A Toolbox for Age-structured Stock Assessment using Catch and Survey data (TASACS)

Dankert W. Skagen and Åsmund Skålevik

FISKEN OG *HAVET*

HAVFORSKNINGSINSTITUTTET
INSTITUTE OF MARINE RESEARCH

PROSJEKTRAPPORT

Rapport: Fisken og Havet	**Nr. - År**

Tittel (norsk/engelsk):

A Toolbox for Age-structured Stock Assessment using Catch and Survey data (TASACS)

En verktøykasse for aldersstrukturerte bestandsberegninger basert på fangst- og overvåkingsdata (TASACS)

Forfatter(e):
Dankert W. Skagen og Åsmund Skålevik

in collaboration with:
Sergei Belikov, Tatyana Prokhorova and Alexander Krysov (PINRO), Tatjana Bulgakova and Dimitrij Vasilyev (VNIRO), and Jens Christian Holst, Daniel Howell, Reidar Toresen and Morten Åsnes (IMR)

Antall sider totalt: 35

Sammendrag (norsk):

TASACS (A Toolbox for Age-structured Stock Assessment using Catch and Survey data) er en samling programvare for å gjøre rutinemessige analytiske bestandsberegninger av fiskebestander. Den består av programmer for å gjøre bestandsbereginger og hjelpeverktøy for å håndtere inngangsdata, organisere arbeidet og presentere resultater og diagnostica. Artikkelen gir en detaljert beskrivelse av beregningsmetodene og en oversikt over det øvrige verktøyet.

Summary (English):

TASACS (A Toolbox for Age-structured Stock Assessment using Catch and Survey data) is a collection of computer programs for performing routine analytic assessments of fish stocks. It consists of programs for making assessments and a working environment for handling input data, organizing the work and displaying results and diagnostics. The paper gives a detailed description of the methods and an outline of the working environment.

Emneord (norsk):	Subject heading (English):
1. Bestandsberegning	1. Stock assessment
2. Programvare	2. Computer software
3.	3.

Prosjektleder   **Faggruppeleder**

Contents

1.   Introduction and background

TASACS is a toolbox of programs to perform analytic assessments of a fish stock. The assessment tools here can be regarded as means to translate catch and survey information over a time range, into stock abundance at age and fishing mortality at age covering the same time range. The assessment programs in TASACS are integrated in a working environment for organizing the work and presentation and book-keeping of the results.

The routines in the toolbox are based on well established theory and procedures, and should cover the needs for assessing most stocks where standard information (catch in numbers at age, weights and maturities at age and survey indices - or alternatively catch per unit of effort - with and/or without age disaggregation) is available. The toolbox provides a selection of implementations of the theory.

The development was triggered by a request from NEAFC to ICES to provide a unified software for analyzing the data for Norwegian Spring Spawning herring (NSSH). The model formulations and the contents of the package is the result of collaboration between scientists from PINRO, VNIRO and IMR. The programs have been written in accordance with the protocols from a series of meetings with these scientists. The coding and testing of the program was done by the authors.

Although triggered by the need for software to asses the NSSH stock, TASACS is designed as a general toolbox to serve the needs for any stock where the standard data are available.
This publication documents the methods and procedures in TASACS. The program, including the source code can be obtained from the authors, and is distributed without restrictions. A user manual is distributed together with the program, which also includes instructions for installation, file formats and a technical description of the program structure.

2.   Overview

The TASACS toolbox has the following elements:

1. The TASACS assessment program with a variety of options.
2. A set of procedures that call the assessment program (1) to perform:
    - Single runs with present parameters
    - Ordinary assessment runs (estimation of parameters)
    - Bootstrap runs
    - Retrospective runs
3. A set of post-processing graphical routines to present results and diagnostics.
4. The TASACS working environment for
    - Inspecting and editing input data and parameters, as well as model settings.
    - Calling tasks (cfr. pt. 2 and 3) to be done
    - Book-keeping of runs and results
    - Presenting results and diagnostics

The user interface is the working environment. The communication between the interface and the assessment program is trough files, which are normally edited in the working environment. The assessment program can also run on its own if the necessary files have be made separately. All files are ascii-files and can be made with any text editor. The Lowestoft format is used for the input data files.

The TASACS assessment program itself is written in Fortran 95, and compiles and runs on any platform. The working environment is written mostly in Delphi, and requires Windows as operating system.

The assessment program uses the following kinds of information:

- Input observations. These are on ascii-files formatted according to the Lowestoft file format standard.
- Parameters, with proposed values and active flags (see Section 3.2.1) for all model parameters, on a parameter-file. If the parameter is declared as active (flag=1), the value is just an initial value for the search.
- Weightings of the individual data (weightings files).
- Basic specifications, including names of input files and model options. This information is assembled on an index file, which is generated by the working environment and used as the link between that and the program.

The TASACS environment has facilities for inspecting and editing the input files in an understandable lay-out, for starting the procedures described above, and for presenting results and diagnostics.

The main output from the assessment program itself is two ASCII- files with the main results in matrix form:

1. Summary.txt: A summary file with stock numbers, fishing mortalities and a summary of biomasses, recruitments and average fishing mortalities over time.
2. Output.txt: This file has the additional information needed for the diagnostics.

The files are read by the post-processing routines, but can also easily be read by other programs, e.g. R-scripts.

The TASACS package is self-contained and can be compiled and run without additional software or libraries. However, it is structured in such a way that the core parts of the program may be called as .dll files by other environments. In the standard distribution, the assessment program itself is in a .dll file, which is seen by the TASACS environment. Likewise, the output is on files that should be easy to read by other programs, to utilize existing software for presentation etc.

3. The TASACS assessment program

The method is basically to construct a model population driven by model parameters and optionally by data, derive expected observations and express the fit of the model to the observed data as an objective function. The user can decide which parameters to estimate, by attaching an 'active flag' to each parameter. The objective function is a quantitative measure of the fit of the model to the data, which formally is a functional on the space of active parameters, given the remaining parameters and the data. An optimization routine searches over the space of active parameters to find the set of active parameters that give the smallest value of the objective function. The model population with this set of parameters is taken as the estimate of the stock.

The model data are generated from the population model and catchability models. Both for the population model, the catchability models and the objective functions several options are available. An outline of the main building blocks is given in Figure 1.
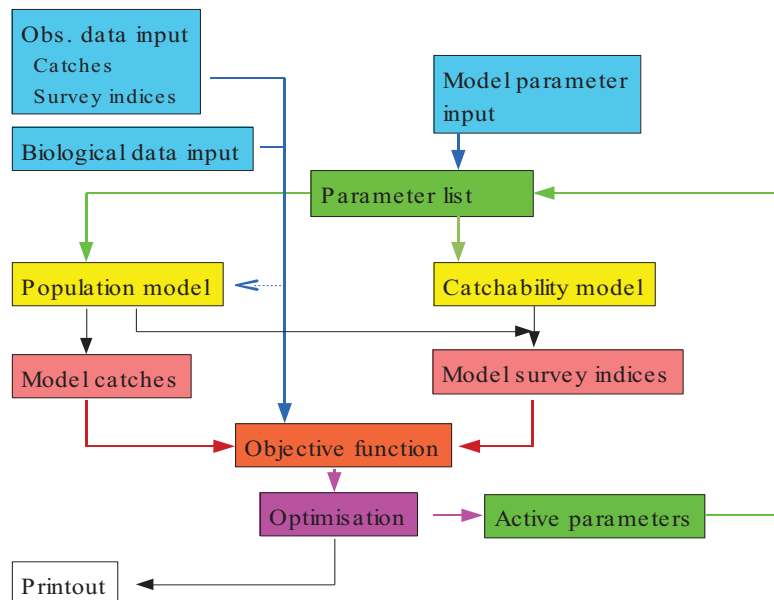
Figure 1. Building blocks in the TASACS assessment program.

In the following, each of these building blocks is described.

3.1 Population models

The population model generates a matrix of stock numbers $N(a,y)$ by age and year, and a corresponding matrix of fishing mortalities $F(a,y)$. Model catch numbers at age and year are generated if they deviate from the observed catches. The lowest age in the age range is the recruiting age. The oldest age is regarded as a plus group in the population. The oldest true age is assumed to be the age below the plus age.

The year range is from the first to the last model year. In the population models that require catch numbers at age (VPA and ISVPA), the model calculates Ns and Fs only up to and including the last year with catch data. If this year is earlier than the last model year, the stock is projected forwards with assumed fishing mortalities. This is done to allow the use of survey data after the last year with catch data even with these population models.

The following population models are available:
1. A VPA
2. An ordinary separable model
3. Two variants of the ISVPA type separable model

3.1.1   The VPA

Input parameters:
- Survivor numbers at all ages in the last year
- Survivor numbers at oldest true age in previous years (optional)
- Natural mortalities $M(a,y)$ at age and year

Input observation data:
- Catch numbers at age by year

Output:
- Primary: Matrix of $N(a,y)$
- Secondary: Matrix of $F(a,y)$ derived from the N-matrix

Calculation of N-value:
 Let $N0(a,y)$ denote the stock numbers $N(a,y)$ at age a at the <u>start</u> of year y, and $N1(a,y)$ the survivors , i.e. the stock number at age a at the <u>end</u> of year y. The survivors $N1(a,y)$ in year y at age a are identical to the numbers $N0 (a+1,y+1)$ at the start of next year. At each time step, $N0$ is calculated from $N1$ using Popes equation (Pope, 1972):

$$N1(a,y) = N0(a+1,y+1) \qquad\qquad\qquad (1a)$$
$$N0(a,y) = N1(a,y)*exp(M(a,y)) + C(a,y)*exp(M(a,y)/2) \qquad\qquad (1b)$$

The start of this back-calculation is the survivors $N1$ at the end of each year class cohort.

The fishing mortalities:
are derived from the stock numbers and the natural mortality as :

$$F(a,y)= log (N0(a,y)/N1(a,y)) - M(a,y)$$

In cases where $N1(a,y) = 0,$ the convention in TASACS is to let F be undefined (see under zero catches below)

Terminal stock numbers:

For obtaining the survivor numbers $N1(a,y)$, the following algorithms apply:

If y is the last year with catch data (Yc)**:**

      Take $N1(a,Yc)$ as a parameter, and derive $N0$ using eqn. (*1b*).

If y is not the last year with catch data, but a is the oldest true age A.

      Here are two alternatives:

      <u>Alternative 1,</u> used if the flag on $N1(A,y)$ = 0, 1 or 2.:

            Take $N1(A,y)$ as a parameter and derive $N0$ using *eqn. 1b*.

      <u>Alternative 2,</u> used if the flag on $N1(A,y)$ = 3:

            Use the catch numbers $C(A,y)$ and fishing mortalities $F(A,y)$ in the terminal year. The stock numbers $N0$ at the start of the terminal year are computed from F and C as:

$$N0(A,y) = C(A,y)*Z(A,y)/F(A,y)/[1\text{-}exp(\text{-}Z(A,y))] \qquad (2)$$

            where $Z(A,y) = F(A,y) + M(A,y)$.

            The following algorithm is used to obtain $F(A,y)$:

            Derive, for each year $y<Yc$:
$$F'(A,y)=F(A,y+1)*Average\{F(a,y)\}/Average\{F(a,y+1)\} \qquad (3)$$

            $F'(A,y)$ is applied as $F(A,y)$ when there are catch data at the oldest age, i.e. when $C(A)>0$.

            If $F(A,Yc)$ is defined as a parameter, that value is used to trigger the process to derive F'. If not, the value of F at the oldest age in the last year where there is a defined F is used.

            If, for $y < Yc\text{-}1$, $F(A,y+1)$ is undefined (see Section 3.2.2), $F'(A,y+1)$ is used instead in eqn. 3.

            The averages in eqn. 3 are taken over all ages in the range that are defined as reference ages, and where the Fs at age are valid in both years. If the *average(F(a,y+1)=0*, or y is the last catch year, take $F'(A,y) = F(a,y)$, *a* being the oldest age with a valid F.

Alternative 1 is the normal way to handle cohorts where supplementary informations is available to scale the cohort. Alternative 2 is the convenient way to handle cohorts where no other data than catch data are available. This procedure always provides a value for $F(A,y)$.

This value is used to derive the terminal N using eqn. 2, unless the rules described in section 3.2.2. on zero catches apply. It assumes that the selection at oldest age relative to the average selection at younger ages is constant, and that the difference in $F(A)$ from one year to the next follows the differences in overall fishing mortality. The procedure has some similarity to that used in ICA, but ICA does not handle undefined fishing mortalities.

3.1.2   The ordinary separable model:

Input parameters:
- Selection at age $S(a)$, all ages
- Annual fishing mortalities $Fy(y)$, all model years
- Survivor numbers at age in the last model year, all true ages
- Survivor numbers at oldest true age in all previous years
- Natural mortalities at age and year

Input observation data:
- None

Output:
- Primary: Matrix of $F(a,y)$ derived from the parameters.
- Matrix of $N(a,y)$
- Model catches.

The fishing mortality $F(a,y)$ at age a in year y is assumed to be a product of a year factor $Fy(y)$ and a selection at age $S(a)$. This defines fishing mortalities at all ages and years as $F(a,y) = Fy(y)*S(a)$, and accordingly, relative stock numbers at age within each year class. The absolute stock numbers is then determined by scaling the cohort to hit a number at age within each year class. In the TASACS implementation, this number is the survivors $N1$ at the end of each year class, which becomes a parameter for each year class. Then for all earlier years, the stock numbers $N0$ at the start of the year are calculated backwards in time as

$$N0(a,y)=N1(a,y)*exp(F(a,y)+M(a,y))$$
and $$N1(a,y) = N0(a+1,y+1)$$

Model catches are derived using the catch equation:

$$C(a,y)= F(a,y)*N(a,y)*(1-exp(-Z(a,y)))/Z(a,y)$$

On the parameter input file, the selection is stated for each year. Normally, the flags are set so that the selection is equal for all years, using the first years' selections at age as active parameters. It is possible, however, to declare selections in later years as active as well, implying different selections in different periods.

Scaling of fishing mortalities in the separable model:

Since the $F(a,y)$ is a product $S(a)*Fy(y)$, the selection has to be on a fixed scale in order to have a unique two-way correspondence between $S(a)$ and $Fy(y)$ on one hand and $F(a,y)$ on the other. In the TASACS program, the selection is rescaled when entering the separable population model, so that the average over the reference ages is *1*. The annual *F*-factor is not touched. Hence, the annual F-factor is interpreted as the normally reported average *F*, while the stated selections at age are interpreted as relative.

3.1.3 ISVPA type separable model:

This model estimates fishing mortalities in a separable model, but it uses the catch data to reduce the numbers of undetermined parameters in an iterative process. ISVPA has several alternative procedures to model the population, the ones represented here corresponds to the 'catch controlled' and the 'effort controlled' versions and 'weighted arithmetic mean procedure' (Vasilyev 2005), but with ordinary fishing mortalities instead of the instantaneous fishing mortality used in the original ISVPA.

A key point in ISVPA is the normalization of selection, where selections at age are scaled so that their sum over all ages is one. This leads to a scaling of selections and yearly F-factors that is different from the general rule in TASACS.

Input parameters:
- Fishing mortality year factor $Fy(Yc)$ in the last year *Yc* with catch data.
- Selection $S(a)$ at age (relative) and annual fishing mortalities in earlier years have to be given as input. They are used as starting values for the iterative search in ISVPA, and will be changed by the ISVPA procedure, but updated values are not copied to the parameter list. The active flag on these input 'parameters' is ignored.
- Natural mortalities at age and year

Input observation data:
- Catch numbers at age by year

Output:
- Primary: Matrix of $N(a,y)$
- Secondary: Matrix of $F(a,y)$ derived from the *N*-matrix

Algorithm:
Let *A* be oldest true age and *Yc* the last year with catch data.

1. Get proposed values for all $F(a,y)$: $F(a,y)=Fy(y)*S(a)$

2. Calculate all $N(A,y)$ and $N(a,Yc)$ from the $F(a,y)$ and the observed catches $C(a,y)$, using eqn (1).

$$N(a,y) = C(a,y)*Z(a,y)/F(a,y)/[1-exp(-Z(a,y))] \qquad (1)$$

where $Z(a,y) = F(a,y) + M(a,y)$ and $M(a,y)$ is natural mortality.

If the catch is 0, the N is 0.

3. Calculate the other $N(a,y)$ backwards for each year class:
    If $N(a+1,y+1)$ >0 and $C(a,y)$>0, then $N(a,y)$ is computed by (eqn.2)

    $$N(a,y) = N(a+1,y+1)*exp(M(a,y) + C(a,y)*exp(M(a,y)/2) \qquad (2)$$

    If $N(a+1,y+1)$ >0 and $C(a,y)$=0, than $N(a,y)$ is computed by (eqn.2).
    If $N(a+1,y+1)$ =0 and $C(a,y)$=0, then $N(a,y)$ = 0.
    If $N(a+1,y+1)$ =0 and $C(a,y)$>0, then $N(a,y)$ is computed by (eqn.1).

4. Calculate the $F(a,y)$ as $F(a,y) = log\{N(a,y)/N(a+1,y+1)\}$ - $M(a,y)$.
    If $F(a,y)$ is undefined because N is 0, or $F(a,y)=0$ because $C(a,y)=0$, use $F(a,y) = (a)*F(y)$.

5. Calculate new selections at age $S(a)$:
    Age loop (except oldest age): $S'(a) = 1/Ny*\Sigma_y\{F(a,y)/F(y)\}$ using all years.
    The $F(y)$ are from the previous iteration.
    Selection at oldest age: $S(A) = S(A-1)$
    Age loop: Normalize the $S(a)$ to get $S'(a)$: $S'(a) = S(a)/ \Sigma_a S(a)$

6. Calculate new $Fy(y)$ as $Fy(y) =1/N_A*\Sigma_a[F(a,y)/S(a)]$ for all $y<Y$.
    The sum is over all ages a.

7. Derive smooth selections $Ssmooth$ as the weighted average between the new $S(a)$ and the $S(a)$ from the previous iteration:

    $S(a)smooth = K*S(a)new+(1-K)*S(a)old, 0<K <1$.
    The value of K is a compromise: a bigger K makes the convergence faster, a smaller K suppresses possible oscillations more strongly.

8. New $Fy(y)$ are calculated as a weighted average between $Fy(y)$ and the previous $Fy(y)$:
    $Fy(y)=K*Fy(y)new+(1-K)*Fy(y)old$

9. Calculate model catches from catch equation 3 using $F(a,y)=S(a)Fy(y)$

    $$C(a,y)= F(a,y)*N(a,y)*(1-exp(-Z(a,y))/Z(a,y) \qquad (3)$$

10. Calculate log squared differences: $[log\{C_{obs}(a,y)/C_{model}(a,y)\}]^2$ (or use some other measure of closeness of fit to catch-at-age data)

11. Compare log squared differences from the current iteration with the three last iterations. Convergence criterium: The maximum log squared difference difference is < 10E-5. If not converged, goto 1.

In Pt.5, different vectors of $S(a)$ for different intervals of years can be specified and applied in the respective years when estimating $Fy(y)$ in Pt.6 . The normalization and inter-iteration smoothing is applied to each vector of $S(a)$.

The procedure produces unbiased estimates of $F(a,y)$ when assuming a lognormal error distribution in the separable approximation, since:

$\sum_a\{logF(a,y)-log[Fy(y)S(a)]\}=0$ for each year y,

and

$\sum_y\{logF(a,y)-log[Fy(y)S(a)]\}=0$ for each age group a

This algorithm will render the $Fy(Yc)$ unchanged.

The result is a set of $F(a,y)$ conditional on the $Fy(Yc)$. The fit of the ISVPA to the catches does to some extent depend on the $Fy(Yc)$. However, leaving it fixed when the ISVPA is evoked is standard practise . The model fit to the catches can optionally be included in the overall objective function.

The two versions of ISVPA that are implemented in TASACS, differ in the way the final N-matrix is derived:

In the *catch controlled version*, the set of $N(a,y)$ is derived from the terminal F-values and the observed catches using the VPA algorithm. The final fishing mortalities are then derived from N-values with eqn (4).

$$F(a,y)= log (N(a,y)/N(a+1,y+1)) - M(a,y) \qquad (4)$$

Hence, in this version, the separable fishing mortalities are only used to derive terminal fishing mortalities, the rest of the N-matrix is generated by the VPA algorithm.

In the **effort controlled version**, the output N-values are model N-values (except for those at the end of each year class). Model catches are calculated from the catch equation (3) using the separable $F(a,y)=S(a)F(y)$.

Scaling of the fishing mortalities in ISVPA
The selection at age as stated in the input to ISVPA is rescaled internally when entering the ISVPA to get the sum of selections at age equal to 1. The $Fy(Y)$ given as a parameter to the ISVPA algorithm is also rescaled internally with the inverse of the scaling factor for the selections.

The $Fy(Yc)$ is the only free parameter that the ISVPA sees. Even though $Fy(Yc)$ is rescaled internally, the rescaling is consistent as the selections on the parameter list and in the parameter file are not updated by the ISVPA. Hence, despite the internal rescaling, the $Fy(Yc)$ can be applied as a free parameter.

3.2 Some special features in the population models in TASACS

3.2.1 Specification of population model parameters

Both the population models and the catchability models are basically specified through their structural assumptions (e.g. a separable assumption) and a set of model parameters.

Since TASACS is designed as a toolbox, with several optional population models, an important aim has been consistency of parameter definitions for the various models, despite the different requirements for parameters in the different models. This has two implications:

1. Some parameters will be redundant with some models. A 'relevance filter' in the program ensures that redundant parameters are not included in the optimisation procedure.

2. The definition of parameters is consistent across models. Hence, where N-values are needed as parameters, they represent survivors. The F-level parameters represent average F over the reference ages.

Active parameters and parameter flags

The number of parameters is large, to allow flexibility in the way the model is set up. Estimating every parameter is out of question with the information at hand. In TASACS, the analyst has to decide which parameters to assume at fixed values, and which to estimate. The latter are called 'active parameters'. The decision is expressed by flags attached to each parameter. Only those parameters that are flagged as active are seen by the optimization routine, and hence are actually estimated. It is possible to specify by a flag that the value of a parameter shall equal the similar parameter in the previous year or at the previous age. That allows for specifying constant values over time or age.

All parameters are listed on a parameter list, which is read from a parameter input file. The simplest way to generate this file is by using the editor in the TASACS environment, which ensures that the format of the parameter file gets right. Both values and flags can be set with the editor. During optimization and when the optimization is finished, the current parameters are dumped to a file named paramOut.txt. It is possible to specify this file as the parameter input file. That allows restarting the optimization with the best parameters obtained so far. Parameters that are irrelevant for the population model are not updated on the parameter list or the paramOut.txt file. For example, the selection at age generated by the ISVPA is not transferred to the file. Only parameters which are active, or copies of active parameters (see under flags below) and are relevant for the actual population model, are updated. Therefore, the paramOut.txt file does not necessarily show the current values of all parameters.

Interpretation of the active flags

Flag = 0   Keep the parameter fixed.

Flag = 1   To be optimized by the optimization routine (if it is relevant for the population model used)

Flag = 2   Use the same value as that for the previous parameter on the list.
For example, if this applies to a year factor, the parameter shall have the same value as the year before.

Flag = -1   Use the same value as for the previous age

This flag applies only to age factors in the specification of selections at age and catchabilities at age. This is the way to specify a 'flat' selection or catchability at age.

Flag = 3   Only applicable to Survivors at oldest true age in the VPA. Derive the survivors from F-values, as described in the VPA section 3.1.1.

3.2.2   Handling of zero catches

TASACS has facilities to handle the situation where catch numbers at age are zero. Catch observations with the value zero cause problems in at least two ways:

1. The link between Catch, F and N in the terminal age in the VPA algorithm breaks down if the catch is zero.
2. Some objective functions do not allow zeros (e.g. log SSQ).

In many assessment programs, the problem is fixed by substituting zeros with a small number. The actual value of the 'small number' does matter, in particular if the zero catches are common, as is the case with the NSSH data.

Instead, TASACS has a modification of the objective function, and a special handling of zeros in the VPA algorithm. The modified VPA is described here, the objective function is dealt with in Section 3.4.

The VPA with zero catches

The key to the handling of zero catches in the VPA in TASACS is to allow a fishing mortality to be undefined.

When a catch at age is zero, it can be because there was no fish or because there was no fishery (real or apparent because the catch was not picked up by the sampling). Hence, there may be two situations:

1. *No fish case: N1 = 0, i.e. N* at the end of the year at that age is zero. Then, with no catch we can assume that the *N* at the start of the year (*N0*) is also zero. The fishing mortality is undefined. When going backwards from there in the cohort there will be a last year where there was a non-zero catch. We assume that the last fish was taken in that year. Hence, in the last year with a catch in the year class we have *N0 = C*exp(M/2)* and *N1 = 0* and the fishing mortality is infinite. In TASACS, we leave the *F* undefined in this case.
2. *Fish present but no catch: N1 >0.* Then, according to Popes equation, *N0 = N1*exp(M)* and *F=0.* The information about *N1* must come from somewhere else.

In TASACS, case 2 is assumed if

a. There is catches later on from the cohort, or
b. *N1* at the last age is entered as a parameter with flag = 1 or 2. If so, survey information is allowed to determine the survivors number.

3.2.3 Extending the model beyond the last year with catch data.

A common situation is that there are survey data in the year after the last year with catch data. It may also be of interest to project the stock forwards to get estimates of the stock beyond the last catch data year, e.g. to get input for a projection. TASACS considers a final model year (Y) and a final catch data year (Yc), and has routines to fill in the N-matrix and F-matrix between these years, with slightly different algorithms for each of the population models.

VPA and ISVPA

Internally, these models are run with the last catch year (Yc) as the last year. When leaving the population model, the survivors in the last catch year $N1(a,Yc)$ and fishing mortalities at age $F(a,Yc)$ are available. From there, the stock is projected forwards until the end of year Y. The fishing mortalities are assumed to have the same selection at age as in year Yc, and a year factor $Fy(y)$ that is a parameter for each year $Yc+1 .. Y$. Hence,

$$F(a,y) = F(a,Yc)*Fy(y)/Fy(Yc)$$

Here, $Fy(Y)$ is taken from the parameter list while $Fy(Yc)$ is the average of the estimated $F(a,Yc)$ over the reference ages. If the Fy has flag = 2 in these years, the same fishing mortality as in the last catch year is assumed. It may also be estimated (flag = 1), if there is relevant information to do so. Using flag = 0 is a way of projecting the stock forwards with other fishing mortalities.

Separable model

Here, the population and fishing mortalities are generated by the parameters, so no special action is needed.

Objective functions

In the years after the last catch data years, the catch objective function terms are set to zero. Survey data are included in the objective function as relevant.

Parameters and flags for the years after the last catch year

VPA and ISVPA: Yearly F-factor and N at youngest age are needed. The N-values at youngest age are treated as recruits (i.e. numbers at the start of the year).
Separable: No special consideration.

3.2.4 Estimating the plus group

The stock numbers in the plus group are calculated independently of the population models, after the population model has been run. TASACS have two methods for calculating the stock numbers in the plus group:
- Direct from catch
- As a dynamic pool.

The user decides which method to use. However, when the stock is projected beyond the last catch year, the dynamic pool method always is used.

Direct from catch

Assuming that the fishing mortality at plus age is equal to that at the oldest true age, the stock number is calculated as

$$N0(A+1,y) = C(A+1,y)*Z(A+1,y)/F(A+1,y)/[1-exp(-Z(A+1,y))]$$

As a dynamic pool

Again, it is assumed that the fishing mortality at plus age is equal to that at the oldest true age. The plus group in each year is the sum of what is left over in the plus group from last year, and what is transferred from the oldest true age:

$$N(A+1,y) = N(A+1,y-1)*exp(-Z(A+1,y-1)) + N(A,y-1)*exp(-Z(A,y-1))$$

In the first year, the *N*-value is calculated from the catch.

3.3   Catchability models

The program allows for 3 types of survey observations representing relative abundance measures:

- Yearly indices of abundance in numbers at age.
- Yearly indices of spawning biomass
- Yearly indices of total biomass

Each of these is related to the stock abundance *V* through a catchability function.

I = Q(V) where *V* can be numbers at age or biomass at survey time.

To get abundance at survey time, the abundance at the start of the year is reduced with a fraction of the modelled mortalities that corresponds to the time of the survey. The fraction of mortality before the survey is input information, and separate for *F* and *M*.

The function that is implemented is the power function:

$$I = q*V^b$$

With *b=1*, this is the proportionality function, that is most commonly used. Setting both *b* and *q* equal to one, with flag = 0 implies that the index is taken as an absolute measure of the stock abundance.

Age structured indices *I(a,y)*

Catchability parameters are normally assumed to be dependent on age, but equal for all years. As input parameters, the catchability parameters *q* and *b* are nevertheless represented as separable by age and year, and specified as a year factor (which normally should be fixed at a constant value) and an age factor. Normally, the parameter flags should be set so that the catchability parameters in the first year are the active parameters, and these carry over to subsequent years. However, the age factor can be allowed to vary between years, to pick up changes in survey selectivity by setting the active flag to 1 in the year where it changes. This is equivalent to splitting the survey into two periods.

SSB indices *I_{SSB}(y)*

Catchabilities are normally assumed to be constant over all years, but are represented by year in the parameter file. As for age structured indices, a shift in catchability in some year can be indicated by setting an active flag in that year.

The abundance *V* is the model spawning stock biomass at spawning time, derived from stock numbers at age (reduced by mortality before spawning), weights at age in the stock and proportions mature at age. The mortality before spawning is accounted for when calculating the SSB - the timing of the survey as such is ignored.

Total biomass indices

These are analogous to spawning biomass indices, except that the biomass is calculated without taking proportions mature into account. The weights at age applied are the weights in the stock.

3.4 Objective functions

The objective function is the measure of model fit to observed data. The following objective functions are implemented:
- Weighted log sum of squares
- Median deviations
- AMD
- Gamma likelihood
- Gamma deviance
- Poisson-like

The objective function Φ is a sum of partial objective functions, each originating from one source of data. Each objective function is in turn the weighted sum of contributions $\phi(x, \hat{x})$ from single observations *x* and the corresponding model value $\hat{x}$:

$$\Phi = \sum_{sources} {}^* \Phi_{source} \text{ and } \Phi_{source} = \sum_i w_i \, \phi(x_i \, \hat{x}_i)$$

where w_i is the weighting of the ith individual observation, as described below. The w_i have to be set manually, iterative reweighting as in XSA (Shepherd 1999) is not implemented.

Weighted log sum of squares

$$\phi(x, \hat{x}) = w*\{log((x+\varepsilon)/(\hat{x}+\varepsilon))\}^2$$

The ε is introduced to handle the situation where x =0. Having this term in both the numerator and denominator maintains the minimum at $\hat{x} = x = 0$. The value of ε is input, and typically, it is given a value similar to the resolution in the data.

Median deviations

For each source, sort the $\phi(x_i, \hat{x}_i) = (log((x_i+\varepsilon)/(\hat{x}_i+\varepsilon))^2$

Φ_{source} is the sum of the X central elements. This is a non-parametric measure of closeness, which will ignore outliers. Typically, X=10 is used. A larger X gives a smoother function, but increases the risk of including outliers.

AMD: Median of the absolute deviations from the median of the residuals:

Find the median of the residuals $R(x_i, \hat{x}_i) = (log((x_i+\varepsilon)/(\hat{x}_i+\varepsilon))$ for each source.

Then, AMD = Median of all $|R(x_i, \hat{x}_i) - Median\ of\ R(x_i, \hat{x}_i)|$

Gamma likelihood

The gamma distribution function has two parameters, k (the shape parameter) and θ (the scaling parameter):

$$p(x) = x^{k-1} * \exp(-x*\theta) * \theta k/\Gamma(k)$$

The expectation value and the variance are both the product $k* \theta$. One of them (here k) is estimated as an additional free parameter (one for each index series) and the other follows from the model value $\hat{x}_i$ as $\theta_i = \hat{x}_i/k$.

The negative log likelihood for observing x_i when the model says $\hat{x}_i$ then becomes:

$$\phi(x_i, \hat{x}_i) = (k-1)ln(x_i) - kx_i/\hat{x}_i - k*ln(\hat{x}_i/k) - ln(\Gamma(k))$$

The log of the gamma function $\Gamma(k)$ can be calculated using Stirlings approximation:

$$ln(\Gamma(z)) \sim 1/2*(ln(2\pi)-ln(z)) + z*(ln[z+1/\{12z-1/(10z)\}]-1)$$

(Documented in http://en.wikipedia.org/wiki/Stirling%27s_approximation)

This formulation of the gamma likelihood follows the procedure in Sea Star (Tjelmeland and Lindstrøm 2005) that was used for assessing NSSH previously.

The Gamma Deviance function
is also implemented, as an alternative. This function is

$$\phi(x_i, \hat{x}_i) = -ln(x_i/\hat{x}_i) + (x_i - \hat{x}_i)/\hat{x}_i$$

This function is not a likelihood function, but it is a function with some convenient properties:
- It is 0 when $x = \hat{x}$
- It is steeper when $x > \hat{x}$ then when $x < \hat{x}$, which is opposite from the log SSQ.

Hence, it reacts more strongly to prevent unduly large x, and is more tolerant to to small x.

Poisson-like
This is a simplified version of the Poisson likelihood, and is similar to the gamma function with $\theta = 1$ and integer k. The term relating to $x!$ is left out, since this term is only dependent on the observations. The function ϕ can be used as a proxy in cases where the distribution is multinomial, as its derivative is the same as for the multinomial likelihood function. At large negative deviations, this function is almost flat, which slows down convergence markedly.

$$\phi(x_i, \hat{x}_i) = \quad \hat{x} - x + x*log(x/\hat{x}_i) \quad \text{if } x > 0$$
$$\text{if } x = 0$$

The figure below shows the shape of some of these functions.

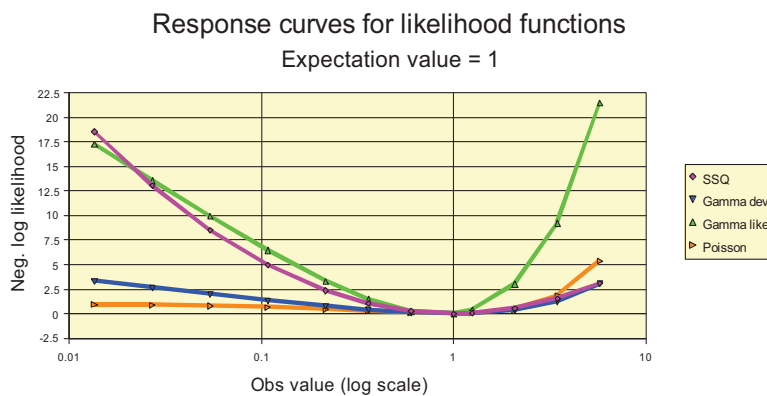Response curves for likelihood functions
Expectation value = 1

Figure 2. The shape of some objective functions in TASACS, as described above.
Green: Gamma likelihood with k = 1.
Orange: Poisson
Blue: Gamma deviance
Pink: log SSQ.

Weighting
The weighting is by weight factors set manually as part of the input. The weights are the product of two factors:
- A common weight given to each data source (e.g. the catches or each survey)
- An individual weight given to each observation

Unless there are external estimates of the variance of input data, it is suggested that they all are regarded as equally likely, leading to a weighting *=1* of the individual data. The weight given to each data source is best used as a means of including or ignoring that source, but may also be used to give proper emphasis on sources with aggregated data, like measurements of biomass.

3.5   Optimization

The optimization process is to find the set of active model parameters that leads to the best fit of the model to the data, i.e. to the lowest value of the objective function. As seen from the optimization procedure perspective, the objective function is a functional on the vector of active parameters. Hence, the optimization suggests a set of active parameters, calls the objective function routine to get the value of the objective function with the current parameters, and examines how the objective function now compares with that at previous attempts. The objective function routine in turn calls the population model and the catchability models to get the modeled observations according to the current parameters.

TASACS uses a simple searching routine to find the optimal set of active parameters. The routine is similar to that in AMCI (Skagen 2005). The searching routine is relatively slow, but has proven to be quite robust, and rarely gets stuck in local minima if the problem is well posed, i.e. if there is sufficient information in the data to estimate the active parameters.

The searching procedure starts with an initial set of active parameters, taken from the parameter input. Each parameter is then increased and decreased with a certain fraction *delta*, and the value of that parameter giving the lowest objective function value is kept. After going through all active parameters, the process is repeated, until a maximum number of rounds has been made, or there is no improvement from one round to the next. In each round, the sequence in which the parameters are examined is drawn randomly, to avoid being trapped by correlated parameters. If the maximum number of rounds is reached, the *delta* is increased. If there is no improvement from one round to the next, the *delta* is decreased. The process is repeated until there is no improvement even at a very low *delta*. The starting value for *delta*, the increase and decrease of *delta* and the maximum number of rounds are set based on experience. At present, starting *delta* is 0.25, *delta* increase factor is 2, *delta* decrease factor is 0.4, number of rounds is 25 and the convergence criterium is *delta* < 0.0001.

Since all changes in parameter values are with a multiplier, the searching process gets trapped if the parameter value is zero. Hence, active parameters should not have zero as initial value.

For each round, the value of delta and the objective function are shown on the screen, and a graphical representation of the *N*-matrix and the *F*-matrix is displayed. Typically, if the information to estimate all parameters properly is insufficient, the search is slow, and goes back and forth between increment and decrement of *delta*, with very small improvement of the objective function for each round. Often this process leads towards trivial optima, i.e.

stock numbers go towards infinity and F-values towards zero. If this happens, the choice of active parameters should be re-considered.

4　Procedures that use the TASACS assessment program

4.1　Population model as it stands

With this option, the population model is simply run once with the parameters as they stand. Results are produced. No optimization is performed.

4.2　Ordinary assessment runs.

This is the normal procedure to perform an assessment. This procedure just calls the TASACS assessment program and runs it, i.e. it estimates the active parameters by optimization.

4.3　Bootstrap runs

The TASACS environment can perform bootstrap runs with stochastic observations data. The following options are available:

- **Non-parametric** around the **observed** data.
- **Parametric** (with a log-normal distribution with specified CV) around the **observed** data
- **Non-parametric** around the **modeled** data
- **Parametric** (with a log-normal distribution with specified CV) around the **modeled** data.
- Replicas of data read from a file.

Options can be chosen individually for each source of data, i.e. for the catch numbers at age and for each of the survey index series.

Non-parametric data are generated by drawing randomly (without replacement) from the collection of unweighed log residuals for that source of data (excluding data with zero weight), and generating new data as

$$Obsnew = Obsorig*exp(log\_residual)$$

Parametric data are generated as

$$Obsnew = Obsorig*exp(\xi+\sigma^2/2),$$

where ξ is a normally distributed random number with mean 0 and specified σ.

Reading data from a file is intended for methodological studies, where e.g. raw survey data are re-sampled and carried through the aggregation process leading to the indices used by the program. The data sets are read in the sequence they stand. If the number of data sets is too small, the file is read over again.

The number of bootstrap replicas is specified by the user, and the results are output to files for further processing. Stock numbers and fishing mortalities at age for the last year can be written to files for use in subsequent stochastic predictions.

4.4 Retrospective runs

Retrospective runs are assessments where the *n* last data years are excluded, to show how the assessment outcome changes as recent data are removed. When *n* years are to be excluded, TASACS does this by removing the *n* last years from each set of data, and moving the parameter flags for the last years *n* years backwards. Both the last catch data year and the last assessment year are moved. This procedure is not fool-proof, but it ensures that e.g. surveys performed after the last catch data year still are treated as such, and it ensures that the adequate parameter flags are used.

The results are output as for the ordinary single runs, and can be post-processed to illustrate the results as wished.

5 Post-processing

The post processing routines are written in Delphi (Pascal). These are stand-alone programs that are called from the TASACS working environment. The only link to the other parts of TASACS is through the files mentioned below.

5.1 Output files

The output from a single assessment run is on 2 files:
1. Standard report file (Summary.txt) , with tables of *F(a,y), N(a,y)* and a summary table with annual Recruitment, TSB, SSB, mean F and reported catch in weight.
2. Output file (Output.txt): Has all the remaining information that is needed for post-processing:

For each data source:
- A table of observed values
- A table of modeled values.
- A table of weightings
- A table of Phi-values (contributions to the objective function from individual data)
- A table of observed values converted to *N*-values

In addition, bootstrap runs and retrospective runs generate files with the main results.

The output files are palced in the Working run directory and are ascii files in a standardized table format.

5.2 Graphical output

The TASACS package has a set of graphical routines, to display the results and some diagnostics.

5.2.1 Output from the summary table (Figure 3)

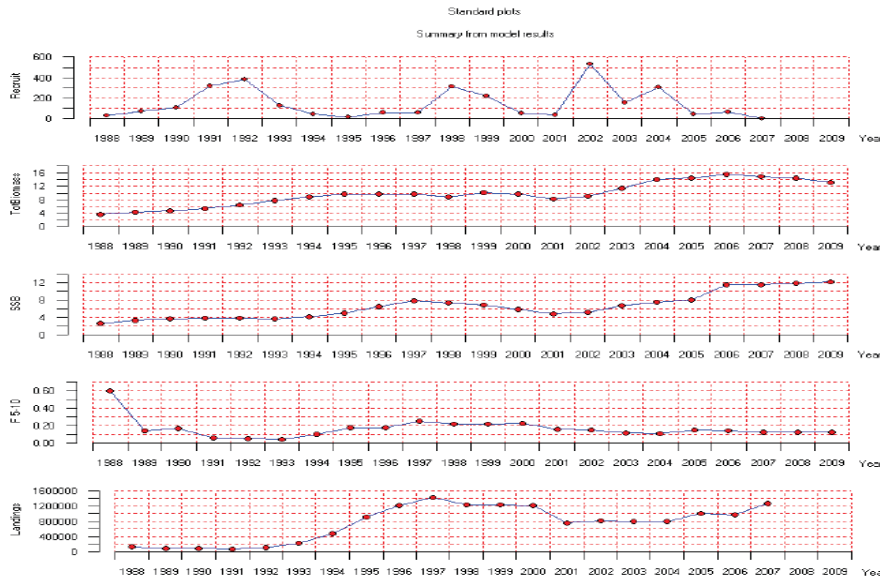This is a set of 4 graphs showing the time course of Recruitment, SSB, F and Catch

Figure 3. Standard plots generated by TASACS.

5.2.2 Residual plots and phi-plots (Figure 4)

Standard residual plots are bubble plots of the log ratio $ln(x(a,y)/ \hat{x}(a,y))$ between each observation x and its modeled counterpart $\hat{x}$. Both unweighted and weighted residuals are shown. In the same plot, there is a display of the ϕ-values which are the actual contributions to the objective function from each observation. Finally, the weightings of the observations are displayed as bubbles. There is one set of figures for each source of data.
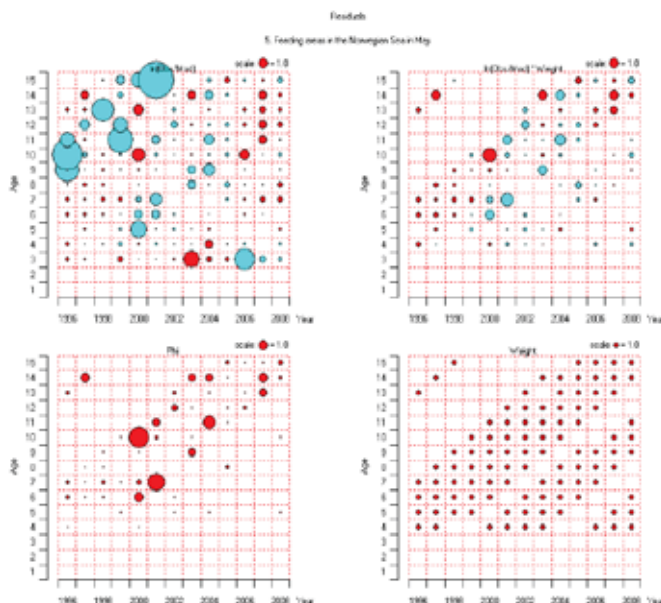
Figure 4. Residual plots generated by TASACS.
Upper panels: Log residuals: Left: unweighted, Right: weighted.
Lower left: Individual contributions to the objective function. Lower right: Display of the applied weightings.

5.2.3 Comparison of distributions of modeled and observed values

Q-Q plots (Figure 5:

These are plots of the cumulated distribution of observed and modeled values. Let x(i)) be the ith value of the cumulated distribution of the observations x_i, $i=1..n_{obs}$, and $\hat{x}_i$. the model value representing this observation. The Q-Q plot is an x-y plot of the x_i versus the $\hat{x}_i$. Plots are made separately for each source of observations.
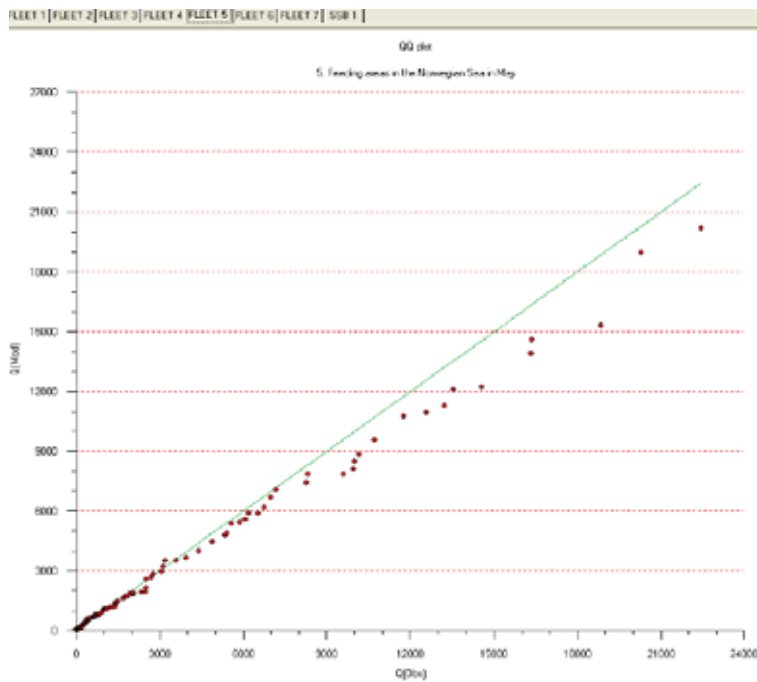

Figure 5. Q-Q plots.

An adaptation of Kolmogorovs test for goodness of fit (Kolmogorov plots - Figure 6):
Kolmogorovs test is in itself a test on the deviation of the empirical cumulated distribution of observations from an assumed parametric distribution. In our case, we regard the modeled values as the parametric model. Their distribution is complex at best, but we have discrete values that have been calculated, and we want them to be as close to the observations as possible. The test looks at the whole ensemble of observations as a distribution, which we also do here.

Let $F(x)$ be the cumulated distribution of x_i, $i=1..n_{obs}$, i.e. $F(x) = 1/n_{obs}*\{number\ of\ x_i < x\}$. Let $F_{mod}(x)$ be defined the same way, on the set of modeled values $\hat{x}$. Then, the Kolmogorov statistic is

$$D = sup_{\ x \in (-\infty,\infty)} |F(x) - Fmod(x)|$$

Procedure:
For each category of observations (i.e. catch numbers at age and year, indices from one survey etc. sort the modeled values. For each of the model values $\hat{x}_i$ (the number i in the

row), count the number j of observed values x that are smaller than $\hat{x}_i$, and note the value $|i/n_{mod} - j/n_{obs}|$. The criterion is the largest of these values.

The plot shows for each observation, the rank *i* of the observation vs. the rank *j* of the corresponding modeled value.
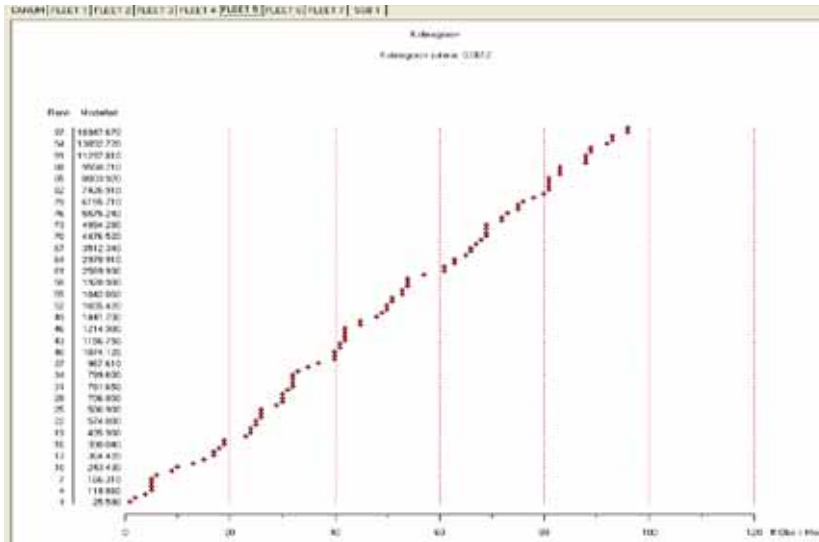

Figure 6. Kolmogorov plots.

5.2.4 Year class plots

These are plots of all information about each year-class, assembled in one graph.

Year class trends (not shown) are plots of all information scaled relative to the largest observed value for that data source in the year class, to give an overview of the available information.

Year class Ns (Figures 7a,b) are plots of all age-disaggregated catch and survey information, translated into N-values by using the current estimates of catchabilities and (for catch observations) of fishing mortalities. Observations with zero weight can be hidden in the plot. This tool has proven to be very useful to uncover conflicts between data from various sources, and the impact of the various data on the final estimates of the population.

5.2.5 Bootstrap output

The plots showing bootstrap results (Figure 8) has 4 panels:
- SSB: 10, 50 and 90 percentiles, and the SSB in the original run.
- F level: 10, 50 and 90 percentiles, and the SSB in the original run.
- The cumulated distribution of objective function values, with the objective function value of the original run indicated.
- A banana-plot with F-level and SSB in the last model year for each replica.
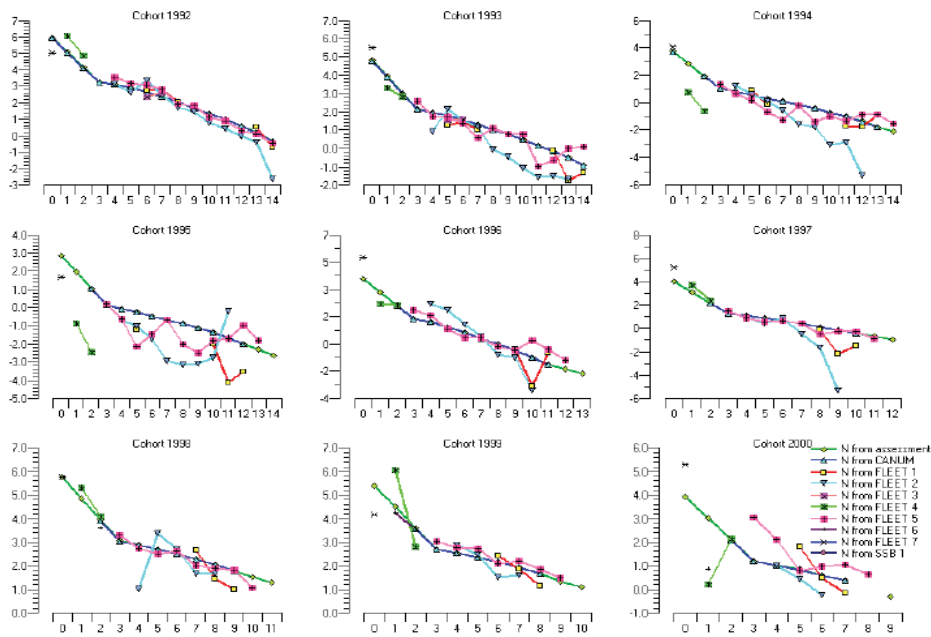
28

Figure 7a. Plots of all data for each year class, converted to N-values. Yearclass 1992-2000
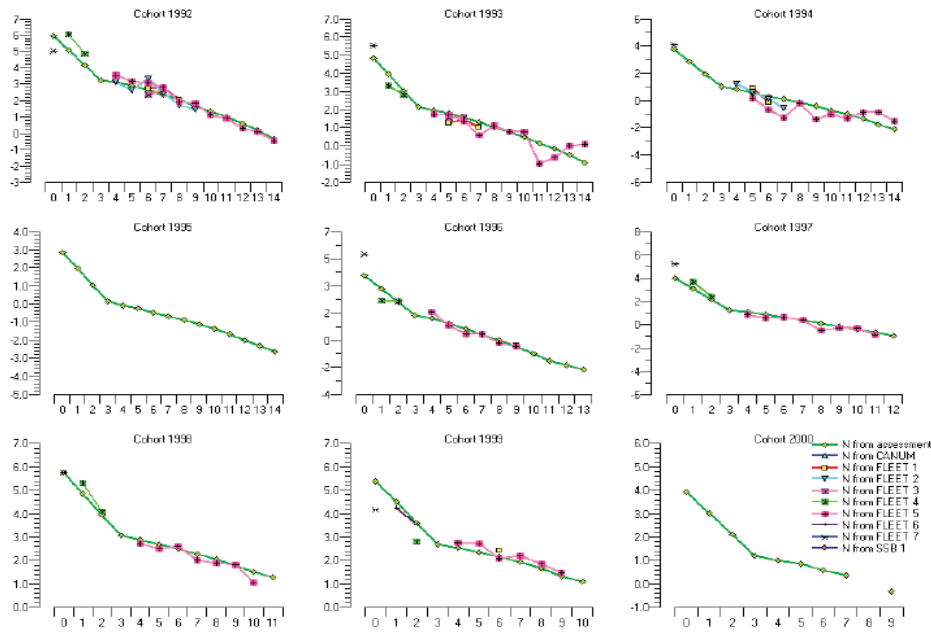

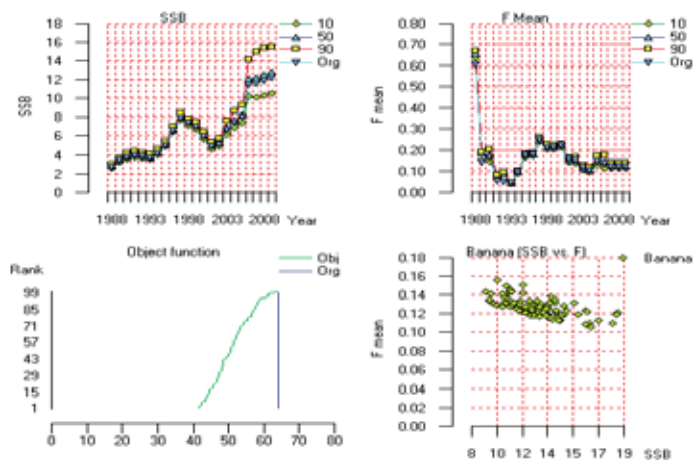**Figure 7b.** As figure 7a, but with data with zero weight removed. Yearclass 1992-2000.

Figure 8. Graphical display of bootstrap results

5.2.6 Retrospective output (Figure 9)

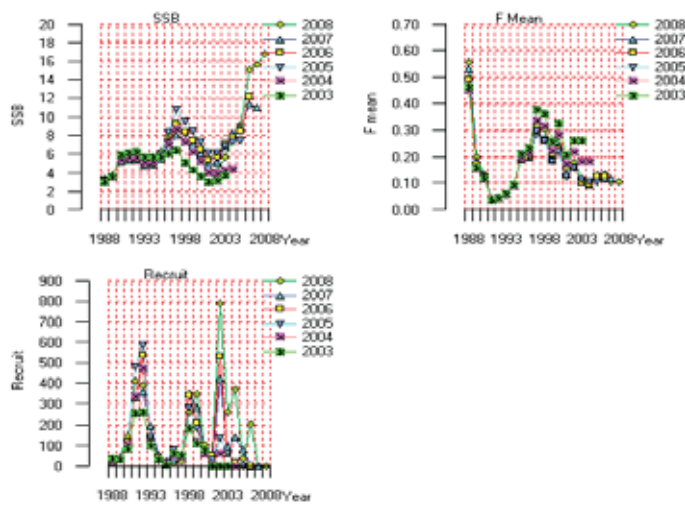Bundle plots showing F-level, SSB and recruitment for each retrospective year.

Figure 9. Retrospective plots.

6 The TASACS working environment

6.1 Projects and runs

The TASACS environment organizes the work in projects and runs. A project is a folder that contains all relevant files relating to one set of data. A project is selected when TASACS is started. The project is linked to the observations data, and it is suggested that a new project is made if these data are revised. The data are in a ***Data*** subfolder.

A run is one specific implementation of TASACS. There is a ***Working run*** subfolder under the project, which holds all files specifying the run, e.g. parameters, weightings, model specifications etc. These files can be edited from the run properties panel (Section 6.5). The Working run folder also holds all results files from that run.

Model settings can be changed in the Working run. A working run, with all settings and results can be archived in the ***Runs archive sub***folder, where each archived run can be given a specific name. By moving an archived run into the Working run, the archived run can be reproduced.

Hence, the working environment is the Working run, and interesting runs can be archived. The purpose of this arrangement is to facilitate book-keeping and avoid confusing runs in the explorative phase of an assessment.

6.2   Data editors

The data are on ASCII files according to the Lowestoft file format. As this format may be difficult to edit in a text editor, TASACS has an editor with a spread-sheet like layout which can be used to control and edit data files (Figure 10). This editor is linked to an editor for weightings of individual observations, and a color code is used to keep track of weightings. The weightings themselves are stored on files that are part of the run specification, but are edited together with the data for practical reasons.

Figure 10. The TASACS data editor panel, here editing a set of age structured survey indices.

6.3   Parameter editors

Parameters are edited with a tool similar to that for the data. Active flags are edited together with the parameters, and indicated in the parameter panel with colors.

6.4   The run properties panel

This is the place to select options for the run. The panel is shown in Figure 11.

To the left, there is a listing of the files that are used for specifying the run, and below are the names of the output files.

The index panel as shown here allows setting years and ages, type of assessment model, some model options and the task to be performed. The lower left part is for choices for each data source, like timing of surveys bootstrap specifications, the choice of objective function applied for this type of source etc. Weights here are common weightings for all data from the source.

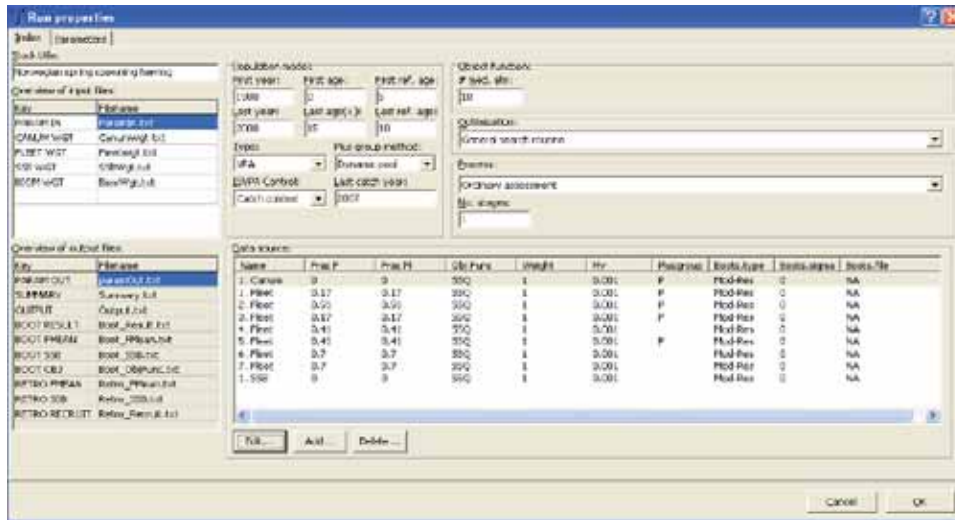The parameter tag leads to the parameter editor.

Figure 11. The Run properties panel.

6.5 The run progress panel

During the optimisation, the searching prcess can be followed, but as values of the objective function and the steps in the earch, and the cuurent N- and F- matrices displaued graphically. An example is shown in Figure 12.
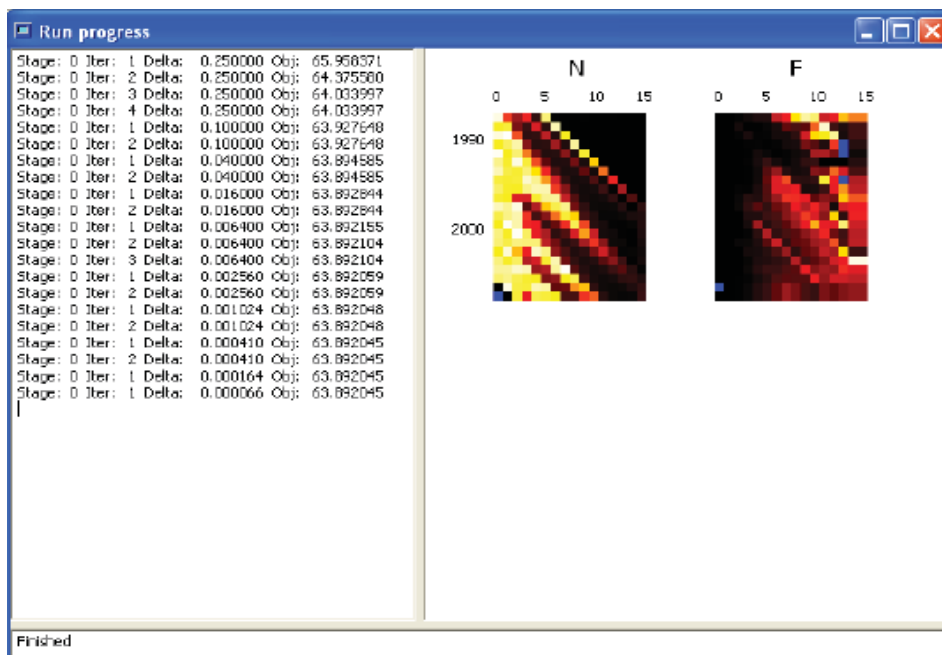

Figure 12. Example of a run progress panel after finishing the optimisation.

7. Discussion

The TASACS toolbox was made because of a specific request. Part of the request was that the development should be in cooperation between Norwegian and Russian scientists. Accordingly, the overall design, as well as details in algorithms were formulated in meetings between scientists from the two nations. This cooperation was successful, and led to a mutual understanding of what should be done.

The coding of the program was done by the authors at IMR, according to agreed specifications. The working environment was largely developed at IMR, but approved in subsequent meetings between Russia and Norway. Some of the software, in particular the graphical routines utilizes routines made previously for other projects.

The program was first used for an ordinary assessment for NSSH at the ICES WGWIDE in Spetember 2008. This assessment was subsequently approved by ICES.

The overall philosophy has been to develop a versatile package that is easy to use, that allows the user to understand the links between data and results, and gives some flexibility with respect to model conditioning.

Data and parameter editors have been included, to reduce the risk of mistakes in the input. By experience, such mistakes easily occur, in particular when editing on 'cryptic' file formats. The interface also encourages the user to excert some book-keeping disipline in the trial and error phase, by organizing the work in projects and runs and have easy facilities for archiving and retrieving of runs.

It has been important to have readily accessible diagnostics, which not only may indicate problems, but also identify their source and their effect. Some of the diagnostics are standard in any assessment package, for example plots of residuals, while others, like the 'Year class N-plots', are not.

The assessment methods themselves are relatively standard.

The VPA as it normally will be set up is close to ADAPT (Gavaris 1988). It does not have facilities like shrinkage and iterative reweighting as for example XSA (Shepherd, 1999).

The separable population model (Fournier and Archibald, 1982) is standard, but allows a good deal flexibility with respect to changes in selection over time and shaping of the selection at age. The combination of a separable model and a VPA, as in ICA (Patterson and Melvin, 1996), is not available. Neither is there any mechanism for gradual change in the selection over time, as in AMCI and time series analyses.

The ISVPA-type models reproduces the original ISVPA (Vasilyev, 2005), but is restricted to a limited number of versions.

The catchability models are also the commonly used ones, but as for the fishing mortalities, multiple periods with different catchabilities can be defined.

The objective functions include the commonly used log SSQ, and some others. Some of these were included at the request of scientists taking part in the design of the program. Altogether, the collection of objective functions allows for a range of assumptions about the error structure in the data.

In TASACS, the objective functions are treated strictly as measure of closeness between modelled and observed data, without any statistical justification. Accordingly, there is no attempt to estimate variances or to weight observations according to inverse variances. This may be considered as a weakness. On the other hand, given the quite complex error structure in most of the data, estimates of variances with simple assumptions about error structure may easily become misleading. In addition, one may argue that if the choice of likelihood function has a large impact on the results, this is a data problem rather than a model problem, and should be solved accordingly. Nevertheless, if there is external information about proper weighting of data, such weightings can and should be applied.

The optimization is by a simple searching routine, taken over from AMCI. The choice was made for several reasons. One was the experience from AMCI that this routine is extremely robust, and not dependent on starting close to the optimum. Secondly, to make the program code open, we wanted to avoid proprietary software that would require licenses to use. Various Newton type routines, coded from 'scratch', tended to fail at times. That may be because of poor solutions of technical problems, but also illustrates that such routines have to be used with some care. In future versions, new attempts may be made. One drawback with not using Newton routines is that the Hessian matrix, which is a by-product of such routines, and the inverse of which provides estimates of variances of the parameters, is not generated.

Instead of deriving uncertainties from the inverse Hessian, bootstrap routines are provided, both parametric and non-parametric. It is also possible to read bootstrapped input data generated externally, for example by re-sampling of raw survey information. This option has proved useful in e.g. analysis of the impact of various kinds of errors in the observations data. The bootstrap estimates of uncertainties are of a different nature from the uncertainties derived from the inverse Hessian, as they may reflect directly the uncertainty in the input data. They do not inform about how well the parameters can be estimated with the information that is present, as is the case with variance estimates from the inverse Hessian.

The input data for which TASACS provides objective functions is restricted to age-disaggregated catch and survey data, and aggregated survey estimates of biomass. Routines to handle other kinds of information, like tag return data, are not provided. Neither is TASACS designed for fleet-disaggregated or area-disaggregated information. Other models can handle such information (AMCI, Gadget etc). Rather than trying to reproduce these methods, we concentrated on making a useful package for analyzing the standard types of data.

8   References

Fournier, D.,Archibald, C.P. (1982) A general theory for analyzing catch at age data. Canadian Jounal of Fisheries and Aquatic Science, 39: 1195-1207.

Gavaris, S. 1988. An adaptive framework for the estimation of population size. Canadian Atlantic Fisheries Scientific Advisory Committee Research Document No. 88/29: 12 pp.

Patterson, K.R., Melvin, G.D. 1996. Integrated catch at age analysis, version 1.2. Scottish Fisheries Research Report 58, 60 pp.

Pope, J. G. 1972. An investigation of the accuracy of virtual population analysis using cohort analysis. ICNAF Research Bulletin, 9: 65-74.

Shepherd, J. G. 1999. Extended survivors analysis: An improved method for the analysis of catch-at-age data and abundance indices. ICES Journal of Marine Science, 56: 584-591.

Skagen, D.W. 2005. AMCI: Model description, Instruction for installation and running, File formats. Unpublished manual available from the author.

Tjelmeland, S., Lindstrøm, U. 2005. An ecosystem element added to the assessment of

Norwegian spring-spawning herring: implementing predation by minke whales. ICES Journal of Marine Science, 62: 285-294.

Vasilyev D. 2005. Key aspects of robust fish stock assessment. VNIRO Publishing, 2005, 105 pp.

HAVFORSKNINGSINSTITUTTET
Institute of Marine Research

Nordnesgaten 50 – Postboks 1870 Nordnes
NO-5817 Bergen
Tlf.: +47 55 23 85 00 – Faks: +47 55 23 85 31
E-post: post@imr.no

HAVFORSKNINGSINSTITUTTET
AVDELING TROMSØ
Sykehusveien 23, Postboks 6404
NO-9294 Tromsø
Tlf.: +47 77 60 97 00 – Faks: +47 77 60 97 01

HAVFORSKNINGSINSTITUTTET
FORSKNINGSSTASJONEN FLØDEVIGEN
Nye Flødevigveien 20
NO-4817 His
Tlf.: +47 37 05 90 00 – Faks: +47 37 05 90 01

HAVFORSKNINGSINSTITUTTET
FORSKNINGSSTASJONEN AUSTEVOLL
NO-5392 Storebø
Tlf.: +47 55 23 85 00 – Faks: +47 56 18 22 22

HAVFORSKNINGSINSTITUTTET
FORSKNINGSSTASJONEN MATRE
NO-5984 Matredal
Tlf.: +47 55 23 85 00 – Faks: +47 56 36 75 85

AVDELING FOR SAMFUNNSKONTAKT
OG KOMMUNIKASJON
Public Relations and Communication
Tlf.: +47 55 23 85 00 – Faks: +47 55 23 85 55
E-post: informasjonen@imr.no

www.imr.no

HAVFORSKNINGSINSTITUTTET
INSTITUTE OF MARINE RESEARCH