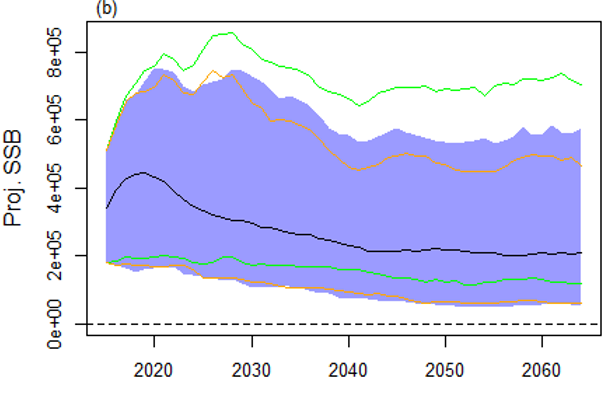


Figure 1. DLMtool design, workflow and MSE process. Bold text represents DLMtool object classes (e.g. Imp, Data). Bold grey text represents the MSE ‘best practice’ guidelines described by Punt et al. (2016) that are numbered in the order they are presented in that paper.

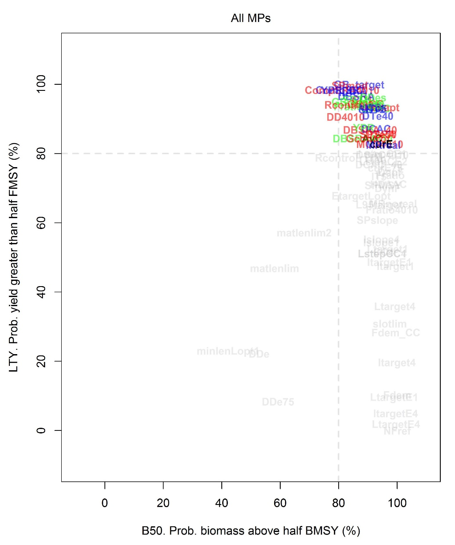
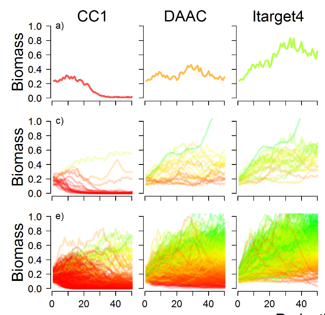
 

Figure 2. Four examples of DLMtool performance outputs.

**Appendix A: DLMtool object classes**

Table App.A.1. The attributes of the Stock object that controls simulated population dynamics.



Table App.A.2. The attributes of the Fleet object that controls simulated fishing dynamics.



Table App.A.3. The attributes of the Obs object that controls the observation model and the generation of simulated data (e.g. bias and imprecision in observations of annual catches).



Table App.A.4. The attributes of the Imp object that controls simulated implementation error (e.g. catch limit overages).



Table App.A.5. The attributes of the OM object that controls the OM simulation 

Table App.A.6. The attributes of the *MSE* object that contains the results and sampled parameters of the MSE analysis. 

Table App.A.7. The attributes of the Data object that contains real or simulated fishery data. When data are simulated, various vectors and arrays have the length nsim (the number of simulations). These are of length 1 (nsim=1) when real fishery data are contained in the data object. Note that any slot of the data object can be unspecified by the user (an NA value).



Table App.A.7. cont.



**Appendix B: Population dynamics model**

**B.1. Conventions**

A wide range of parameters and variables are allowed to vary among simulations (e.g., M, growth rate, recruitment compensation). All parameters are random variables that are sampled across simulations are denoted with a tilde (e.g., ). Hence, each parameter or variable denoted with a tilde represents a sample from a distribution. For example, the symbol  represents which is the sample of the parameter corresponding with the i­­­th simulation, drawn from a distribution function f(), from the operating model parameters θ. Although these parameter values can be drawn from any distribution, by default these are drawn from uniform distributions (Table App.A.1-4).

In some cases parameters and variables are derived by numerical optimization. The notation *opt* is used to represent optimizing a parameter *p*, to obtain the objective *Δ* with respect to existing parameters and variables *θ*: *p* = *opt*(*Δ*| *θ*). For example represents optimization of the catchability *q* in order to obtain depletion given fishing effort *E*, natural mortality rate and unfished recruitment (where , and are all user defined and drawn from distributions).

Management strategy evaluation has two phases: (1) an historical ‘spool-up’ phase where data are generated and dynamics produced that create current conditions (e.g. industrial fishing from 1955 to 2017), and (2) a projection phase where MPs are tested in closed-loop simulation (e.g. a 30 year projection from 2018 to 2047). The last historical year (2017) is referred to as the ‘current year’ *c*, in these appendices.

**B.2. Population dynamics**

An age-structured, spatial model was used to simulate population and fishery dynamics (e.g. Quinn and Deriso 1999, Chapter 8). A summary of user-specified parameters controlling stock dynamics can be found in Tables App.B.1. and App.B.2.

Numbers of individuals in consecutive years are calculated from those from the previous year and age class *a*, after movement within that year (there is no ‘plus group’ and individuals greater than maximum model age *na* are assumed to die):

1)

where the are numbers subject to the total instantaneous mortality rate *Z* and movement rate *Ψ* from area *k* to area *r*:

2)

where *Ψ* is a Markov matrix (for any area ‘from area’ *k*, ) of movement probabilities and total mortality *Z*. The movement matrix *Ψ* is calculated by numerical optimization

3)

according to the fraction of the unfished vulnerable biomass in each area *θd* and the mean probability that individuals remain in the same area between time steps *θv* (the mean value of the positive diagonal rates: where *k* = *r*). This parameterization is intended to be intuitive in a data-limited setting where only broad characterizations of stock viscosity and distribution may be available.

Total mortality rate *Z* is the sum of natural mortality (*M*) and fishing mortality (*F*) rates:

4)

Fishing mortality rate (*F*) calculations are included in Appendix C below. Natural mortality rate can vary among ages and years and is calculated:

5)

where is the mean natural mortality rate of mature individuals in the current year and ages, is the percentage annual increase in *M* over years, *ny* is the number of historical years, is a parameter controlling the exponential decline in *M*-at-age according to weight-at-age *Wa* relative to maximum weight , is an annual log-normal deviation and ΔM is an adjustment factor ensuring that the Lorenzen decline in *M* relative to weight is correctly referenced to the mean mortality rate of mature individuals:

6)

Where *na* is the number of modelled age classes, *nint()* is the nearest integer function and *ma* is the age corresponding with 50% maturity.

The parameterization of *M* expressed in Eqn. 5 was chosen deliberately to allow users the flexibility to include any level of detail in their specification of *M*. Users can only specify mean *M* of mature fish or include any or all of the additional features including slope (, weight dependency (or interannual error ( where appropriate. Where *Stock* object attributes are not specified (e.g. the slope parameter), these features are disabled (e.g. non-time varying *M* is prescribed). By including a matrix of *M* in the population dynamics model that has dimensions for time and age, users can specify their own custom *M* matrices using the custom parameters feature (the *cpars* attribute, *OM@cpars*).

By default, DLMtool models growth according to von Bertalanffy model:

7)

where *κy* is the growth rate, *Ly,∞* is the maximum length and *t0* is the theoretical age where length is zero. The growth rate and maximum length parameters have year subscripts because, similarly to *M*, these can vary according to user-specified slope parameters:

8)

9)

The maturity of individuals is specified by length according to a logistic model with parameters for the length at 50% maturity () and the slope of the logistic curve (. In order to maintain a constant length at 50% maturity with changing growth, the ratio of length at 50% maturity to maximum length for the current year is maintained over all years of a simulation:

10)

11)

To ensure that maturity is more easily specified by the user there are two inputs to the *Stock* object: and the length increment to 95% maturity which then requires optimization to find :

12)

Maturity at age calculated using the length at age *Ly,a*:

13)

where is the age at 50% vulnerability given by the inverse von-Bertalanffy growth curve:

14)

and the slope of the age-at-maturity curve is calculated by numerical optimization:

15)

The numbers of individuals recruited to the first age group *Ny,a=1,r* in each year *y*, and area *r* is calculated using a Beverton-Holt stock-recruitment relationship with log-normal recruitment deviations (Table App. B.2):

16)

where the spawning biomass in a given year and area is the summation over ages of the maturity at age *m*, weight at age *W*, and numbers at age *N*:

17)

and the parameters and are given by:

18)

19)

The steepness (recruitment compensation) parameter , is user-defined. Unfished spawning biomass is calculated from unfished recruitment and survival to age *a*:

20)

Weight-at-age *Wa*, is assumed to be related to length by:

21)

As an alternative to the Beverton-Holt stock-recruitment relationship, users can specify the Ricker model using the *Stock* attribute ‘SRrel’:

22)

Log-normal recruitment deviations include both error and temporal autocorrelation. A series of initial error terms were sampled from a log-normal distribution with mean 1 and standard deviation :

23)

To these initial error terms, temporal autocorrelation was added:

24)

The final recruitment deviations can also include cyclical recruitment changes over time according to a sin function:

25)

where controls the wavelength of the transformed sin wave (2π is the wave length of an untransformed sin wave), alters the amplitude of the wave (which has a maximum value of 1 and is symmetrical around 1) and the term is a random uniform variable between 0 and 1, which is used to randomize the position of the sinusoidal recruitment fluctuations among simulations. When is set to zero or not specified . The principal purpose of including recruitment regime shifts is to add an additional test of robustness in case there is further need to discriminate among MPs that otherwise perform similarly.

To initialize numbers at age (first historical year), the regional numbers at age were calculated according to unfished recruitment , log-normal recruitment deviations the equilibrium fraction of the stock in each area *r* under unfished conditions *dr*

26)

where *dr* is found by iteratively multiplying an initial guess for the vector *dr* by the movement matrix until *dr* has converged meeting the condition:

27) *dr = dr*

Table App.B.1. Sampled parameters controlling stock dynamics (*Stock* object)

|  |  |  |  |
| --- | --- | --- | --- |
| **Symbol** | **Controls:** | **Description** | **Object@Attribute** |
|  | Stock viscosity | Mean probability of individuals staying in the same area between years | Stock@Prob\_staying |
|  | Spatial distribution | Unfished fraction of individuals in each area | Stock@Frac\_area\_1 (2 area model) |
|  | Area size | Geographical size of each area | Stock@Size\_area\_1  (2 area model) |
|  | Natural mortality | Mean instantaneous natural mortality rate of mature individuals in the current year | Stock@M |
|  | Post release mortality rate | Fraction of released fish that die (size independent) | Stock@Fdisc |
|  | M gradient | Per cent annual change in natural mortality rate | Stock@Mgrad |
|  | Declining M with increasing weight | Exponential parameter controlling the relationship between natural mortality rate at age and weight at age | Stock@Mexp |
|  | Maximum length | Mean maximum length in current year | Stock@Linf |
|  | Maximum length gradient | Per cent annual change in maximum length | Stock@Linfgrad |
|  | Growth rate | von Bertalanffy growth parameter *κ* in current year | Stock@K |
|  | Age at zero length | Theoretical parameter of the von Bertalanffy growth model | Stock@t0 |
|  | gradient | Per cent annual change in growth rate | Stock@Kgrad |
|  | Maturity | Length at 50% maturity | Stock@L50 |
|  | Maturity | Length increment to 95% maturity | Stock@L50\_95 |
|  | Steepness | The fraction of unfished recruitment at 1/5 unfished spawning biomass (Beverton-Holt) | Stock@h |
|  | Unfished recruitment | Mean number of recruits under unfished conditions | Stock@R0 |
|  | Stock depletion in current year | Current spawning stock biomass relative to unfished levels | Stock@D |
|  |  |  |  |

Table App.B.2. Single value parameters controlling stock dynamics (*Stock* object)

|  |  |  |  |
| --- | --- | --- | --- |
| **Symbol** | **Description** | **Object@Attribute** | **Type** |
| *na* | Maximum number of age classes used in population dynamics calculations | Stock@maxage | Single integer value |
|  | Stock recruitment relationship | Stock@SRrel | Single value  1: Beverton-Holt  2: Ricker |
|  | Length-weight parameter | Stock@a | Single value |
|  | Length-weight parameter | Stock@b | Single value |
|  |  |  |  |

Table App.B.3. Sampled parameters controlling variability in stock dynamics (*Stock* object)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Symbol** | **Description** | **Default distribution** | **Sampled parameter** | **Object@Attribute** |
|  | Inter-annual variability in natural mortality rate |  |  | Stock@Msd |
|  | Inter-annual variability in von Bertalanffy growth rate |  |  | Stock@Ksd |
|  | Inter-annual variability in maximum length |  |  | Stock@Linfsd |
|  | Inter-annual variability in recruitment |  |  | Stock@Perr |
| Temporal autocorrelation in recruitment |  |  | Stock@AC |
| Period (wavelength) of cyclical recruitment |  |  | Stock@Period |
| Amplitude of cyclical recruitment |  | Stock@Amplitude |
|  |  |  |  |  |

**Appendix C. Fishing dynamics**

Fishing mortality rate *F* is calculated according to a catchability coefficient, annual effort *E*, spatial distribution of effort *δE*, age-selectivity *s*, the retention rate (probability of retaining a fish given it is caught) *R,* the discard mortality rate *Mdisc* (fraction of released fish that die) and the geographical size of each fished area :

28)

The catchability coefficient is calculated by numerical optimization such that stock depletion in the current year matches user-specified depletion (spawning biomass relative to unfished levels):

29)

Meeting the condition:

30)

The overall retention rate at age *R*, is a combination of an age-specific retention *r* with a maximum value of 1, and a flat-rate of discarding :

31)

Separating overall retention *R*, into age-specific *r*, and overall rates of discarding *γ*, allows users to adjust these separately for example examining the influence of a new size-limit (*r*) whilst acknowledging that a fixed fraction of the catch γ will still be discarded.

The spatial distribution of effort is proportional to the distribution of vulnerable biomass *V* raised to the power of a spatial targeting parameter :

32)

When is 1, fishing effort is distributed in proportion to vulnerable biomass (the default assumption). For values of higher than 1, fishing occurs preferentially in areas of higher vulnerable biomass. When is zero, fishing effort is distributed uniformly irrespective of vulnerable biomass. Vulnerable biomass is the product of numbers *N*, weight *w* and age selectivity *s*:

33)

The vulnerability at age, *sy,a,* was calculated from a double-normal selectivity curve and model predicted length-at-age.

34)

The length at maximum selectivity is user defined (Table App.C.1). The standard deviation parameters of the ascending limb and was given by user specified length at 5% vulnerability :

35)

While the standard deviation of the descending limb was given by user specified vulnerability of fish of length (Eqn. 8):

36)

Age-specific retention *r*, is modelled using the same double-normal curve as selectivity:

37)

The length at maximum retention is user-specified (Table App.C.1). The standard deviation parameters of the ascending limb and was given by user-specified length at 5% retention :

38)

While the standard deviation of the descending limb was given by user-specified retention of fish of length (Eqn. 8):

39)

In historical simulations, catch in numbers *C*, are calculated using the Baranov equation:

40)

Note that the fishing mortality rate calculation (Eqn. 28) includes discard mortality that is not included in the calculation of catches in this equation.

In projected years when TACs (limits on the weight of landings) are recommended by MPs, the equations are reversed and fishing mortality rates are calculated from prescribed catches. To distribute catches over areas and ages it is first necessary to calculate the distribution of vulnerable biomass (post retention because this is used to distribute landings) across ages and areas.

41)

The realized catches are the TAC recommendations accounting for implementation error in the TAC *ITAC* (an improper fraction, see Appendix E for the implementation error modelling), retention rates *R* and post-release mortality rate *Mdisc*:

42)

Fishing mortality rates are then calculated from these realized catches subject to the constraint that they do not exceed user-specified *Fmax*.

43)

In cases where MPs make TAE (Total Annual Effort, e.g. 100 boat days, 1000 pot hours), these recommendations are expressed as a fraction *TAEMP,y* of the total annual effort in the current year *cy* (the last historical year, e.g. 2017). These TAE recommendations are subject to implementation error *ITAE* (an improper fraction, see Appendix E for the implementation error modelling):

44)

This equation also includes time varying catchability (projected years only) which is determined by the estimated catchability in the final year accounting for trend and log-normal inter-annual variability :

45)

Some MPs prescribe size limits. In these cases, in projection years (*y > c*) the retention-at-age curve (Eqns. 37 - 39) is calculated from MP parameters controlling the length at 5% retention , length at 100% retention and retention of the fish at maximum length subject to implementation error in the size limit *ISL,y* (an improper fraction that can lead to the retention of smaller or larger fish than prescribed by the MP, Appendix E):

46)

47)

48)

In cases where users wish to specify a slot-limit, a knife-edge upper bound on retention can be specified using *Lupper\_MP*, for which retention is zero for fish of greater length. Note that selectivity remains constant in the future projections regardless of size limits.

Some MPs prescribe spatial effort controls that can include regional reductions or spatial closures. These MPs provide a vector of spatial fractional closures *SCMP* and the fraction of the effort displaced from closed areas that is reallocated to open areas. Since this MP is static over projected years (a constant effort distribution recommendation) it is based on the effort distribution in the current year. First it is necessary to characterize the new fishing distribution over areas. This is based on Eqn. 32 which calculates effort distribution δ based on the spatial distribution of vulnerable biomass *V* and the targeting parameter . The new distribution of fishing effort is proportional to in the current year *c*, and the vector of spatial closures *SCMP*:

49)

The summation of these terms provides the fraction of current effort in open areas:

50)

The MP also specifies the fraction *RAMP* of the effort in closed areas (), that is reallocated to open areas. For MPs that alter the spatial distribution of effort, for all projected years *y*, the distribution of effort (previously Eqn. 32) is calculated:

51)

When *RAMP* = 0, there is no effort reallocation to open areas and effort is reduced in accordance with the regional extent of closure defined by *SCMP* (i.e. . When *RAMP* =1 (the default), there is full reallocation of displaced effort to open areas (i.e. . Currently there is no implementation error and the distribution of spatial effort prescribed by an MP is assumed to be followed perfectly.

Table App.C.1. Sampled parameters controlling fleet dynamics (*Fleet* object)

|  |  |  |
| --- | --- | --- |
| **Symbol** | **Description** | **Object@Attribute** |
|  | Shortest length at 100% vulnerability | Fleet@LFS |
|  | Shortest length at 5% vulnerability (ascending limb of double-normal vulnerability) | Fleet@L5 |
|  | Vulnerability of the maximum length fish in the current model year | Fleet@Vmaxlen |
|  | Shortest length at 100% retention (*r*) | Fleet@LFR |
|  | Shortest length at 5% retention (ascending limb of double-normal retention *r*) | Fleet@LR5 |
|  | Retention rate (*r*) of the maximum length fish in the current model year | Fleet@Rmaxlen |
|  | Controls the degree of proportionality between the distribution of vulnerable biomass and effort | Fleet@Spat\_targ |
|  | Overall discarding rate | Fleet@DR |
|  | Per cent annual change in catchability in projected years | Fleet@qinc |
|  |  |  |

Table App.C.2. Other parameters controlling fleet dynamics (*Fleet* object)

|  |  |  |  |
| --- | --- | --- | --- |
| **Symbol** | **Description** | **Object@Attribute** | **Type** |
| *c* | Number of historical years | Fleet@nyears | Single integer value |
|  |  |  |  |

Table App.C.3. Sampled parameters controlling variability in fleet dynamics (*Fleet* object)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Symbol** | **Description** | **Default distribution** | **Sampled parameter** | **Object@Attribute** | **Default distribution** |
|  | Inter-annual variability in projected |  |  | Stock@qcv | U(LB, UB) |
|  |  |  |  |  |  |

Table App.C.4. Operating model parameters controlling MSE forward projections (*OM* object)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Symbol** | **Population dynamics** | **Description** | **Object@Attribute** | **Default value** |
|  | Maximum fishing mortality rate | In any year, area and age class this is the maximum instantaneous rate of fishing. | OM@Fmax | 0.9 |
|  | Random seed | Ensures exact reproducibility of MSE results | OM@seed | 1 |
|  | Update interval | Controls the duration (number of time steps, e.g. years) between new MP recommendations. | OM@interval | 3 |
|  | Number of projection years | The number of years over which MPs are tested in closed-loop simulation | OM@proyears | 50 |
|  |  |  |  |  |

**Appendix D: Observation model**

In order to test MPs in closed loop simulation it is necessary to simulate the various types of data required by these MPs. There are three fundamental types of data: (1) time series data (e.g. annual catches from 1970-2017), (2) single value variables and parameters and (3) catch composition data (e.g. size or age samples).

***Time series data***

Time series data are simulated that include consistent biases (e.g. catch under reporting) in addition to error (e.g. lognormal observation error in annual catches). Single value variables and parameters include biases (for example consistent underestimation of natural mortality rate or steepness of the stock-recruitment relationship).

The example of observed annual catches is used to describe how time series bias and error is handled in DLMtool. Annual observed catches are calculated by multiplying simulated catch in numbers-at-age *C*, by weight-at-age *W*, and adding observation error and bias through a factor term *ω*:

52)

The catch factor includes both bias and imprecision in observations.

53)

where bias is an improper fraction (e.g. = 1.2 is equivalent to a 20% positive bias) and the lognormal error term *ε*, was drawn from a standard normal distribution whose standard deviation was sampled at random in each simulation:

54)

By default DLMtool samples simulation-specific observation error from a uniform distribution.

55)

and bias from a log-normal distribution:

56)

57)

This convention means that the user can specify an unbiased (e.g. low and therefore sampled values of close to 1) or biased (e.g. high and therefore sampled values of substantially lower or higher than 1) time series that can be observed with a low degree of error (e.g. low sampled values of specified by lower *LBC* and *UBC*) or high degree of error (e.g. high sampled values of specified by higher *LBC* and *UBC*). Table App D.1. details all time series data that are simulated, their biases and observation errors using the (parameter controlling extent of bias) and *LBC* and *UBC* (range of observation error) conventions described here.

Other time series data are simulated in the same way (Table App.D.1). These include estimates of absolute biomass calculated from annual vulnerable biomass *V* (Eqn 33) (, , *LBB* and *UBB*). Stock depletion calculated from spawning biomass relative to unfished levels (Eqn 30) (, , *LBD* and *UBD*). Annual total fishing mortality rate calculated in Eqns 28, 43 and 44 (, , *LBF* and *UBF*). Relative abundance indices calculated from time series of spawning biomass *S* (Eqn 17) (, , *LBI* and *UBI*). The last of these differs because it is simulated with an additional parameter controlling non-linearity in the relationship between spawning biomass and the index:

58)

where is the index factor that, like annual catches (Eqns 52 - 57 ) includes bias and imprecision in the index observations and *beta* is the hyperstability-hyperdepletion parameter. When *beta* is 1 the index is linearly related to spawning biomass *S*. When *beta* is greater than 1 the index is hyper deplete and moves faster than true spawning biomass changes. When *beta* is lower than 1 the index is hyperstable and moves slower than true spawning biomass changes. Note that in every year *y* that the index is recalculated, it is re-normalised to have a mean value of 1 over all years.

***Single value variables and parameters***

Unlike time-series data, single value variables and parameters are assumed to have a fixed bias over the entire projected time series. For example growth rate *κ* may be consistently over or under-estimated:

59)

60)

61)

A number of slots in the *Obs* object control consistent biases in observed quantities that are described in further detail in Table App.D.2.

***Catch composition data***

Two types of catch composition observations are simulated, catches by age class by year (CAA) and catches by length class by year (CAL).

These observation models use a simple multinomial model that accounts for effective sample size (the number of independent observations). For both CAA and CAL observation models the user specifies an average annual number of samples (number of individuals measured for example) and the annual effective sample size. For example, *ESSCAA* independent catch samples at age (e.g. 20 per year) are sampled in proportion *p* to the spatial catch-at-age predicted by the model *C* (Eqns. 40 and 42):

62)

For each year, the frequency of samples at age are inflated to match the total sample size *nCAA* and rounded to the nearest integer:

63)

Due to rounding, this relatively simple model generates frequency at age data that is approximately equal to (but not always exactly) the average annual sample size:

64)

For example, for 10 age classes, *nCAA* = 200 and *nESS* = 45, 90% of simulations sampled equal to 199, 200 or 201 and less than 1% of simulation were less than 198 or greater than 202.

Table App.D.1. Observation model attributes for simulating biased and imprecise time series data (*Obs* object)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Time series data type** | **Sampled bias parameter** | **Object@attribute** | **Default distribution** | **Sampled error parameter** | **Object@Attribute** |
| Annual catches *Cobs* |  | Obs@Cbiascv | LN(1, ) |  | Obs@Cobs |
| Index of relative Abundance *Iobs* |  | Obs@Ibiascv | LN(1, ) |  | Obs@Iobs |
| *beta* | Obs@beta | U(LB, UB) |  |  |
| Stock depletion *Dobs* |  | Obs@Dbiascv | LN(1, ) |  | Obs@Dobs |
| Current absolute biomass *Bobs* |  | Obs@Btbiascv | U(LBbC, UBbC) |  | Obs@Btobs |
|  |  |  |  |  |  |

Table App.D.2. Observation model parameters controlling consistent biases in variables and parameters.

|  |  |  |
| --- | --- | --- |
| **Time series data type** | **Sampled bias parameter** | **Object@attribute** |
| Mean natural mortality rate of mature fish *Mobs* |  | Obs@Mbiascv |
| von Bertalanffy growth parameter *κobs* |  | Obs@Kbiascv |
| von Bertalanffy age at zero length |  | Obs@t0biascv |
| Maximum length |  | Obs@Linfcv |
| Length at first capture |  | Obs@LFCbiascv |
| Length at first capture |  | Obs@LFSbiascv |
| Length at 50% maturity |  | Obs@LenMbiascv |
| Reference catch levels (Maximum Sustainable Yield) *Crefobs* |  | Obs@Crefbiascv |
| Reference biomass levels (BMSY) *Brefobs* |  | Obs@Brefbiascv |
| Reference abundance index levels (BMSY relative to unfished) *Iobs* |  | Obs@Irefbiascv |
| MSY fishing mortality rate *FMSYobs* |  | Obs@FMSYbiascv |
| MSY fishing mortality rate relative to natural mortality rate of mature fish *FMSY\_Mobs* |  | Obs@FMSY\_Mbiascv |
| Biomass at MSY relative to unfished biomass *BMSY\_B0obs* |  | Obs@BMSY\_B0biascv |
| Recruitment compensation (steepness) |  | Obs@hbiascv |
| Recent recruitment strength |  | Obs@recbiascv |
|  |  |  |

**Appendix E: Implementation error model**

Three types of implementation error are included in DLMtool which relate to MPs that provide management advice in terms of Total Allowable Catch (TAC), Total Allowable Effort (TAE) and size limits (SL). Parameters for these are included in the *Imp* (implementation error) object. Similar to time-series data, these implementation error models can simulate both consistent overages / underages and also inter-annual variability (e.g. TAC underages of 10% that vary between 5% and 15%). The effect of these implementation errors on fishing dynamics are described in Eqns 42, 44, and 46 , respectively. For example TAC implementation error *ITAC,y* is the product of a constant fraction of the TAC taken sampled for each simulation and a degree of inter-annual variability controlled by .

53)

where is an improper fraction (e.g. = 0.7 is equivalent to 30% catch underages) and the lognormal error term *ε*, is drawn from a standard normal distribution whose standard deviation was sampled at random in each simulation:

54)

By default DLMtool samples simulation-specific variability from a uniform distribution.

55)

and mean fraction of recommendation from a log-normal distribution:

56)

57)

Using the same equations, implementation error for TAEs (*ITAE,y,* , , *LBTAE* and *UBTAE*) and size limits (*ISL,y,* , , *LBSL* and *UBSL*) were also simulated. Details about these parameters and their corresponding *Imp* object attributes are included in Table App.E.1.

Table App.E.1. Implementation model attributes for simulating implementation error in TACs, TAEs and size limts (*Imp* object). In this context the fraction parameter refers to consistent under or overages.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Recommendation type** | **Sampled fraction parameter** | **Object@attribute** | **Default distribution** | **Sampled variability parameter** | **Object@Attribute** |
| Total Annual Catch (TAC) |  | Imp@TACFrac | LN(1, ) |  | Imp@TACSD |
| Total Annual Effort (TAE) |  | Imp@EFrac | LN(1, ) |  | Imp@ESD |
| Size limit |  | Imp@SizeLimFrac | LN(1, ) |  | Obs@Dobs |
|  |  |  |  |  |  |

**Appendix F: Overview of Management Procedures**

Table App.F.1. Overview of output control MPs (class *Output*) that provide TAC recommendations.

|  |  |
| --- | --- |
| **MP code** | **Description** |
| AvC | TAC is average historical catches |
| BK | Beddington and Kirkwood (2005) approach for estimating FMSY sets the TAC using an estimate of absolute current abundance. |
| BK\_CC | As BK but using a catch-curve analysis to estimate current abundance |
| CC1 | Constant Catch MP of Geromont and Butterworth (2014) |
| CC4 | Constant Catch MP of Geromont and Butterworth (2014) |
| CompSRA | TAC is calculated from an estimate of FMSY and current abundance provided by catch at age composition data. |
| DCAC | Depletion Adjusted Average Catch (MacCall 2009) |
| DBSRA | Depletion-Based Stock Reduction Analysis (Dick and MacCall 2011) |
| DBSRA\_40 | DBSRA assuming that stock depletion is constant at 40% of unfished |
| DAAC | Depletion Adjusted Average Catch, a version of DCAC with an additional correction for depletion |
| DCAC4010 | A version of DCAC with a 40-10 harvest control rule |
| DCAC\_40 | DCAC assuming depletion is constant at 40% of unfished |
| DD | Delay-Difference stock assessment |
| DD4010 | Delay-Difference stock assessment with a 40-10 harvest control rule |
| Fratio | TAC is set to an estimate of FMSY multiplied by an estimate of current stock size |
| Fratio\_CC | As Fratio but the estimate of current stock size comes from a catch curve analysis |
| GB\_CC | Geromont and Butterworth (2015b) constant catch MP |
| GB\_slope | Geromont and Butterworth (2015b) slope of relative abundance index MP |
| GB\_target | Geromont and Butterworth (2015b) target CPUE and catch MP |
| Gcontrol | Searches for biomass with high surplus production by finding flat gradients in the relationship between surplus production and biomass. Smooths historical catch and indices. Based on theory of Maunder (2014). |
| HDAAC | Hybrid Depletion Adjusted Average Catch. This is DCAC for biomass levels higher than BMSY and DAAC for biomass levels lower than BMSY. |
| Islope1 | Adjusts the TAC to achieve constant CPUE (Geromont and Butterworth 2015a) |
| Islope4 | As I slope 1 but is slower to respond and aims for a lower fraction of historical catches. |
| IT10 | TACs are adjusted by up to 10% to reach a target index level |
| IT5 | TACs are adjusted by up to 5% to reach a target index level |
| Itarget1 | TAC adjusted to reach a target relative abundance index level (Geromont and Butterworth 2015a) |
| Itarget4 | As Itarget1 but aims for a higher relative abundance |
| ITM | Similar to IT10 but uses an estimate of natural mortality rate to select the number of years for smoothing observations of the index and catches. |
| L95target | TAC is adjusted to meet a target mean length of fish caught |
| LBSPR\_ItTAC | Length-based spawning potential ratio assessment is used to adjust TAC |
| LstepCC1 | TAC is adjusted based on the mean length of recent catches (Geromont and Butteworth 2015a) |
| LstepCC4 | As LstepCC1 but reference TAC level is lower (ie makes adjustments relative to a lower TAC) |
| Ltarget1 | Incrementally adjusts the TAE to reach a target mean length in catches (Geromont and Butterworth 2015a) |
| Ltarget4 | As LtargetE1 but more biological precautionary aiming for a higher mean length of catches and uses a lower reference TAC level (Geromont and Butterworth 2015a) |
| MCD | The TAC recommendation is twice average historical catches multiplied by depletion (Harford and Carruthers 2017) |
| MCD4010 | As MCD but has an additional 40-10 harvest control rule |
| Rcontrol | Similar to Gcontrol but an additional approximation of intrinsic rate of increase is used to quantify surplus production |
| Rcontrol2 | As Rcontrol but a quadratic relationship between surplus production and biomass is imposed |
| SBT1 | Similar to Southern Bluefin Tuna MP1 (CCSBT 2011) |
| SBT2 | Similar to Sourthern Bluefin Tuna MP2 (CCSBT 2011) |
| SPMSY | Catch trends are used to infer depletion and set a TAC (operational version of the concept of Martell and Froese 2015) |
| SPslope | Simpler version of Gcontrol that uses slope in the index and catches to infer surplus production and make TAC adjustments to reach a productive stock size (Maunder 2014). |
| SPSRA | A simple stock reduction analysis is used to |
| YPR | Calculates FMSY from yield per recruit analysis and then sets the TAC according to a recent absolute abundance estimate. |
| YPR\_CC | As YPR but catch curve analysis and an estimate of natural mortality rate are used to estimate recent absolute abundance. |
|  |  |

Table App.F.2. Overview of input control MPs (class *Input*) that provide TAE recommendations, size limits and time-area closures.

|  |  |
| --- | --- |
| **MP code** | **Description** |
| curE | TAE is constant at current effort levels |
| curE75 | TAE is constant at 75% current effort levels |
| DDe | Delay Difference stock assessment making TAE recommendations consistent with FMSY |
| DDe75 | Delay Difference stock assessment making TAE recommendations consistent with 75% FMSY |
| DDes | TAE searching version of Delay Difference stock assessment |
| DTe40 | TAE recommendations are modified to reach 40 per cent stock depletion |
| DTe50 | TAE recommendations are modified to reach 50 per cent stock depletion |
| EtargetLopt | TAE is increased / decreased if mean length is higher / lower than target level |
| ItargetE1 | TAE adjusted to reach a target relative abundance index level (Geromont and Butterworth 2015a) |
| ItargetE4 | As ItargetE1 but aims for a higher relative abundance |
| ITe10 | TAE adjusted by up to 10% per year to reach a target level of the relative abundance index |
| ITe5 | As ITe10 but with up to 5% changes in TAE among years |
| LBSPR\_ItEff | Length based spawner per recruit model that iteratively adjusts TAE |
| LBSPR\_ItSel | Length based spawner per recruit model that iteratively adjusts retention at length |
| LstepCE1 | Incrementally adjusts the TAE according to the mean length of recent catches. |
| LstepCE2 | As LstepCE1 but less biological precautionary |
| LtargetE1 | Incrementally adjusts the TAE to reach a target mean length in catches (Geromont and Butterworth 2015a) |
| LtargetE4 | As LtargetE1 but more biological precautionary aiming for a higher mean length of catches and uses a lower reference TAC level (Geromont and Butterworth 2015a) |
| matlenlim | Retention at length is set according to the size at maturity curve |
| matlenlim2 | Retention at length is set slightly higher than the size at maturity curve |
| minlenLopt1 | Sets the minimum length of fish caught to a fraction of the length that maximises the biomass |
| MRnoreal | An marine reserve in an area with no reallocation of fishing effort |
| Mrreal | An marine reserve in an area with full reallocation of fishing effort |
| slotlim | Sets an example slot limit |
|  |  |

**Appendix G. Custom parameters**

When a DLMtool MSE runs, under default conditions parameter are sampled from (mostly uniform) random distributions and these are used to construct various vector and matrices for undertaking calculations. Often this results in a vector the same length as the number of simulations. The *cpars* slot of the *OM* object can be used to override these values with any user-specified values (which may arise from any sampling distribution and be cross correlated, for example). If 5 simulations were specified in the OM object (OM@nsim = 5) a user could custom specify current depletion () for these simulations, for example: OM@cpars$dep = rnorm(5, 0.5,0.01). In this case, a sample of 5 current depletion values were drawn from a very precise normal distribution with mean 0.5 and standard deviation 0.01. Tables App.G.1 – 3 describe each variable that may be overridden by custom parameters including their nomenclature in this paper (e.g.) and the internal nomenclature in the DLMtool R code (e.g. dep).

In some cases users can override arrays that are produced inside the DLMtool MSE function itself to specify more complex operating model hypotheses. For example,e the result of all the parameter draws for growth parameters is a 3 dimensional array (Len\_age) which provides the mean length at age of individuals for each simulation, year and age class (Table App.G.1). Using these ‘internal’ *cpars* attributes users can specify complex patterns of natural mortality rate (Marray) and recruitment strength (Perr) (Table App.G.1) in addition to exploitation rate characteristics such as fishing pressure (Find) and vulnerability (V) (Table App.G.2).

Table App. G.1. A list of custom population dynamics parameters that may be specified by the user which over-rides the default sampling of parameter values (from the *Stock* object). The terms *nsim*, *ny*, and *na* refer to the number of simulations, years and ages, respectively.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Description** | **Internal DLMtool parameter name** | **Dimensions** |
|  | Stock depletion in current year | dep | Vector *nsim* long |
|  | Inter-annual variability in recruitment (lognormal St.Dev) | procsd | Vector *nsim* long |
|  | Temporal autocorrelation in recruitment | AC | Vector *nsim* long |
|  | Mean instantaneous natural mortality rate of mature individuals in the current year | M | Vector *nsim* long |
|  | Inter-annual variability in natural mortality rate (lognormal St.Dev) | Msd | Vector *nsim* long |
|  | Per cent annual change in natural mortality rate | Mgrad | Vector *nsim* long |
|  | The fraction of unfished recruitment at 1/5 unfished spawning biomass (Beverton-Holt) | hs | Vector *nsim* long |
|  | Mean maximum length in current year | Linf | Vector *nsim* long |
|  | Inter-annual variability in maximum length (lognormal St.Dev) | Linfsd | Vector *nsim* long |
|  | Per cent annual change in maximum length | Linfgrad | Vector *nsim* long |
|  | von Bertalanffy growth parameter *κ* in current year | K | Vector *nsim* long |
|  | Inter-annual variability in von Bertalanffy growth rate (lognormal St.Dev) | Ksd | Vector *nsim* long |
|  | Per cent annual change in growth rate | Kgrad | Vector *nsim* long |
|  | Mean theoretical length at age zero | t0 | Vector *nsim* long |
|  | Length at 50% maturity | L50 | Vector *nsim* long |
|  | Length increment to 95% maturity | L50\_95 | Vector *nsim* long |
|  | Unfished fraction of individuals in area 1 | Frac\_area\_1 | Vector *nsim* long |
|  | Mean probability of individuals staying in the same area between years | Prob\_staying | Vector *nsim* long |
|  | Relative size (fraction) of area 1 | Size\_area\_1 | Vector *nsim* long |
|  | Unfished recruitment | R0 | Vector *nsim* long |
|  | Mortality rate of discarded fish | Fdisc | Vector *nsim* long |
| internal | Maturity at age array (overrides and ) | Mat\_age | 3D array *nsim* by *ny* by *na* |
| internal | Weight at age by year (overrides ) | Wt\_age | 3D array *nsim* by *ny* by *na* |
| internal | Length at age by year (overrides ) | Len\_age | 3D array *nsim* by *ny* by *na* |
| internal | Recruitment strength (assumed to arise from a lognormal random variable with mean 1) (overrides , , , ) | Perr | Matrix *nsim* by *ny* |
| internal | Mortality rate at age by year (, | Marray | 3D array *nsim* by *ny* by *na* |
|  |  |  |  |

Table App. G.2. As Table App.G.1 but for fishing dynamics parameters of the *Fleet* object.

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter** | **Description** | **Internal DLMtool parameter name** | **Dimensions** |
|  | Controls the degree of proportionality between the distribution of vulnerable biomass and effort | Spat\_targ | Vector *nsim* long |
|  | Per cent annual change in catchability in projected years | qinc | Vector *nsim* long |
|  | Inter-annual variability in catchability (log normal St.Dev) | qcv | Vector *nsim* long |
|  | Shortest length at 5% vulnerability (ascending limb of double-normal vulnerability) | L5 | Vector *nsim* long |
|  | Shortest length at 100% vulnerability | LFS | Vector *nsim* long |
|  | Vulnerability of the maximum length fish in the current model year | Vmaxlen | Vector *nsim* long |
|  | Shortest length at 5% retention (ascending limb of double-normal retention *r*) | LR5 | Vector *nsim* long |
|  | Shortest length at 100% retention (*r*) | LFR | Vector *nsim* long |
|  | Retention rate (*r*) of the maximum length fish in the current model year | Rmaxlen | Vector *nsim* long |
|  | Overall discarding rate | DR | Vector *nsim* long |
| < internal > | Vulnerability at age array (overrides | V | 3D array *nsim* by *ny* by *na* |
| < internal > | Apical (rate of the most vulnerably age class) fishing mortality rate by year | Find | Matrix *nsim* by *ny* |
|  |  |  |  |

Table App. G.2. As Table App.G.1 but for observation model parameters of the *Obs* object.

|  |  |  |
| --- | --- | --- |
| **Parameter** | **Description** | **Internal DLMtool parameter name** |
|  | Observation error of annual catches (lognormally distributed) | Csd |
|  | The degree of bias in annual catch observations (the mean bias is sampled from a lognormal distribution with mean 1) | Cbias |
|  | Average number of annual samples of catch at age. | CAA\_nsamp |
|  | Effective sample size (number of independent observations of) annual catch at age. | CAA\_ESS |
|  | Average number of annual samples of catch at length. | CAL\_nsamp |
|  | Effective sample size (number of independent observations of) annual catch at length. | CAL\_ESS |
| *beta* | The hyperstability/hyperdepletion parameter controlling linearity between biomass trends and an observed relative abundance index. | Betas |
|  | Observation error of annual catches (lognormally distributed) | Isd |
|  | Observation error of annual catches (lognormally distributed) | Derr |
|  | The degree of bias in annual catch observations (the mean bias is sampled from a lognormal distribution with mean 1) | Dbias |
|  | As above for natural mortality rate of mature individuals. | Mbias |
|  | As above for the ratio of FMSY fishing mortality rate to natural mortality rate | FMSY\_Mbias |
|  | As above for length at 50% maturity | lenMbias |
|  | As above for length at 5% vulnerability | LFCbias |
|  | As above for length at 100% vulerabilty | LFSbias |
|  | Observation error of annual catches (lognormally distributed) | Aerr |
|  | The degree of bias in observation of annual absolute biomass (the mean bias is sampled from a lognormal distribution with mean 1) | Abias |
|  | As above for the von Bertalanffy growth parameter | Kbias |
|  | As above for the theoretical length at age zero | t0bias |
|  | As above for average maximum length | Linfbias |
|  | As above for the index at MSY levels | Irefbias |
|  | As above for MSY catch levels | Crefbias |
|  | As above for biomass at MSY levels | Brefbias |
|  | As above for the strength of recent recruitments. | Recsd |
|  |  |  |

**References**

Beddington, J.R., Kirkwood, G.P., 2005. The estimation of potential yield and stockstatus using life history parameters. Philos. Trans. R. Soc. Lond. B: Biol. Sci. 360,163–170.

CCSBT. 2011. Report of the Sixteenth Meeting of the Scientific Committee. Commission for the Conservation of Southern Bluefin Tuna, Bali, Indonesia. https://www.ccsbt.org/sites/default/files/userfiles/file/docs\_english/meetings/meeting\_reports/ccsbt\_18/report\_of\_SC16.pdf (last accessed 5 October 2017).

Geromont, H. F., and Butterworth, D. S. 2015a. Complex assessments or simple management procedures for efficient fisheries management: A comparative study. ICES Journal of Marine Science, 72: 262–274.

Geromont, H. F., and Butterworth, D. S. 2015b. Generic management procedures for data-poor fisheries; forecasting with few data. ICES Journal of Marine Science, 72: 251–261.

Maunder, M. N. 2014. Management Strategy Evaluation (MSE) Implementation in Stock Synthesis: Application to Pacific Bluefin Tuna. Fifth Meeting of the Scientific Advisory Committee. Inter-American Tropical Tuna Commission. SAC-05-10b. http://www.iattc.org/Meetings/Meetings2014/MAYSAC/PDFs/SAC-05-10b-Management-Strategy-Evaluation.pdf (last accessed 5 October 2016).

Quinn, T.J., Deriso, R.B. 1999. Quantitative fish dynamics. Oxford University Press, New York.