CCSBT-MP/0304/06

CSIRO
MARINE RESEARCH

RESULTS FROM INITIAL TESTING OF SOME CANDIDATE MANAGEMENT PROCEDURES FOR SOUTHERN BLUEFIN TUNA

Tom Polacheck
Dan Ricard
Paige Eveson
Marinelle Basson
Dale Kolody

Prepared for the CCSBT 2nd Management Procedure Workshop
7-9 and 13-14 April 2003, Queenstown, New Zealand, CCSBT-MP/0304/06

Table of Contents

List of Appendices

Abstract

Results are presented from testing a large number of candidate management procedures for the SBT stock based on eleven fundamentally different decision rules. Eight of the rules involve feedback mechanisms, while the three that do not are presented as reference cases to assist in the evaluation process and to understand the limits in performance that can be achieved under the set of initial operating model scenarios. Initial testing of candidate management procedures indicates that substantial improvement in average performance could be achieved by adopting a feedback approach. Within any decision rule, a wide range of performance is achievable in terms of the trade-off in catch versus stock status (e.g. rebuilding) by varying the tuning parameters. Similar average performance can be achieved from two different decision rules, but with quite different performance within particular operating model scenarios.

Nevertheless, given the wide range of uncertainty about the SBT stock dynamics embedded in the initial trials, there appears to be substantial limits to what can be achieved from feedback rules to both ensure that the CCSBT rebuilding target is met where possible (or at least that some rebuilding of the stock occurs) and simultaneously ensure that catches are not unnecessarily foregone. Recommendation on a choice of management procedure will require clarification of management objectives in terms of robustness and risk combined with a process of evaluation that takes into account these objectives and the plausibility of different scenarios.

Introduction

The agreed approach for the development and evaluation of a management procedure for the stock of southern bluefin tuna (SBT) by the Commission for the Conservation of Southern Bluefin Tuna (CCSBT) involves the simulation testing of a range of candidate management procedures. The testing process involves the development of operating models which simulate both the underlying dynamics of the population and fishery, and the data from the fishery that can be used by the management procedure for setting future TACs. The operating model is used to project what future catches and population sizes would be under the application of a particular management procedure. A range of operating model scenarios is used in the testing process to represent the underlying uncertainty about the stock and the observational errors in the fishery data. The specific dynamics of the stock and fishery in an operating model scenario are unknown by a candidate management procedure. Thus, by testing procedures against the set of scenarios, their performance in the face of uncertainty can be evaluated. For the purpose of comparing performance, performance indicators of the stock and fishery (e.g. total catches, spawning biomass in relationship to some reference level, etc) are defined and calculated from the simulation projections.

At the CCSBT Stock Assessment Group meeting, a set of 8 operating model scenarios were defined for the initial testing of candidate management procedures for SBT (Anon 2002)[1]. Performance indicators for this initial testing were also defined. Computer code which implemented these operating model scenarios, and which performed the actual projections was developed by Vivian Haist (Haist et al. 2002,

[1] At the 2002 CCSBT SAG meeting a ninth scenario involving an MCMC set of operating model scenarios was defined, but this scenario was not implemented in the initial set of test scenarios which were distributed.

personal communications). The software developed was generic in the sense that it could be interfaced with user specified management procedures. This software was distributed to members of the CCSBT Scientific Committee to allow a range of candidate management procedures to be developed and their performance tested with the initial set of operating models. We have developed a number of candidate management procedures in order to explore how different types of information and feedback mechanisms affect trade-offs in performance. This paper presents results from initial testing of these procedures.

Methods

Eleven different decisions rules were defined as the underlying basis for candidate management procedures. A decision rule is defined here as a basic algorithm that can be used to determines the TAC in the next year given available information (e.g. past catches, CPUE trends, etc.). All of the decision rules considered here have “control” or “tuning” parameters that need to be specified before they can actually be used. These determine the actual TAC given the algorithm and the available information. We define a management procedure to be a fully specified decision rule (i.e. specific values assigned to the tuning parameters). Thus, for any general decision rule there are potentially an infinite number of possible versions or candidate management procedures depending upon the specific values of the tuning parameters. For example, setting the TAC to a constant value constitutes the simplest decision rule that one might consider. In this case, there would be one tuning parameter (i.e. the actual constant value for the TAC) and any specific value for this constant would constitute a candidate management procedure.

A generic feature built into all of the decision rules that we developed was a tuning parameter that controlled the frequency with which the TAC could be changed and a tuning parameter that controlled the maximum change in the TAC allowed between consecutive years. In nearly all the testing that we performed, we set the frequency to be yearly and the maximum level of change to be 20%. Therefore, unless specifically noted, all of the results presented here incorporate these constraints. Fixing these constraints provided a common basis for comparisons. However, we anticipate that at a later stage in the testing process it may be worthwhile to vary these constraints. Also, in all our initial testing we have maintained a constant distribution of the TAC among fisheries at the current level. This was done to facilitate comparison of the basic performance of different decision rules. In addition, there was little basis for evaluating performance if this was varied without any guidance from the Commission on what might constitute an appropriate performance measure.

The eleven decision rules were given the following names:

1. Const (constant catch)
2. Incr (continuous increase)
3. Decr (continuous decrease)
4. CPUE (trend in CPUE)
5. CPUE_age (age-based trends in CPUE)
6. Mean CPUE
7. Stinky (CPUE-based decisions with CPUE-based reference points)
8. Fox
9. Fox_Var
10. Kaltac

11. Comp_FC (composite CPUE and Fox)

Rules 1-3 involve no feedback. They are based on either constant catch or steadily increasing or decreasing catches. They have been included because they provide useful references for evaluating the performance of rules with an actual feedback component and also for understanding the limits in performance that can be achieved under the set of initial operating model scenarios. In particular, the constant catch decision rule provided a standard for evaluating the degree to which the feedback control mechanism incorporated into a particular decision rule improved (or degraded) performance. Rules 4-7 are empirically CPUE based rules in that they use changes in CPUE as the primary information for adjusting the TAC and do this outside of any underlying population dynamics model. Rules 8-10 are "model" based in that an underlying population dynamics or assessment model is fit to the available information and the estimated model parameters are used to determine the TAC. Rule 11 is a composite rule in that it combines the TAC specified from two or more basic decision rules into a single TAC. Detailed descriptions of all rules are contained in the appendices.

Decisions rules were tested using the projection software developed and provided by Vivian Haist for this purpose. Decision rules were tested over the first three uncertainty hierarchies defined by the initial management procedure workshop (Anon 2002a). The graphical summaries and associated software developed by Eveson and Ricard (2003) were used to evaluate the performance of a decision rule as the tuning parameters were varied and to compare results between different decision rules.

Results and Discussion

The performance of each decision rule was explored over a range of tuning parameter values, and a summary of the results is presented in the appendix in which the rule is described. In the current paper, we have not attempted to provide an overall ranking of the performance of the different rules that we have considered. This is because the basis for such a ranking across the range of operating model scenarios has not been decided upon and is likely to be dependent upon both the general approach and the specific criteria adopted for synthesising results (see Polacheck and Kolody, 2003). Instead, we limit our discussion here to some general observations and conclusions.

As expected, variation in the tuning parameters yields a negative trade-off in performance between catch indicators and biological status indicators (i.e. higher catches yield lower stock status). This was true both within a given operating model scenario and when averages were calculated across scenarios (see the various summary plots within each appendix). The trade-offs tend to be smooth and small changes in the catch performance indicators yield approximately linear changes in stock status indicators (over the range of interest). In general, at low catch levels, the degradation in stock status performance as catches increase tends to be small (i.e. relatively large gains in catch are associated with relatively small decreases in the stock status indicators). However, at higher catch levels, the stock status indicators tend to decrease at a proportionately greater rate. Differences were found in how different rules make this trade-off with each of the scenarios.

Decisions rules that involved a feedback component in setting the TAC tended to result in improved performance when averaged over all operating model scenarios relative to the constant catch case. Examples of this can be seen in Figure 1. In this figure, the performances of the Kaltac and Stinky decision rules are compared to a constant catch decision rule in which all three decision rules were selectedto give approximately the same average catch. In this case, the average performance of the two feedback rules in terms of the stock status biomass indicators was substantially higher than that achieved by the constant catch decision rule. Note, however, that performance in terms of variability in TAC was worse for the feedback rules, as it would have to be, given that there is no variability in catch in the constant case.

The average performance of the two feedback candidate management procedures in Figure 1 in terms of stock status are approximately equal. However, they achieve this in different ways. Both procedures reduce catches for the least productive scenarios (h3M10 and h3M15) on average to the same extent and also increase catches to the nearly the same average extent for the most productive scenario. However, a lot more variability exists between the two procedures for scenarios with intermediate productivity (i.e. with a steepness of 0.60). In particular, the Kaltac procedure reduces catches on average for the h6M15 and h6M15d1 scenarios while Stinky increases them. For these two scenarios, Kaltac appears “overly” conservative in that higher catches could have been taken and the CCSBT recovery goal still met. Note, however, that this is compensated for in the overall average with Stinky being relatively more “conservative” than Kaltac on other scenarios.

It is important to note that although two candidate procedures can yield the same performance in terms of average catch and overall stock status indicators, they can achieve this in different ways in terms of the catch and spawning stock biomass time series. This is illustrated in Figures 2 and 3 in which worm plots (time trajectories) for catch and SSB are plotted for the h3M10 scenario for the versions of Stinky and Kaltac illustrated in Figure 1. In this case, both decision rules yield very similar overall performance. However, by the end of the 20-year period Kaltac has “learned” that this is a low productive stock and reduced catches in all runs to low levels (arguably too low). In contrast, Stinky has a large amount of variability in the catch trajectories between runs. In some cases, catches are increasing at the end of the time series while in others they are decreasing. Similarly, in all of the runs with Kaltac, SSB is increasing at the end of the projection period. In contrast, Stinky has a much wider range of performance with respect to SSB, with the stock collapsing in some runs.

The results from these initial attempts at developing a management procedure for SBT have demonstrated the difficulty in developing a procedure that can provide good performance with respect to catches and still ensure “acceptable” behaviour with respect to the stock status indicators over all scenarios. Ideally, a decision rule will yield appropriately low catches for low productive scenarios and high catches for productive ones. (In terms of the summary graphics (e.g. Figure 1), if a unique rebuilding target could always be attained, the stars in the left hand panels would ideally be flat with the rays approaching horizontal.) However, achieving this ideal has proved illusive and procedures tend to be either too “conservative” with respect to the highly productive scenarios or too “aggressive” in the low productive scenarios. “Conservative” is used in the sense of overshooting (in some cases substantially) the CCSBT rebuilding target and thus having foregone catches. “Aggressive” is used in the sense that the stock did not recover to the rebuilding target and/or the final SSB was below its current level.

In our initial testing, there was not a clear distinction in performance among empirically based CPUE decision rules or among rules derived from fitting a population dynamics model (e.g. Kaltac and Stinky provide similar average performance). In our initial attempts, we have made only limited use of the information on the age distribution of the catches. As such, the potential for the catch at age data to improve performance is not clear, particularly in the context of a population dynamics model. Similarly, we have only performed limited testing of composite rules. In particular, it may be worth exploring composite rules that are a non-linear combination of the TACs resulting from other rules. Time did not permit further development of either alternative catch at age based or non-linear composite decision rules. We think that further development and exploration of additional types of decision rules would be worthwhile.

Results from the continuous increasing and continuous decreasing rules (Appendices 2 and 3) provide some insight into the problem of designing a decision rule that will yield “good” performance across all scenarios. Thus, in order for the low productivity scenarios to not result in SSB being below its current level in 20 years, substantial cuts in catches relative to current levels are required. For example, if no cut in catch is made during the first five years, a cut of ~10% in each of the subsequent 15 years is required in deterministic runs of the scenarios with steepness 0.3 to ensure that SSB at the end of the 20 years is no lower then its current level. Similarly, for the high steepness scenarios, if catches are held constant for the first five years, an increase of over 10-15% is required for the SSB in 2020 to not overshoot the 1980 rebuilding target. Thus, unless reasonably decisive and consistent TAC trajectories are initiated soon, it is unlikely that a candidate management procedure will be able to yield “good” behaviour across all scenarios. However, the information available to a candidate management procedure in the first few years would be quite similar across different scenarios, particularly if no large changes in TACs occur (Figure 4-8). Note, however, that in the current implementation there is a discontinuity in the simulated CPUE series in the first year of the projection (i.e. between 2002 and 2003) such that each scenario has a distinct value which appears to be related to the productivity in that scenario[2].

In conclusion, initial testing of candidate management procedures indicates that considerable improvement in average performance can be achieved by adopting a feedback approach. However, given the wide range of uncertainty about the SBT stock dynamics embedded in the initial trials, there appears to be substantial limitations in simultaneously ensuring that the CCSBT rebuilding target is met (or at least some that some rebuilding of the stock occurs) and that the resource is not under utilized in terms of catch. Recommendations on a choice of management procedure will require clarification of management objectives in terms of robustness and risk combined with a process of evaluation that takes into account these objectives and the plausibility of different scenarios.

---

[2] This discontinuity appears to be the result of some problem in the specifications of the operating models and would not be expected to occur in reality. It is not clear how this discontinuity affected the performance of the candidate management procedures investigated in this paper.

Literature Cited

Anon. 2002. Report of the third Stock Assessment Group Meeting. 3-7 September, 2003. Canberra, Australia. CCSBT.

Eveson, P. and D. Ricard. 2003. An overview of potential graphics for evaluating the performance of candidate management procedures for southern bluefin tuna. CCSBT-MP/0304/05.

Haist, V., A. Parma, and J. Ianelli. 2002. Initial specifications of operating models for southern bluefin tuna management procedure evaluation. CCSBT-SC/0209/7.

Polacheck, T. and D. Kolody. 2003. Synthesising performance of candidate management procedures over different operating model scenarios. CCSBT-MP/0304/08.

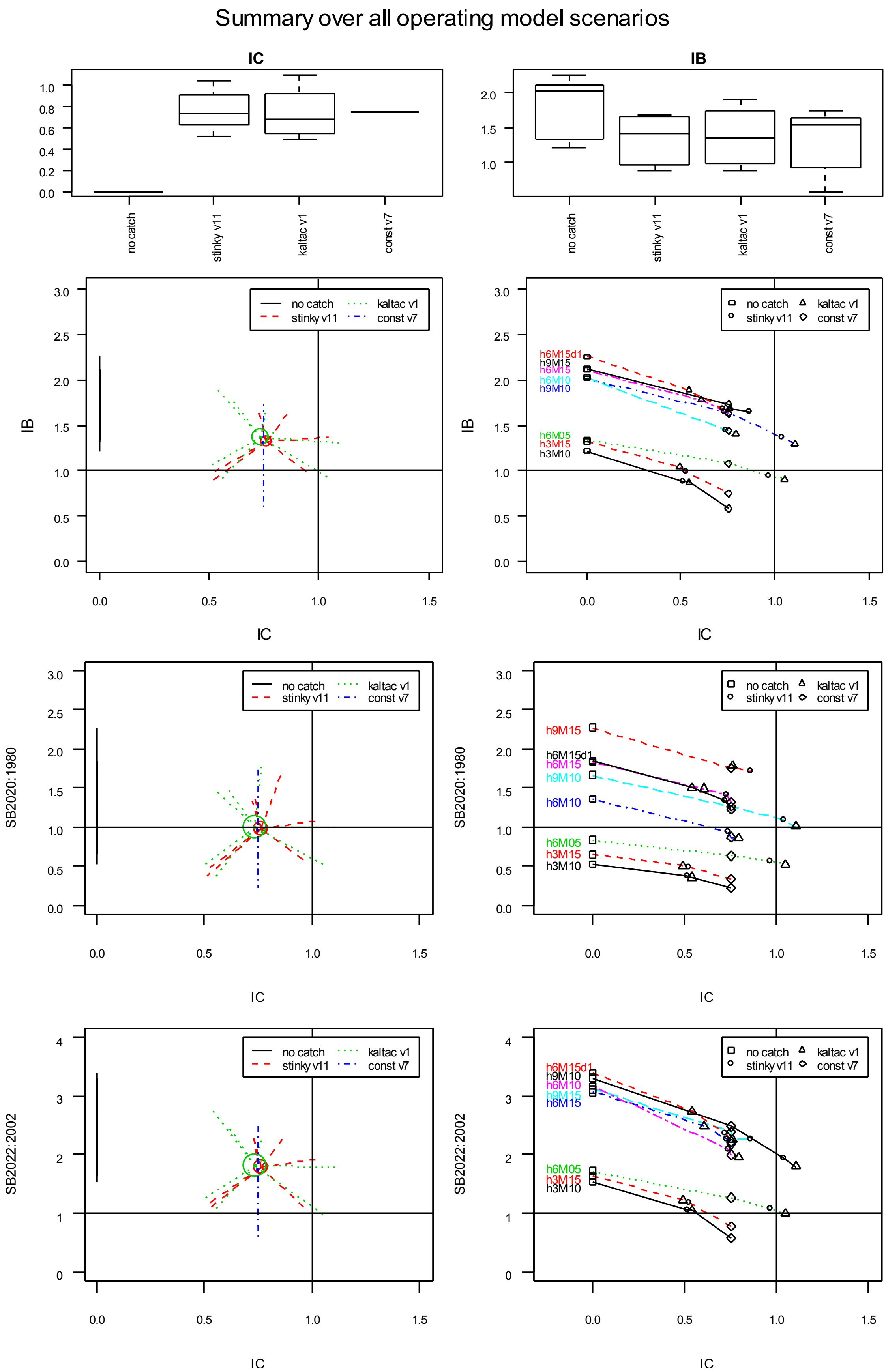


Figure 1: Summary results for three different decision rules in which the tuning parameters were chosen to provide approximately the same average catch across the eight initial operating model scenarios. (See Eveson and Ricard, 2003 for details of the summary plots).

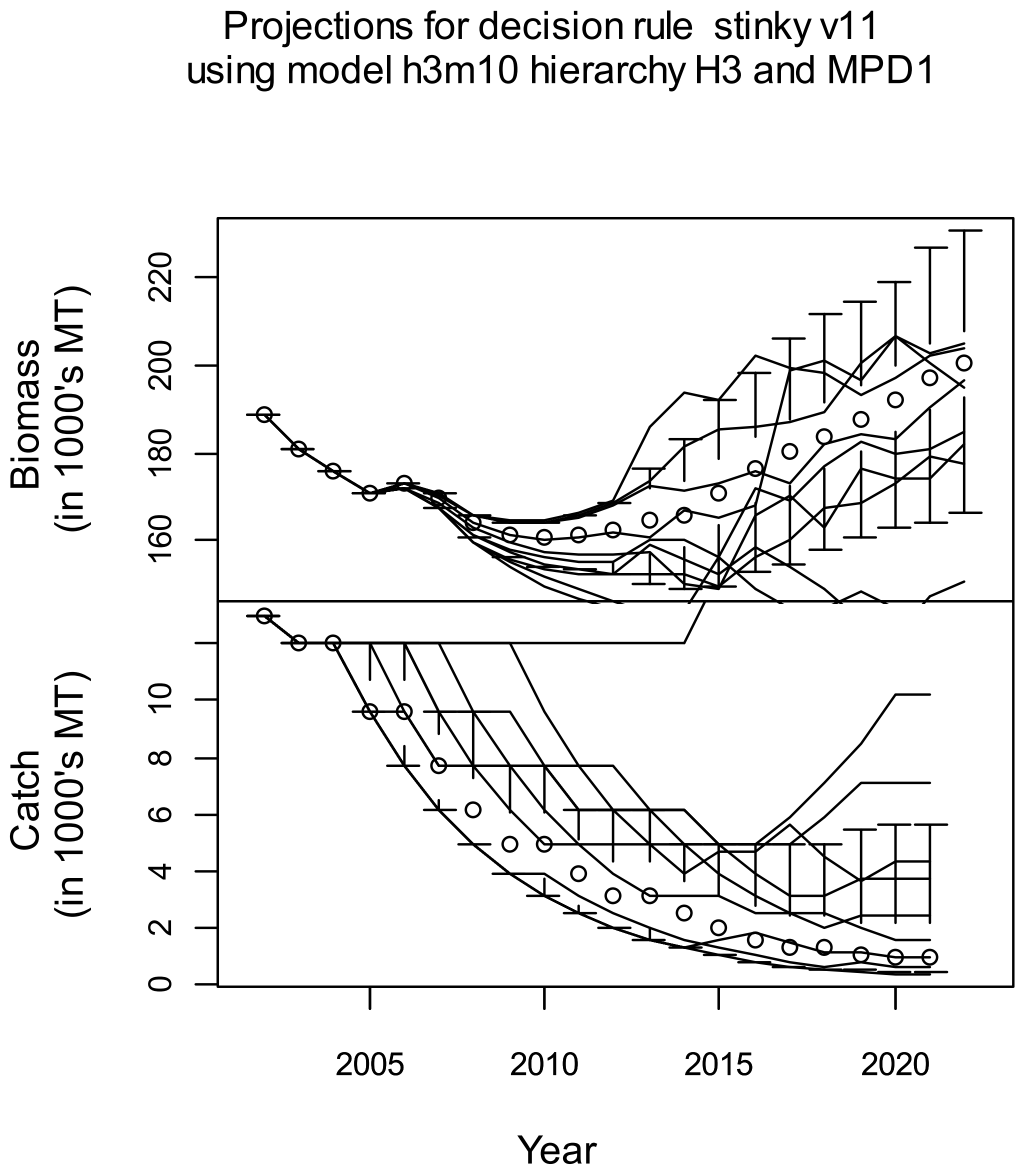


Figure 2: Worm plots for the Stinky decision rule shown in Figure 1 (i.e. Stinky v11) under the h3M10 operating model scenario.

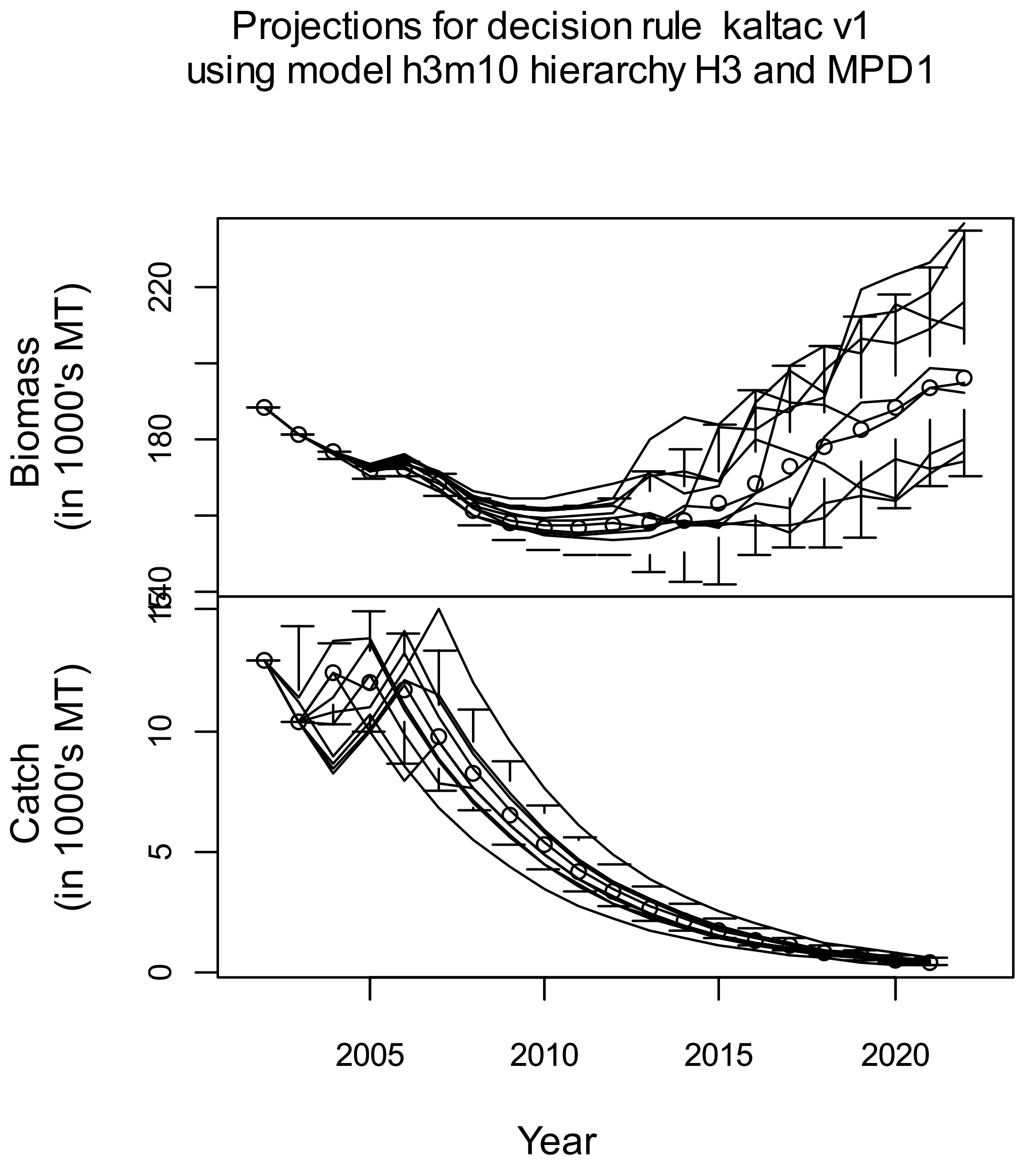


Figure 3: Worm plots for the Kaltac decision rule shown in Figure 1 (i.e. Kaltac v1) under the h3M10 operating model scenario.

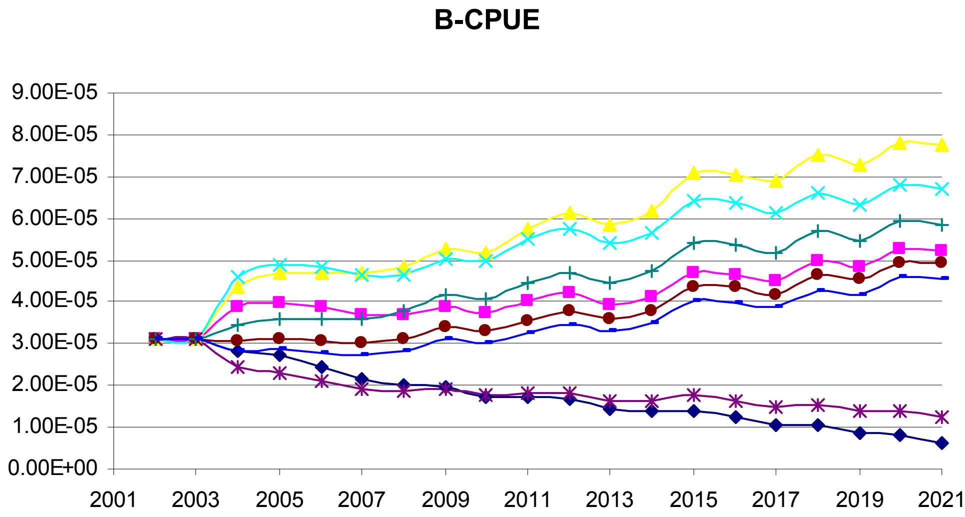


Figure 4: Comparison of the CPUE trend in biomass for deterministic runs with constant current catches for the eight initials operating model scenarios.

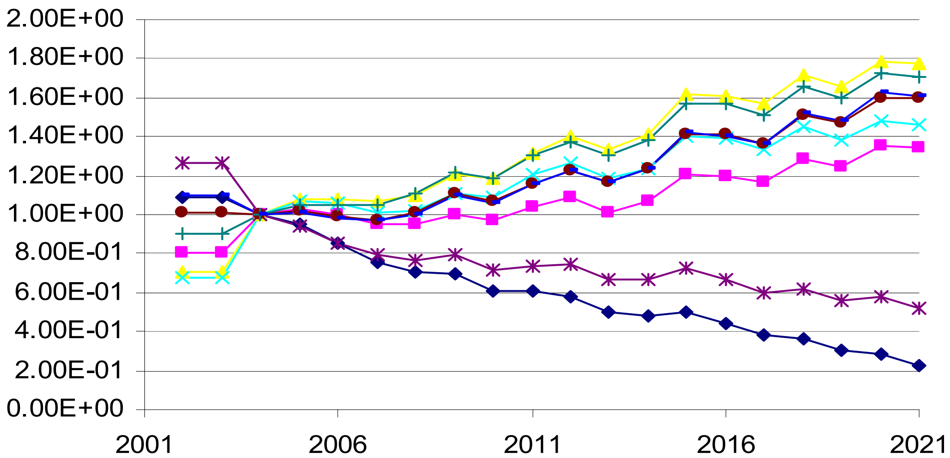


Figure 5: Comparison of the CPUE trend in biomass for deterministic runs with constant current catches for the eight initials operating model scenarios but with the CPUE indices standardised to the 2003 value.

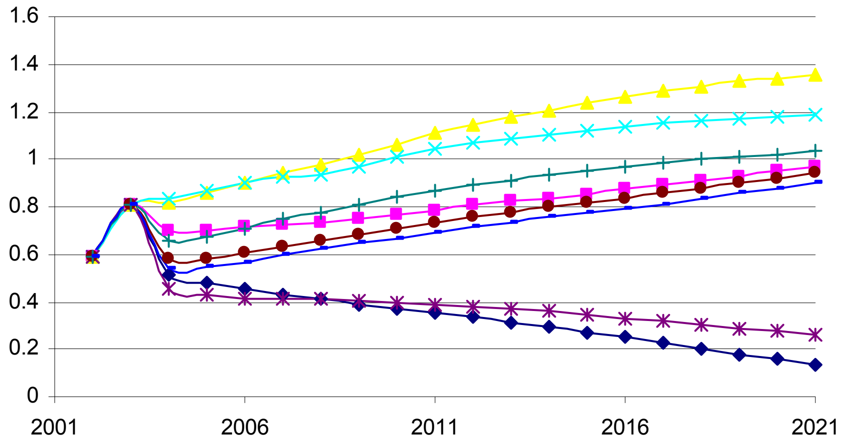


Figure 5: Comparison of the CPUE trend in numbers for deterministic runs with constant current catches for the eight initials operating model scenarios.

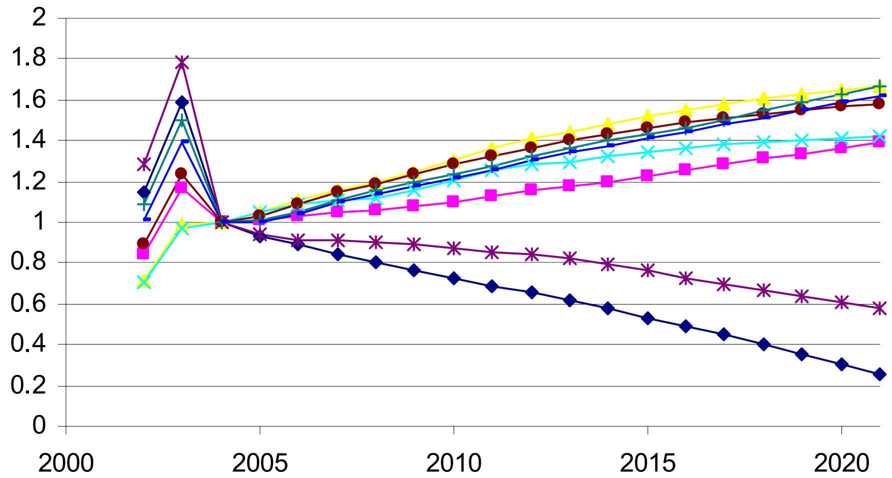


Figure 6: Comparison of the CPUE trend in numbers for deterministic runs with constant current catches for the eight initials operating model scenarios but with the CPUE indices standardised to the 2003 value.

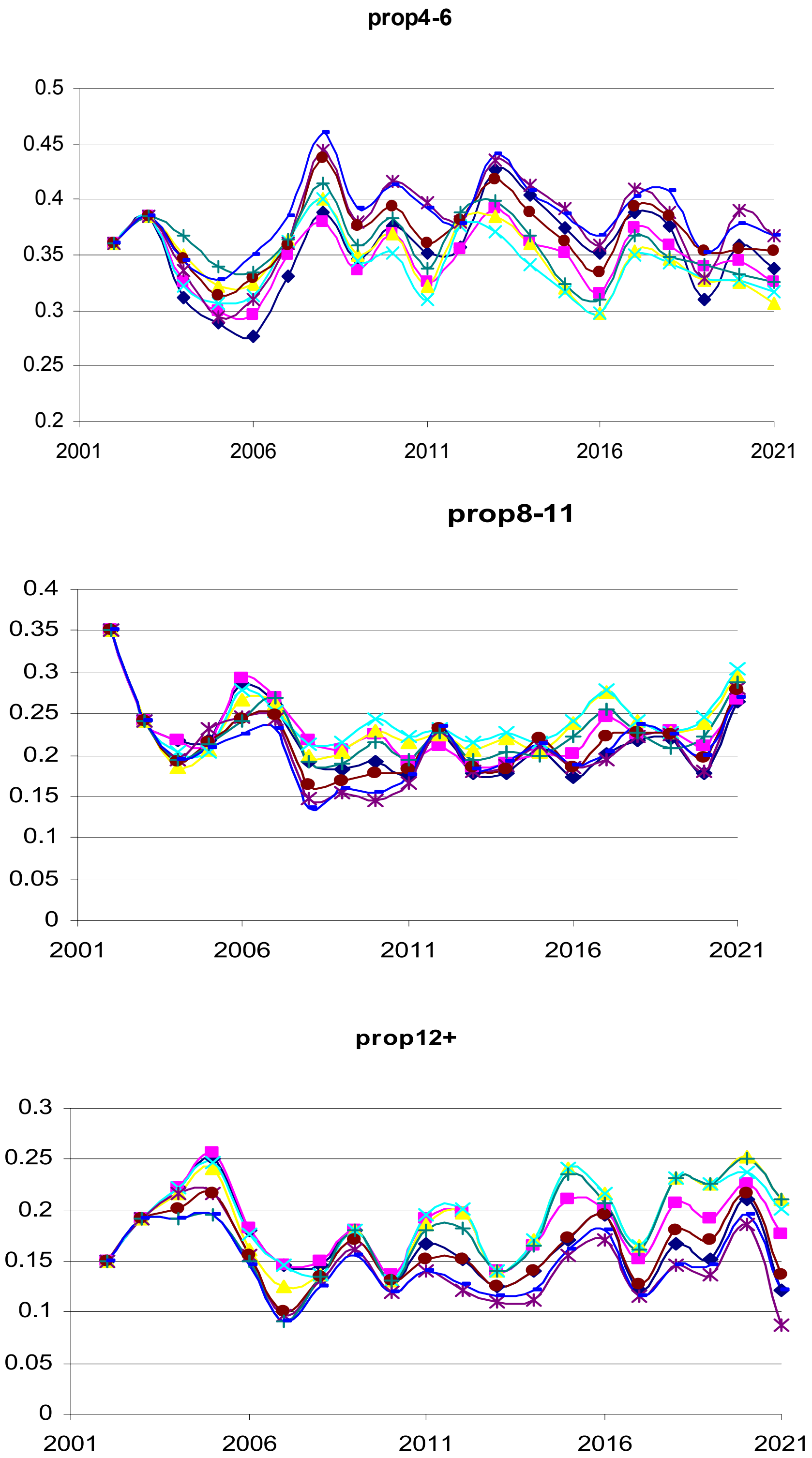


Figure 7: Comparison of the estimated proportion at age in the catch for deterministic runs with constant current catches for the eight initials operating model scenarios.

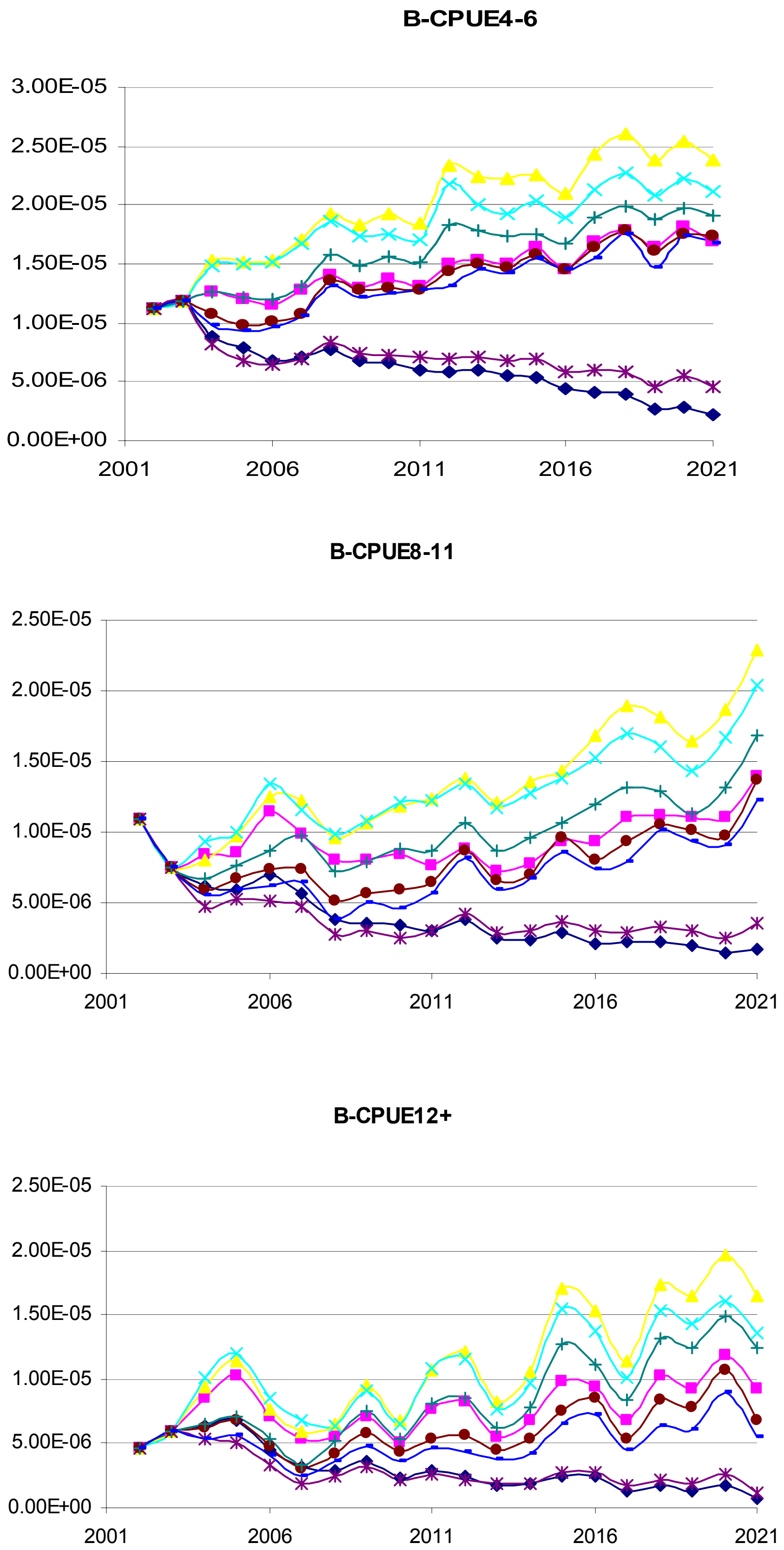


Figure 8: Comparison of the estimated CPUE age group indices (in biomass) for deterministic runs with constant current catches for the eight initials operating model scenarios.

CSIRO
MARINE RESEARCH

APPENDICES FOR:

RESULTS FROM INITIAL TESTING OF SOME CANDIDATE MANAGEMENT PROCEDURES FOR SOUTHERN BLUEFIN TUNA

Tom Polacheck
Dan Ricard
Paige Eveson
Marinelle Basson
Dale Kolody

Prepared for the CCSBT 2nd Management Procedure Workshop
7-9 and 13-14 April 2003, Queenstown, New Zealand, CCSBT-MP/0304/06

A1. Const (constant catch)

A1.1. Description of rule

A1.1.1. Overview

Const simply sets the TAC to a constant value *C* in all years, where *C* is a tuning parameter of the rule. Const is meant to serve as a reference point for evaluating other decision rules; the zero catch case (*C* = 0) is particularly useful in this regard.

A1.1.2. Mathematical description

For year *t*, TAC[*t*] = *C*

A1.1.3. Versions (tuning parameter values)

Version	*C*
1	Catch for 2001 (MT)*
2	0
3	5000
4	10000
5	15000
6	20000

*The total catch value for 2001 ranges from 15714.7 to 16196.8 MT depending on the operating model scenario.

A1.2. Performance of rule

A1.2.1. Overview

Consider the case of zero catch first. Even with no catch, the management goal of rebuilding the spawning stock biomass to the 1980 level by 2020 cannot be met under operating model scenarios h3m10, h3m15 and h6m05. Contrarily, the spawning stock biomass in 2020 is more than double the 1980 level under operating model h9m15. This highlights the large influence that the operating model assumptions can have on the results.

At the other extreme, with a catch of 20000 MT in every year, the management rebuilding goal can only be met in one operating model scenario (h9m15), although two others come very close (h6m15 and h9m10). Moreover, the stock crashes under operating model scenario h3m10.

Intermediate catch values between 0 and 20000 MT behave predictably. The results are best summarized through the graphs shown below.

A1.2.2. Graphics

The graphs are fairly self-explanatory. Note that const version 2 is called “no catch” for easy interpretation because it is included as a reference case on the graphs for all decision rules. Under h3m10, const v6 cannot maintain a catch of 20000 MT in the final years, hence the non-zero value for the inter-annual catch variability statistic, and the deviation in the catch statistics from a constant value. The worm plots are shown for version 1 (keeping the TAC at 2001 catch levels) under model scenarios h3m10 and h9m15. The biomass trend and amount of variability will obviously vary depending on the version and model scenario.

Figure A1-1a.

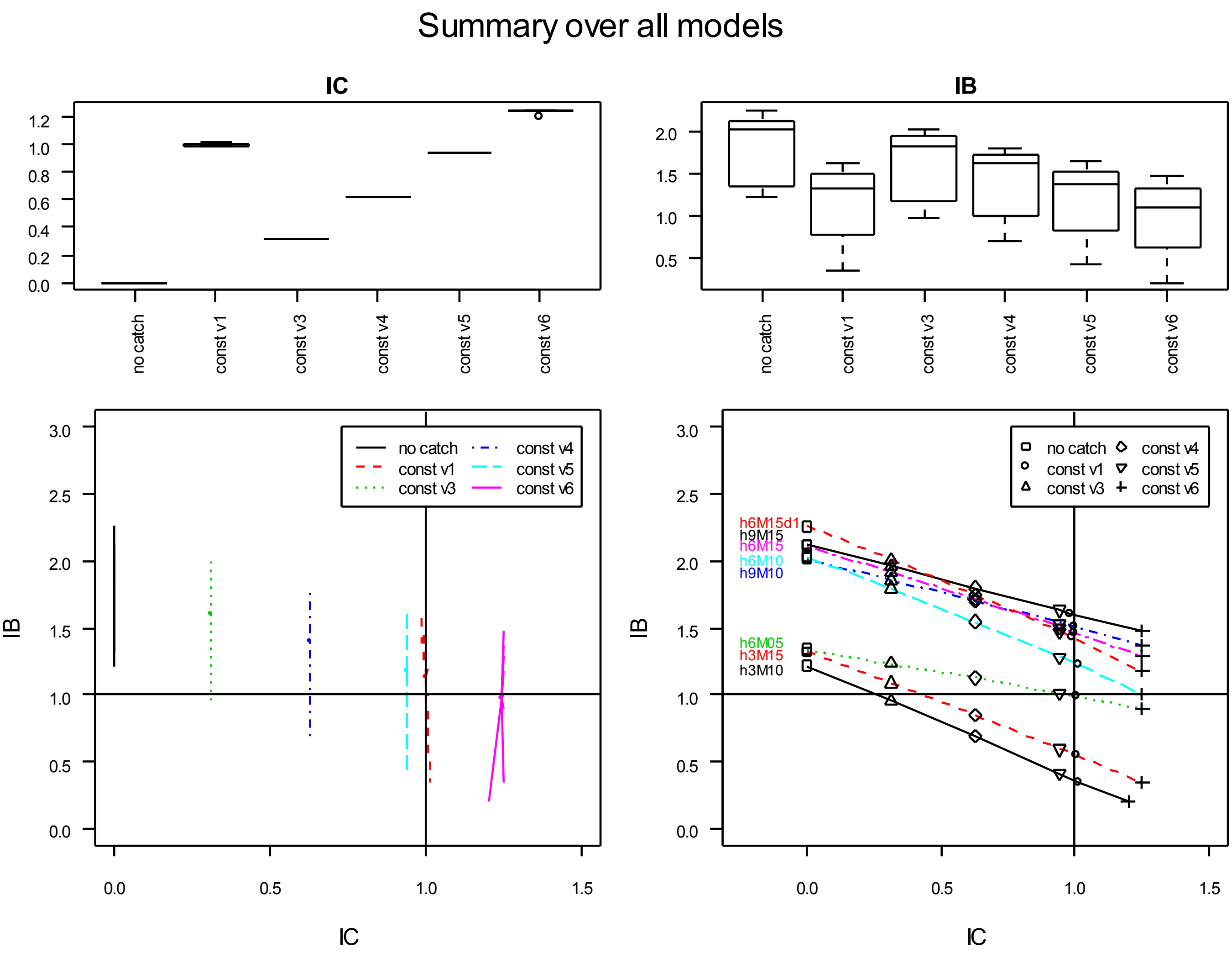

Summary over all models
IC
IB
no catch
const v1
const v3
const v4
const v5
const v6
h6M15d1
h9M15
h6M15
h6M10
h9M10
h6M05
h3M15
h3M10
IB
IC

Figure A1-1b.

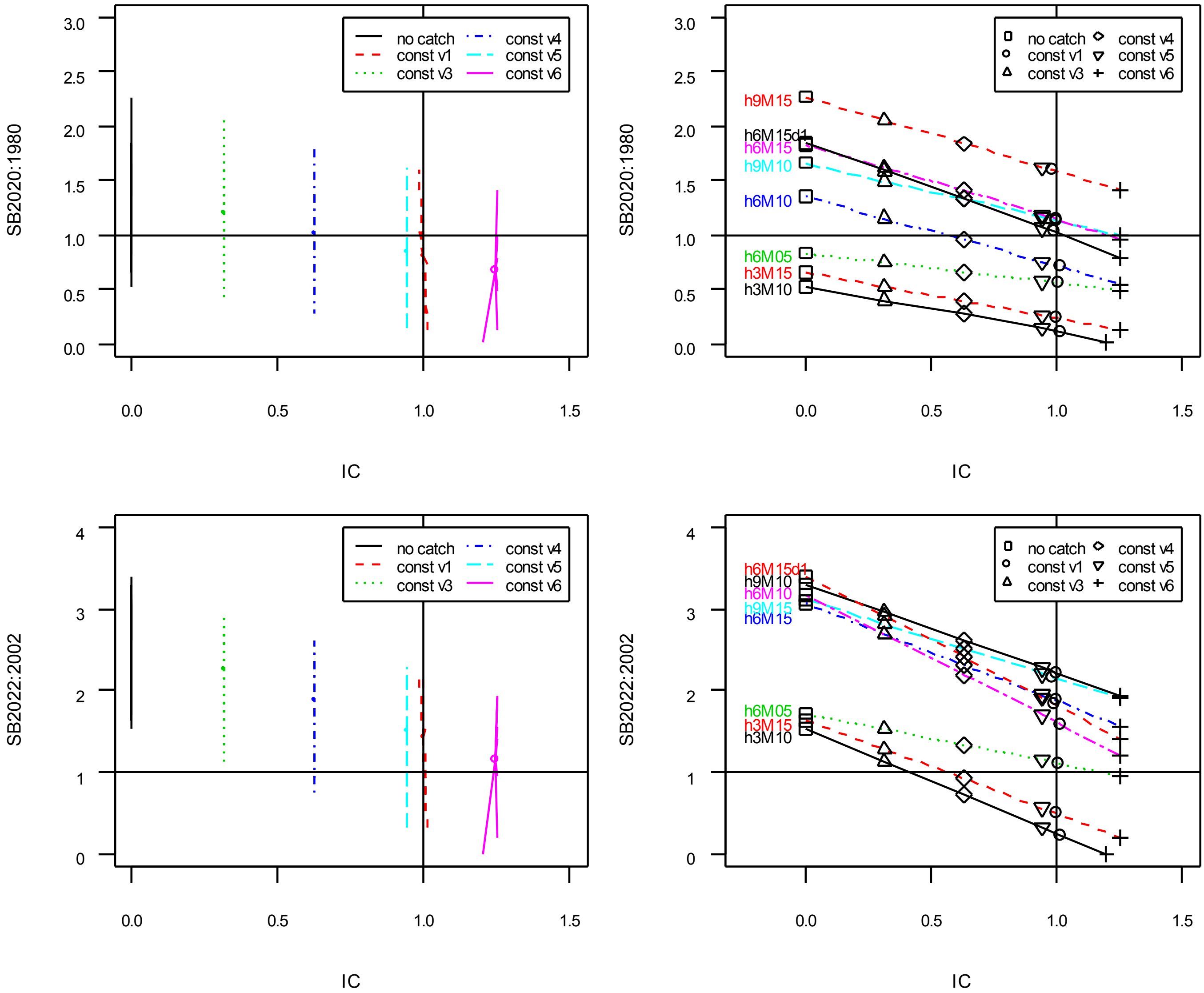
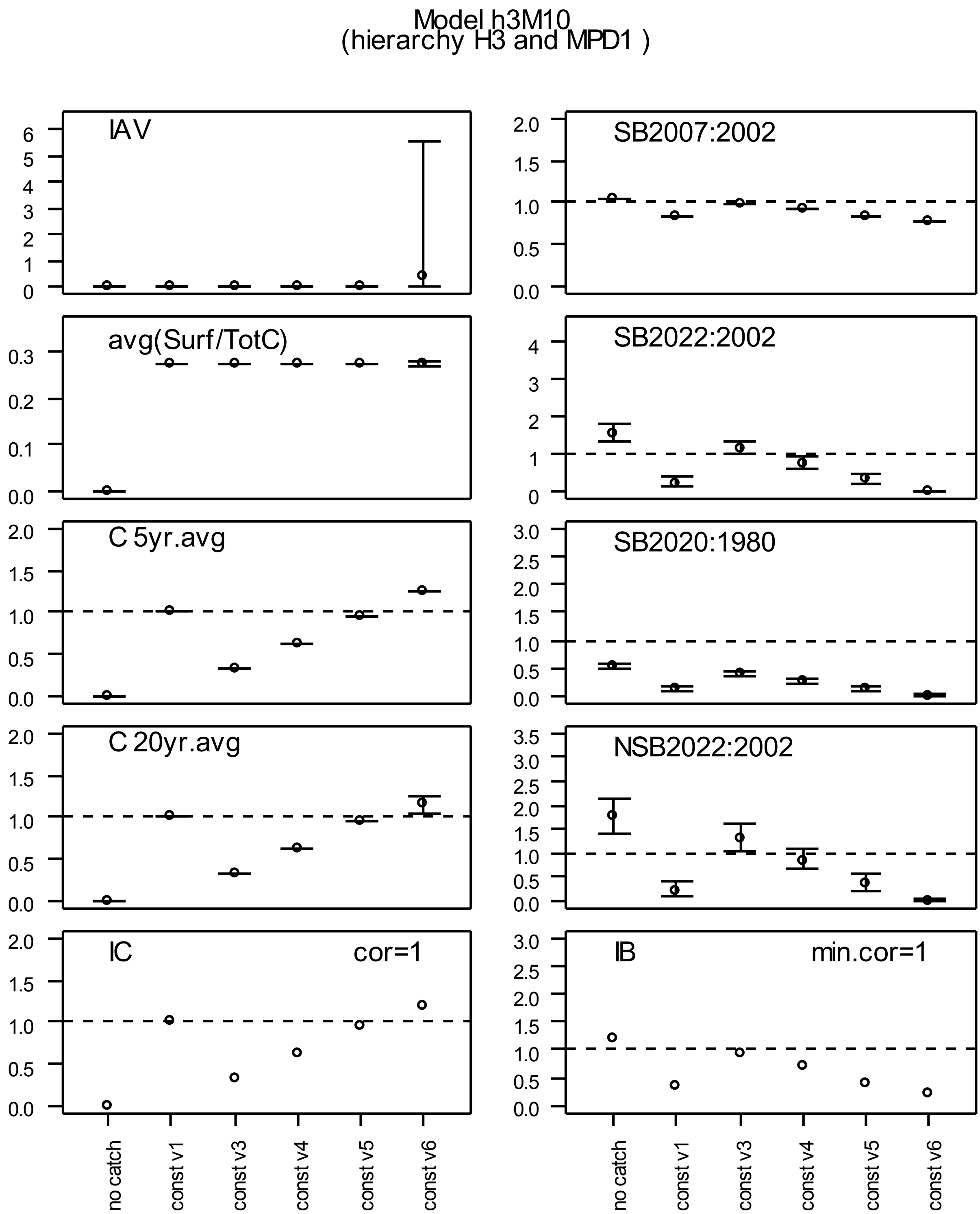

no catch
const v1
const v3
const v4
const v5
const v6
SB2020:1980
IC
h9M15
h6M15d1
h6M15
h9M10
h6M10
h6M05
h3M15
h3M10
SB2022:2002
h6M15d1
h9M10
h6M10
h9M15
h6M15
h6M05
h3M15
h3M10

Figure A1-2.



Model h3M10
(hierarchy H3 and MPD1)
IAV
avg(Surf/TotC)
C 5yr.avg
C 20yr.avg
IC
cor=1
SB2007:2002
SB2022:2002
SB2020:1980
NSB2022:2002
IB
min.cor=1
no catch
const v1
const v3
const v4
const v5
const v6

Figure A1-3.

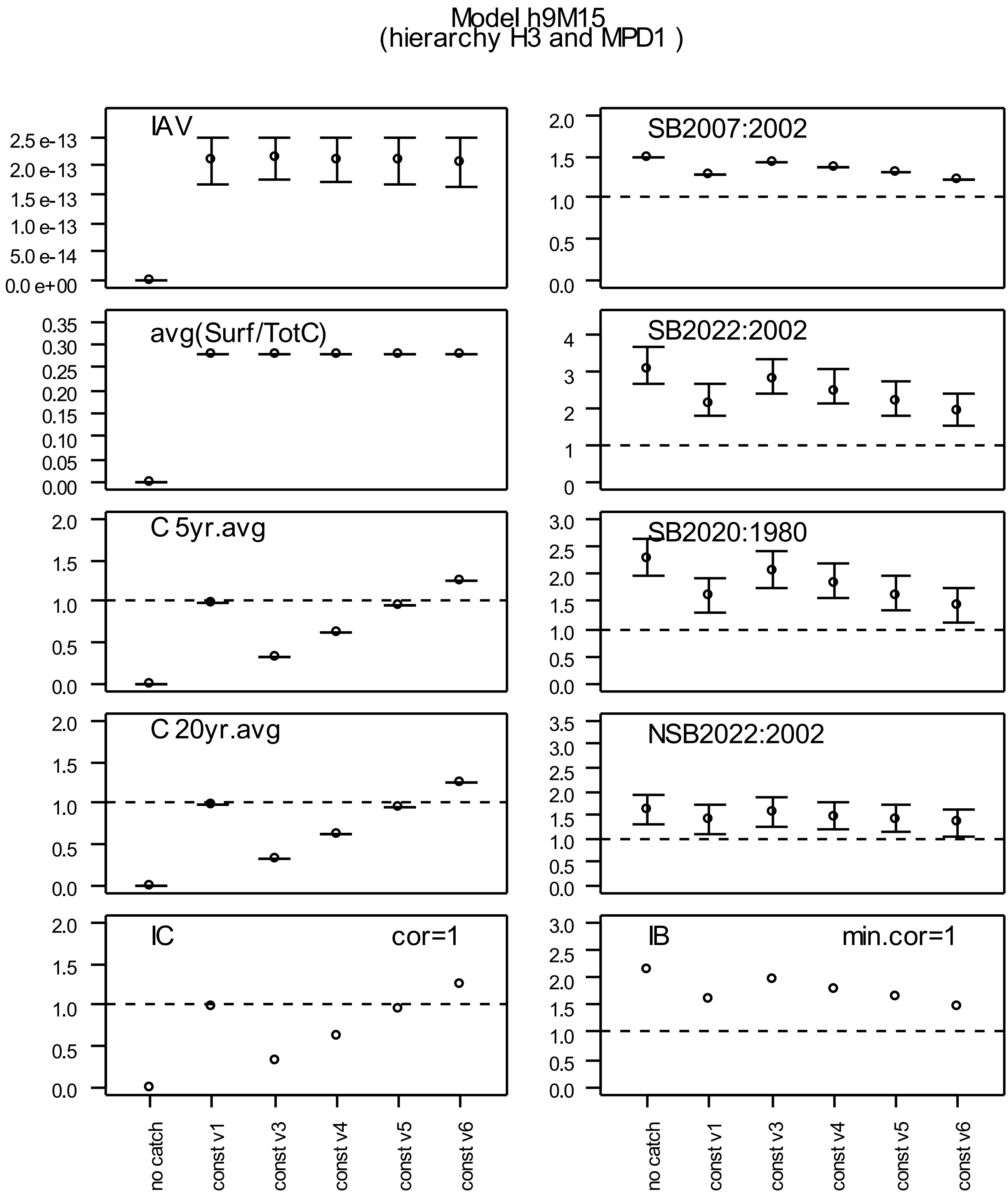

Model h9M15
(hierarchy H3 and MPD1)
IAV
avg(Surf/TotC)
C 5yr.avg
C 20yr.avg
IC
cor=1
SB2007:2002
SB2022:2002
SB2020:1980
NSB2022:2002
IB
min.cor=1
no catch
const v1
const v3
const v4
const v5
const v6

Figure A1-4.

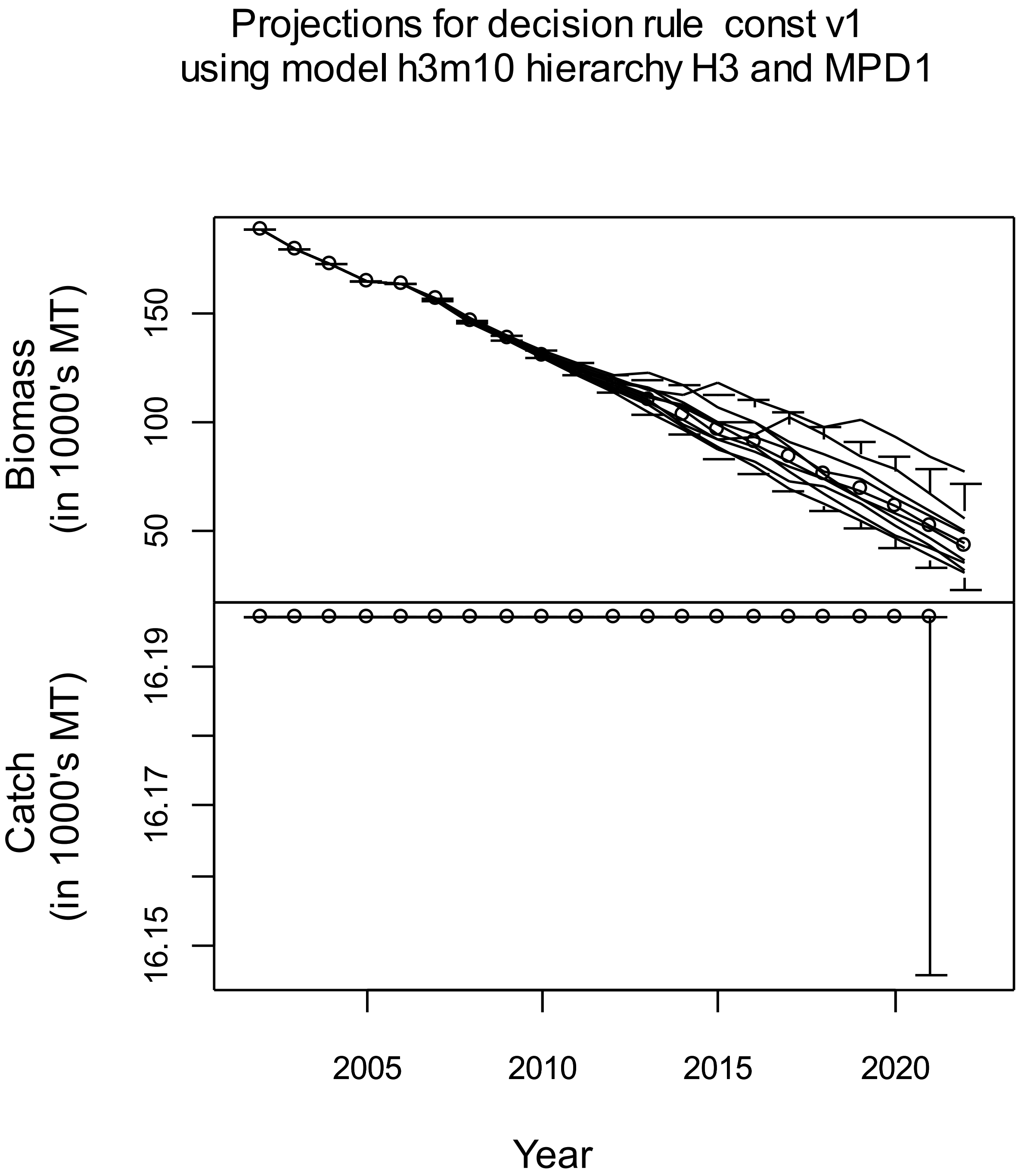
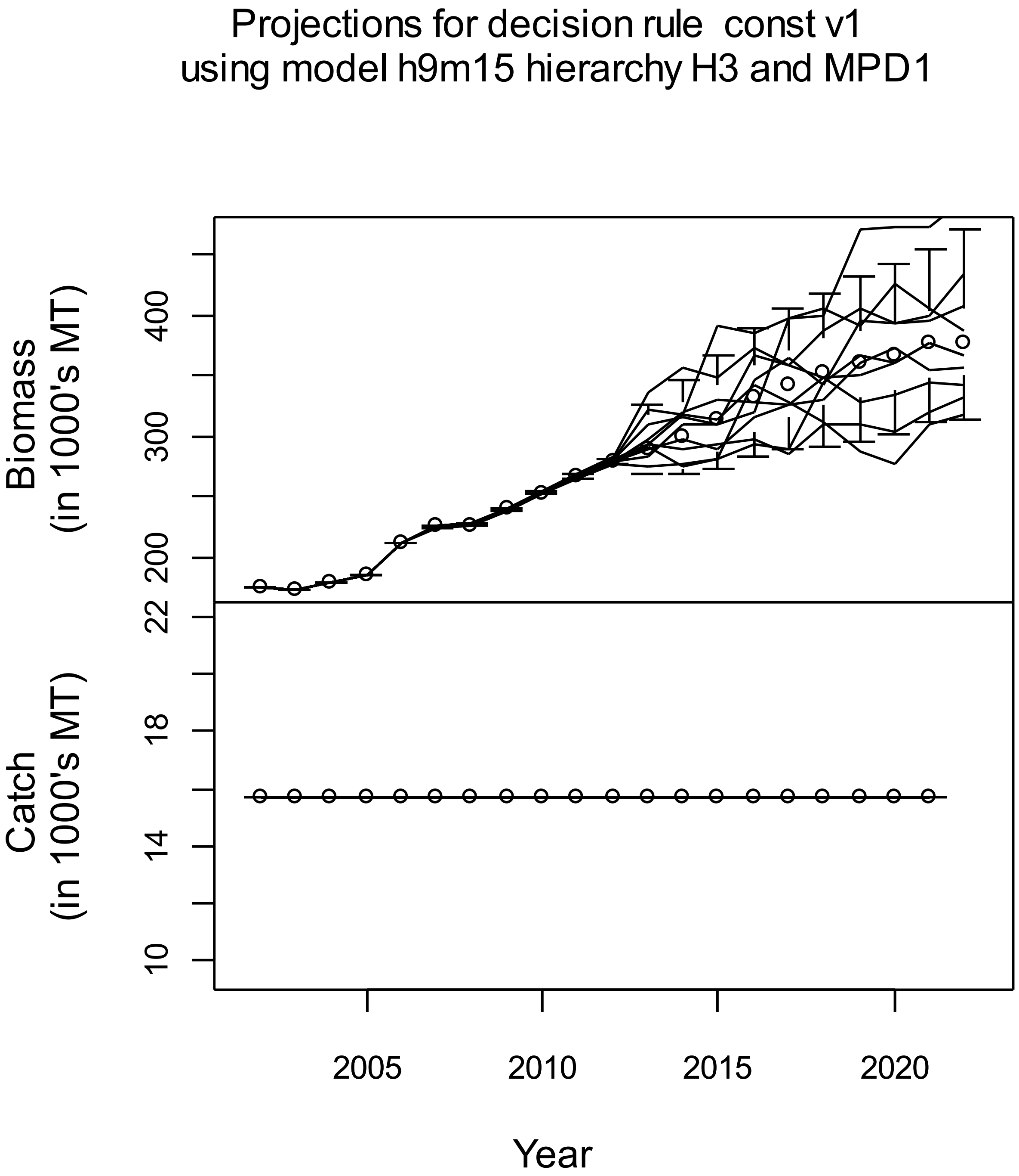

Projections for decision rule const v1
using model h3m10 hierarchy H3 and MPD1
Biomass
(in 1000's MT)
50
100
150
Catch
(in 1000's MT)
16.15
16.17
16.19
2005
2010
2015
2020
Year

Figure A1-5.


Projections for decision rule const v1
using model h9m15 hierarchy H3 and MPD1
Biomass
(in 1000's MT)
200
300
400
Catch
(in 1000's MT)
10
14
18
22
2005
2010
2015
2020
Year

A2. Incr (continuous catch increase)

A2.1. Description of rule

A2.1.1. Overview

For exploratory purposes only. This rule increase the TAC each year.

A2.1.2. Mathematical description

TAC increases every year. The increase is either linear:

$$TAC_{y+1} = TAC_y + INCR$$

or exponential:

$$TAC_{y+1} = TAC_y * mult$$

A2.1.3. Versions (tuning parameter values)

Rule version	***Details***		
v1	linear increase	$INCR = 0.075 * TAC_{first\_yr-1}$	current TAC for 0 years
v2	linear increase	$INCR = 0.075 * TAC_{first\_yr-1}$	current TAC for 5 years
v3	linear increase	$INCR = 0.10 * TAC_{first\_yr-1}$	current TAC for 0 years
v8	linear increase	$INCR = 0.15 * TAC_{first\_yr-1}$	current TAC for 5 years
v15	exponential increase	$mult = 1.05$	current TAC for 0 years
v20	exponential increase	$mult = 1.10$	current TAC for 5 years

A2.2. Performance of rule

A2.2.1. Overview

The rule is used to establish a reference case.

A2.2.2. Graphics

Figure A2-1a.

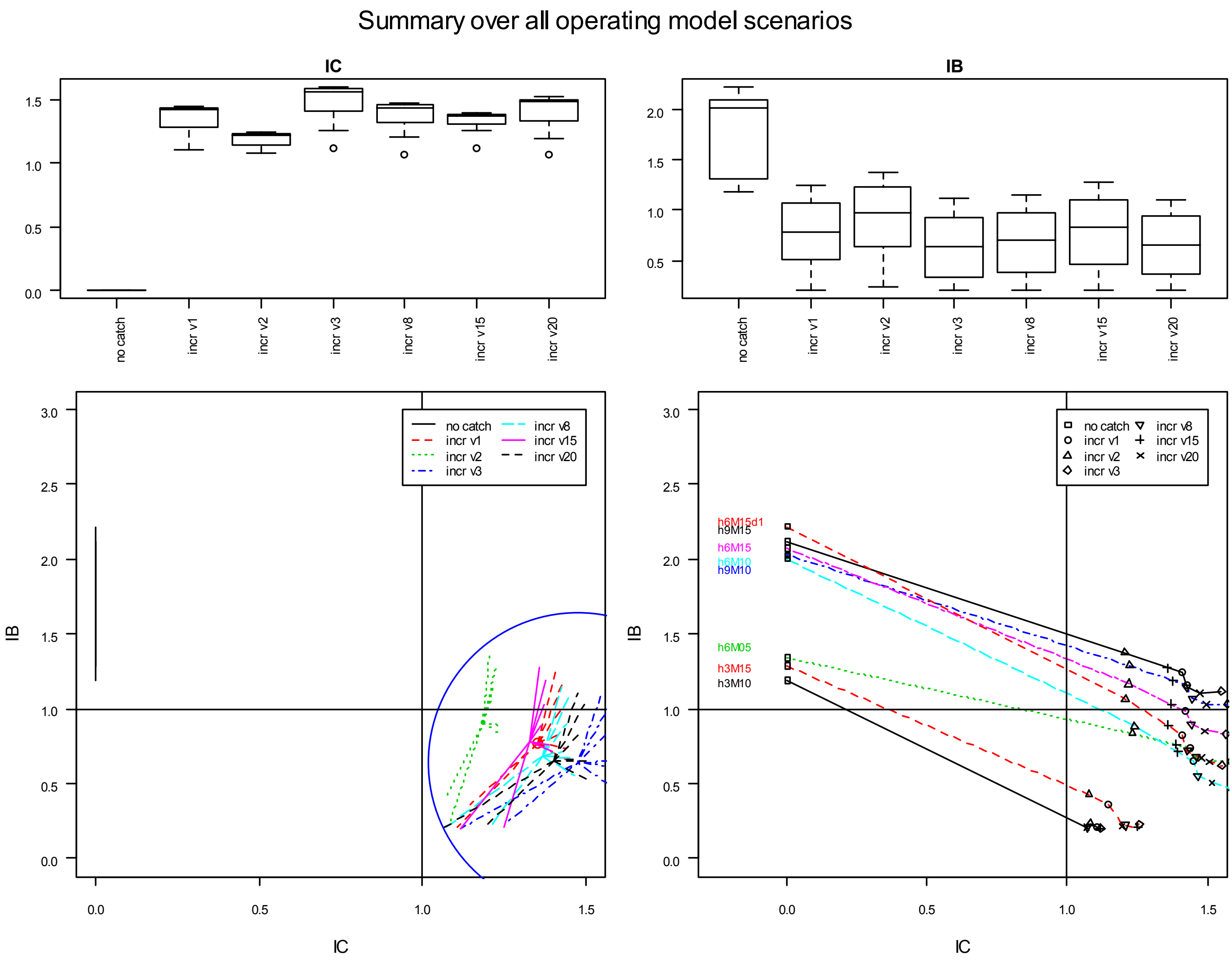

Summary over all operating model scenarios
IC
IB
no catch
incr v1
incr v2
incr v3
incr v8
incr v15
incr v20
h6M15d1
h9M15
h6M15
h6M10
h9M10
h6M05
h3M15
h3M10

Figure A2-1b.

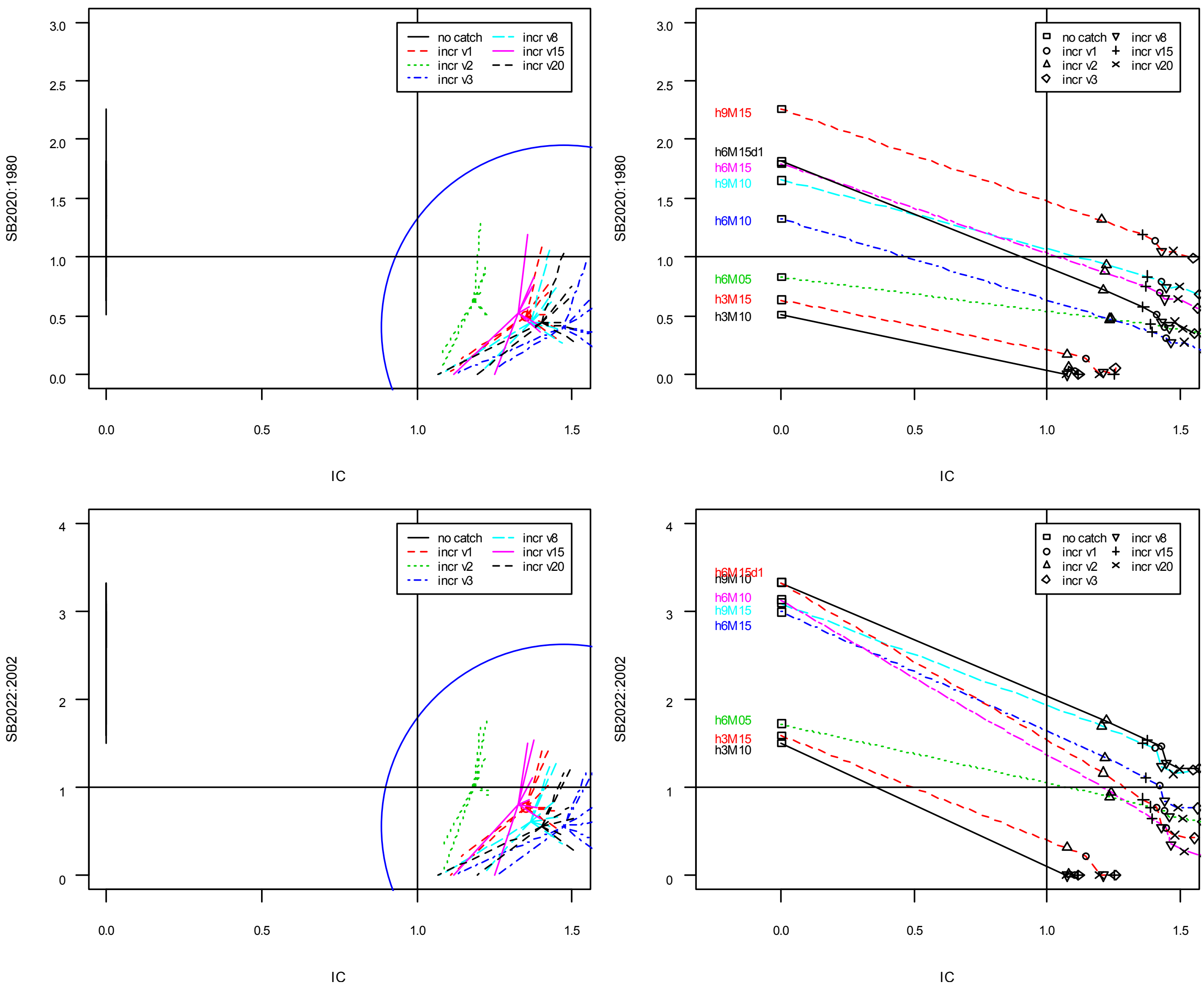

no catch
incr v1
incr v2
incr v3
incr v8
incr v15
incr v20
SB2020:1980
SB2022:2002
IC
h9M15
h6M15d1
h6M15
h9M10
h6M10
h6M05
h3M15
h3M10

Figure A2-2.

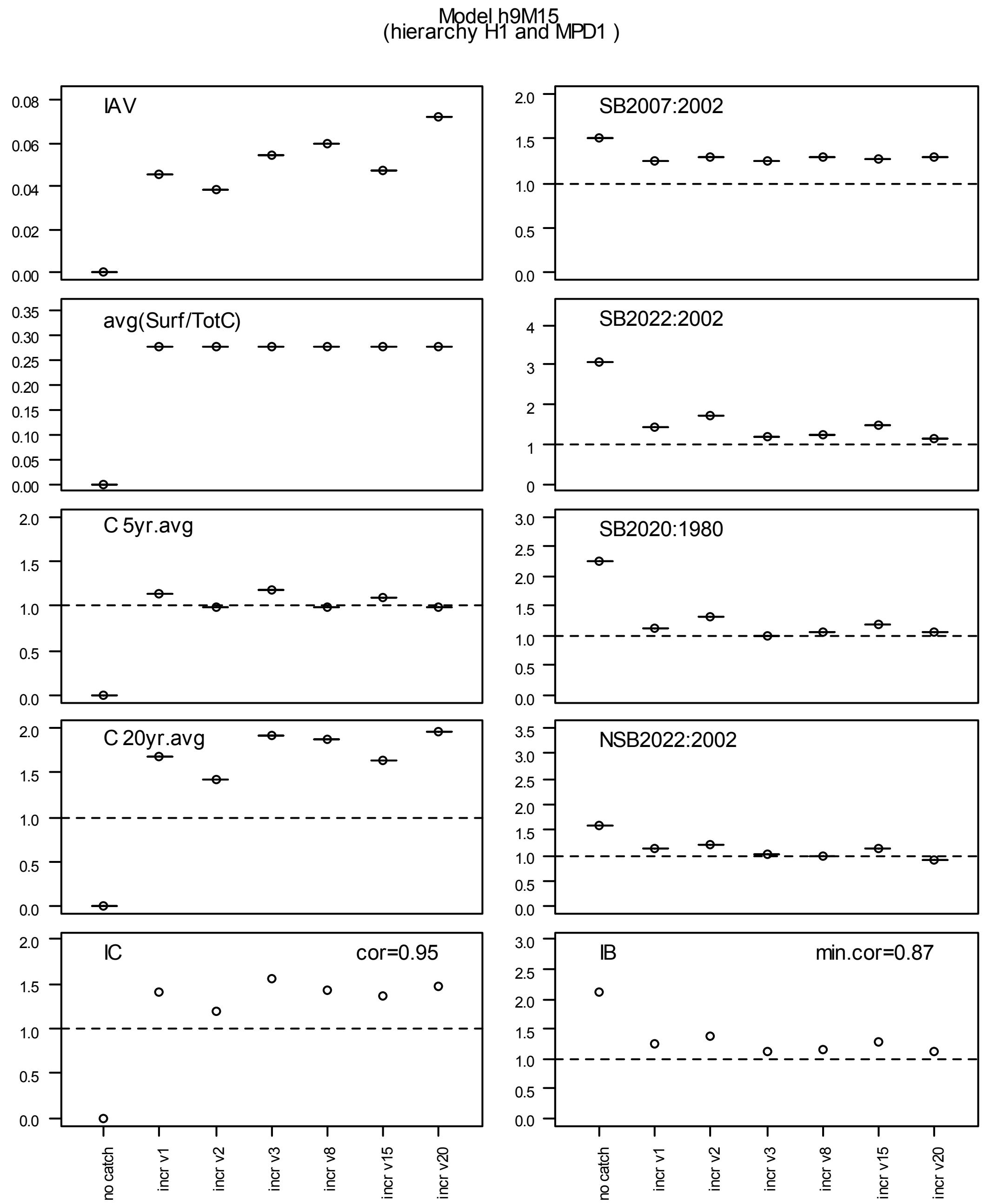

Model h9M15
(hierarchy H1 and MPD1)
IAV
SB2007:2002
avg(Surf/TotC)
SB2022:2002
C 5yr.avg
SB2020:1980
C 20yr.avg
NSB2022:2002
IC
cor=0.95
IB
min.cor=0.87
no catch
incr v1
incr v2
incr v3
incr v8
incr v15
incr v20

Figure A2-3.

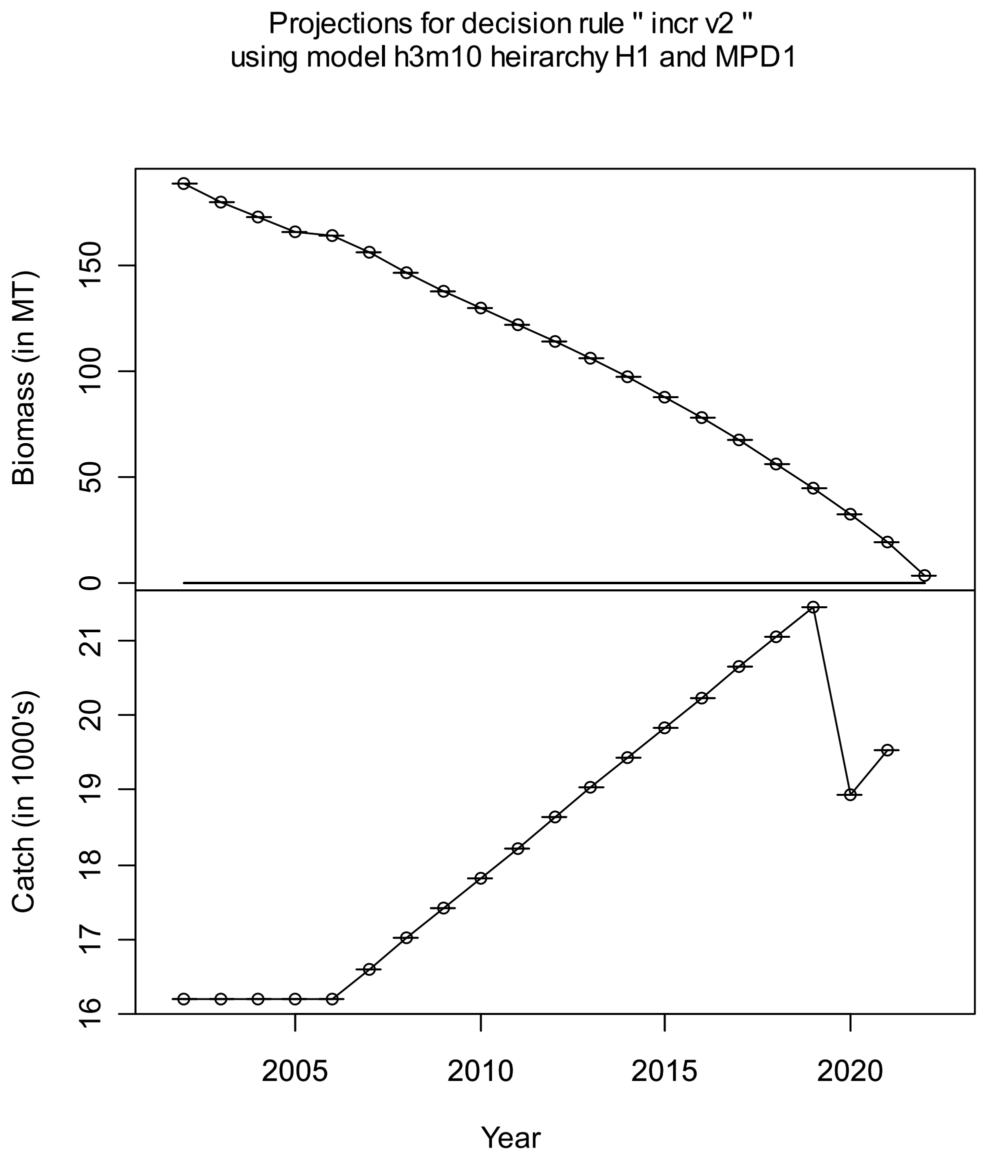
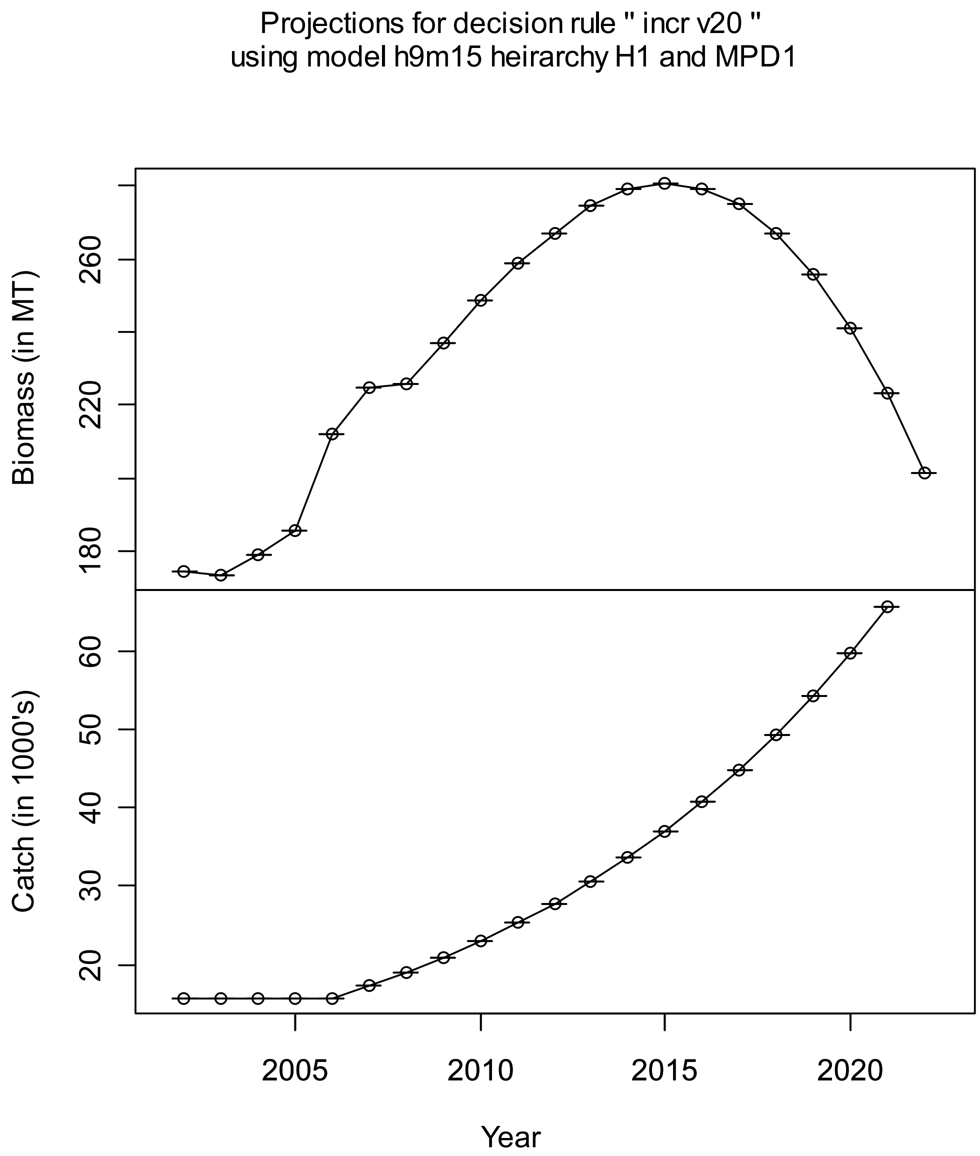

Projections for decision rule " incr v2 "
using model h3m10 heirarchy H1 and MPD1
Biomass (in MT)
0
50
100
150
Catch (in 1000's)
16
17
18
19
20
21
2005
2010
2015
2020
Year

Figure A2-4.


Projections for decision rule " incr v20 "
using model h9m15 heirarchy H1 and MPD1
Biomass (in MT)
180
220
260
Catch (in 1000's)
20
30
40
50
60
2005
2010
2015
2020
Year

A3. Decr (continuous catch decrease)

A3.1. Description of rule

A3.1.1. Overview

The TAC is decreased every year.

A3.1.2. Mathematical description

TAC decreases every year. The decrease is either linear:

$$TAC_{y+1} = TAC_y - DECR$$

or exponential:

$$TAC_{y+1} = TAC_y * mult$$

A3.1.3. Versions (tuning parameter values)

Rule version	***Details***		
v1	linear decrease	$DECR = 0.10 * TAC_{first\_yr-1}$	decrease every year
v2	linear decrease	$DECR = 0.20 * TAC_{first\_yr-1}$	decrease every year
v3	linear decrease	$DECR = 0.10 * TAC_{first\_yr-1}$	decrease every 5 years
v4	linear decrease	$DECR = 0.20 * TAC_{first\_yr-1}$	decrease every 5 years
v5	exponential decrease	$mult = 0.90$	decrease every year
v6	exponential decrease	$mult = 0.80$	decrease every year
v7	exponential decrease	$mult = 0.90$	decrease every 5 years
v8	exponential decrease	$mult = 0.80$	decrease every 5 years

A3.2. Performance of rule

A3.2.1. Overview

This rule is used to establish a reference case.

A3.2.2. Graphics

Figure A3-1a.

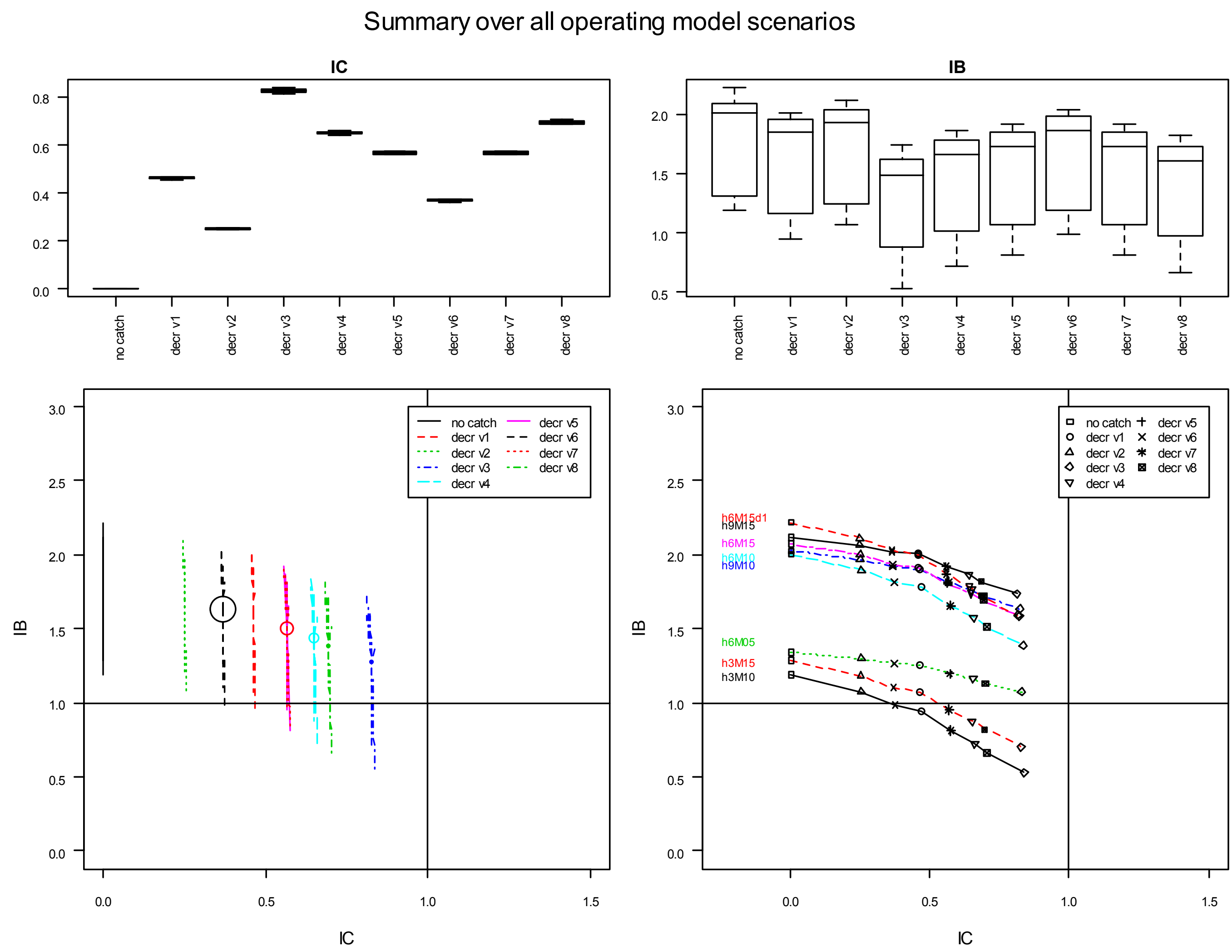

Summary over all operating model scenarios
IC
IB
no catch
decr v1
decr v2
decr v3
decr v4
decr v5
decr v6
decr v7
decr v8
h6M15d1
h9M15
h6M15
h6M10
h9M10
h6M05
h3M15
h3M10

Figure A3-1b.

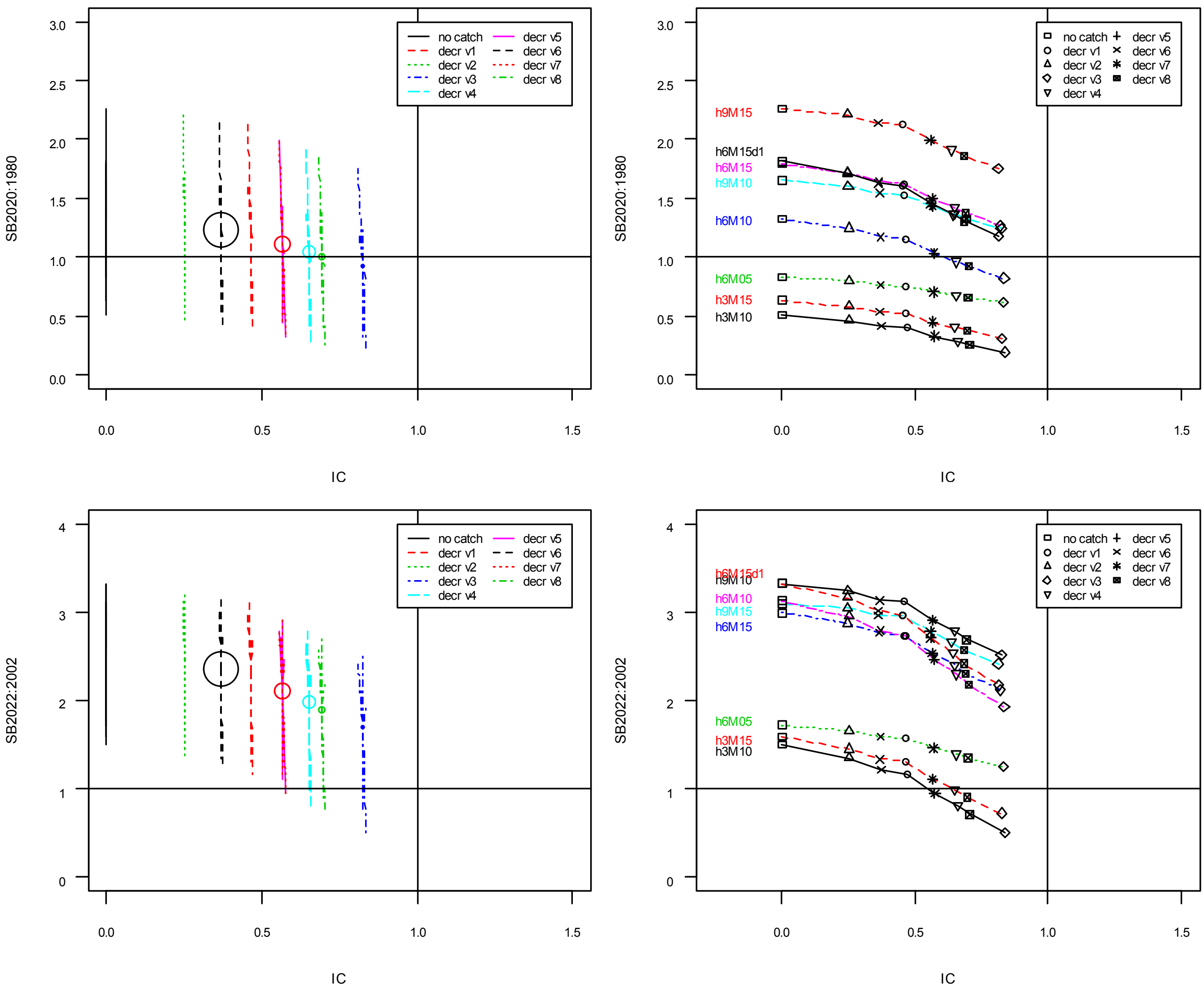

SB2020:1980
SB2022:2002
IC
no catch
decr v1
decr v2
decr v3
decr v4
decr v5
decr v6
decr v7
decr v8
h9M15
h6M15d1
h6M15
h9M10
h6M10
h6M05
h3M15
h3M10
0.0
0.5
1.0
1.5
2.0
2.5
3.0
0
1
2
3
4

Figure A3-2.

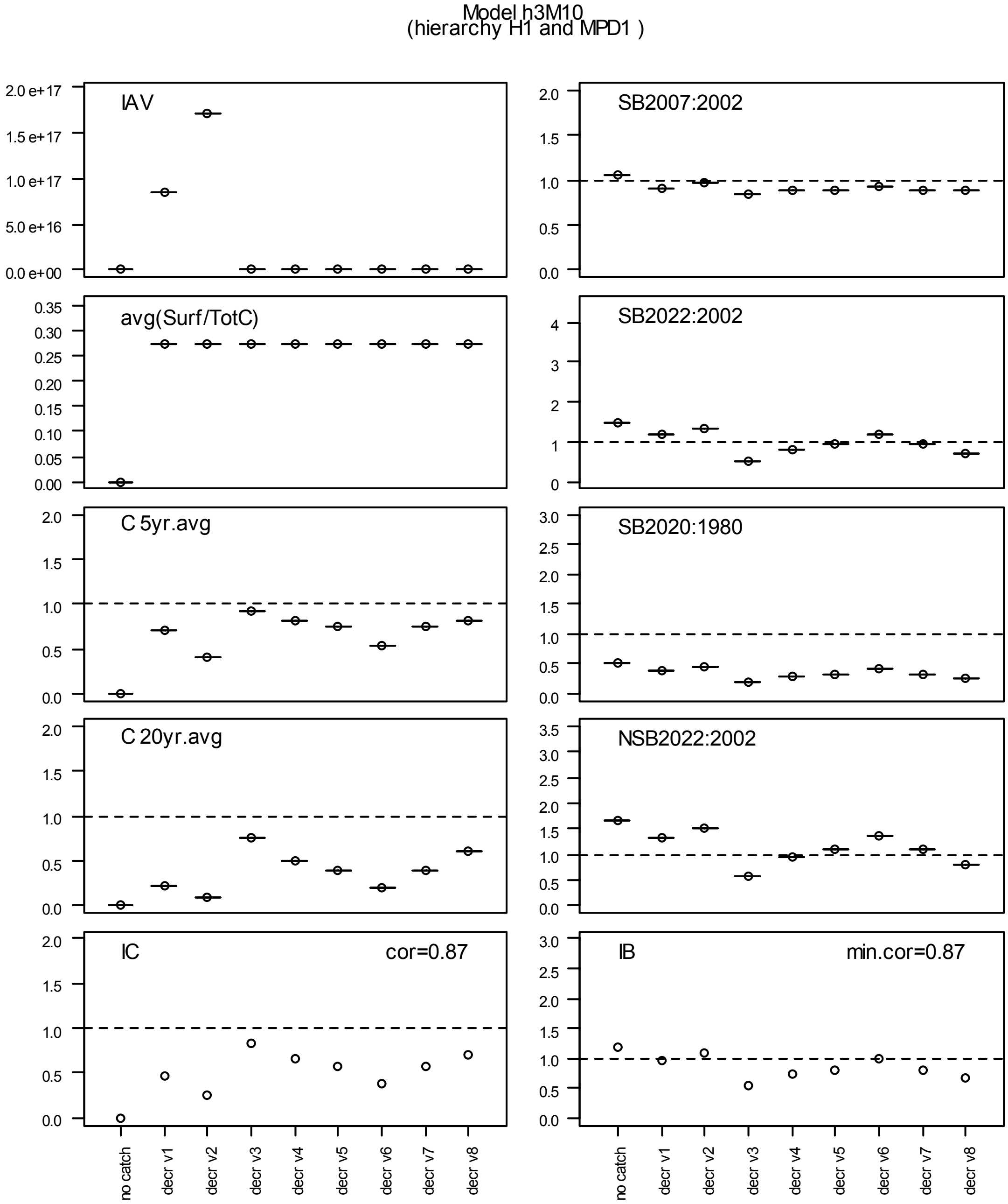

Model h3M10
(hierarchy H1 and MPD1)
IAV
SB2007:2002
avg(Surf/TotC)
SB2022:2002
C 5yr.avg
SB2020:1980
C 20yr.avg
NSB2022:2002
IC
cor=0.87
IB
min.cor=0.87
no catch
decr v1
decr v2
decr v3
decr v4
decr v5
decr v6
decr v7
decr v8

Figure A3-3.

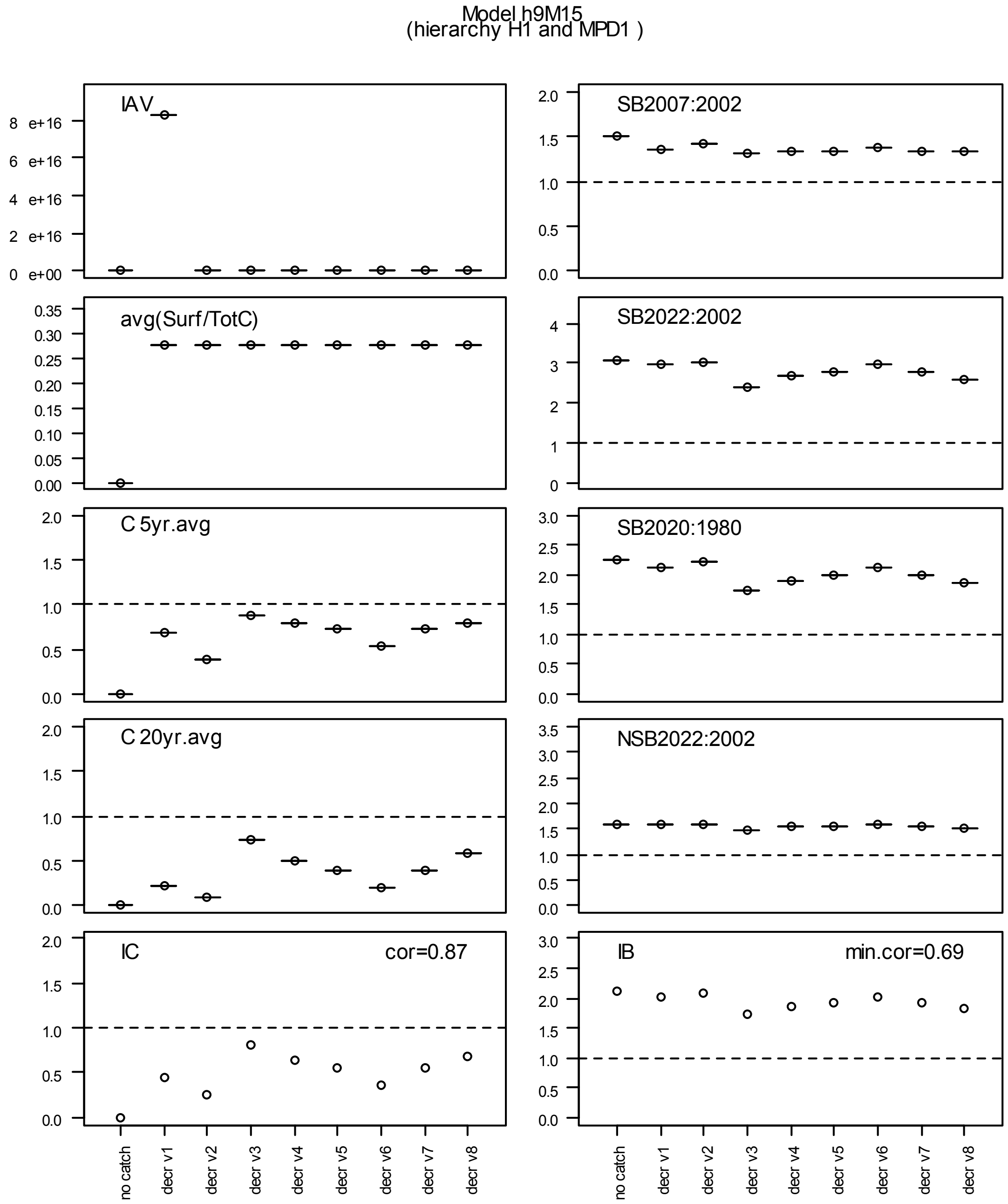

Model h9M15
(hierarchy H1 and MPD1)
IAV
SB2007:2002
avg(Surf/TotC)
SB2022:2002
C 5yr.avg
SB2020:1980
C 20yr.avg
NSB2022:2002
IC
cor=0.87
IB
min.cor=0.69
no catch
decr v1
decr v2
decr v3
decr v4
decr v5
decr v6
decr v7
decr v8

Figure A3-4.

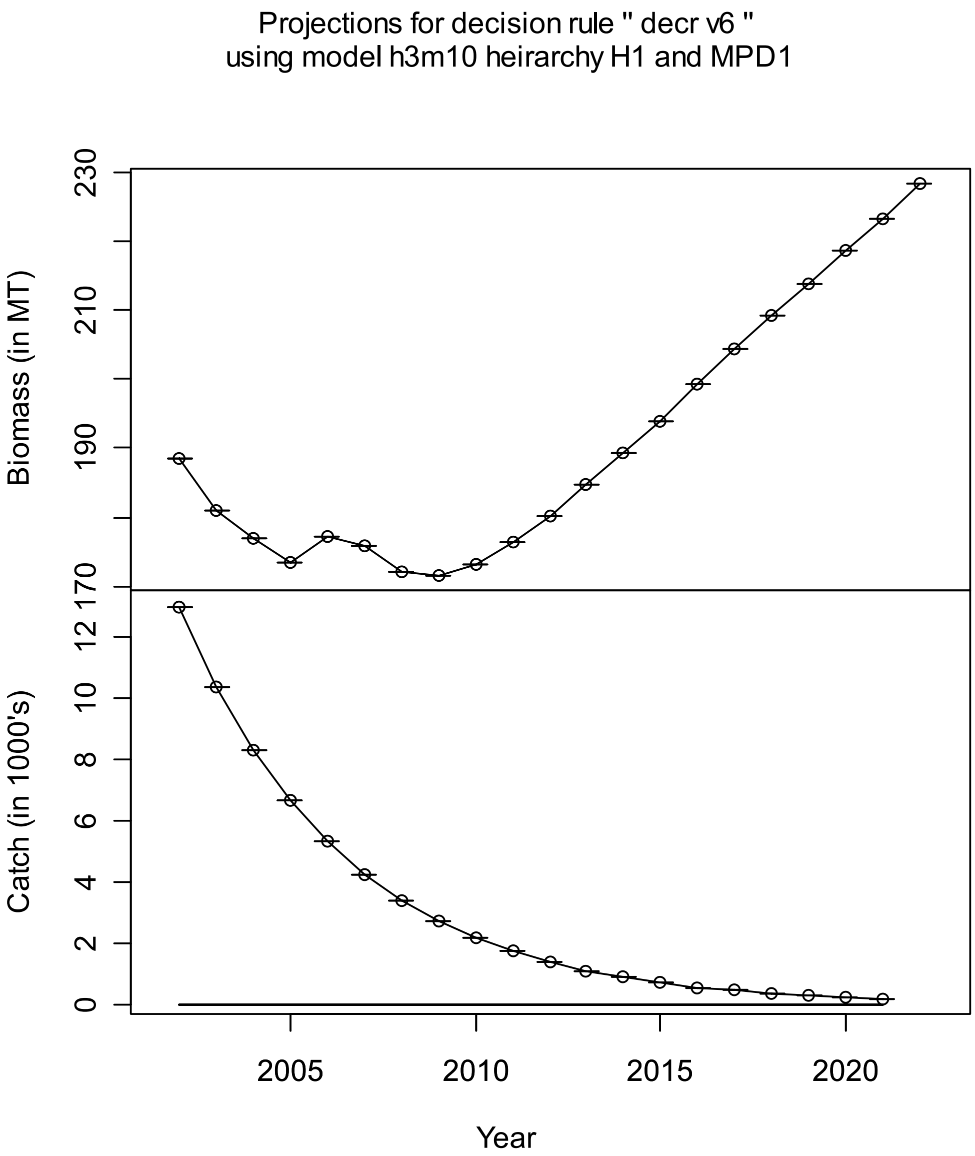
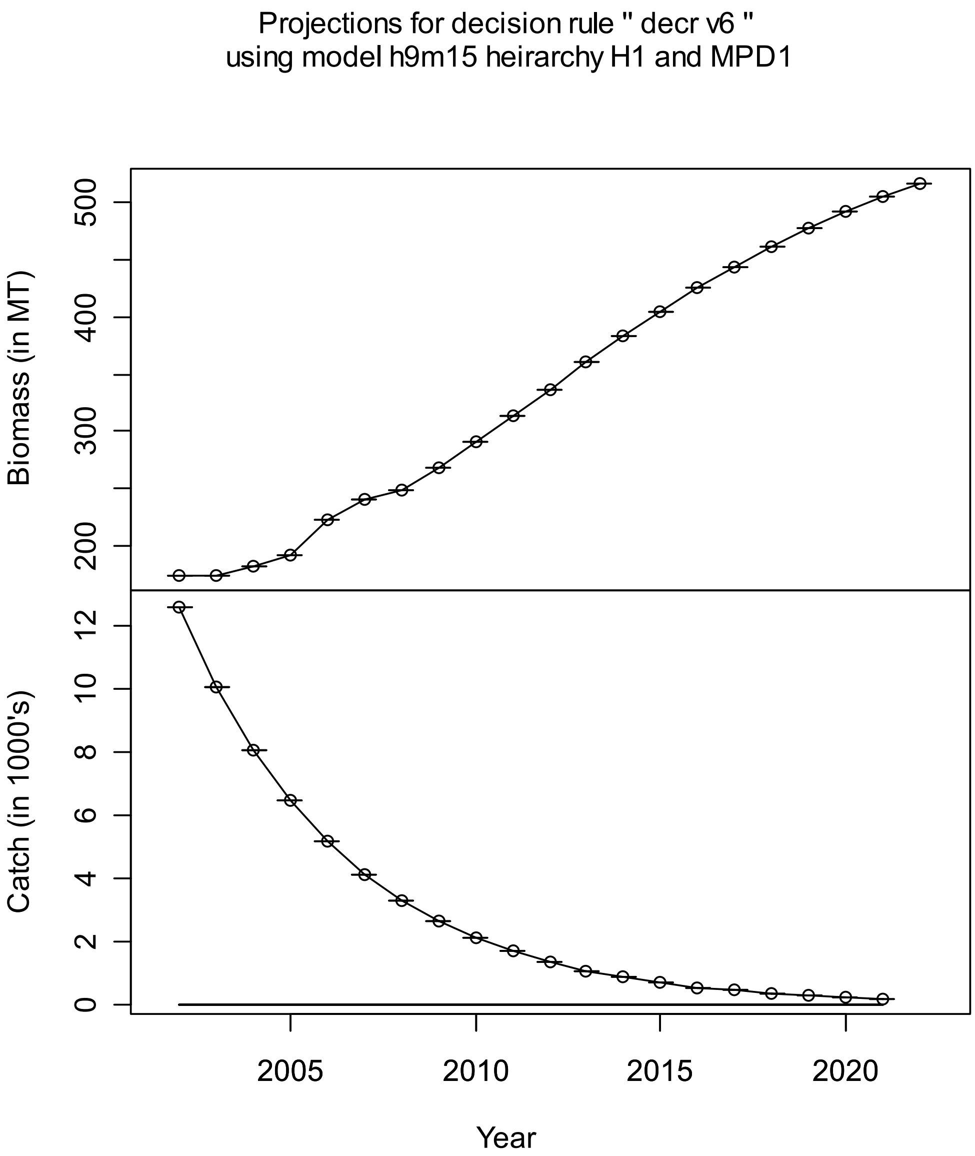

Projections for decision rule " decr v6 "
using model h3m10 heirarchy H1 and MPD1
Biomass (in MT)
170
190
210
230
Catch (in 1000's)
0
2
4
6
8
10
12
2005
2010
2015
2020
Year

Figure A3-5.


Projections for decision rule " decr v6 "
using model h9m15 heirarchy H1 and MPD1
Biomass (in MT)
200
300
400
500
Catch (in 1000's)
0
2
4
6
8
10
12
2005
2010
2015
2020
Year

A4. MeanCPUE (based on average LL1 CPUE)

A4.1. Description of rule

A4.1.1. Overview

MeanCPUE is a simple decision rule based on the CPUE abundance of fish in age classes *a* and above (denoted CPUE_*a*+). In brief, the most recently available value of CPUE_*a*+ (which is the value for current year minus 2) is compared with the average value of CPUE_*a*+ over the previous *n* years. Both *a* and *n* are tuning parameters of the rule. If the most recent value has changed from the previous *n*-year average by less than a minimum percent (*min_change*), then the TAC is left unchanged from the previous year. If it has increased or decreased by more than *min_change*, but less than a maximum percent (*max_change*), then the TAC is increased or decreased proportionally by how much CPUE_*a*+ has changed. Finally, if CPUE_*a*+ has increased by more than *max_change*, then the TAC is increased by 1+*max_change* times; similarly, if it has decreased by more than *max_change*, then the TAC is decreased by $1-$*max_change* times. The values of *min_change* and *max_change* are also tuning parameters of the rule.

A4.1.2. Mathematical description

Let *t* be the current year, and define

$$Z = \frac{CPUE\_a^{+}[t-2]}{\frac{1}{n}\sum_{i=t-2-n}^{t-3} CPUE\_a^{+}[i]}$$

Then, the MP can be expressed in pseudo code as follows:

if (abs $(Z-1) \le min\_change$)
    TAC[t]=TAC[$t-1$];
else if ($Z > (1+min\_change)$ & $Z \le (1+max\_change)$)
    TAC[t]=Z*TAC[$t-1$];
else if ($Z > (1+max\_change)$)
    TAC[t]=$(1+max\_change)$*TAC[$t-1$];
else if ($Z < (1-min\_change)$ & $Z \ge (1-max\_change)$)
    TAC[t]=Z*TAC[$t-1$];
else if ($Z < (1-max\_change)$)
    TAC[t]=$(1-max\_change)$*TAC[$t-1$]

A4.1.3. Versions (tuning parameter values)

Version	a	n	min_change	max_change*
1	0	5	0.05	0.2
2	8	5	0.05	0.2
3	12	5	0.05	0.2
4	0	5	0.1	0.2
5	8	5	0.1	0.2
6	12	5	0.1	0.2

7	0	10	0.1	0.2
8	8	10	0.1	0.2
9	12	10	0.1	0.2

* It was agreed among CSIRO participants that for all MP's the maximum change allowed in the TAC from one year to the next should be 20% (or 0.2).

A4.2. Performance of rule

A4.2.1. Overview

MeanCPUE results in relatively high catch levels but does not perform very well in terms of stock rebuilding. Only under the most productive operating model scenarios does this rule tend to meet the goal of rebuilding the spawning stock biomass to the 1980 level by 2020. Additionally, for 3 of the 8 operating model scenarios, there is a projected decline in spawning stock biomass between the starting year of the projection period (2002) and the final year (2022). These statements are true in general, regardless of the version (i.e. tuning parameters).

Some observations regarding how this rule responds to changes in the tuning parameters can be made. As *a* was increased from 0 to 8 to 12 years (recall that *a* determines the minimum age class that will be included in the CPUE values being compared), the biomass indicators tended to decrease while the catch indicators increased. Increasing the *min_change* parameter from 0.05 to 0.1, such that a 10% instead of a 5% change in the CPUE was required before the TAC was changed, resulted in some fairly significant increases in the catch indicators with relatively small decreases in the biomass indicators, which is a desirable property. The effect of changing the number of years over which the CPUE was averaged from 5 years to 10 years varied depending on the value of *a*.

A4.2.2. Graphics

For clarity, only versions 1, 3, 5 and 8 are included on the summary graphs below. These four versions were selected because they represent the range of results that were obtained. On the graphs for specific operating model scenarios (h3m10 and h9m15 are shown), all nine versions of the rule are included. The worm plots are shown for version 5 under model scenarios h3m10 and h915, but the general trends in biomass and catch are fairly typical of all versions under these scenarios (the trends will vary under different scenarios but the two extremes are shown).

Figure A4-1a.

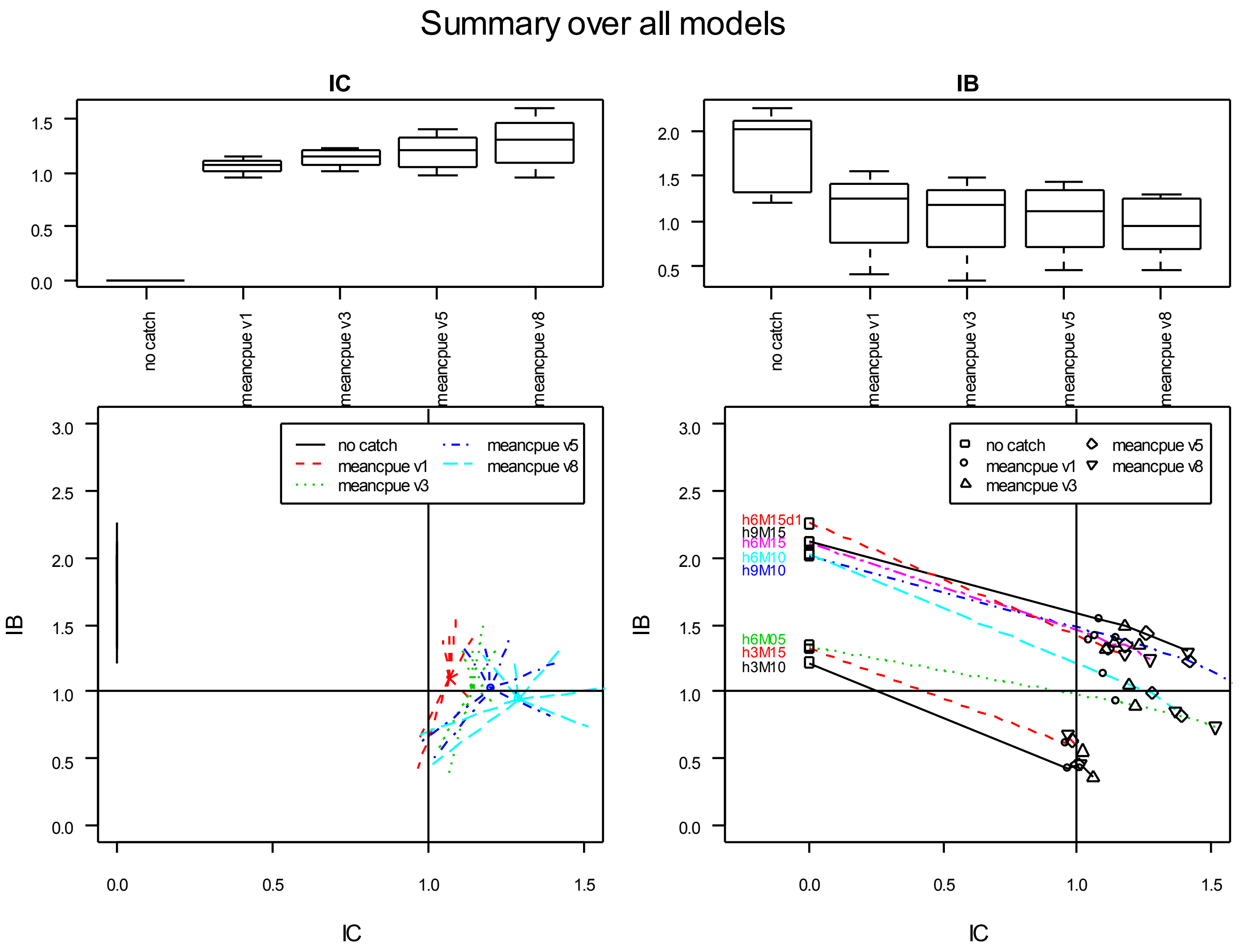

Summary over all models
IC
IB
no catch
meancpue v1
meancpue v3
meancpue v5
meancpue v8
h6M15d1
h9M15
h6M15
h6M10
h9M10
h6M05
h3M15
h3M10

Figure A4-1b.

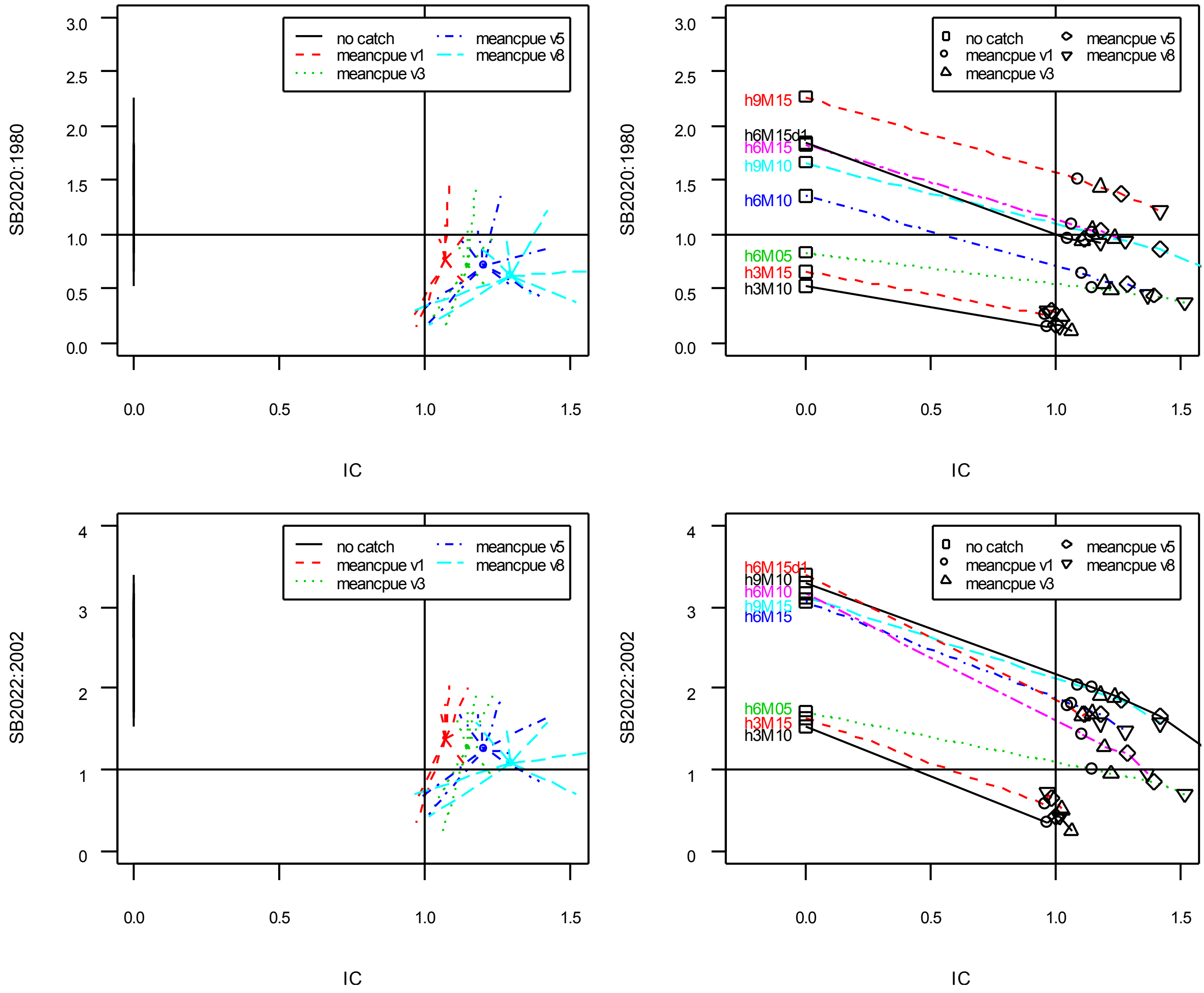
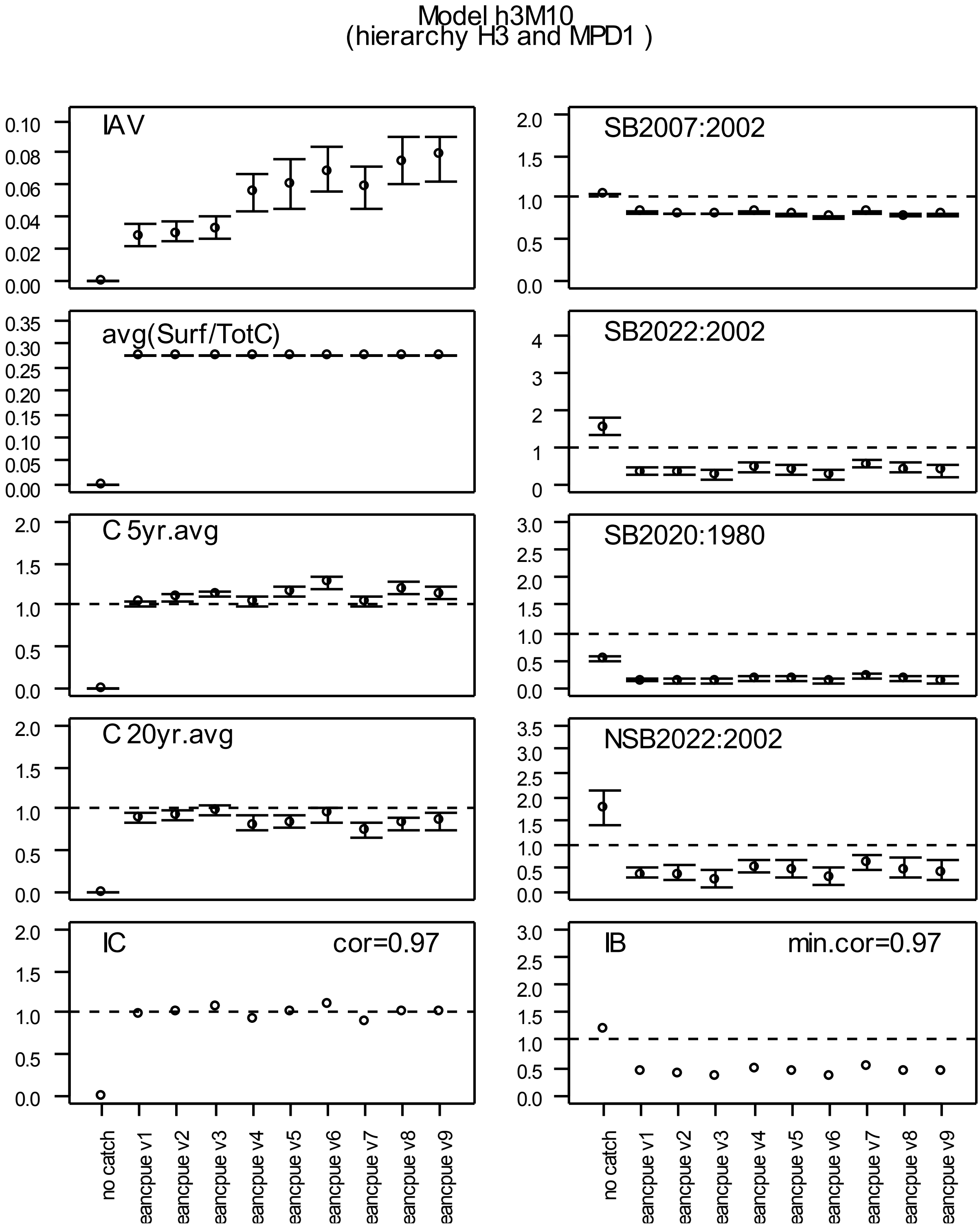

no catch
meancpue v1
meancpue v3
meancpue v5
meancpue v8
SB2020:1980
SB2022:2002
IC
h9M15
h6M15d1
h6M15
h9M10
h6M10
h6M05
h3M15
h3M10

Figure A4-2.



Model h3M10
(hierarchy H3 and MPD1)
IAV
SB2007:2002
avg(Surf/TotC)
SB2022:2002
C 5yr.avg
SB2020:1980
C 20yr.avg
NSB2022:2002
IC
cor=0.97
IB
min.cor=0.97
no catch
eancpue v1
eancpue v2
eancpue v3
eancpue v4
eancpue v5
eancpue v6
eancpue v7
eancpue v8
eancpue v9

Figure A4-3.

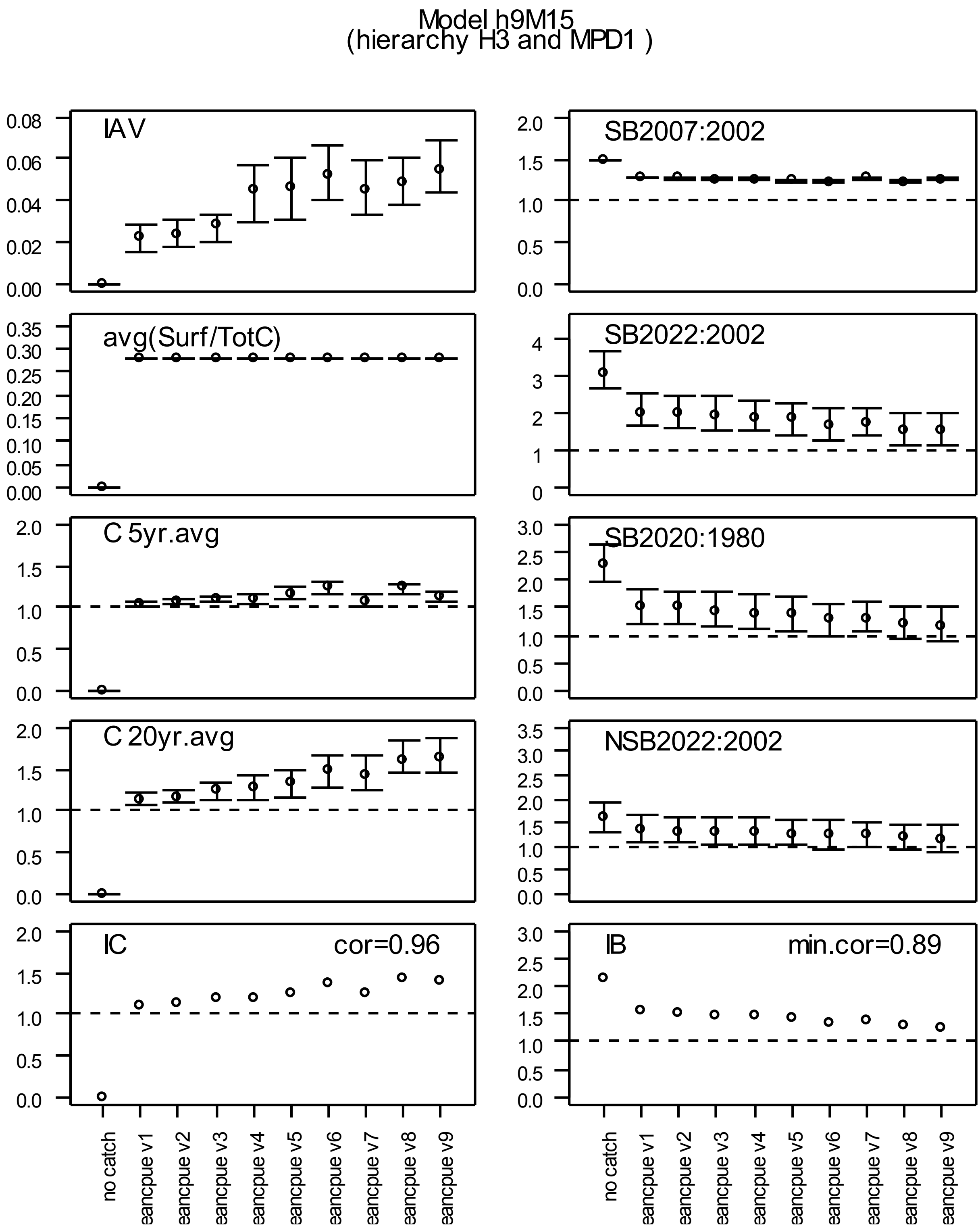

Model h9M15
(hierarchy H3 and MPD1)
IAV
SB2007:2002
avg(Surf/TotC)
SB2022:2002
C 5yr.avg
SB2020:1980
C 20yr.avg
NSB2022:2002
IC
cor=0.96
IB
min.cor=0.89
no catch
eancpue v1
eancpue v2
eancpue v3
eancpue v4
eancpue v5
eancpue v6
eancpue v7
eancpue v8
eancpue v9

Figure A4-4.

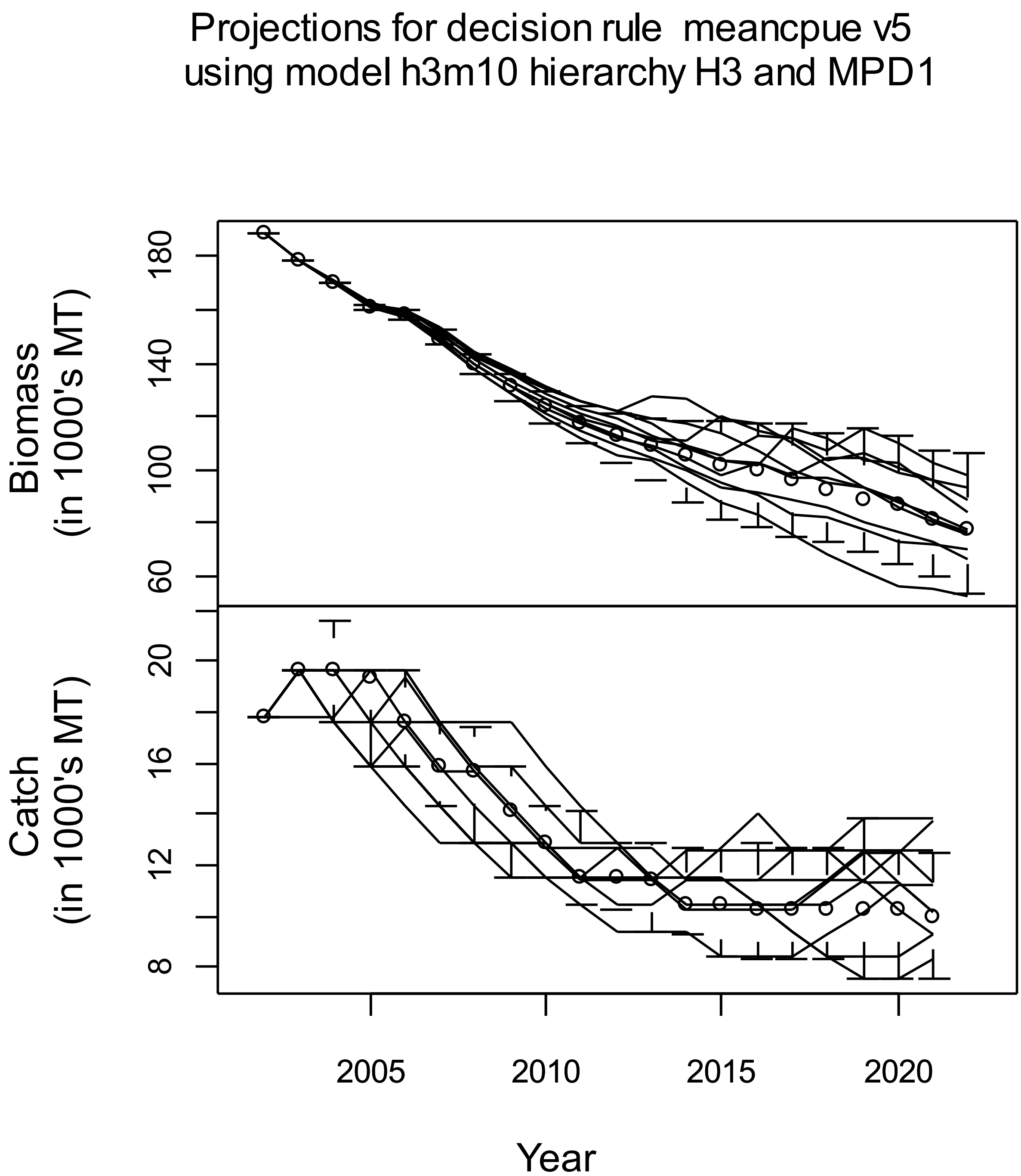
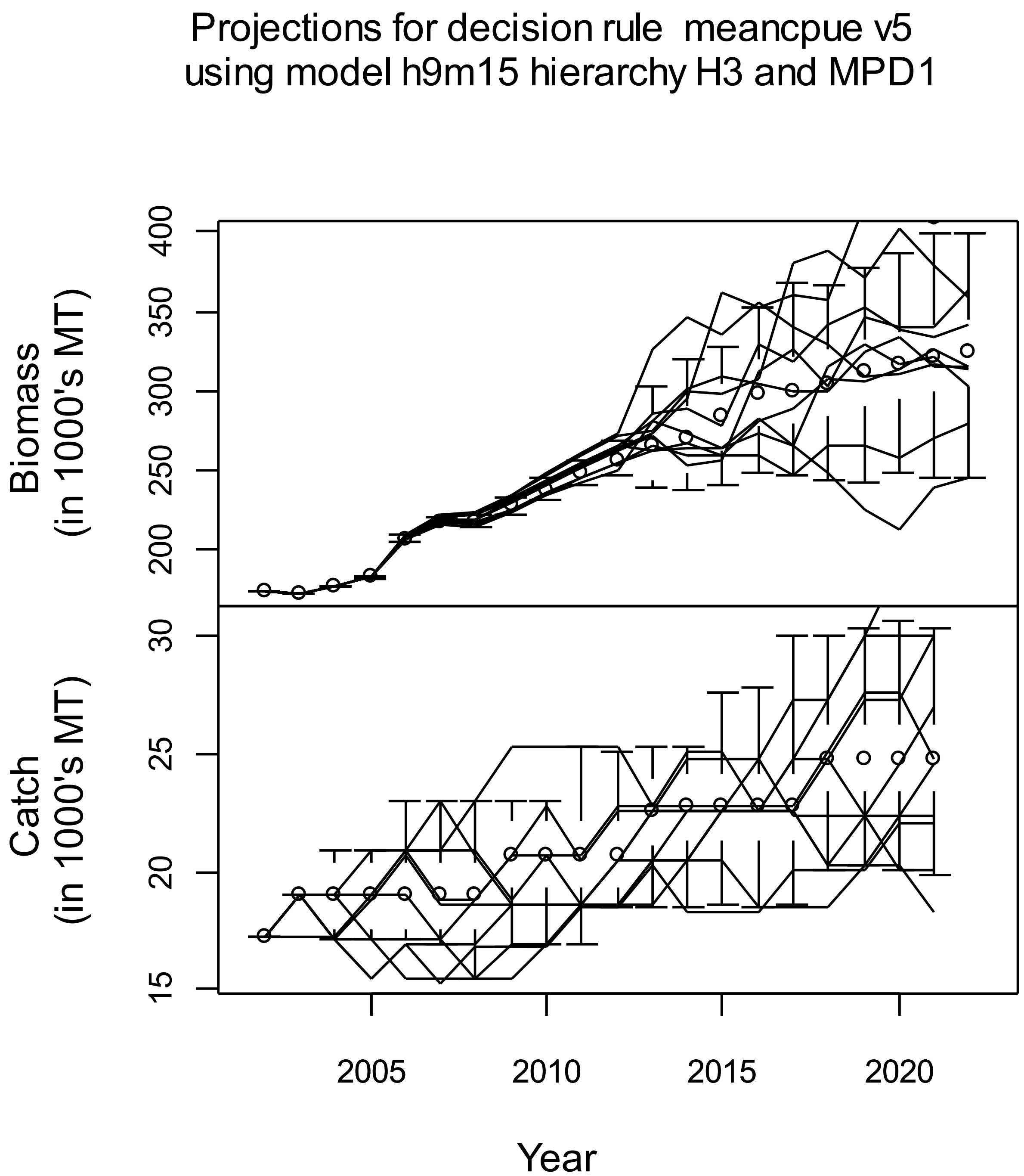

Projections for decision rule meancpue v5
using model h3m10 hierarchy H3 and MPD1
Biomass
(in 1000's MT)
180
140
100
60
Catch
(in 1000's MT)
20
16
12
8
2005
2010
2015
2020
Year

Figure A4-5.


Projections for decision rule meancpue v5
using model h9m15 hierarchy H3 and MPD1
Biomass
(in 1000's MT)
200
250
300
350
400
Catch
(in 1000's MT)
15
20
25
30
2005
2010
2015
2020
Year

A5. CPUE (based on trend in LL1 CPUE)

A5.1. Description of the rule

A5.1.1. Overview

This is the example rule presented in the OM User's Manual. The trend in the log(nominal CPUE) over the last *n* years is used to adjust the TAC.

A5.1.2. Mathematical description

$$TAC_{y+1} = \omega TAC_y + (1-\omega) TAC_y (1+\kappa\lambda_n)$$

where,

ω is the carryover parameter

κ is the weight given to the log(CPUE) slope

λ_n is the slope of the regression of log(CPUE) vs. time over the last *n* years

A5.1.3. Versions (tuning parameter values)

Rule version	Details			
v4	$\omega = 0.8$	$\kappa = 0.75$	$n = 5$	TAC every year
v5	$\omega = 0.8$	$\kappa = 1.0$	$n = 5$	TAC every year
v7	$\omega = 0.8$	$\kappa = 2.5$	$n = 5$	TAC every year
v8	$\omega = 0.8$	$\kappa = 5.0$	$n = 5$	TAC every year
v9	$\omega = 0.8$	$\kappa = 10$	$n = 5$	TAC every year

A5.2. Performance of rule

A5.2.1. Overview

Initial test showed that values below 0.8 for the carryover parameter had no effect on the performance of this decision rule. The log(CPUE) trend used by the decision rule never indicates that a decrease in TAC of more than 20% is necessary.

Rule performance is similar for a regression time window of either 5 or 10 years.

For versions where $\kappa \le 0.75$ there is large stock biomass variability but virtually no variability in catch. Some variability exists for $\kappa = 1.0$ but there is no associated decrease in stock biomass variability.

A potential improvement to this rule would treat positive and negative trends differently. The main shortcoming of this rule comes from the fact that it does not reduce the catch enough to avoid crashing the unproductive OM scenarios (h3M10 and h3M15). If the CPUE trend was used in conjunction with an indicator of the stock productivity (estimate the *r* parameter in a Fox model for example), the responsiveness of the CPUE decision rule could be adjusted accordingly.

A5.2.2. Graphics

Figure A5-1a.

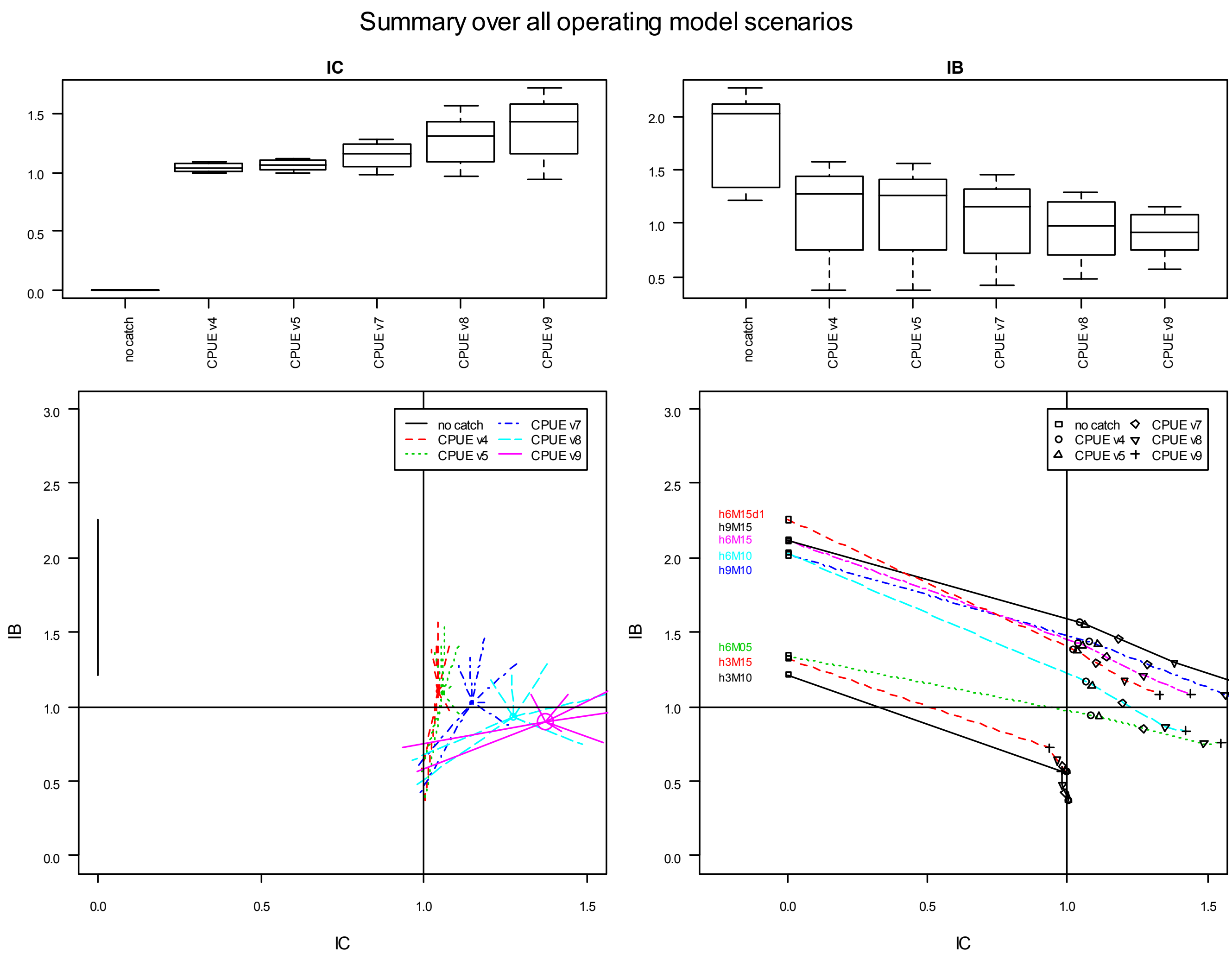

Summary over all operating model scenarios
IC
IB
no catch
CPUE v4
CPUE v5
CPUE v7
CPUE v8
CPUE v9
h6M15d1
h9M15
h6M15
h6M10
h9M10
h6M05
h3M15
h3M10
IC
IB

Figure A5-1b.

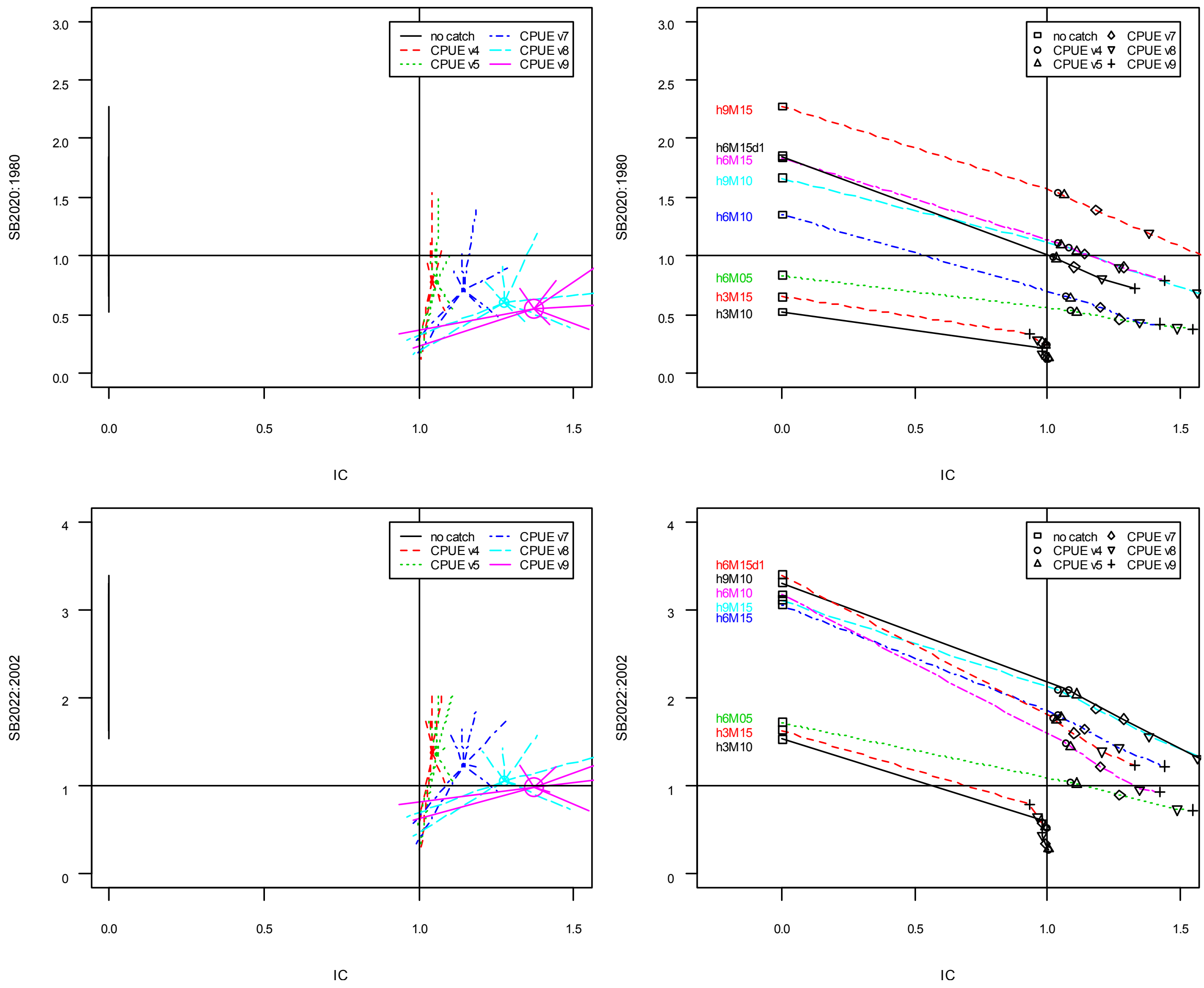
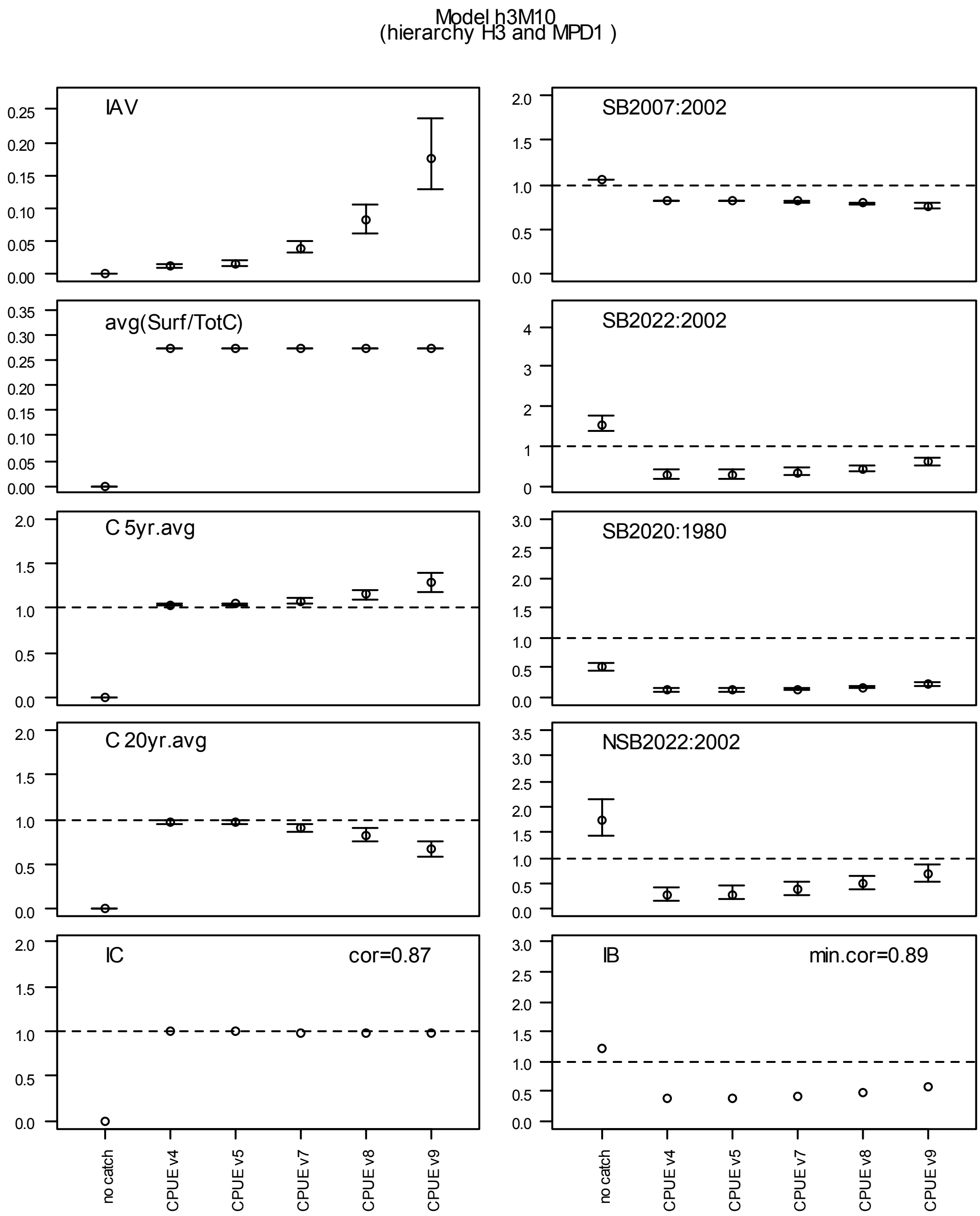

SB2020:1980
IC
no catch
CPUE v4
CPUE v5
CPUE v7
CPUE v8
CPUE v9
h9M15
h6M15d1
h6M15
h9M10
h6M10
h6M05
h3M15
h3M10
SB2022:2002
h6M15d1
h9M10
h6M10
h9M15
h6M15
h6M05
h3M15
h3M10

Figure A5-2.



Model h3M10
(hierarchy H3 and MPD1)
IAV
SB2007:2002
avg(Surf/TotC)
SB2022:2002
C 5yr.avg
SB2020:1980
C 20yr.avg
NSB2022:2002
IC
cor=0.87
IB
min.cor=0.89
no catch
CPUE v4
CPUE v5
CPUE v7
CPUE v8
CPUE v9

Figure A5-3.

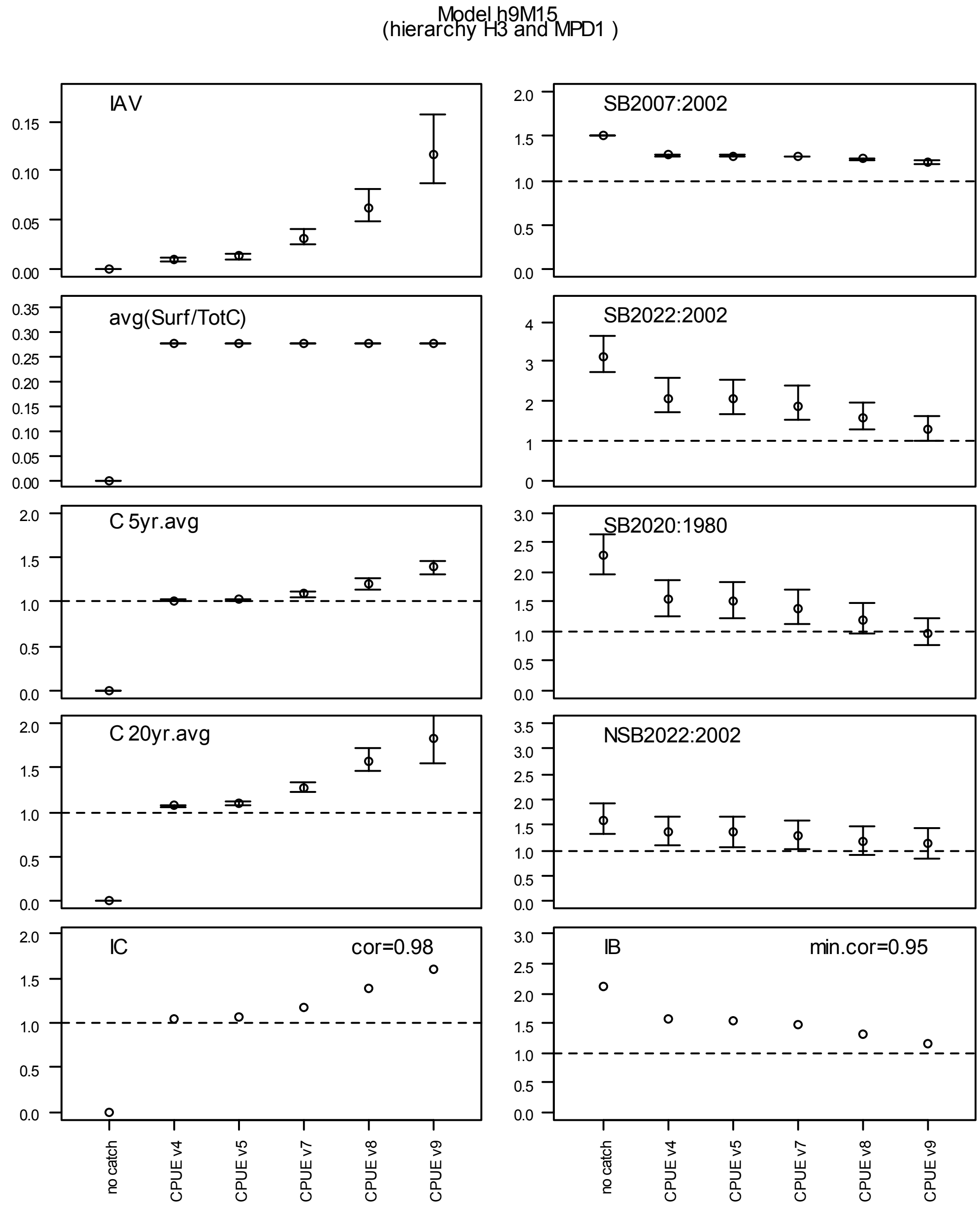

Model h9M15
(hierarchy H3 and MPD1)
IAV
SB2007:2002
avg(Surf/TotC)
SB2022:2002
C 5yr.avg
SB2020:1980
C 20yr.avg
NSB2022:2002
IC
cor=0.98
IB
min.cor=0.95
no catch
CPUE v4
CPUE v5
CPUE v7
CPUE v8
CPUE v9

Figure A5-4.

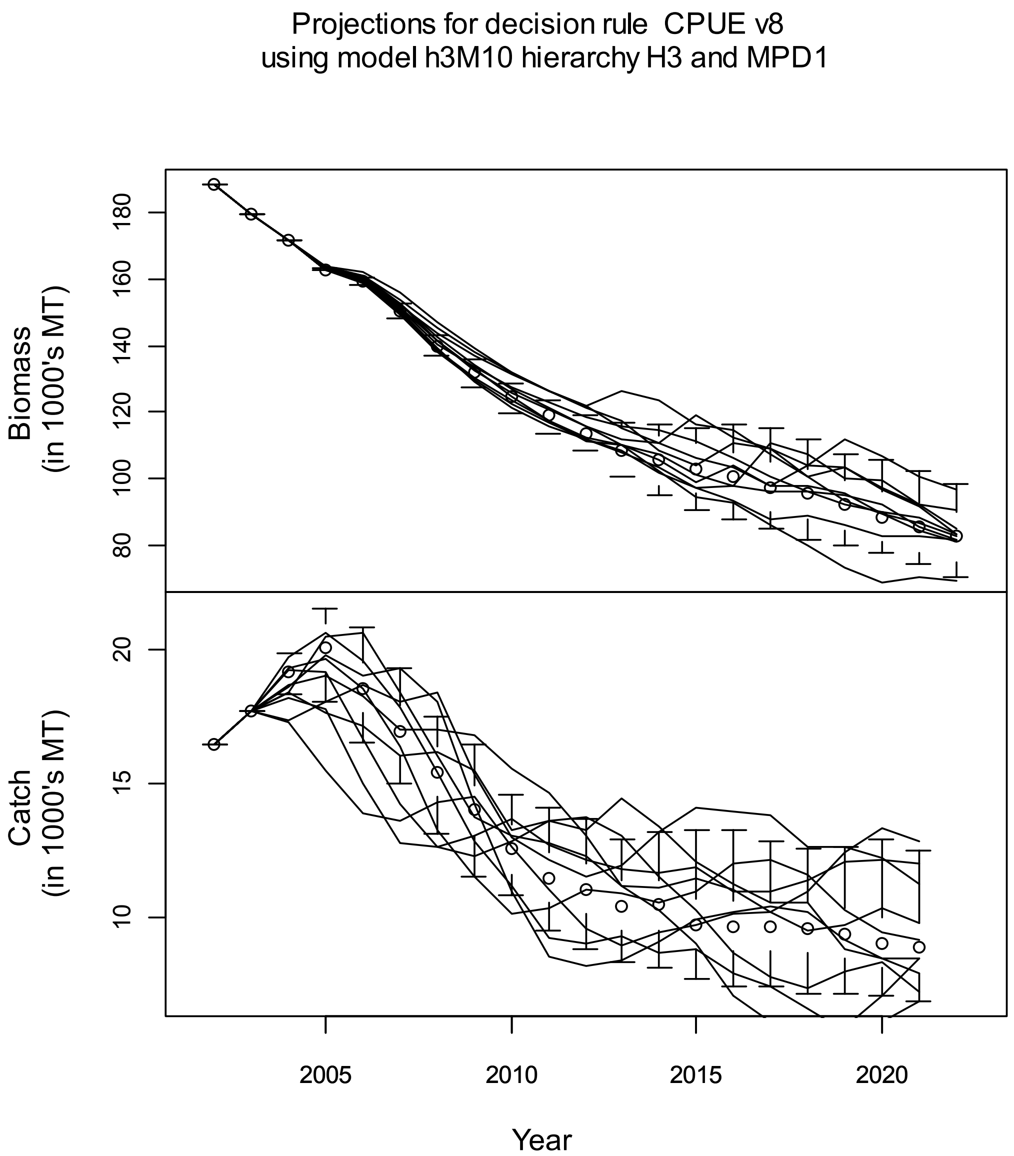
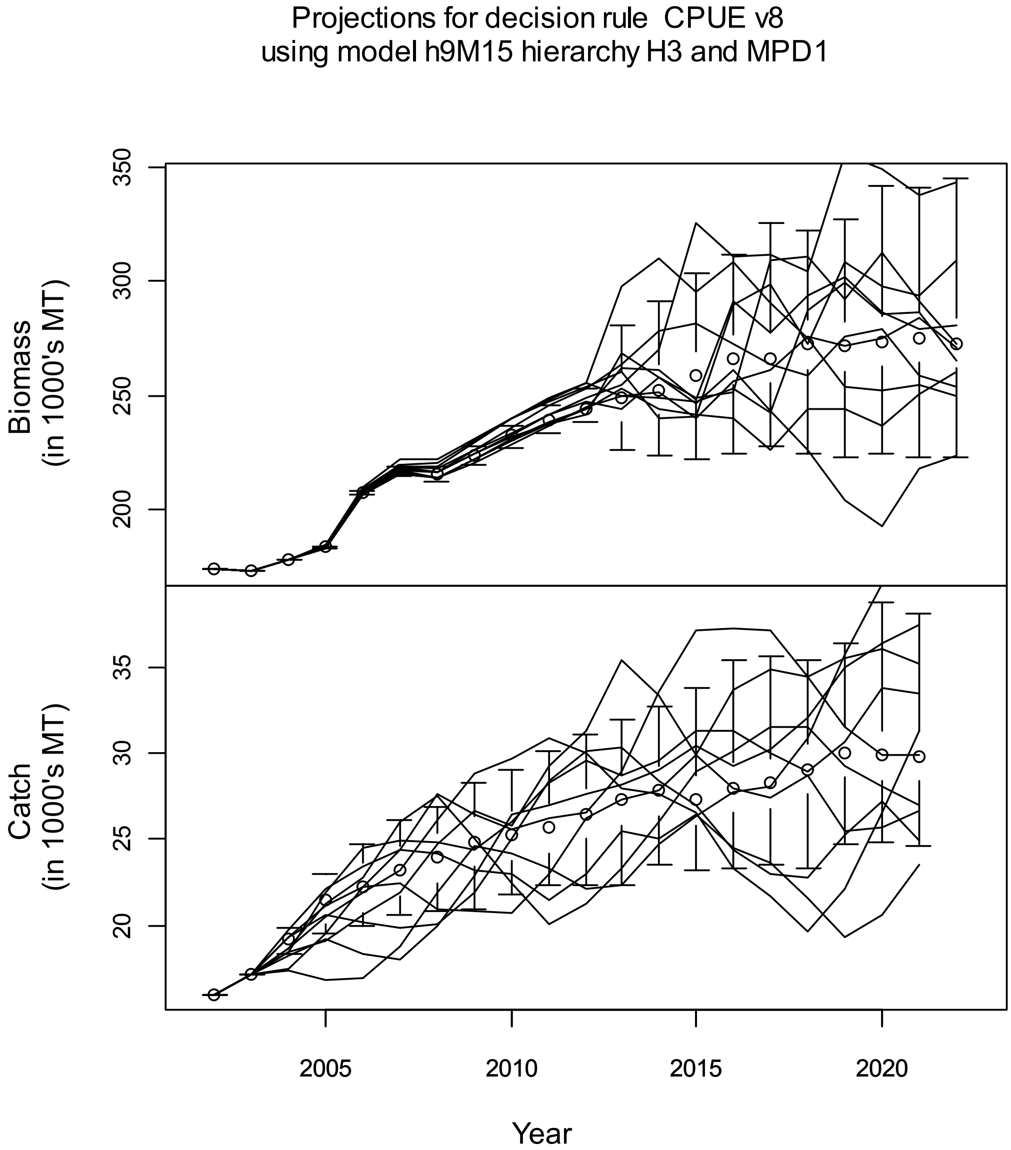

Projections for decision rule CPUE v8
using model h3M10 hierarchy H3 and MPD1
Biomass
(in 1000's MT)
80
100
120
140
160
180
Catch
(in 1000's MT)
10
15
20
2005
2010
2015
2020
Year

Figure A5-5.


Projections for decision rule CPUE v8
using model h9M15 hierarchy H3 and MPD1
Biomass
(in 1000's MT)
350
300
250
200
Catch
(in 1000's MT)
35
30
25
20
2005
2010
2015
2020
Year

A6. CPUE_age (age-aggregated CPUE)

A6.1. Description of the rule

A6.1.1. Overview

This rule builds upon "CPUE". Age-aggregated CPUE is used instead of using the nominal CPUE.
The idea is to use signals from 3 usual age aggregations, ages4-6, ages8-11 and age12+ to orient the TAC.

A6.1.2. Mathematical description

$$TAC_{y+1} = \omega TAC_y + (1-\omega)TAC_y\left(1+\kappa^{4-6}\lambda_x^{4-6}\right) + (1-\omega)TAC_y\left(1+\kappa^{8-11}\lambda_y^{8-11}\right) + (1-\omega)TAC_y\left(1+\kappa^{12+}\lambda_z^{12+}\right)$$

where,

ω is the carryover parameter

λ_x^{4-6} is the slope of the regression of log(ages4-6 CPUE) vs. time over the last x years

κ^{4-6} is the weight given to λ_x^{4-6}

λ_y^{8-11} is the slope of the regression of log(ages8-11 CPUE) vs. time over the last y years

κ^{8-11} is the weight given to λ_y^{8-11}

λ_z^{12+} is the slope of the regression of log(ages12+ CPUE) vs. time over the last z years

κ^{12+} is the weight given to λ_z^{12+}

A6.1.3. Versions (tuning parameter values)

All $\omega = 0.8$.

Rule version	Details			
v4	$\sum \kappa = 0.5$	$\kappa^{4-6} = 0.225$	$\kappa^{8-11} = 0.05$	$\kappa^{12+} = 0.225$
v5	$\sum \kappa = 1.0$	$\kappa^{4-6} = 0.8$	$\kappa^{8-11} = 0.1$	$\kappa^{12+} = 0.1$
v7	$\sum \kappa = 1.0$	$\kappa^{4-6} = 0.1$	$\kappa^{8-11} = 0.1$	$\kappa^{12+} = 0.8$
v9	$\sum \kappa = 1.1$	$\kappa^{4-6} = 0.9$	$\kappa^{8-11} = 0.1$	$\kappa^{12+} = 0.1$
v11	$\sum \kappa = 1.1$	$\kappa^{4-6} = 0.1$	$\kappa^{8-11} = 0.1$	$\kappa^{12+} = 0.9$
v13	$\sum \kappa = 1.25$	$\kappa^{4-6} = 1.0$	$\kappa^{8-11} = 0.125$	$\kappa^{12+} = 0.125$
v15	$\sum \kappa = 1.25$	$\kappa^{4-6} = 0.125$	$\kappa^{8-11} = 0.125$	$\kappa^{12+} = 1.0$

A6.2. Performance of rules

A6.2.1. Overview

Using age-aggregate CPUE time series yields results similar in nature to those obtained using decision rule CPUE. Fox example, the biomass and catch variability remain high under all rule versions explored. Plus, the rule is overly aggressive under unproductive OM scenarios while overshooting the management objective under productive ones.

However, this rule is useful in seeing how the results change when emphasis for setting the TAC is put on the CPUE trend from different age-classes. All other things equal, putting emphasis on the 12+ age group resulted in higher catch and lower biomass than when emphasis was placed on the younger age 4-6 group. Thus, while the performance of this rule as implemented here is relatively poor, it shows the potential of using age-based CPUE time series in conjunction with other rules.

A6.2.2. Graphics

Figure A6-1a.

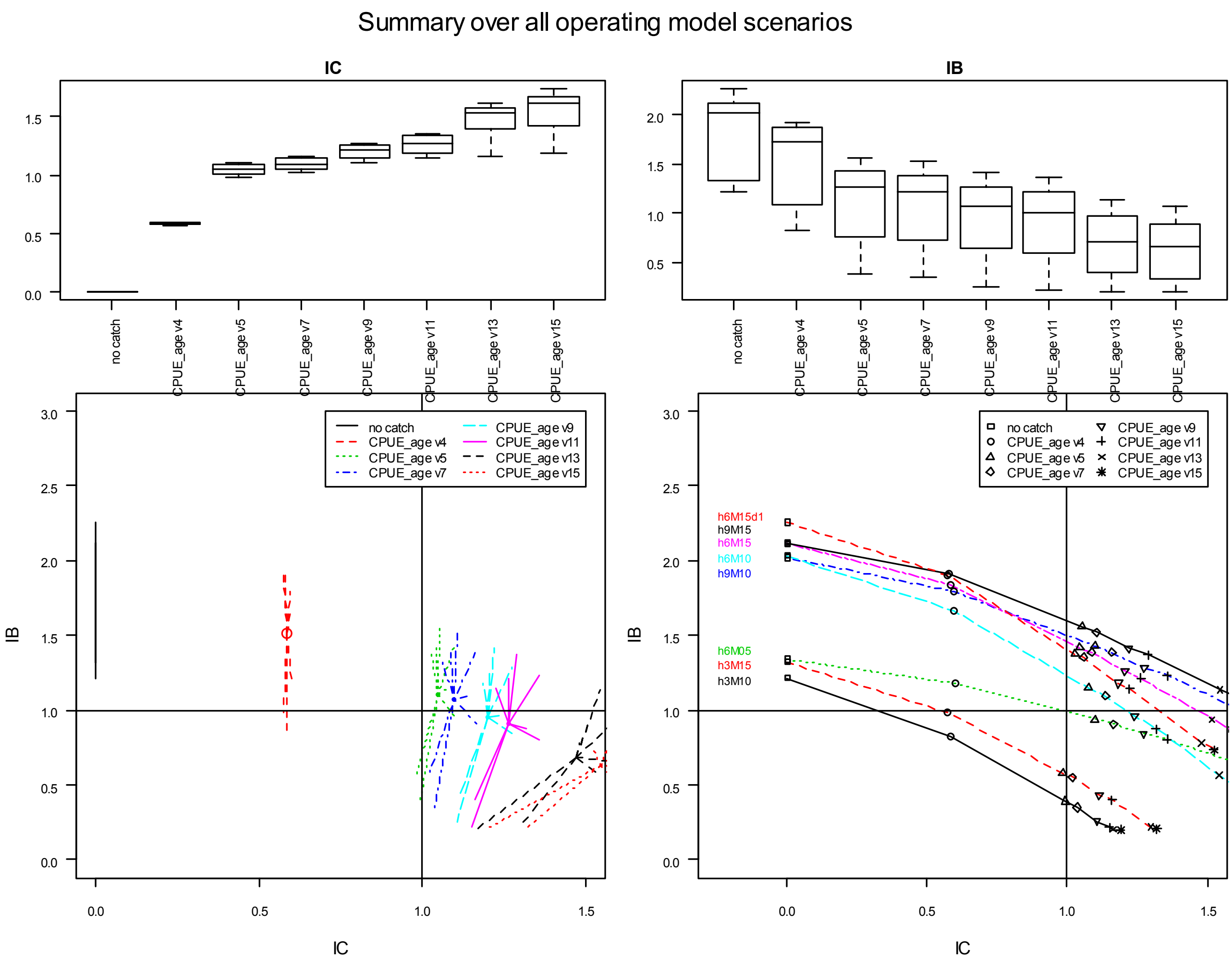

Summary over all operating model scenarios
IC
IB
no catch
CPUE_age v4
CPUE_age v5
CPUE_age v7
CPUE_age v9
CPUE_age v11
CPUE_age v13
CPUE_age v15
h6M15d1
h9M15
h6M15
h6M10
h9M10
h6M05
h3M15
h3M10

Figure A6-1b.

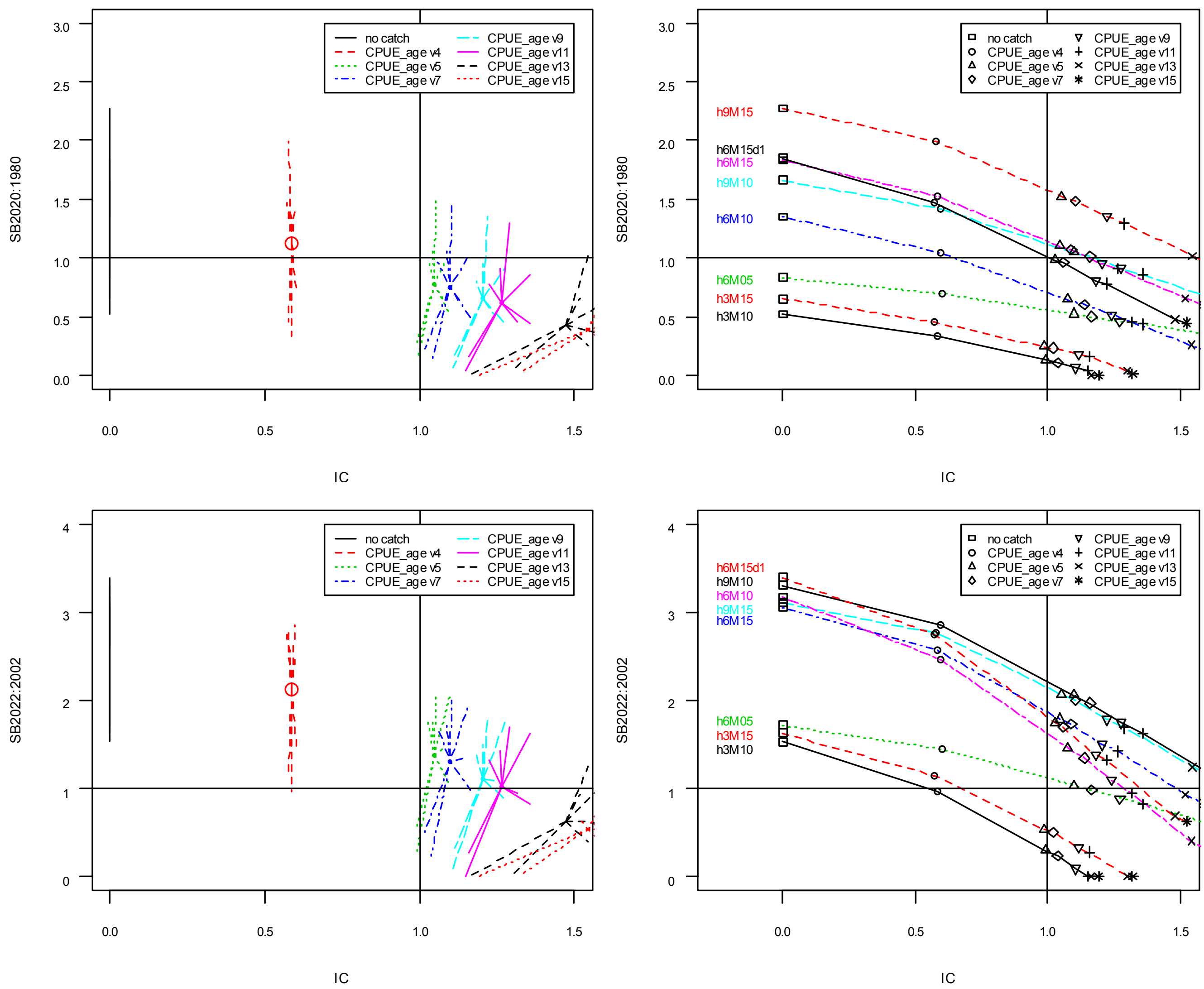

no catch
CPUE_age v4
CPUE_age v5
CPUE_age v7
CPUE_age v9
CPUE_age v11
CPUE_age v13
CPUE_age v15
SB2020:1980
SB2022:2002
IC
h9M15
h6M15d1
h6M15
h9M10
h6M10
h6M05
h3M15
h3M10
h9M15
h6M15

Figure A6-2.

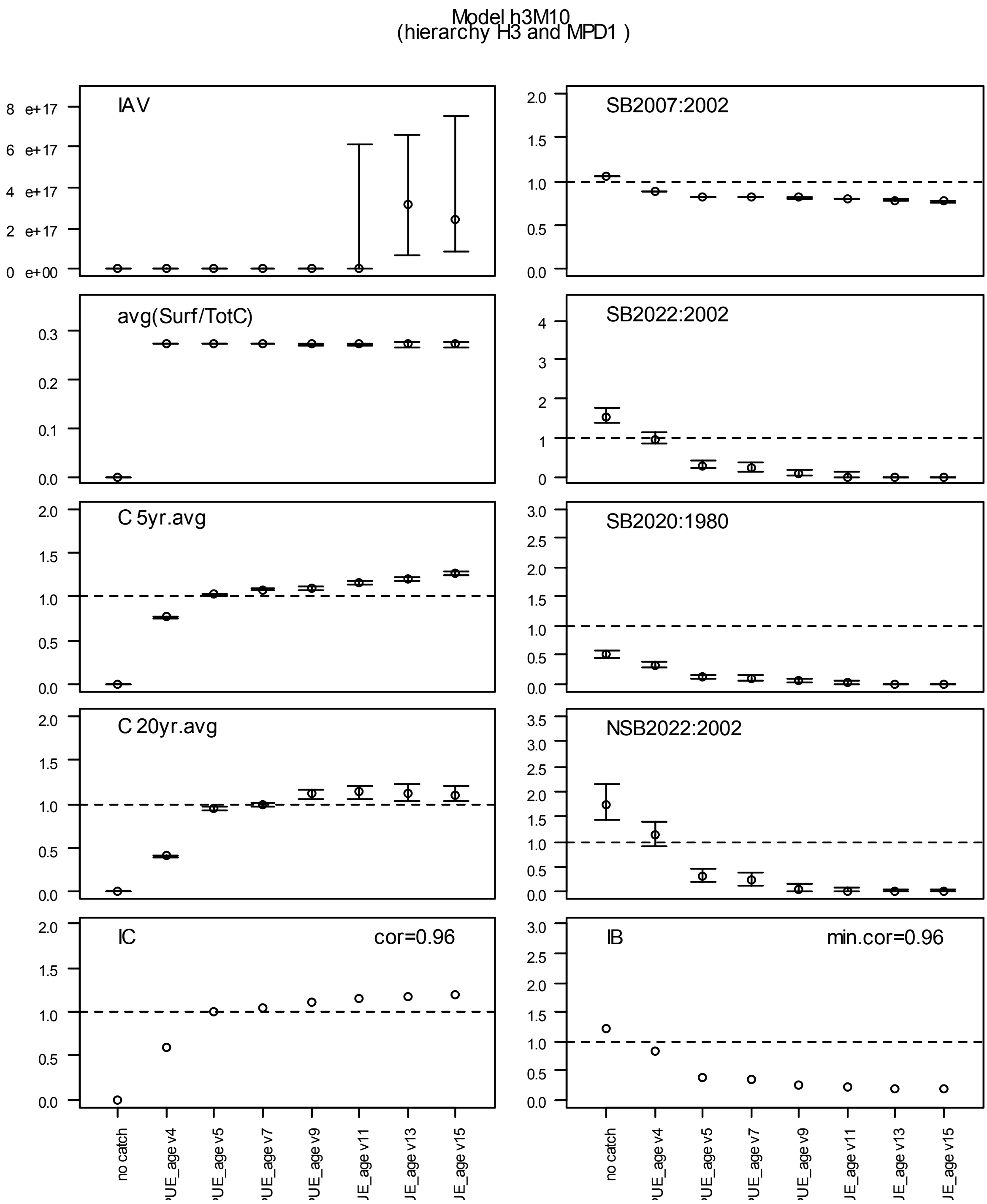

Model h3M10
(hierarchy H3 and MPD1)
IAV
SB2007:2002
avg(Surf/TotC)
SB2022:2002
C 5yr.avg
SB2020:1980
C 20yr.avg
NSB2022:2002
IC
cor=0.96
IB
min.cor=0.96
no catch

Figure A6-3.

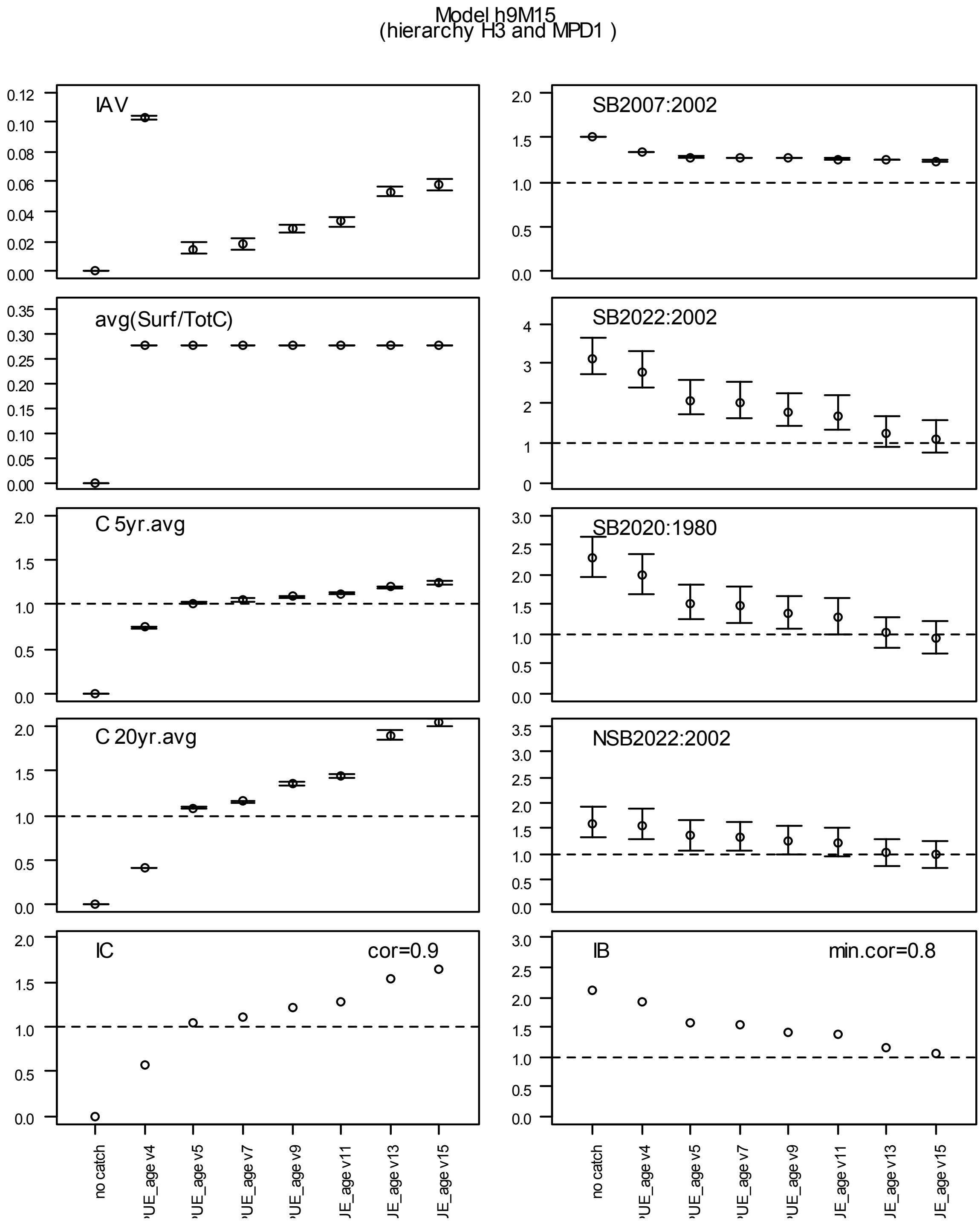

Model h9M15
(hierarchy H3 and MPD1)
IAV
SB2007:2002
avg(Surf/TotC)
SB2022:2002
C 5yr.avg
SB2020:1980
C 20yr.avg
NSB2022:2002
IC
cor=0.9
IB
min.cor=0.8
no catch

Figure A6-4.

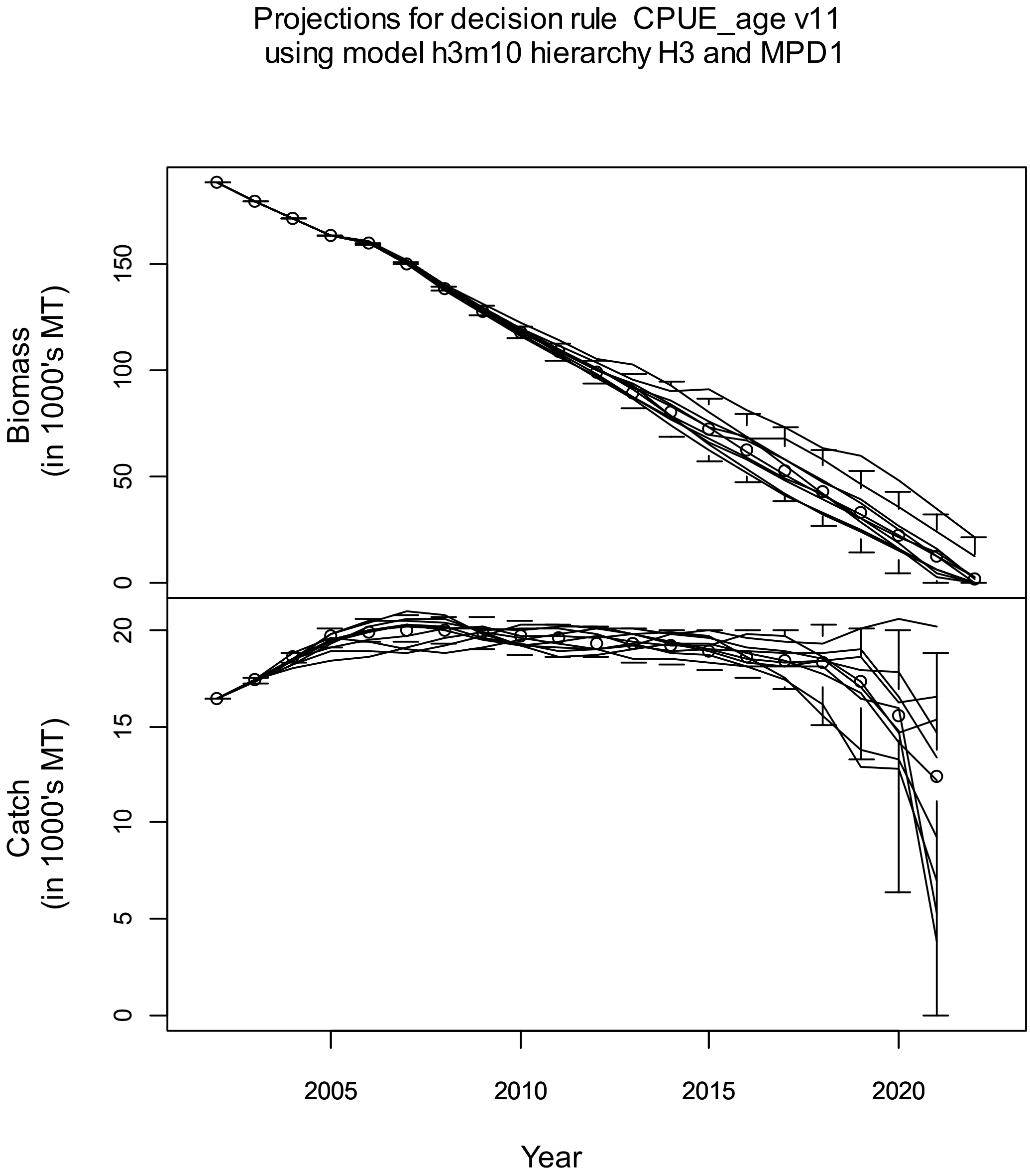
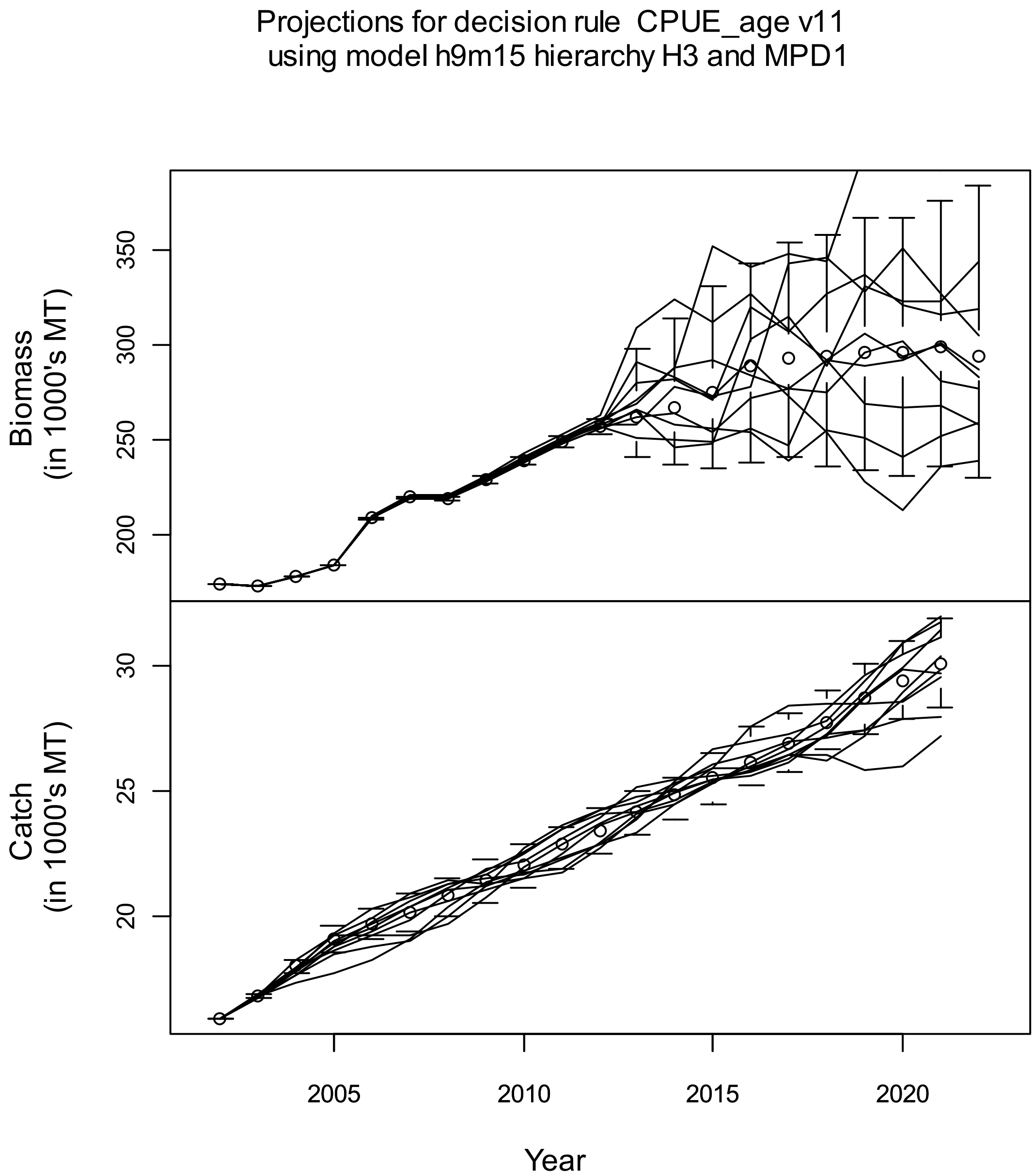

Projections for decision rule CPUE_age v11
using model h3m10 hierarchy H3 and MPD1
Biomass
(in 1000's MT)
0
50
100
150
Catch
(in 1000's MT)
0
5
10
15
20
2005
2010
2015
2020
Year

Figure A6-5.


Projections for decision rule CPUE_age v11
using model h9m15 hierarchy H3 and MPD1
Biomass
(in 1000's MT)
200
250
300
350
Catch
(in 1000's MT)
20
25
30
2005
2010
2015
2020
Year

A7. Stinky (a simple CPUE-based decision rule utilizing absolute rebuilding targets)

A7.1. Description of rule

A7.1.1. Overview: the Stinky rationale

Stinky is a simple decision rule that was intended to explicitly explore rebuilding strategies that forgo short-term catch opportunity for longer-term overall catch increases. The decision rule is based on aggregate longline CPUE biomass indices and considers a number of points:

- There are currently 5, 18 and 20 y time horizons defined to be of interest, and none of the performance indicators defined to date consider economic discounting. (Thus catch in years 16-20 is equally valuable to catch in years 6-10 and Stinky explores the potential for rebuilding, by delaying early catch increases despite evidence of stock increasing).
- We believe the stock to be currently over-fished, with an asymmetrical biological risk function such that short term over-fishing may have a large impact on future production, while lost catch opportunity can be realized at a future date. It is hoped that aggregate biomass CPUE provides sufficient information to prevent short-term over-fishing.
- There is a recognition that the rebuilding target of attaining 1980 spawning biomass by 2020 may be impossible, and an alternative target relative to current biomass level may be specified (compromiseEndTarget below).
- unlike most other rules described in this document (exception of KalTAC), Stinky uses "absolute reference point" decisions, eg CPUE(y)/CPUE(1980)
- There seems to be limited scope for developing decision rules that deviate substantially from the Consvervation / Catch relationships for any particular operating model scenario. Stinky attempts to reduce the range of conservation variability among operating model scenarios, by increasing the variability in catch. (eg flatten the starburst in the risk/catch bivariate summary plots below).

A7.1.2. Implementation details

There are 6 control parameters defined (some are redundant in some versions):

TACInitial – initial TAC prior to firstRebuildYear (can be held constant)
firstRebuildYear – first year of aggressive rebuilding
lastRebuildYear – last year of aggressive rebuilding
TAC2021 – devious option to hammer the stock in the last year of the simulation
maxTACChange – maximum constraint to TAC change between consecutive years
compromiseEndTarget – given the low probability of rebuilding to 1980 levels by 2020, this is a rebuilding target relative to 2002 biomass (active after lastRebuildYear) that should be > 1.

The rule uses simple conditional statements that are most easily represented by the actual implementation code (yr = year for which TAC is to be set; && = logical AND) :

```
double TACTmp;
TACTmp=quota(current_yr-1);  //by default

if(yr<firstRebuildYear) TACTmp=TACInitial;

if(yr >= firstRebuildYear && yr <= lastRebuildYear){
    if(CPUE(yr-2) < CPUE(2002)) TACTmp=0.75*quota(yr-1);
    if(CPUE(yr-2) > 0.8*CPUE(1980)) TACTmp = 1.1*quota(yr-1);
}

if(yr > lastRebuildYear  && yr < 2021){
    if(CPUE(yr-2) > 1.1*compromiseEndTarget*CPUE(2002)){
      TACTmp=1.4*quota(yr-1);
    }
    if(CPUE(yr-2)<0.9*compromiseEndTarget*CPUE(2002)){
      TACTmp=0.6*quota(yr-1);
    }
}

if(yr==2021) { //option to do devious things in final year
    if(TAC2021>0)TACTmp=TAC2021;
}

TAC=TACTmp;


//restrict magnitude of change in TAC
if(TACTmp/quota(yr-1)<(1-maxTACChange)){
    TAC = (1-maxTACChange)*quota(yr-1);
}
if(TACTmp/quota(current_yr-1)>(1+maxTACChange)){
    TAC = (1+maxTACChange)*quota(current_yr-1);
}
```

A7.1.3. Versions (tuning parameter values)

(blank values below indicate that a particular parameter is redundant because of the other parameter values)

Rule version	initial TAC	FirstYearRebuild	lastYearRebuild	maxTAC Change	TAC2021	compromise EndTarget
v1		2000	2025	0.2		
v2		2000	2014	0.2		1.25
v3		2000	2000	0.2		1.05
v4		2000	2000	0.2		1.25
v5	8000	2008	2025	0.2		
v6	8000	2008	2014	0.2		1.25
v7	8000	2004	2014	0.2		1.25
v8	8000	2004	2014	10		1.25

v9	12000	2008	2025	0.2		
v10	12000	2008	2014	0.2		1.25
v11	12000	2004	2014	0.2		1.25
v12	12000	2004	2014	10		1.25
v13	16000	2008	2025	0.2		
v14	16000	2008	2014	0.2		1.25
v15	16000	2004	2014	0.2		1.25
v16	16000	2004	2014	10		1.25
v17*	13000	2004	2015	1000000	1000000	1.5

* a particularly devious and impractical option that tests the scale on the performance indicators graphs

A7.2. Stinky Performance

A7.2.1. Overview

The defined versions of Stinky spanned a substantial range of the Conservation/Catch trade-off space (examples illustrated in Fig. A7-1 to A7-4; the mean performance of versions not shown are all closely aligned along a curved line joining v5 and v14). Worm plots of typical biomass and catch trajectories for the extreme operating model scenarios (H3M10 and H9M15) are indicated in fig. A7-5 and A7-6. for Stinky-v11. Stinky v11 was chosen for comparative purposes on the basis that the "mean" recovery performance was roughly in line with management rebuilding objectives (ie (SSB(2020)/SSB(1980)) and none of the operating model scenarios indicated a mean decline in biomass from 2002 levels (although individual realizations within some scenarios did decline). A few points are evident:

- The initialTAC value had the greatest influence on the Conservation/Catch performance. "Mean recovery" across Operating model scenarios was only achieved in decision rule versions that included a substantial catch decrease in the early years.

- Choice of rebuilding period and constraining the change in TAC between consecutive years to 20% did not result in substantially changed performance

- The effect of compromiseEndTarget was not really explored.

- Stinky tends to have lost economic opportunity on OM scenarios h9M15, and to a lesser extent h6M15 and h6M15d. (ie SSB rebuilding substantially beyond 1980 levels)

- Stinky tends to take higher than average catch in h6M05 (relative to the other OM scenarios) despite falling short on the rebuilding objectives.

- It is not actually clear whether the explicit rebuilding strategy (forgoing short term catch for greater medium term yield) improved performance in the manner intended.

- It would be interesting to investigate to what degree the difficulty in achieving the rebuilding target is a result of using a relative abundance index (CPUE) that is not consistent with "actual" SSB (particularly in 1980). Brief exploration of a similar decision rule based on age-structured CPUE did not result in any obvious improvements in performance.

A7.2.2. Graphics

Fig. A7-1a. Management Procedure Summary Performance: 5 representative versions of Stinky (tested with 8 operating models)

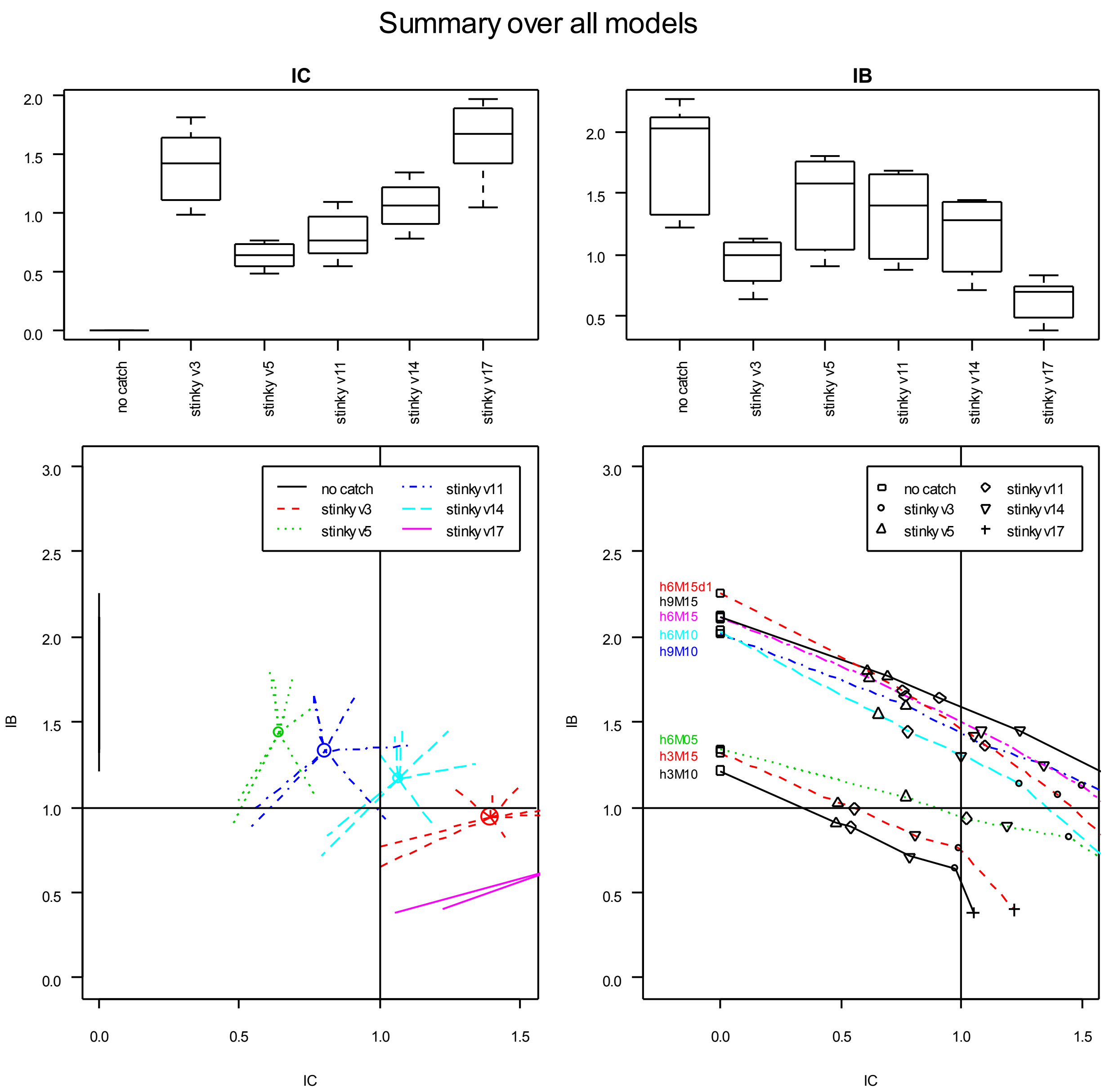


Fig. A7-1b. Management Procedure Summary Performance: 5 representative versions of Stinky tested with 8 operating models (Hierarchy 3)

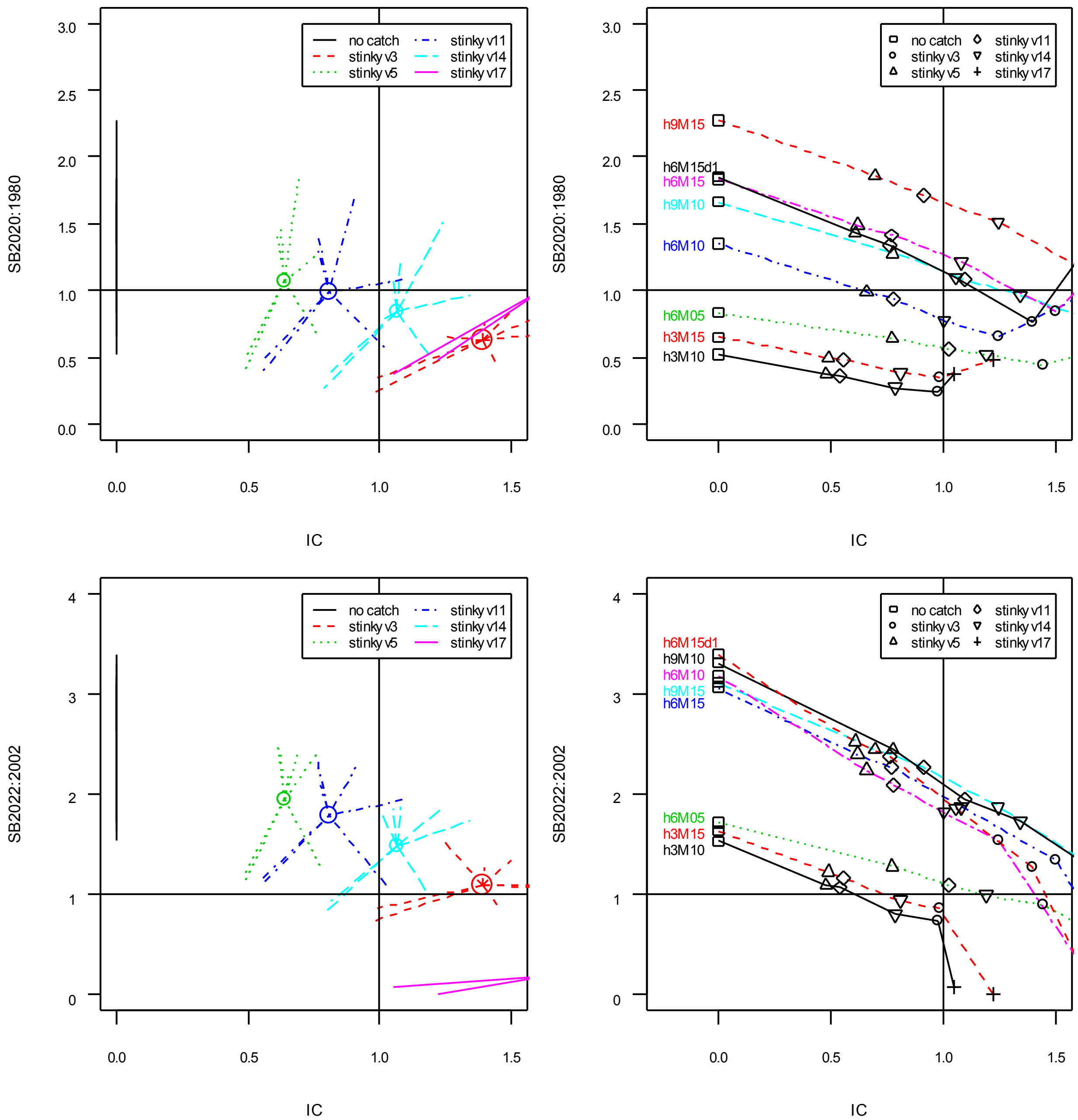


Fig. A7-2. Management Procedure Summary Performance: 5 representative versions of Stinky tested against Operating Model scenario H3M10 (Hierarchy 3)

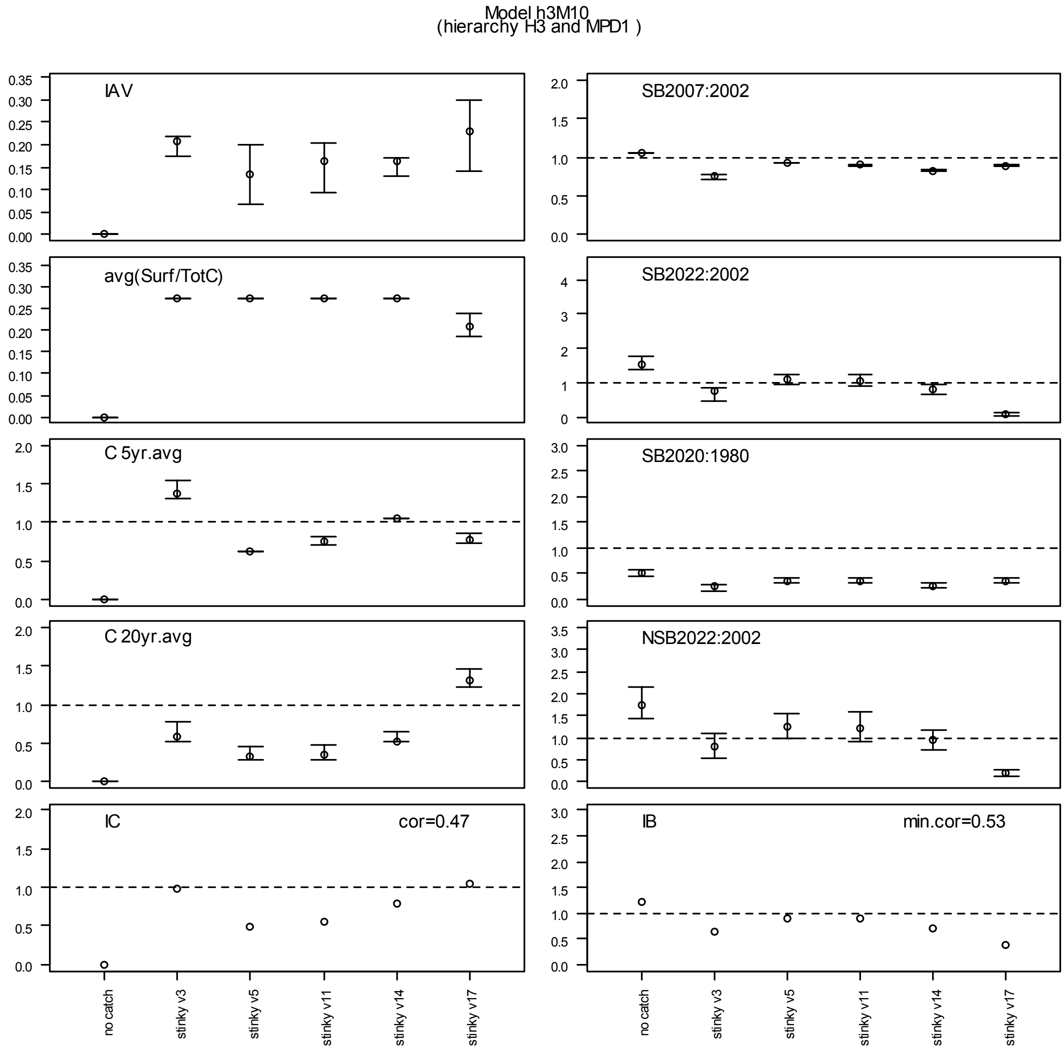


Fig. A7-3. Management Procedure Summary Performance: 5 representative versions of Stinky tested against Operating Model scenario H9M15 (Hierarchy 3)

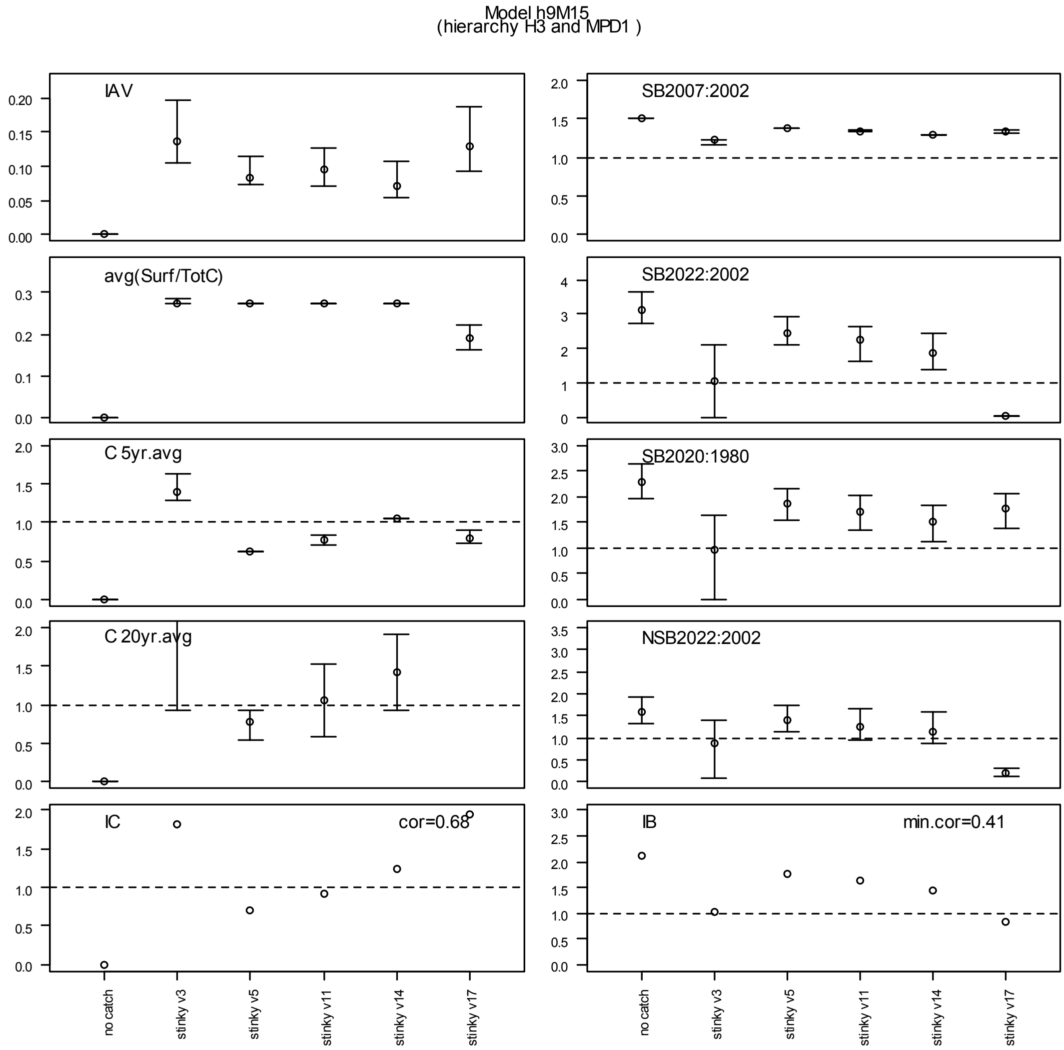


Fig. A7-4. Management Procedure Stinky-v11 summary catch and biomass worm plots for Operating Model scenario H3M10 (Hierarchy 3)

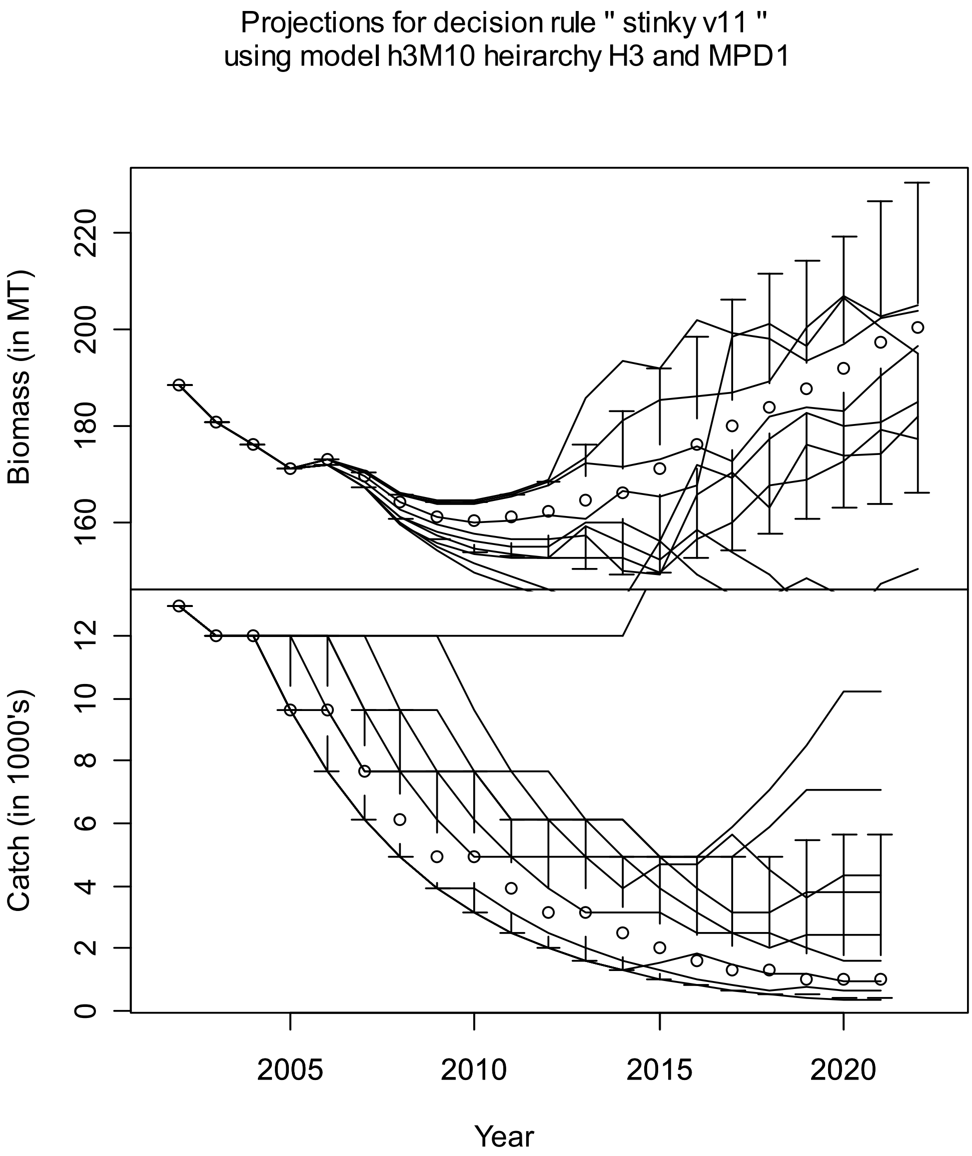
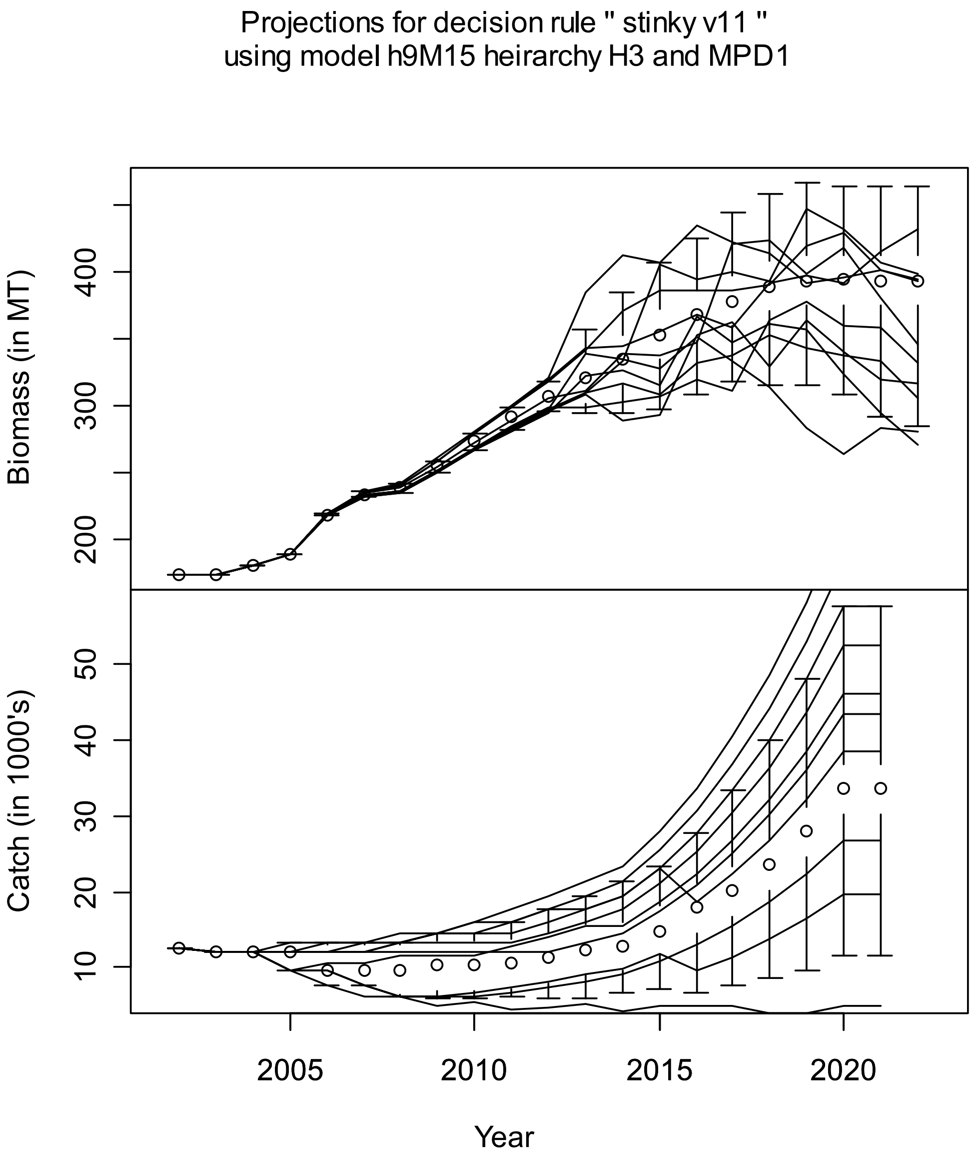


Fig. A7-5. Management Procedure Stinky-v11 summary catch and biomass worm plots for Operating Model scenario H9M15 (Hierarchy 3)



A8. Fox (Fox production model)

A8.1. Description of rule

A8.1.1. Overview

This decision rule fits a Fox surplus production model to the nominal LL1 CPUE. The parameter estimates of *r* and *k* are used to compute MSY and B_{MSY}. The rule sets the TAC to a certain fraction of the exploitation ratio ($F_{MSY} = MSY / B_{MSY}$). This fraction is determined based on the estimated ratio B_y / B_{MSY}.

A8.1.2. Mathematical description

$$TAC_{y+1} = \omega TAC_y + (1-\omega)(B_y (M * F_{MSY}))$$
$$M_y = \eta(B_y / B_{MSY}) + \beta - \eta$$

A8.1.3. Versions (tuning parameter values)

$\omega = 0.8$ for all versions

Rule versions	Details	
v1	$\beta = 0.9$	$\eta = 1.2$
v4	$\beta = 0.7$	$\eta = 1.2$
v7	$\beta = 0.9$	$\eta = 0.9$
v10	$\beta = 0.7$	$\eta = 0.9$

A8.2. Performance of rule

A8.2.1. Overview

The current estimates of stock status are well below B_{MSY} and, subsequently, the rule reduces the TAC in the early years of the simulations. The catch reduction is constrained by the carryover parameter.

This rule is very conservative and performs well in terms of the biomass indicators. The management objective to rebuild the stock to 1980 levels by 2020 is achieved for all OM scenarios that achieve the objective under no catch. However, the catch is unnecessarily reduced for productive OM scenarios.

A potential improvement to this rule would be to use the *r* parameter estimate to adjust the aggressiveness of the catch. This could potentially allow the rule to "soak up" the biomass in excess of the 1980 level by increasing the catch. Decision rule "Fox_var" (below) is a first attempt at implementing such a "learning" rule.

A8.2.2. Graphics

Figure A8-1a.

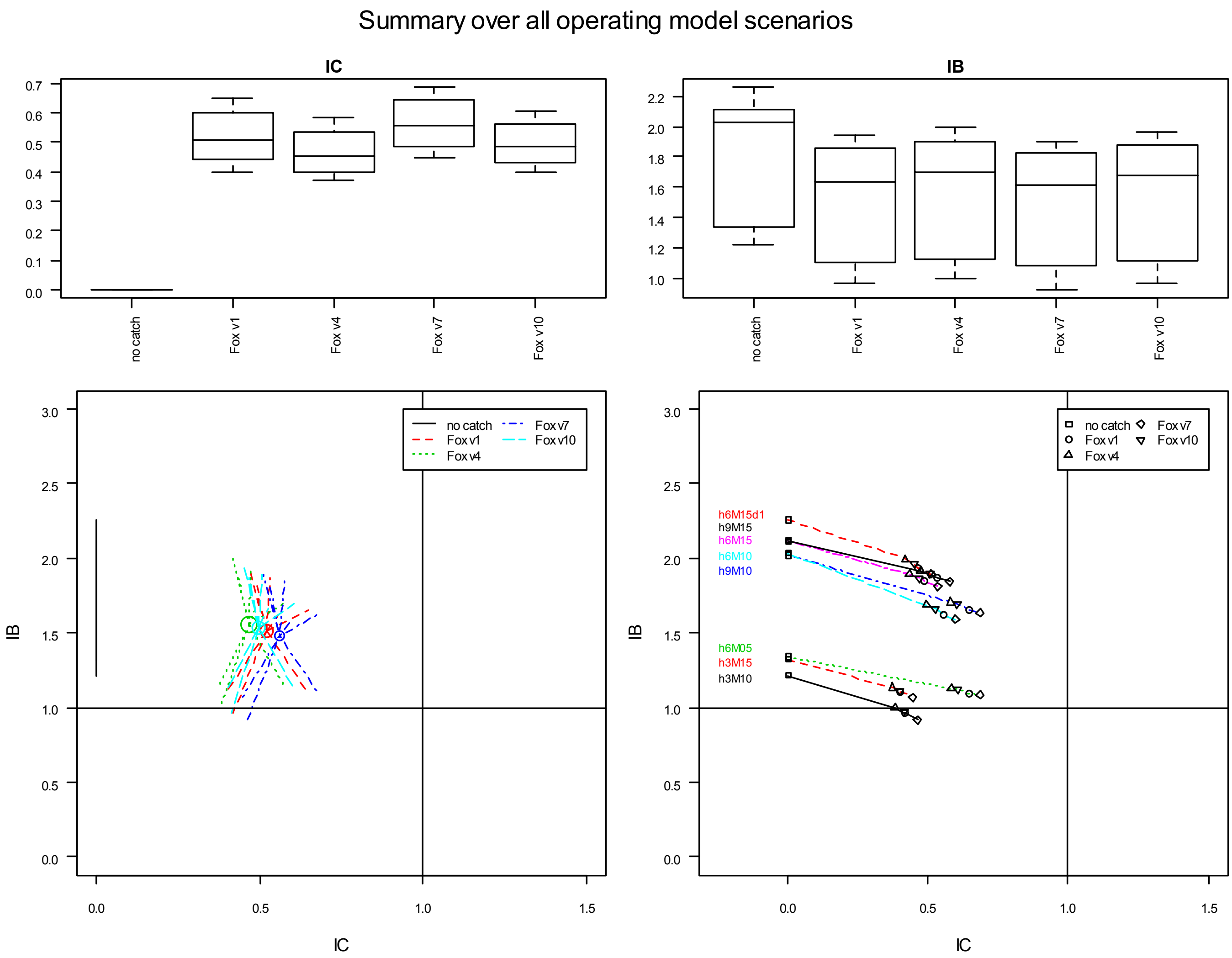

Summary over all operating model scenarios
IC
IB
no catch
Fox v1
Fox v4
Fox v7
Fox v10
h6M15d1
h9M15
h6M15
h6M10
h9M10
h6M05
h3M15
h3M10
IC
IB
IC
IB

Figure A8-1b.

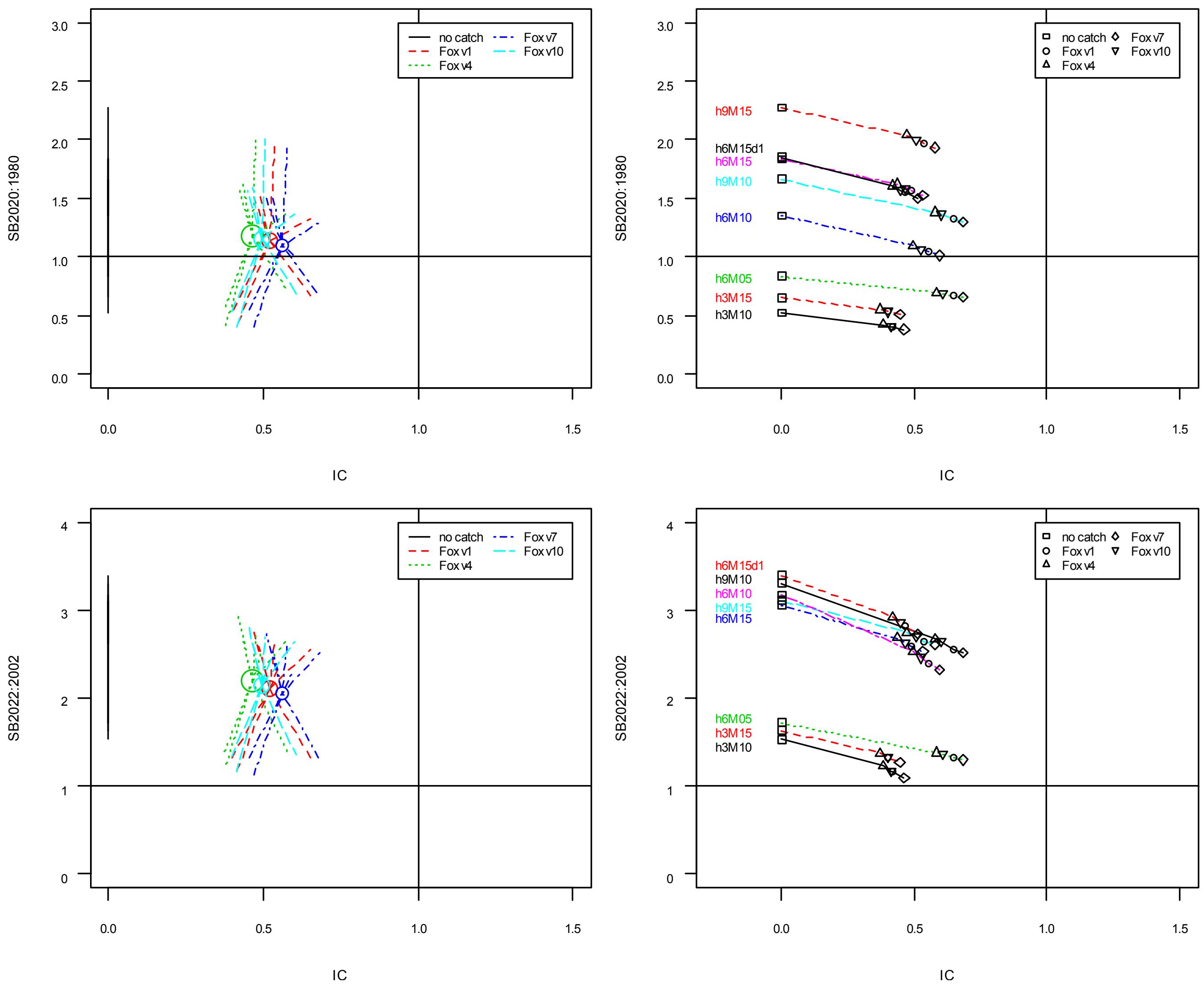

no catch
Fox v1
Fox v4
Fox v7
Fox v10
SB2020:1980
IC
h9M15
h6M15d1
h6M15
h9M10
h6M10
h6M05
h3M15
h3M10
SB2022:2002
h6M15d1
h9M10
h6M10
h9M15
h6M15
h6M05
h3M15
h3M10
0.0
0.5
1.0
1.5
2.0
2.5
3.0
0
1
2
3
4

Figure A8-2.

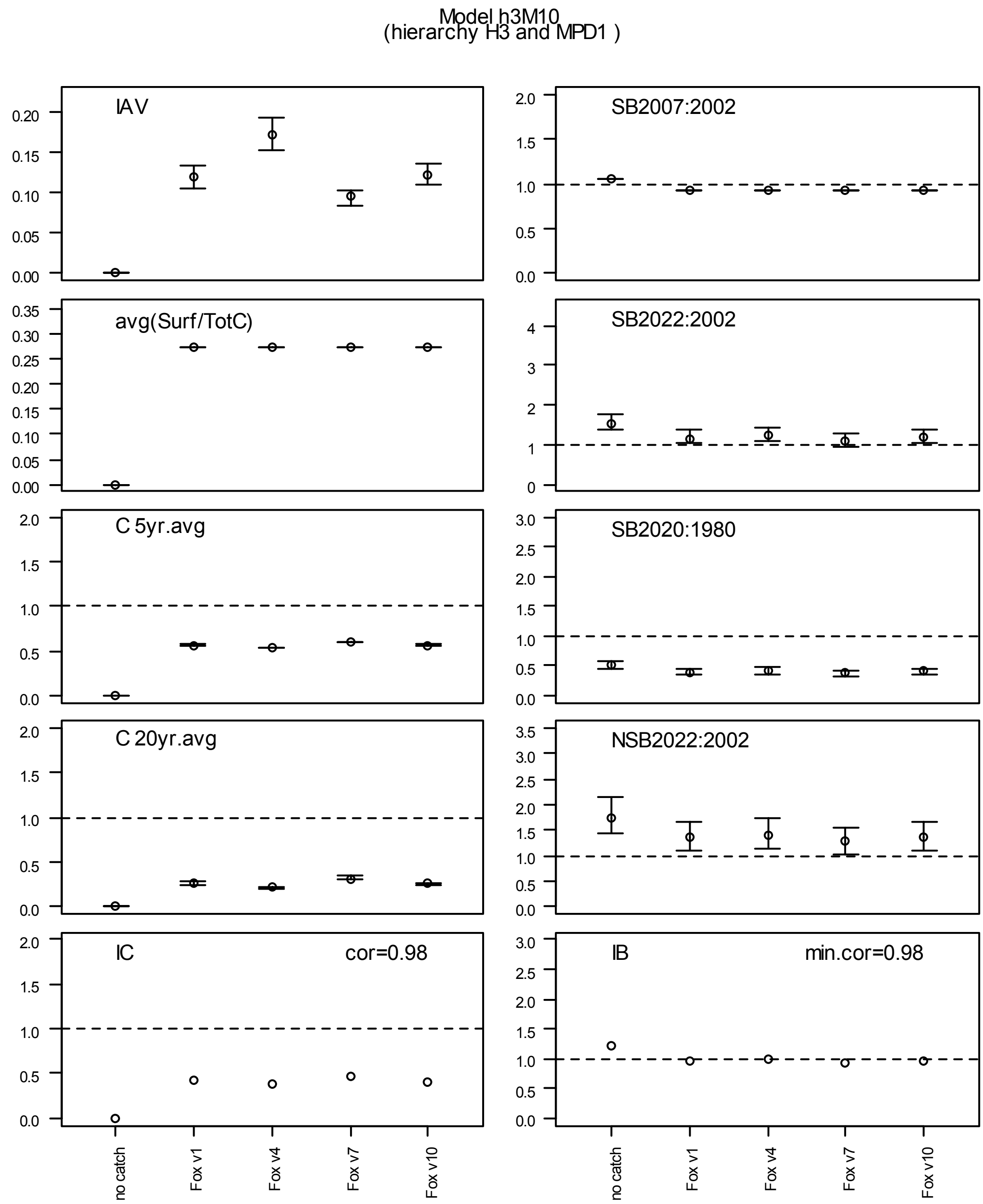

Model h3M10
(hierarchy H3 and MPD1)
IAV
avg(Surf/TotC)
C 5yr.avg
C 20yr.avg
IC
cor=0.98
SB2007:2002
SB2022:2002
SB2020:1980
NSB2022:2002
IB
min.cor=0.98
no catch
Fox v1
Fox v4
Fox v7
Fox v10

Figure A8-3.

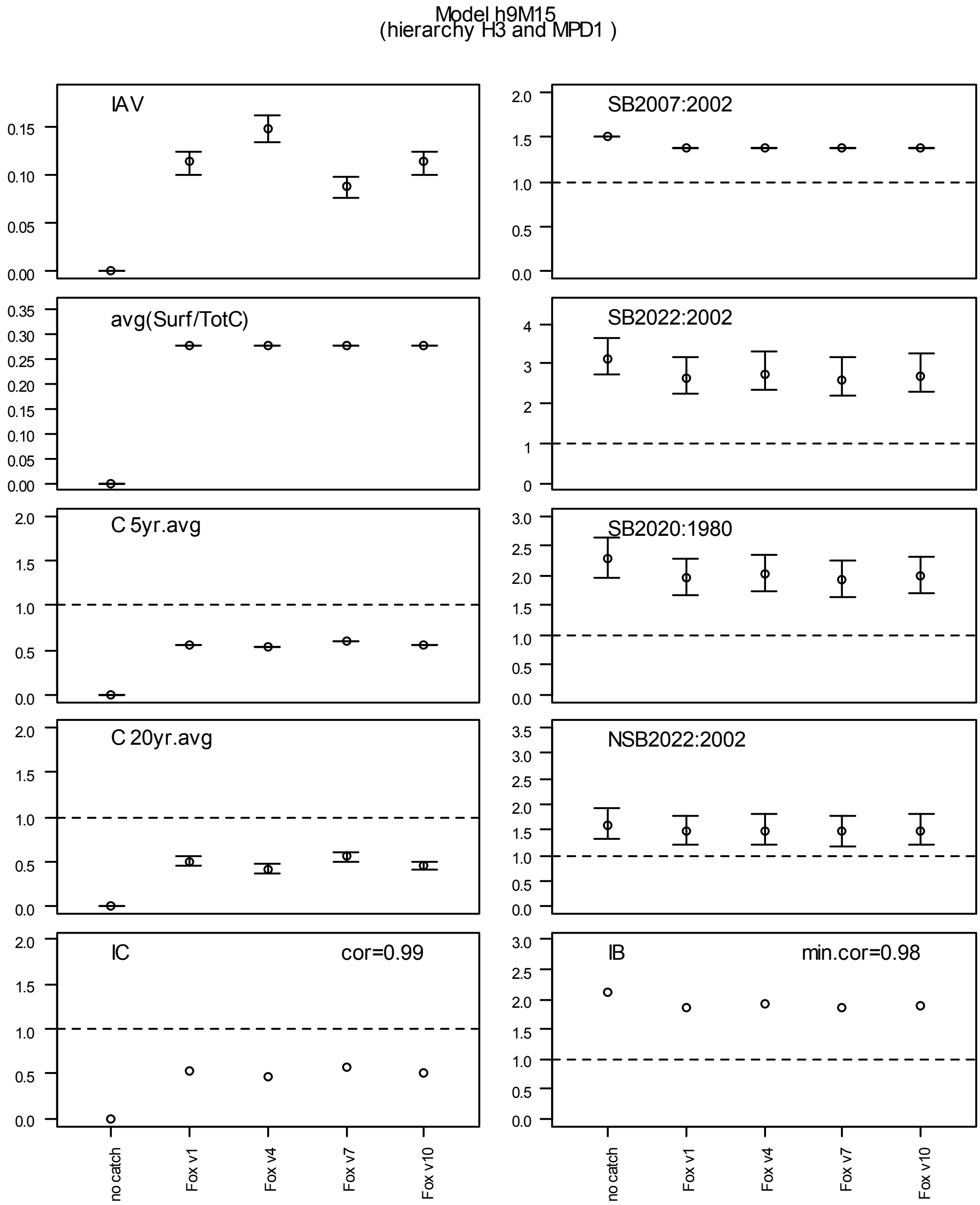

Model h9M15
(hierarchy H3 and MPD1)
IAV
SB2007:2002
avg(Surf/TotC)
SB2022:2002
C 5yr.avg
SB2020:1980
C 20yr.avg
NSB2022:2002
IC
cor=0.99
IB
min.cor=0.98
no catch
Fox v1
Fox v4
Fox v7
Fox v10

Figure A8-4.

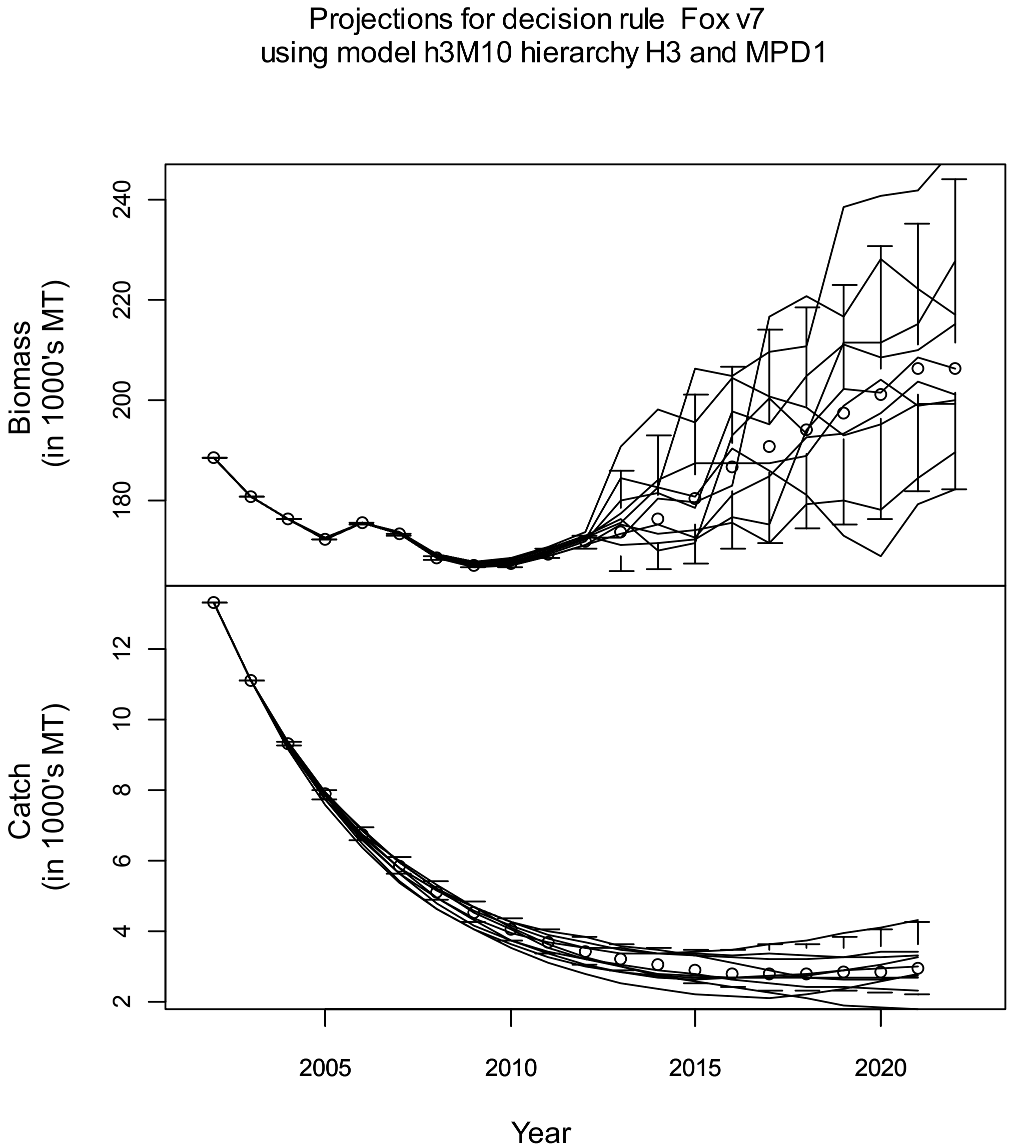
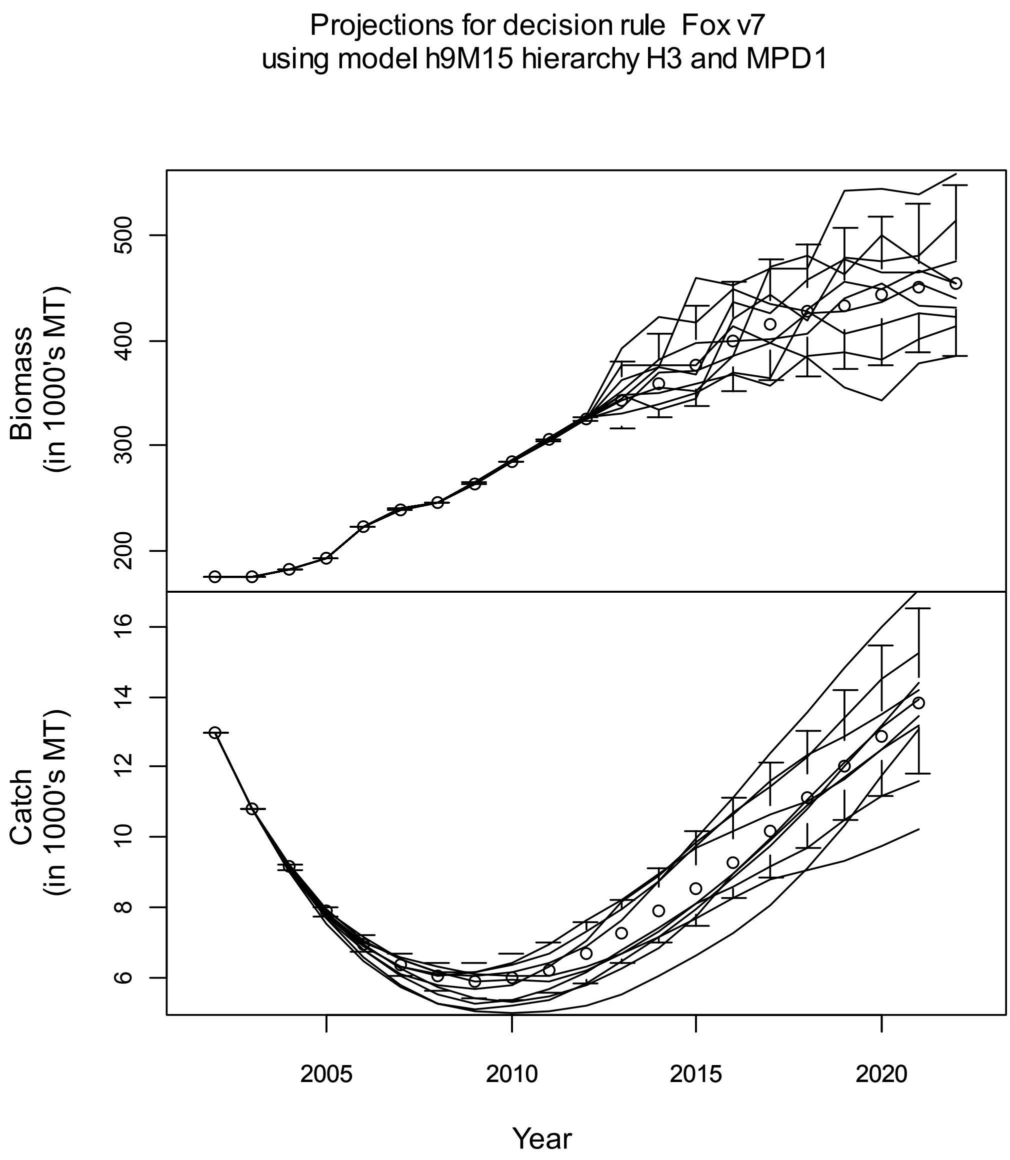

Projections for decision rule Fox v7
using model h3M10 hierarchy H3 and MPD1
Biomass
(in 1000's MT)
240
220
200
180
Catch
(in 1000's MT)
12
10
8
6
4
2
2005
2010
2015
2020
Year

Figure A8-5.


Projections for decision rule Fox v7
using model h9M15 hierarchy H3 and MPD1
Biomass
(in 1000's MT)
200
300
400
500
Catch
(in 1000's MT)
6
8
10
12
14
16
2005
2010
2015
2020
Year

A9. Fox_var (Fox production model using productivity parameter estimates to adjust TAC)

A9.1. Description of the rule

A9.1.1. Overview

This rule makes use of the estimates of the Fox "*r*" parameter as the simulation progresses to set the TAC. Initial results showed that the estimates of Fox's "*r*" parameter were increasing with time when dealing with productive OM scenarios. This rule attempts at using this information to more aggressively harvest without overshooting the rebuilding objective.

A9.1.2. Mathematical description

This rule is the same as "Fox" with an additional scaling factor M_2:

$$TAC_{y+1} = \omega TAC_y + (1-\omega)\left(B_y\left(M_{1,y} * M_{2,y} * F_{MSY}\right)\right)$$

$$M_{y,1} = \eta\left(B_y / B_{MSY}\right) + \beta - \eta$$

$$M_{2,y} = 1 / \left(1 + \tau\left(1 - \frac{r_y}{r_{initial}}\right)\right)$$

A9.1.3. Versions (tuning parameter values)

$\omega = 0.8$, $\beta = 1.0$ for all.

Rule version	Details	
v1	$\eta = 0.5$	$\tau = 1.5$
v3	$\eta = 0.8$	$\tau = 1.5$
v5	$\eta = 0.7$	$\tau = 2.0$

A9.2. Performance of rule

A9.2.1. Overview

Just as for decision rule "Fox", the current estimates of stock status are well below B_{MSY} and the rule reduces the TAC in earlier years.

The estimates of "*r*" allow the rule to roughly distinguish between productive and unproductive cases. However, the overall productivity of the stock is determined by a variety of factors, including the steepness of the SR relationship as well as the mortality schedule. The Fox model is very simple and lumps these two processes together in the "*r*" parameter. It is therefore difficult for the rule to distinguish between OM scenarios of overall intermediate and high productivity. A given version might be well suited for a given OM scenario but the decision rule is not robust enough to provide similar performance with other scenarios. For example, under OM scenario h6M10 and h9M10, "Fox_var" version 5 achieves a substantial catch level while almost fulfilling the rebuilding management objective. This same version performs poorly with other OM scenarios, overshooting the management objective for h9M15, h6M15 and h6M15d1. Still this rule shows potential since it performs better than "Fox" and seems to be able to capture some sort of signal about the different OM scenarios.

A9.2.2. Graphics

Figure A9-1a.

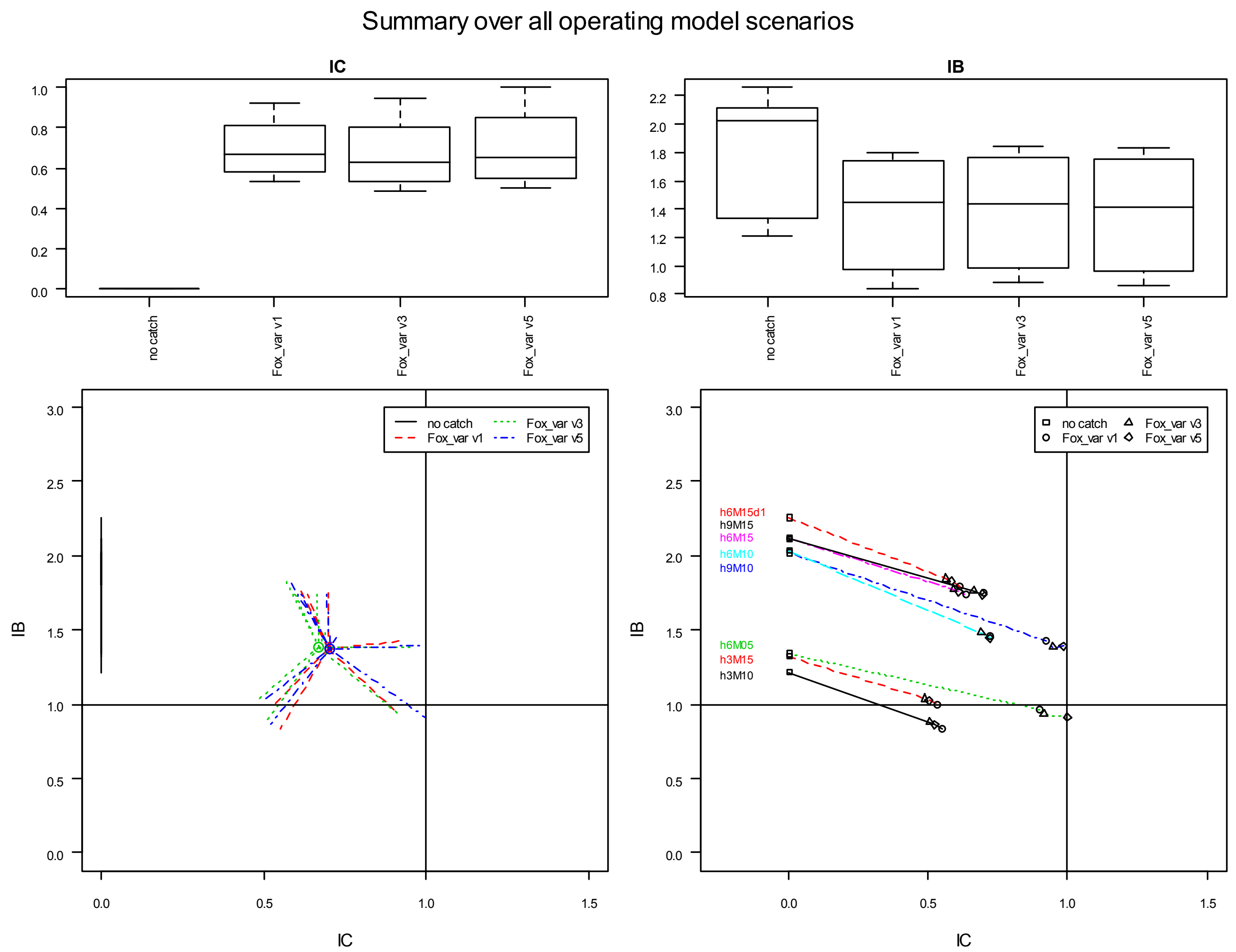

Summary over all operating model scenarios
IC
IB
no catch
Fox_var v1
Fox_var v3
Fox_var v5
h6M15d1
h9M15
h6M15
h6M10
h9M10
h6M05
h3M15
h3M10

Figure A9-1b.

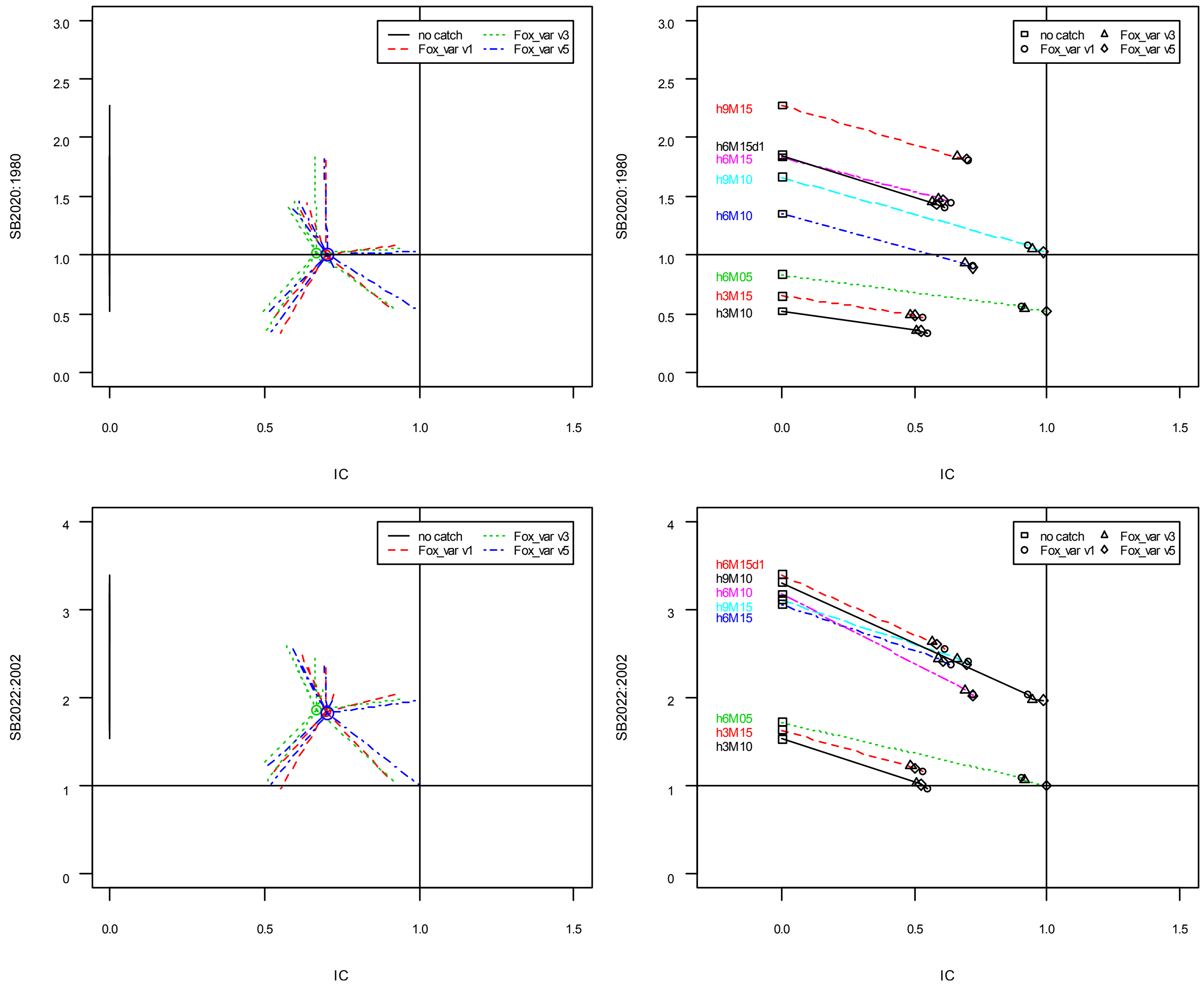

no catch
Fox_var v1
Fox_var v3
Fox_var v5
SB2020:1980
SB2022:2002
IC
h9M15
h6M15d1
h6M15
h9M10
h6M10
h6M05
h3M15
h3M10
h6M15d1
h9M10
h6M10
h9M15
h6M15
h6M05
h3M15
h3M10

Figure A9-2.

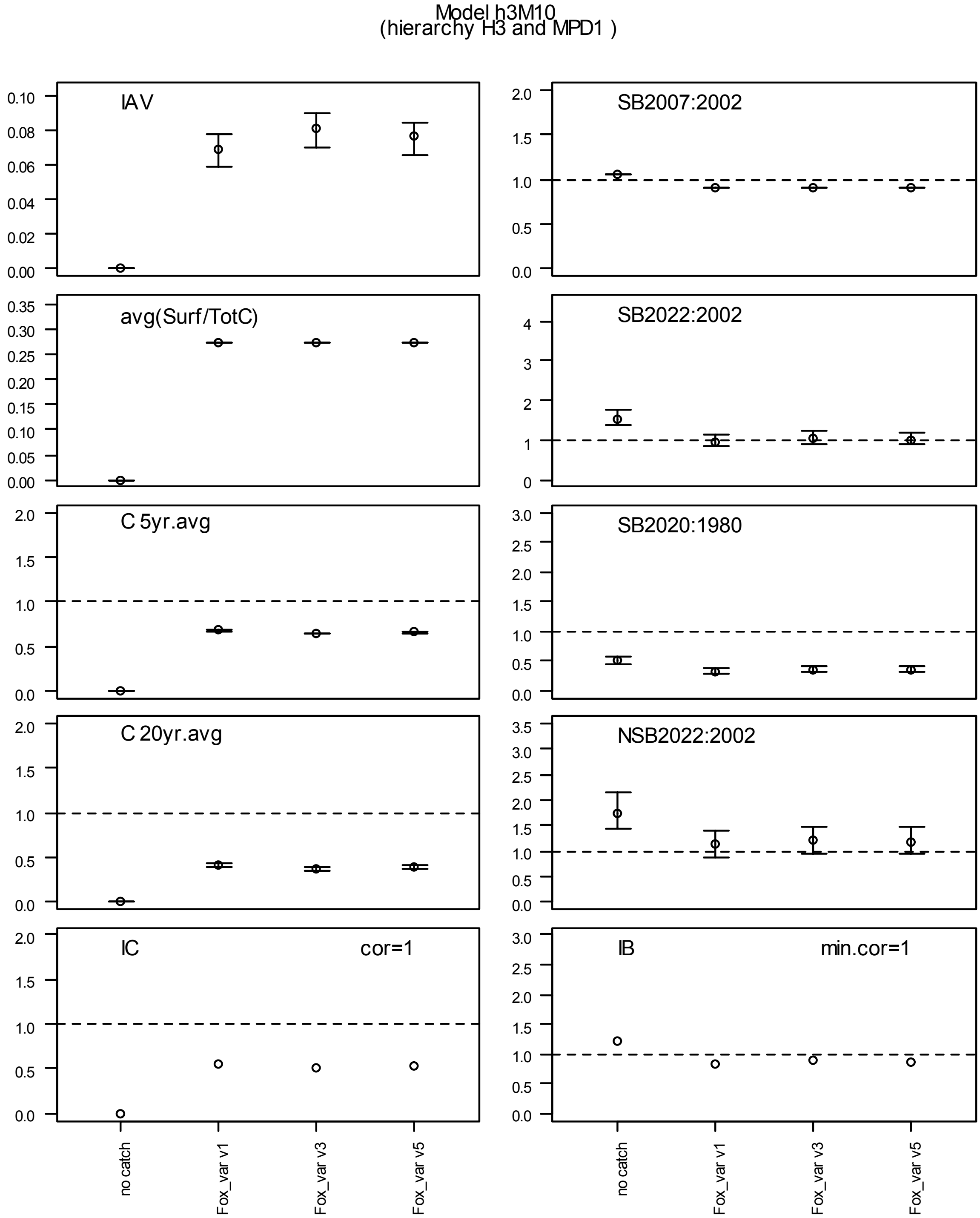

Model h3M10
(hierarchy H3 and MPD1)
IAV
SB2007:2002
avg(Surf/TotC)
SB2022:2002
C 5yr.avg
SB2020:1980
C 20yr.avg
NSB2022:2002
IC
cor=1
IB
min.cor=1
no catch
Fox_var v1
Fox_var v3
Fox_var v5

Figure A9-3.

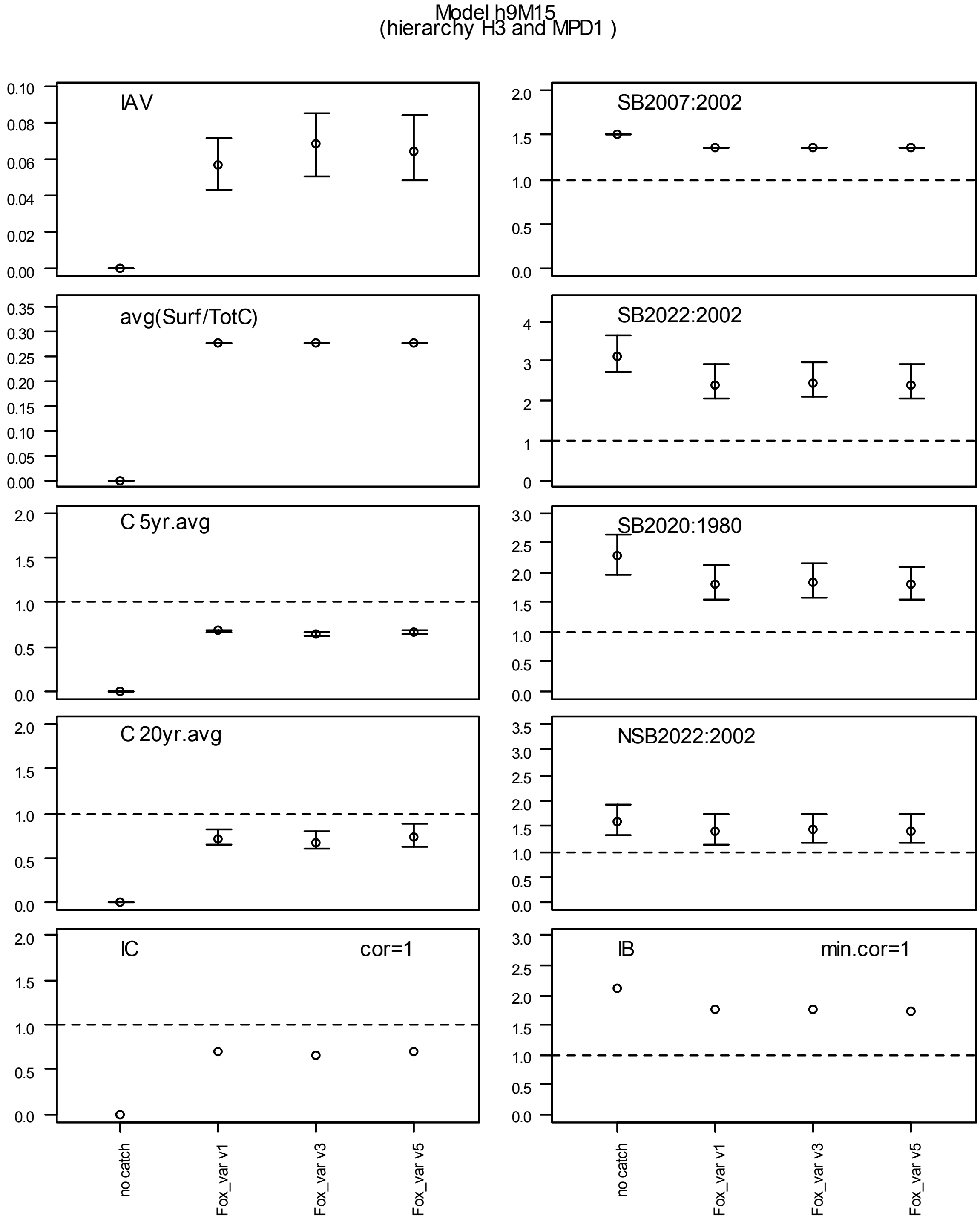

Model h9M15
(hierarchy H3 and MPD1)
IAV
SB2007:2002
avg(Surf/TotC)
SB2022:2002
C 5yr.avg
SB2020:1980
C 20yr.avg
NSB2022:2002
IC
cor=1
IB
min.cor=1
no catch
Fox_var v1
Fox_var v3
Fox_var v5

Figure A9-4.

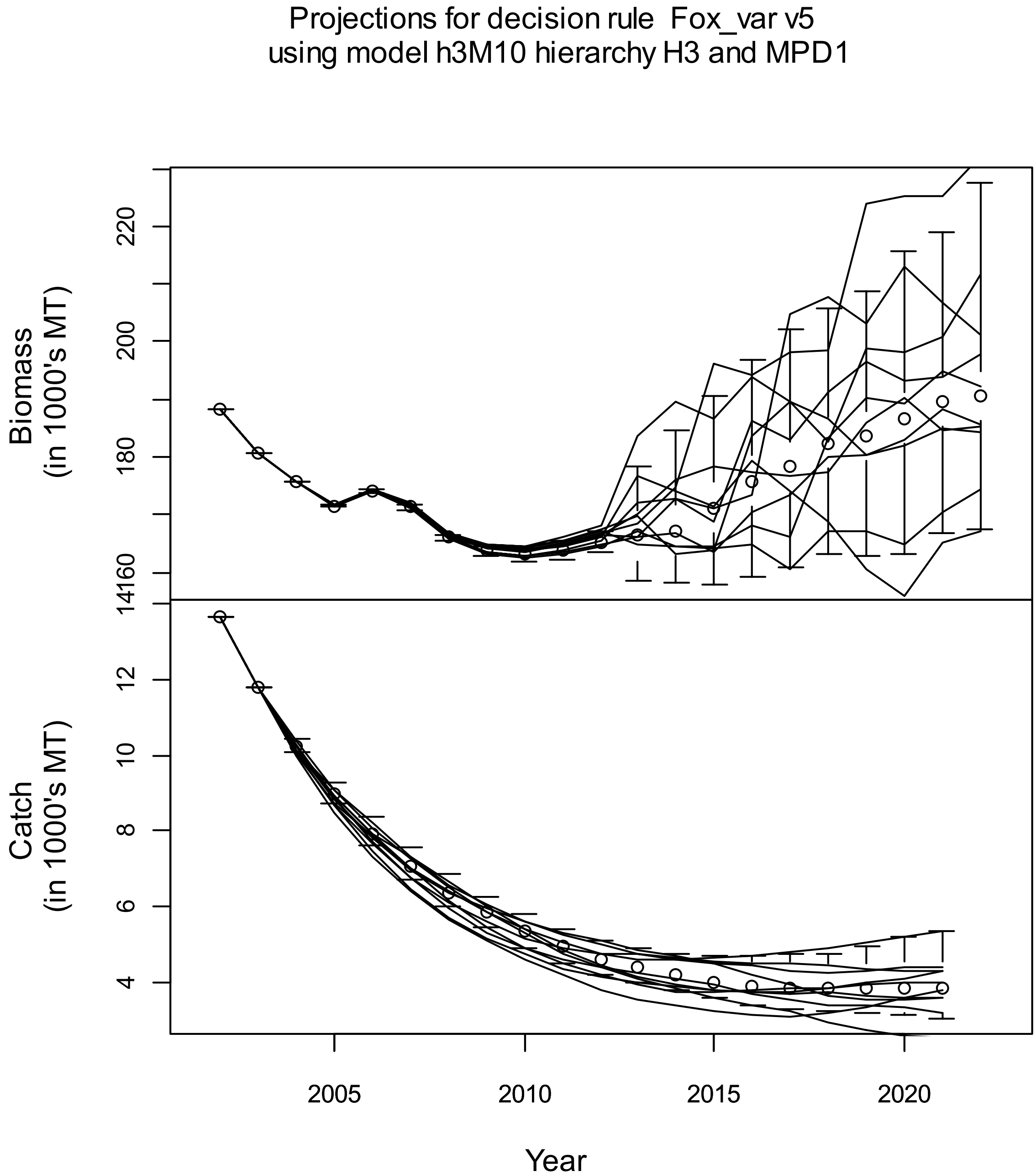
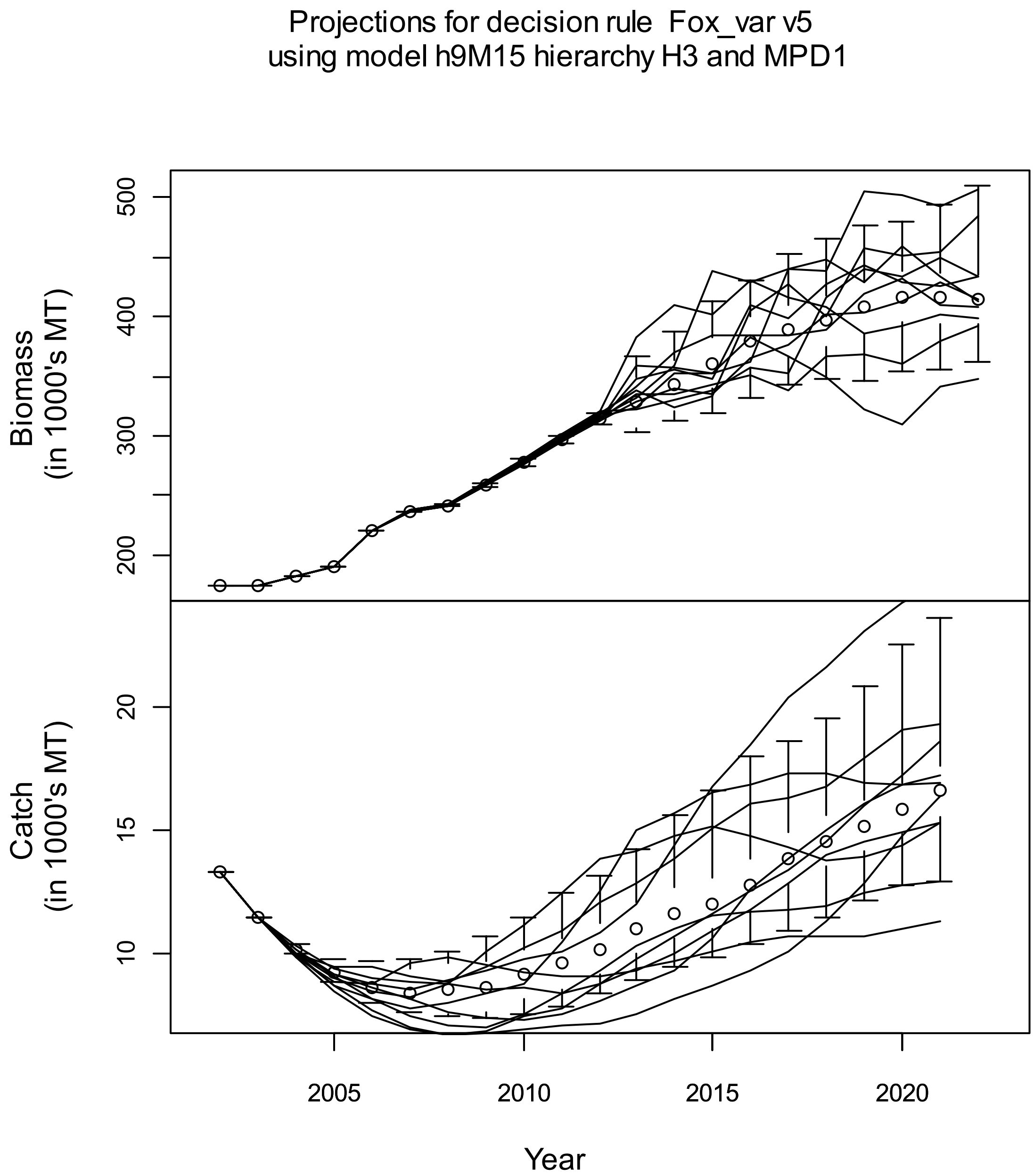

Projections for decision rule Fox_var v5
using model h3M10 hierarchy H3 and MPD1
Biomass
(in 1000's MT)
160
180
200
220
Catch
(in 1000's MT)
4
6
8
10
12
14
2005
2010
2015
2020
Year

Figure A9-5.


Projections for decision rule Fox_var v5
using model h9M15 hierarchy H3 and MPD1
Biomass
(in 1000's MT)
200
300
400
500
Catch
(in 1000's MT)
10
15
20
2005
2010
2015
2020
Year

A10. Kaltac (based on Kalman Filter 'assessment')

A10.1. Description of rule

A10.1.1. Overview

The 'kaltac' decision rule is based on a Kalman Filter 'assessment'. Although the Kalman Filter provides estimates of population numbers, it is the ratio of the current population size to the population size in 1980 that is in fact used by the rule. The rule aims to rebuild the population to the 1980 level by the end of the simulation period. A TAC is determined in such a way that the projected population size, after that TAC has been taken, would lie on a line connecting the population size in the last year of the assessment with the 1980 population level at the end of the simulation period. This is, however, done in terms of the ratios of population size in a given year to that in 1980. Note that the same logic is applied for cases where the current ratio is below 1 and for cases where it is above 1.

<u>The 'assessment'</u>

The assessment consists of fitting a Kalman filter to CPUE indices. The time-series of CPUE (1969-2000) is short for a Kalman filtering approach, which is why the aim is NOT to obtain reliable estimates of absolute abundance, but rather estimates of relative abundance, particularly of the 'current' stock size compared with stock size in the past (e.g. in 1980). A Kalman filter with 2 states representing 'recruits' and 'adults' is fitted to CPUE indices (in numbers). Work presented here defines 'recruits' as age 5 and 'adults' as ages 6 and older. Clearly, other definitions can be considered. The data are therefore CPUE for age 5, and CPUE for age 6+ (i.e. a plus group), as well as, the total annual catches in numbers. The state equations are:

$$R_{t+1} = R_t + \varepsilon_t$$
$$A_{t+1} = A_t.s + R_t - C_t + \varepsilon_t$$

where the ε_t are N(0,sigmaQ) distributed and s is adult survival, assumed to be known and set to exp(-0.2) in all runs presented here. C_t are the annual catches in terms of numbers (all converted to millions). Recruitment has been treated as a random walk process, but other assumptions could be considered.

The observation equations associated with CPUE for recruits ($y_{t,1}$) and adults ($y_{t,2}$) are:

$$y_{t,1}=q_a.\text{qratio}.R_t + \eta_t$$
$$y_{t,2}=q_a.A_t +\eta_t$$

where η_t are N(0,sigmaH) distributed. Assume further that:

a) covariances in sigmaQ, and sigmaH are 0, and write the measurement and process error variance matrices as:

sigmaH = σ^2.I

and sigmaQ= σ^2.Vratio.I

b) input qratio (part of the .dat file); write q_r=q_a.qratio, i.e. qratio represents the ratio of Recruit over Adult catchability

c) input Vratio (part of the .dat file); Vratio represents the ratio of process to observation variance.

There are some issues with regard to starting values for the states, A_0 and R_0 (i.e. essentially A_{1968} and R_{1968}, the states prior to the first CPUE observations). One option, if estimating these quantities, is to use a diffuse prior. Simulations, however, showed that with such a short data series (32 years) there is very little hope of reliably estimating all the unknowns. As a first cut, I have therefore made some less than ideal assumptions. No doubt the treatment can be "cleaned up", refined and possibly improved. For example, although I estimate the initial state, it is set up as a bounded variable to avoid negative population sizes. Also, the variance matrix of the initial state is assumed to be 0, so that the starting states are assumed to be certain. Given that results are NOT used as estimates of absolute abundance, this approach is likely to perform reasonably well.

The 4 parameters that are estimated are:
A0, R0, σ^2 and q
Outputs include estimates of the states at each time step. In the current version, I have not smoothed the estimates (i.e. a 'backwards' filter based on all observations has NOT been applied).

A10.1.2. Mathematical description of the rule

The rule, called 'kaltac', is designed to continually aim at being at the 1980 biomass at the end of the simulation period. Recall that the target is set in terms of SSB, whereas the Kalman filter works in terms of numbers. Initial versions ignored this difference in units and aimed at having population <u>numbers</u> in 2021 over population <u>numbers</u> in 1980, equal to 1. Performance was, however, poor in terms of the biomass ratios, as output in the .sum files (in particular, the B2020/B1980 was well above 1 for the productive scenarios). The current rule makes two adjustments to obtain a proxy for SSB: first, 'adult' numbers are turned into numbers of age 10 and older by using the proportions at age in the input data; second, numbers of age 10+ are turned into biomass by using proportions at age and the length-weight equation to calculate an approximate mean weight of age 10+. Note that the choice of 10+ is arbitrary, and other ages could be tried , such as 12+. The product of the two adjustment factors, d_t, is applied to the estimates of adult numbers (age 6+) in year t, to obtain a 'proxy' spawning biomass, A_t:

$$A^*_t = A_t.d_t$$

This is done for the last year in the assessment (current year -2) and for 1980.

The rule calculates the catch required in the current year to put the biomass ratio (in current year +1 to that in 1980) on a line connecting the population ratio at the end of the assessment (current year - 2) and 1.0 (the target ratio for 2021). This essentially requires forward projections for 3 years from the current estimate of population size (using the state equations, but ignoring the variances). The projections are as follows, with t=current year:

$A_{t-1} = s.A_{t-2} + R_{t-2} - C_{t-2}$
$A_t = s.A_{t-1} + \text{meanR} - TAC_{t-1}$
$A_{t+1} = s.A_t + \text{meanR} - X$

where X is the unknown TAC we're trying to determine, and meanR is the mean of the last 5 recruitments (i.e. R_{t-6}, Rt_{-5},...,R_{t-2}).

Although it would be more in keeping with the Kalman Filter approach to use R_{t-2} in all the projections (and to consider the variance too), this seemed to lead to more conservative TACs than an approach based on a 5-year mean. Other sophistications for dealing with recruitment can be considered in further work. It should also be noted that the current rule does not fully exploit the Kalman filter properties, such as estimates of the covariance matrices of the states.

The unknown TAC is calculated via the following set of equations where A^* denotes the 'proxy' spawning biomass described above, where T is the last year in the simulations:

$$\text{slope}=(1-A^*_{t-2}/A^*_{1980})/(T-(t-2))$$
$$\text{intercept}=1-\text{slope}.T$$
$$X = s.A_{t-1}+\text{meanR} - (t.\text{slope}+\text{interc}).A_{1980}.(d_{1980}/d_{t-2})$$

The rule has been implemented with a constraint on the percentage change in TAC allowed from year to year. When this is incorporated via a term 'maxchange' (e.g. maxchange=0.2, or 20%), then this can be formulated as follows (again, t=current year):

$$TAC_t = TAC_{t-1} + (X-TAC_{t-1})/\text{abs}[X-TAC_{t-1}]*\min(\text{abs}[X-TAC_{t-1}],\text{maxchange}*TAC_{t-1})$$

A10.1.3. Rule versions ('tuning' parameter values)

The key differences between versions lie in assumptions about qratio and Vratio. Recall that qratio is the ratio of recruit to adult q (q_r/q_a) where recruits are age 5 and adults age 6+ in this case. Vratio is the (process error variance)/(observation error variance).

For all versions, the TAC is set every year. For versions 1 to 5, the TAC is not constrained at the start of the period, but for versions 6 and 7, the TAC is constrained to be at least at the level of the current catch (i.e. the TAC is fixed at current catch unless an increase is indicated) for several years (see below). The parameter governing how many years, is called "startfixyrs". In all cases the maximum interannual change is 20%. Versions for kaltac are summarised below:

Version	qratio	Vratio	
1	1	1	
2	0.5	1	
3	2.0	1	
4	1	0.5	
5	1	2.0	
6	1	1	but with startfixyrs=5
7	1	1	but with startfixyrs=10
8	0.1	1	

Version 1 is considered as a 'base case' run.
Versions 2,3, and 8 explore changes in the q ratio.
Versions 4 and 5 explore changes in the variance ratio.
Versions 6 and 7 explore different periods of fixed catches at the start of the 20-year simulation period.

Fixed parameters, unless otherwise stated are:
- starting values in the Kalman Filter,
- mortality, s=exp(-m) in the Kalman Filter; m is assumed to be 0.2 per annum ('adults')
- the maximum allowed inter-annual change (+ or - 20%, calculated from $(C_{t+1} - C_t)/(C_t)$ in the rule, but from $(C_{t+1} - C_t)/(C_{t+1})$ in the RESULTS)
- TAC is calculated, and potentially changed, annually

In the text below, I use the word 'models' for the 8 scenarios, i.e. for h3M10, h6M05 etc.

A10.2. Performance of Kaltac rule

A10.2.1. Overview

Hierarchy 1 was used to develop the rule, to weed out poor performing 'tuning' parameter values, and to check the strength of effects of different values of the tuning parameters. Versions are therefore only compared in hierarchy 1. Following these comparisons, hierarchy 3 results are presented for version 1.

a) Version 1

Version 1 is used as a 'base case', but this does NOT mean that the parameters in that version are considered to be better than any others, but it is useful to make comparisons of changes relative to version 1. Figure A10-1 illustrates the performance of kaltac under version 1, for all models. Behaviour is very much as one might expect. For the low productivity models, the catches are low, average IAV is high because the catch is just being reduced by the maximum amount each year, and biomass performance is good relative to 2002 (biomass declines at first because of the constraint on the magnitude of catch reductions, but then recovers to 2002 levels by 2022), but still poor relative to 1980. Note, however, that even under no catches for the next 20 years, the biomass does not manage to rebuild to 1980 levels by 2022 under the LOW productivity scenarios.

For the higher productivity scenarios, the catches are relatively high (e.g. h9M10, h6M05), but biomass performance is good. For the high mortality scenarios (h6M15, h9M15 and h6M15d1) it is, however, noticeable that the biomass performance is arguably higher than need be (at the end of the period biomass is close to 1.5 times the 1980 biomass), and catches lower than need be.

Figure A10-2 shows the results in terms of trajectories of biomass (bworms in the .sum files) and catch (cworms in the .sum files). Recall that at hierarchy 1 there is only 1 trajectory for each model.

It would be interesting to see what happens to catches over a longer than 20 year time period.
(Note, we could only get software running for 25 years, not for longer periods. When running it for 25 years, the rule aims to be at B1980 at the END, so the catch trajectory is slightly different. One may want to think about how to build in the time frame of where one is 'aiming' to be and when, as well as, what happens when one gets there!)

b) Effect of q-ratio

To compare the effect of a change in the assumed q ratio between the recruit index and the 'adult' index, consider results for two scenarios (models): a low productivity model (h3m10), and a high productivity model (h9m15). Versions 8,2,1 and 3 have been plotted in this sequence, since they represent q ratios of: 0.1, 0.5, 1.0 and 2.0 respectively.

Figures A10-3a and b show great similarity in results for all versions, except possibly version 3 which assumes a much higher q-value for the recruit series than for the adult series (in fact, $q_r = 2*q_a$). This seems rather unlikely, but should nevertheless be explored. Summary results for all the models (Figure A10-4) again show that only version 3 is quite different (low catches and relatively high biomass).

c) Effect of different variance ratios

The summary plot (Figure A10-5) shows very little difference between the 3 versions (i.e. observation error is 1/2, the same or 2 times the process error). One could argue that a wider range should be considered. Limited Kalman Filter simulation suggest that it would be very difficult to reliably estimate both variances, so any indication either that results are not sensitive to this parameter, and/or estimates (or guesstimates) of a realistic range of ratios would be useful.

d) Effect of fixing catches at the start of the series

The next set of results compare runs with the first 0, 5 or 10 years of catches fixed at the current catch (~15.1 thousand tonnes) level, unless an increase is indicated by the rule. The versions are: v1 (0 years), v6 (5 years) ,v7 (10 years). All other parameters are the same as in version 1 (i.e. q ratio = 1, Vratio=1). The summary plot (Figure A10-6) shows that, as expected, the longer the catch is kept fixed, the higher the overall catches, but for lower biomass performance. For some scenarios (models) there is a clear non-linearity in the change in the composite 'IB' measure as the fixed catch period increases.

e) Comparison of results for all 8 versions of kaltac

The overall comparisons (i.e. over all models) are summarised in Figure A10-7. The 2 versions which are most different are versions 3 (q ratio=2) and 7 (keep catch at least at current level over the first 10 years). As noted before, version 3 stands out as having low catches, but having very good biomass performance. Recall that this version assumes that the recruit to adult CPUE index catchability coefficient is 2 (i.e. $q_r = 2*q_a$). This is not really a rule "tuning" parameter, but an assumed input to the Kalman Filter "assessment", and such an assumption may be unrealistic.

On the whole, one notices that the relationships between the different versions are still basically linear in the catch vs. biomass trade-off. The other key observation is that in the high productivity scenarios, or models, the rule tends to do extremely well in terms of rebuilding biomass (biomass in 2022 is up to 2 times that in 1980). This suggests that there could be some scope for increasing the catch, but mainly in the high productivity cases, which means that the assessment and rule need to be able to better distinguish between the low and high productivity scenarios, and as early as possible.

f) Version 1, hierarchy 3

Figure A10-8 shows results for all models, and Figure A10-9 shows overall results (summarised over models). A comparison of figures A10-8 and A10-1 (i.e. hierarchy 3 and 1 results) shows that the deterministic results can be different from the median results of the stochastic runs. For example, model h9M10, hierarchy 1, 20 year average catch is above the current catch, whereas the median under hierarchy 3 is below the current catch. This also means that the median SSB values are bigger than the deterministic counterparts.

Trajectories of biomass and catch (percentiles and 10 'worms') are shown for 3 of the models (h3m10, h9m15 and h6m05) in Figure A10-10. These plots show that the summary statistics do not fully reflect the dynamics. For example, with model h3M10, the biomass drops to its lowest values around 2014 or 2015, and the 10th percentile is well below the current biomass. This represents a risk which needs to be considered.

g) Version 6, hierarchy 3

Recall that the only difference between version 1 and version 6 is that the TAC is fixed at the current catch level for 5 years (or increased) in version 6. Figure A10-9 shows that the combined performance statistics in terms of biomass are not much different (v6 is only slightly lower than v1) between the two versions. The catch performance appears to be better under version 6 than version 1. Trajectories, however, show that the biomass in version 6 dips much lower than in version 1. For example, for model h3M10, the ratio of the lowest biomass (between 2002 and 2022) to biomass in 2002 is 90% for version 1, but only 74% for version 6 when considering the median. In terms of the 10th percentile, these percentages are 89% (v1) and 69% (v6). This could be seen as a greater risk associated with version 6.

This also illustrates that the summary measures do not reflect the full dynamics, and additional performance measures, such as the lowest biomass or number of times the biomass is below some reference level should be considered.

A10.2.2. Discussion

Although the kaltac rule seems to do reasonably well, there is scope for trying and testing variations. The first weakness appears to be the fact that in the high productivity scenarios, the biomass at the end of the simulation period is well above the target level. This is likely to be partly due to the way in which estimates of numbers are converted to a 'proxy' estimate of SSB, and partly due to limits on the interannual change in TAC. Recall that the current approach essentially assumes that the proportions at age in the catch also reflect the proportions at age in the population. This assumption could be very poor, and alternative ways of estimating SSB from total numbers could be considered.

There is scope for making better use of the estimates of uncertainty from the Kalman filter (i.e. the covariance matrix of the 'states'), but outcomes may depend on how reliable or useful these estimates are given the short time series, particularly at the start of simulations. Estimates of uncertainty in the population size could be used when comparing the current population status with the target (1980 SSB), and when generating recruitment in the forward projections. There is also merit in testing different model assumptions about recruitment with the aim to 'pick' up whether one is in a low or high productivity scenario.

The two ‘states’ in the Kalman filter were defined as recruits (5 year olds) and adults (6 year olds and older). The CPUE data are therefore constructed as CPUE for age 5 and CPUE for age 6 and older (i.e. as a plus group). It would be easy to change this to, for example: recruits = 6 year olds, adults=7 year olds and older, and to test the performance of such an assessment with the kaltac rule.

Simulations which end in 2022 provide relatively limited scope for full exploration of the dynamics of the rule when biomass is ABOVE the 1980 target SSB. Simulations of longer duration should ideally also be conducted, but could not be done due to a software problem (in the SBT projection model).

A10.2.3. Graphics

Figures referred to in the text follow.

Figure A10-1. Decision rule kaltac version 1 (hierarchy H1 and MPD1)

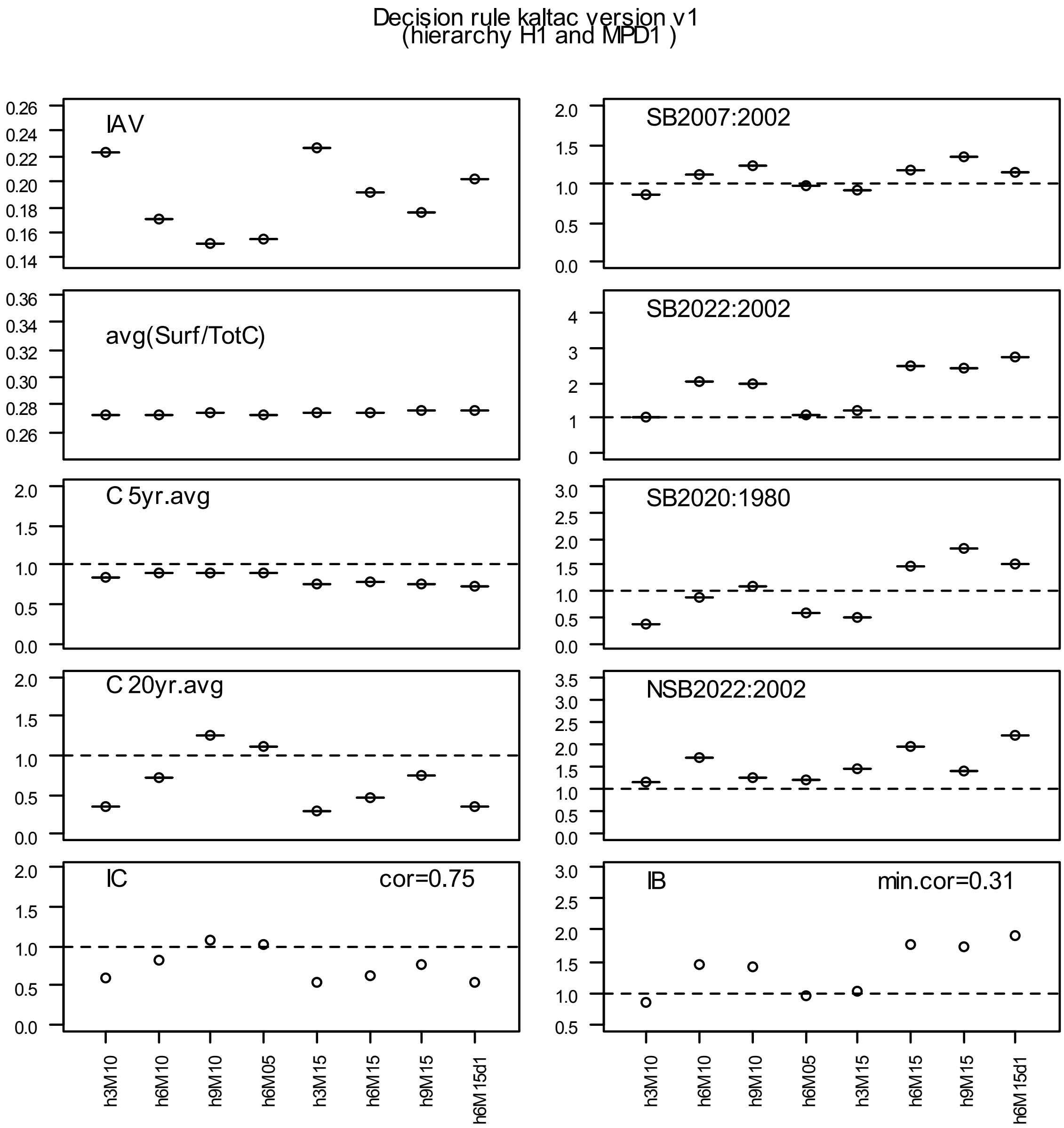


Figure A10-2 a and b. Biomass and Catch trajectories for kaltac, version 1 under hierarchy 1. (bworms and cworms from .sum file)

Figure A10-2a

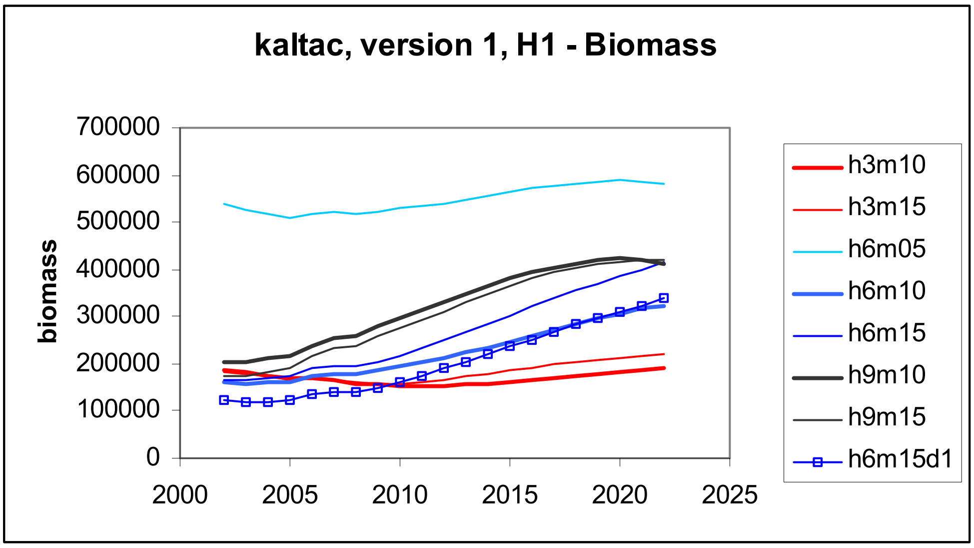


Figure A10-2b

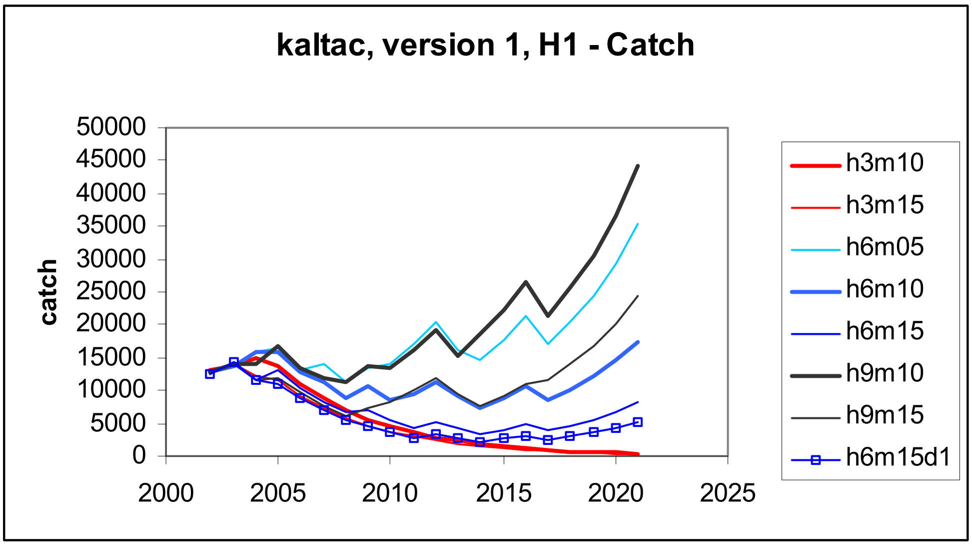


Fig A10-3 Effects of changes in qratio for kaltac rule (versions 8,2,1,3), hierarchy 1, (a) model h3M10 and (b) model h9M15.

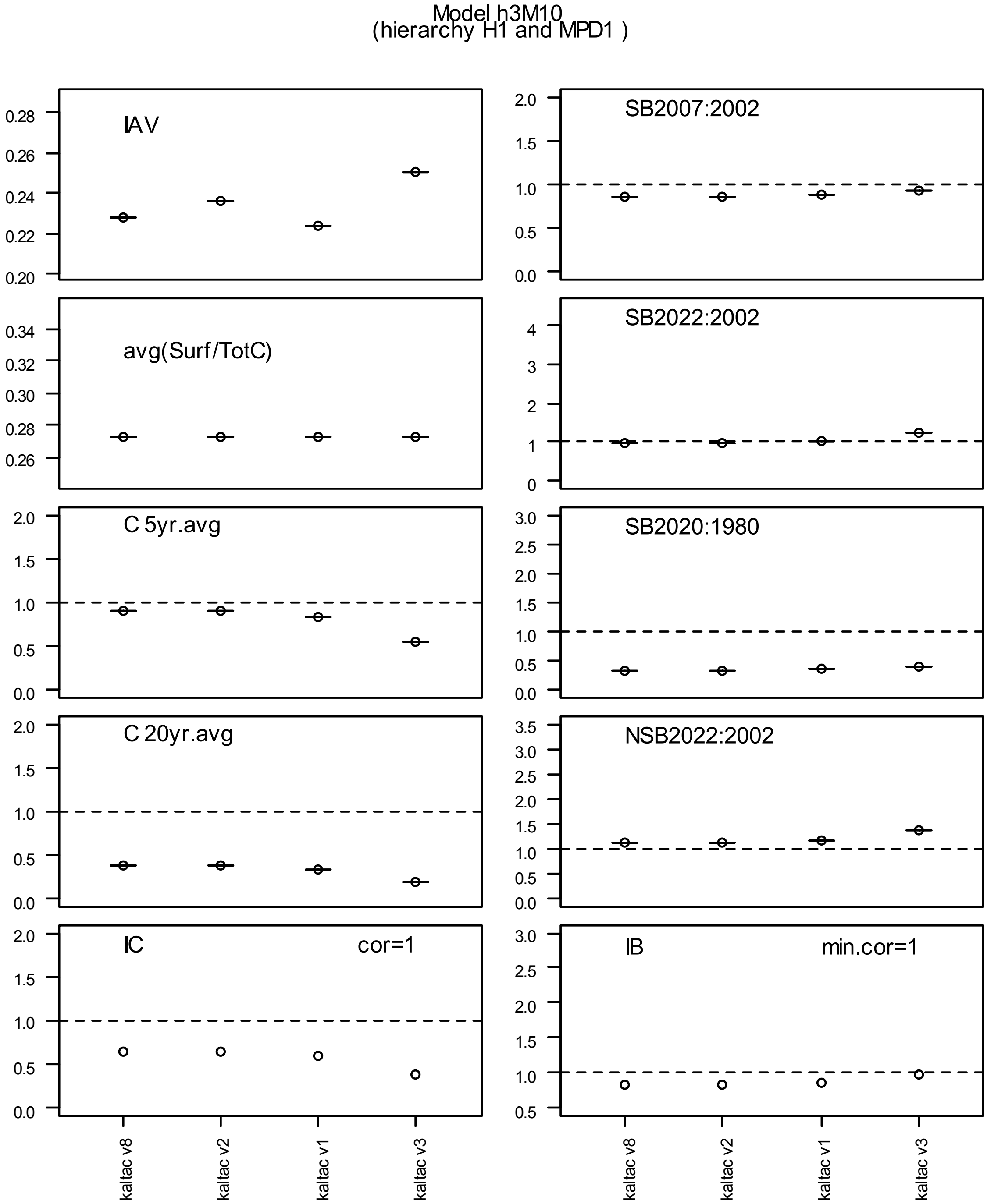


Figure 10-3 (b)

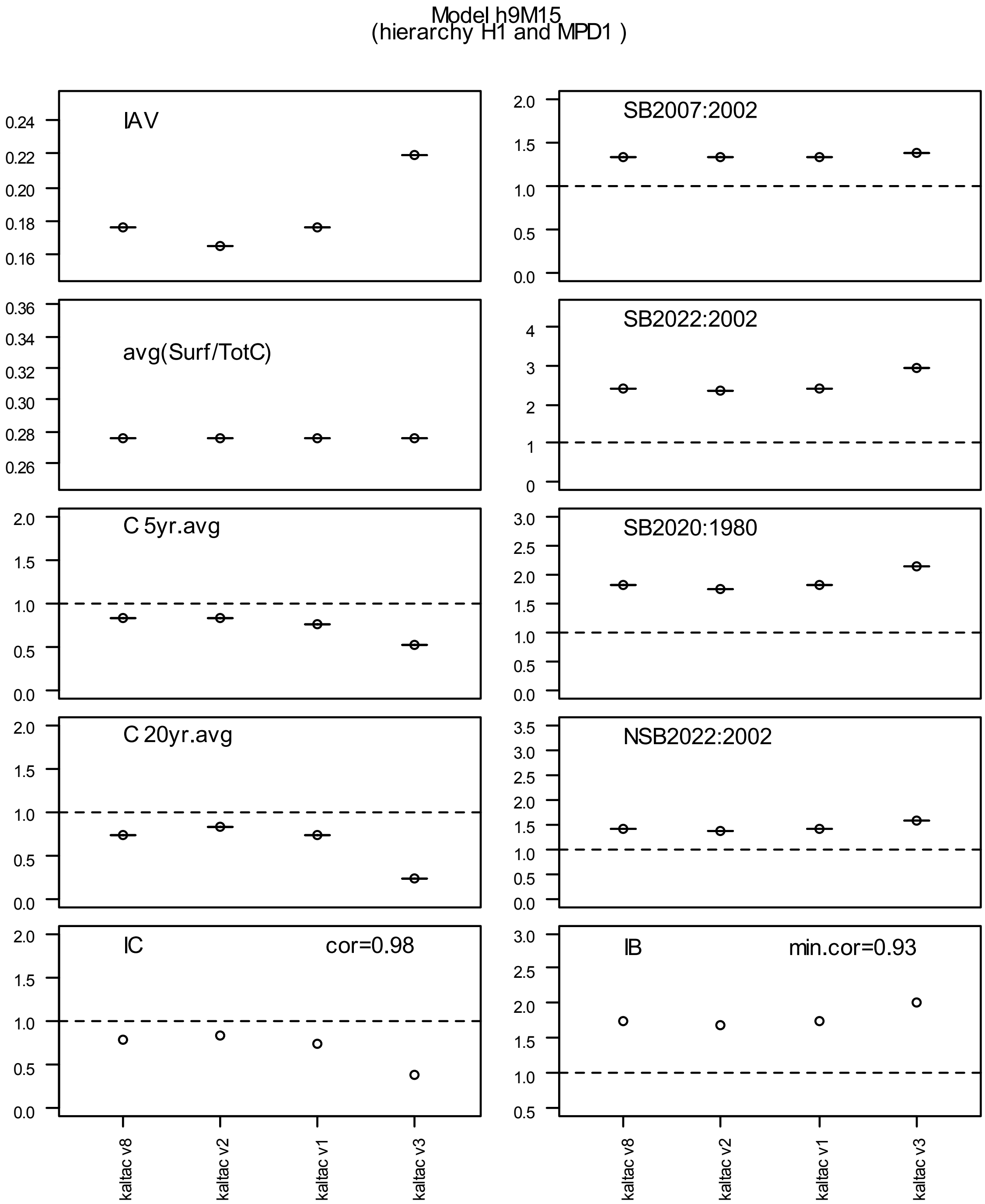

Model h9M15
(hierarchy H1 and MPD1)
IAV
SB2007:2002
avg(Surf/TotC)
SB2022:2002
C 5yr.avg
SB2020:1980
C 20yr.avg
NSB2022:2002
IC
cor=0.98
IB
min.cor=0.93
kaltac v8
kaltac v2
kaltac v1
kaltac v3

Figure A10-4. Kaltac rule, versions 8,2,1,3 i.e. changes in qratio, summary over all models, hierarchy 1.

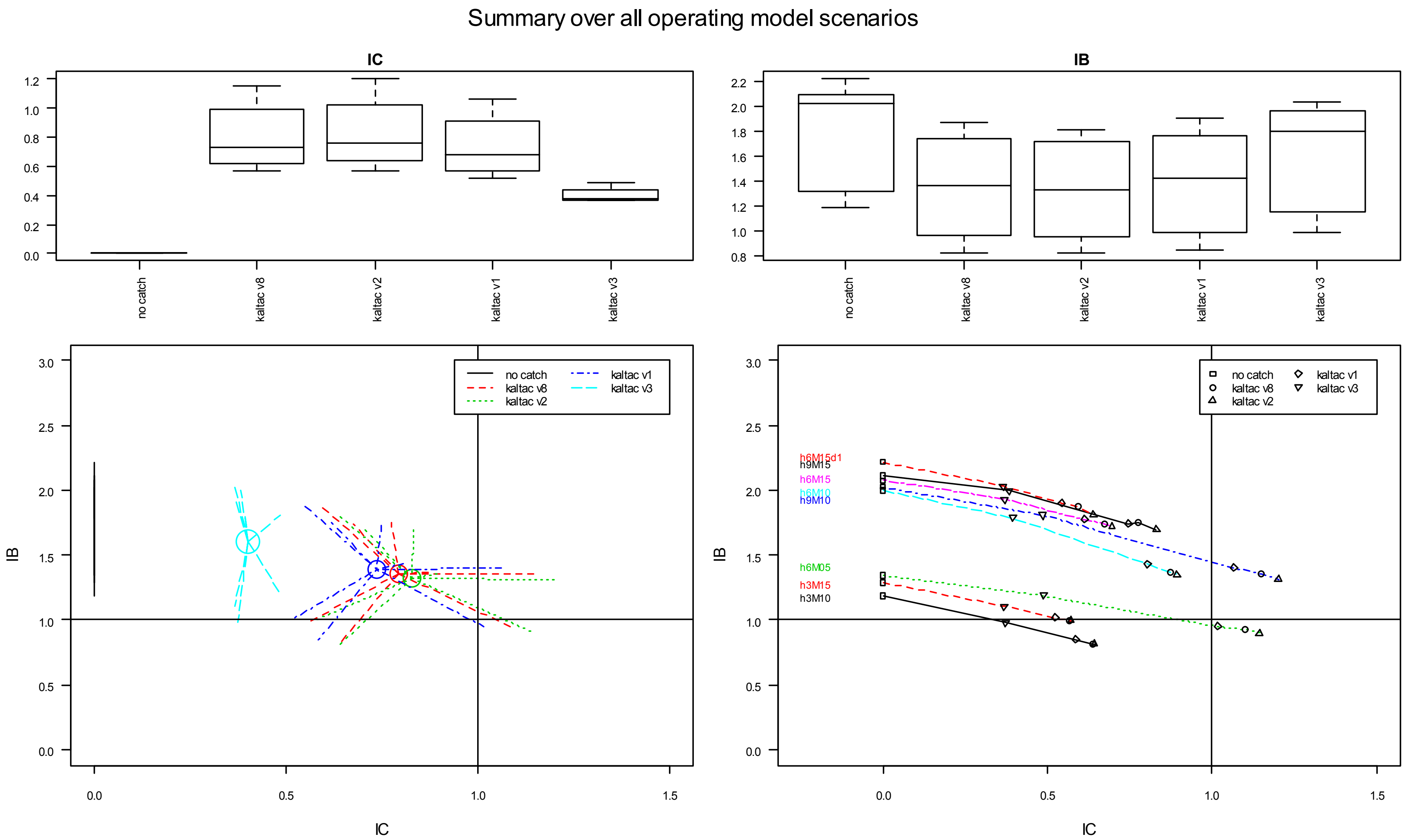


Figure A10-5. Kaltac rule, versions 4,1,5 i.e. changes in variance ratio, summary over all models, hierarchy 1.

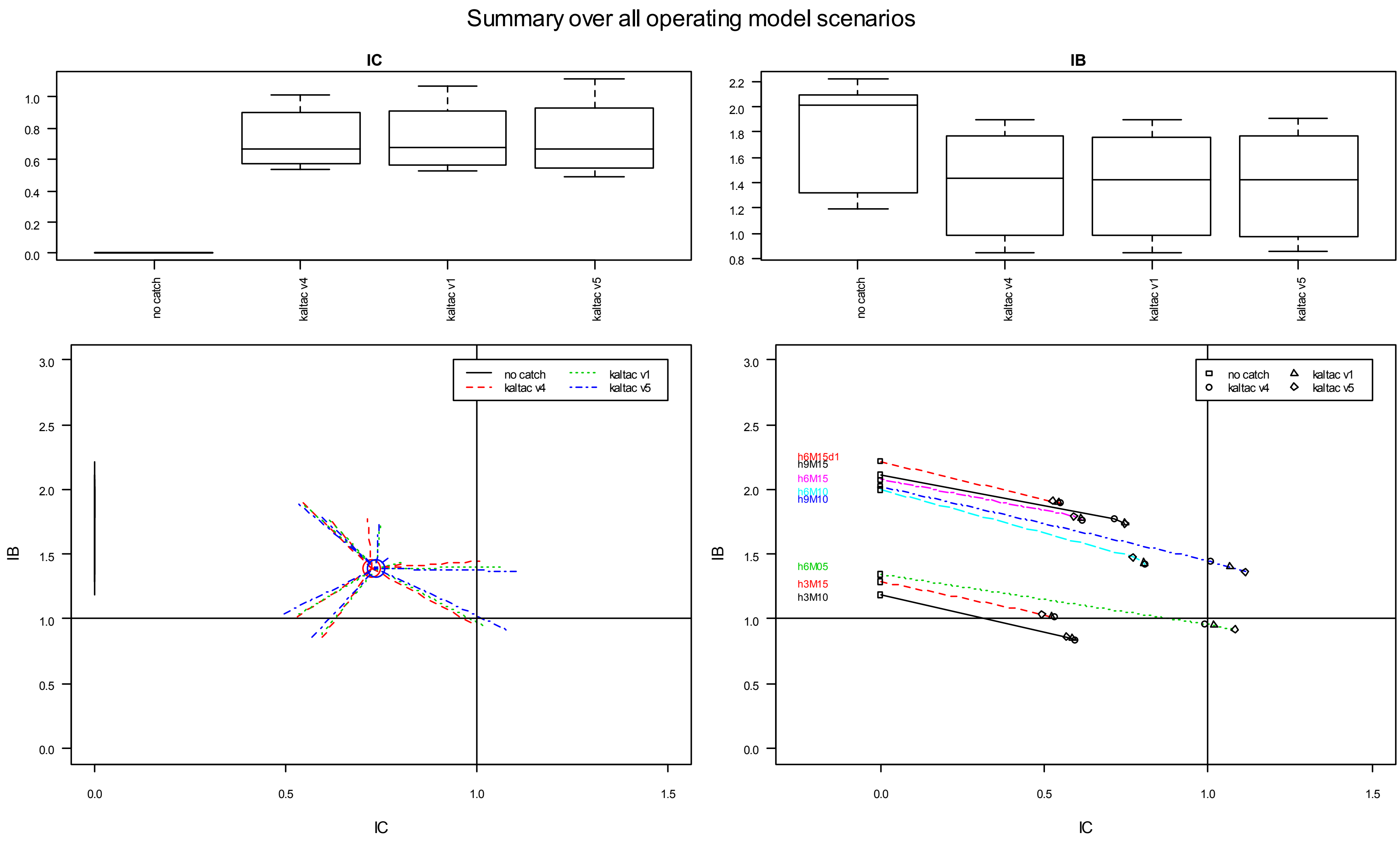


Figure A10-6. Kaltac rule, versions 1,6,7 i.e. changes in number of years for which catches are kept at least at the current catch level, summary over all models, hierarchy 1.

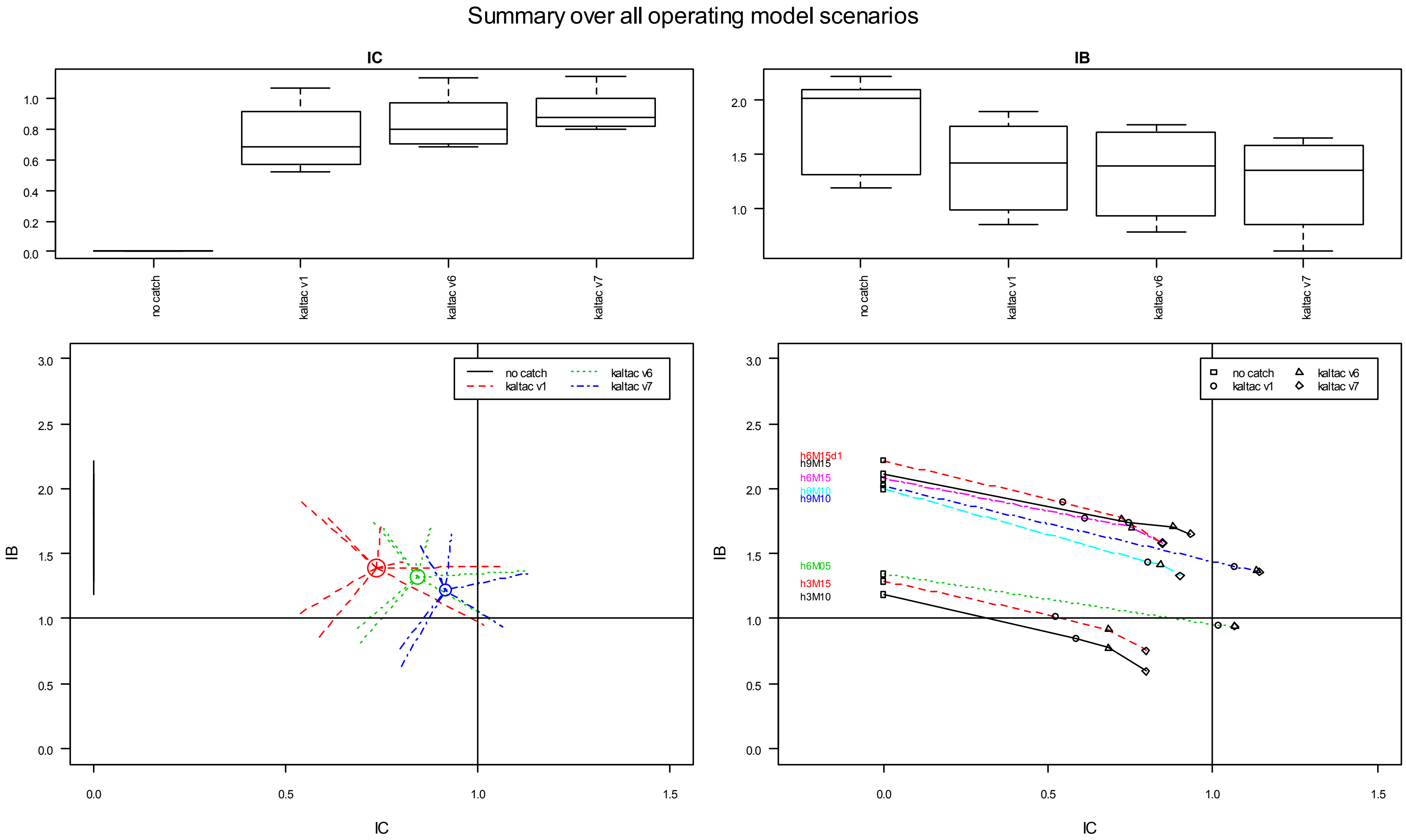


Figure A10-7. Kaltac rule, versions 1 to 8, summary over all models, hierarchy 1.

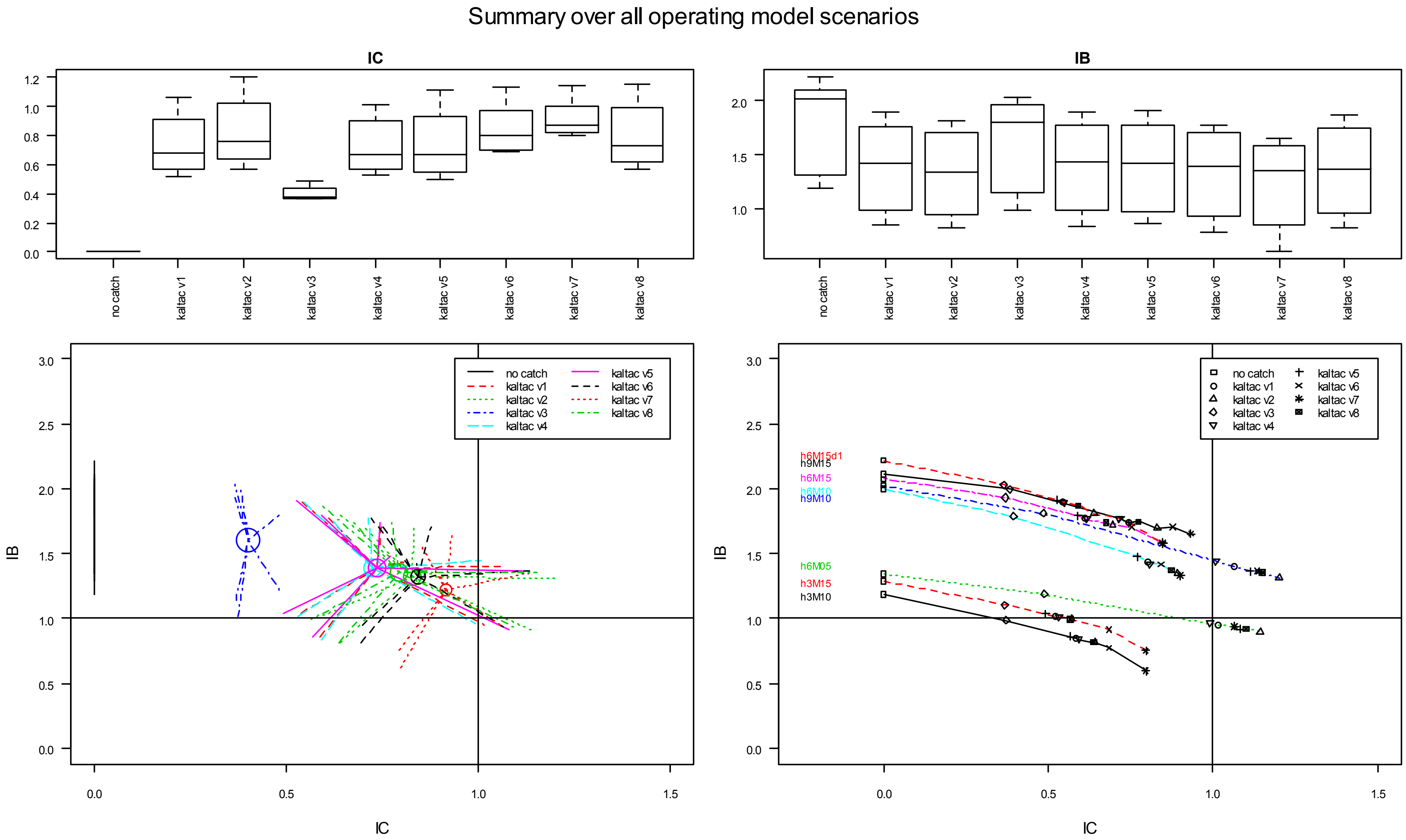


Figure A10-7 continued

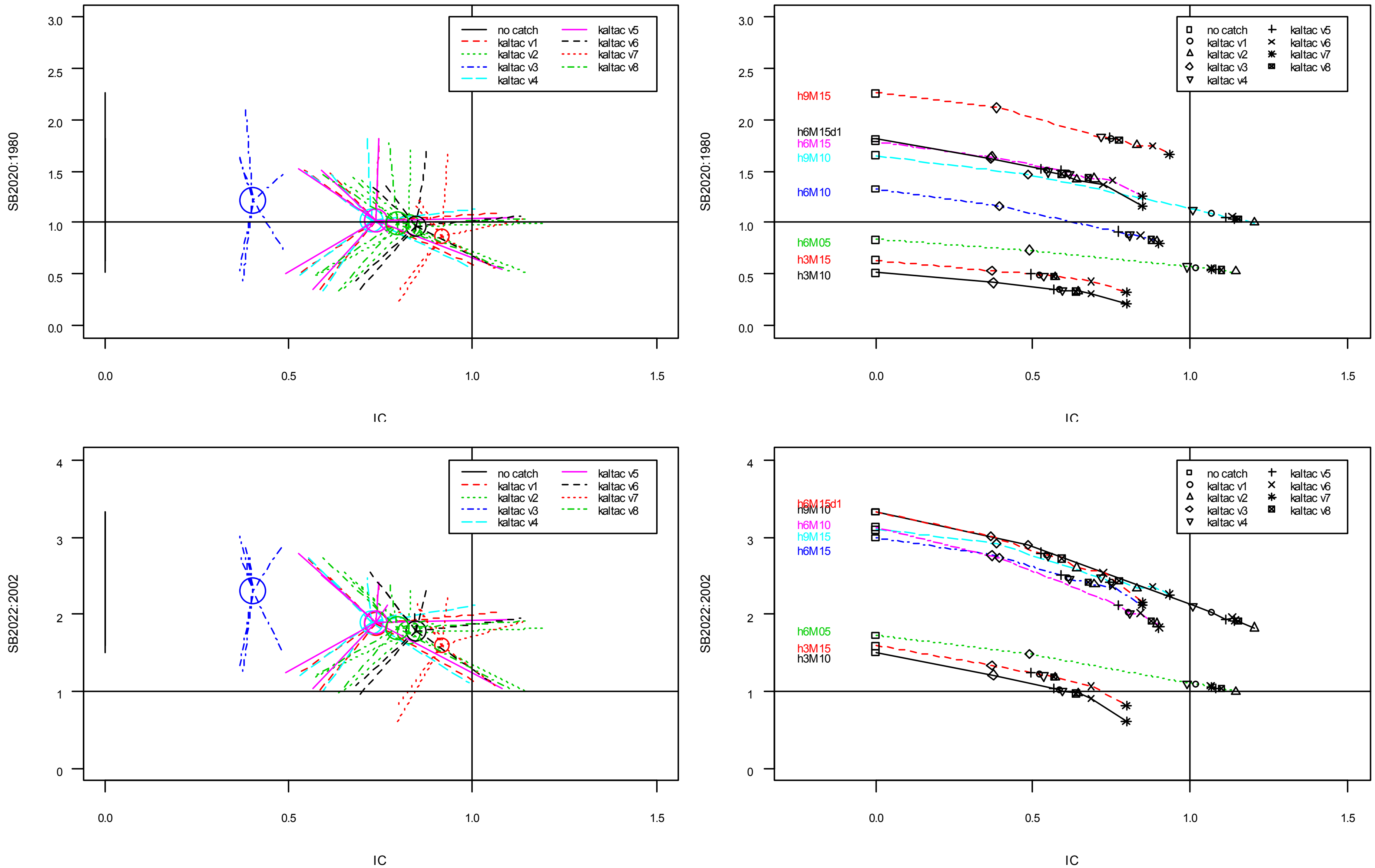


Figure A10-8. Kaltac rule, version 1 for all models, hierarchy 3.

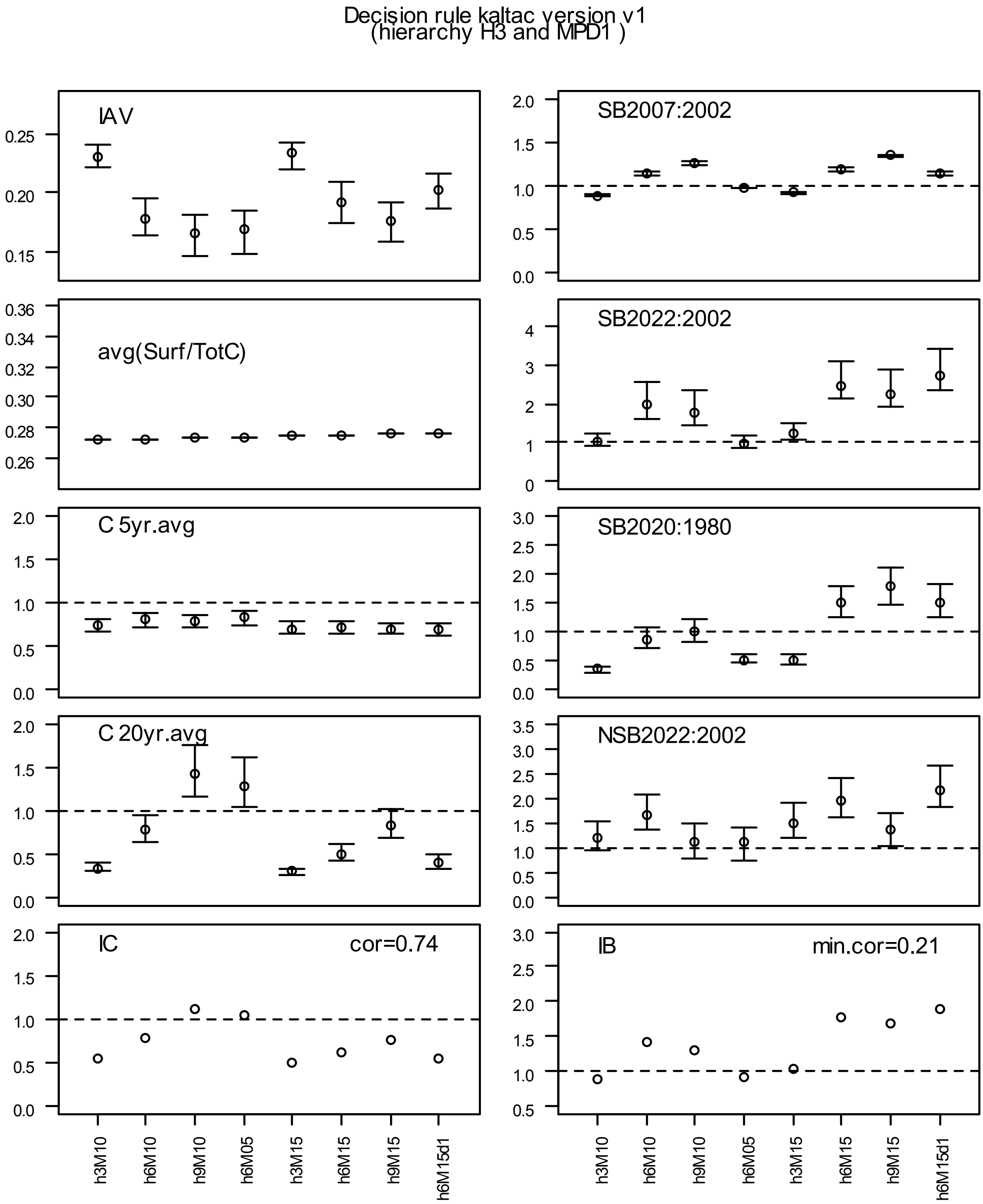


Figure A10-9. Kaltac rule, versions 1 and 6, hierarchy 3 summarised over all operating model scenarios.

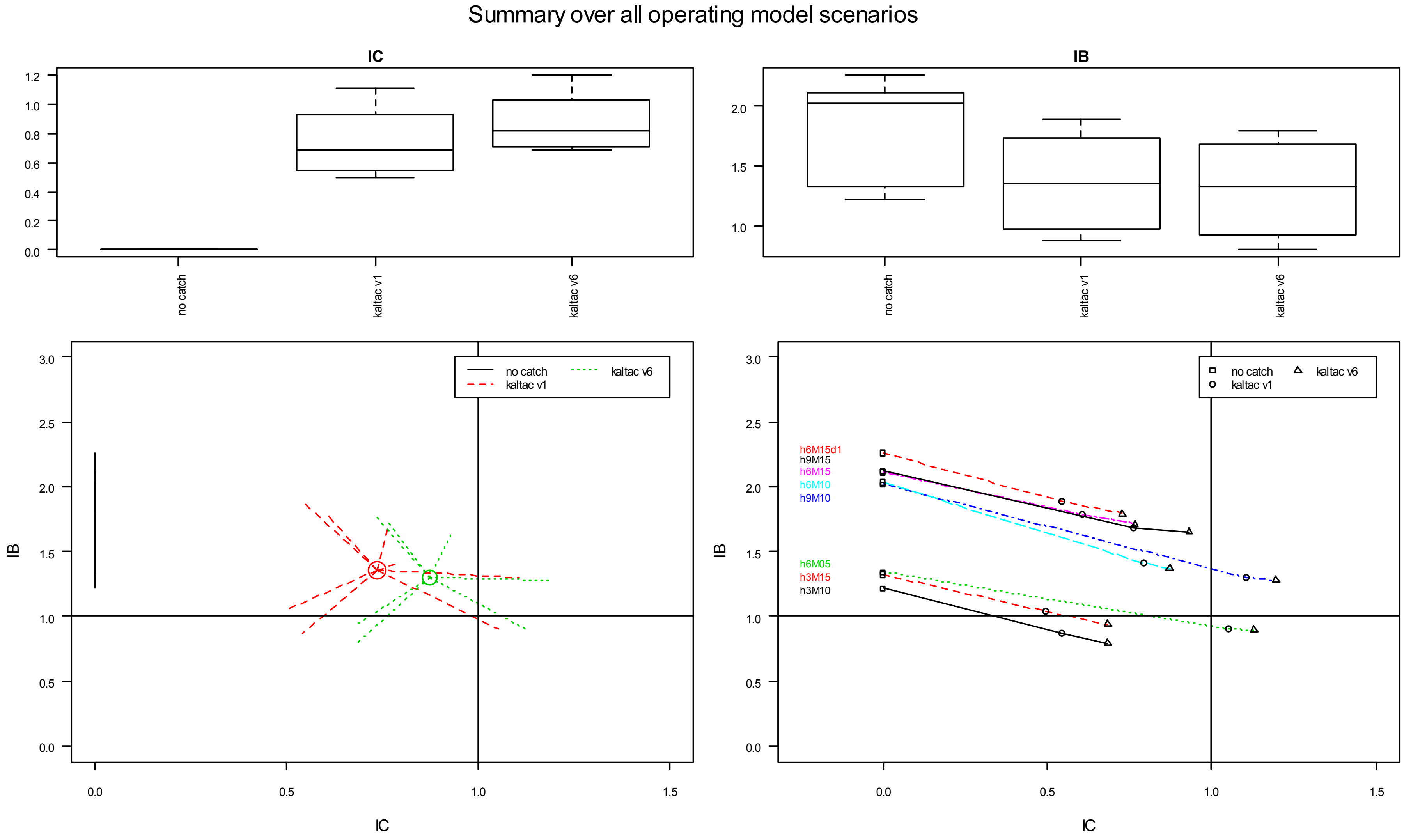


Figure A10-9 continued

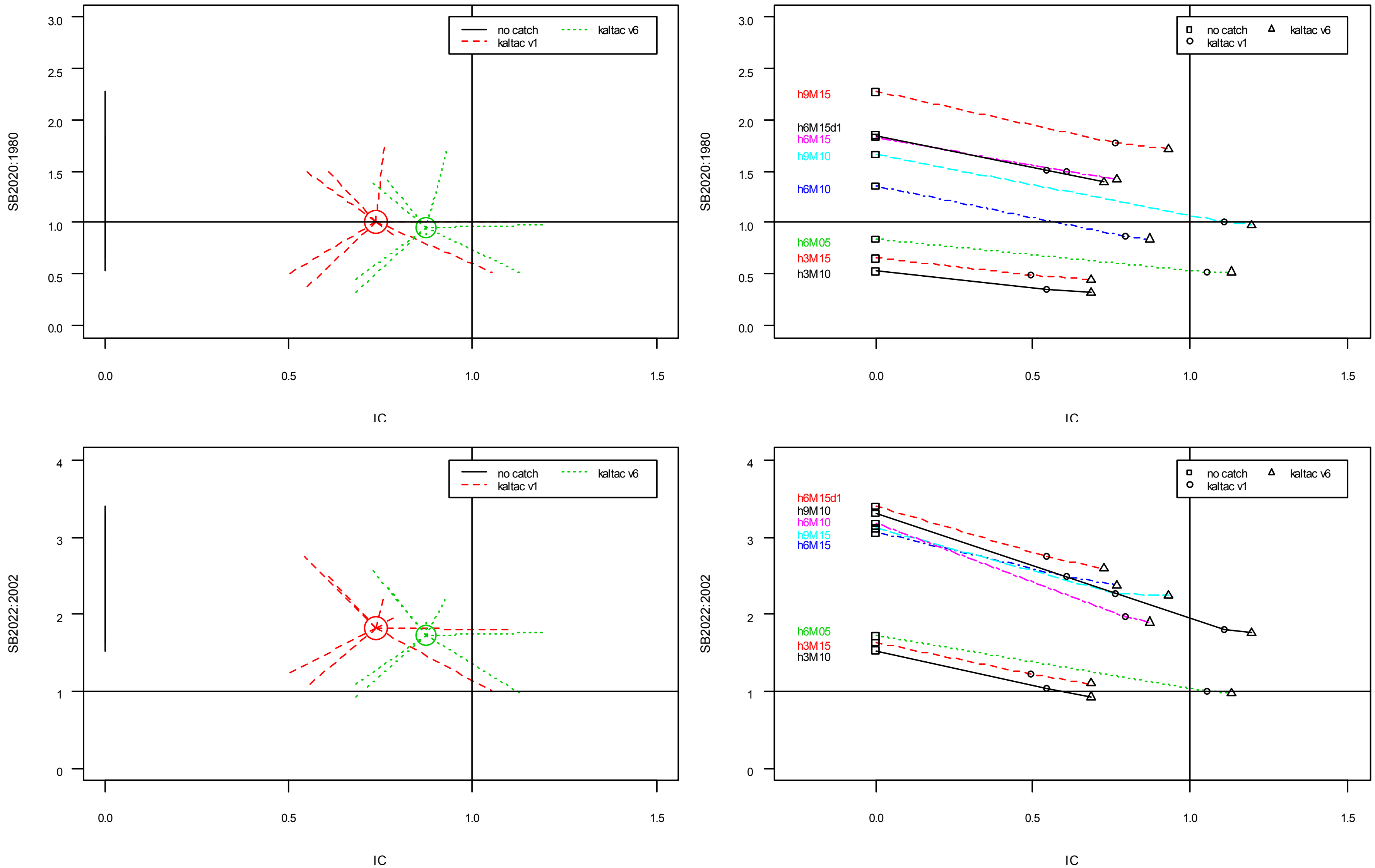
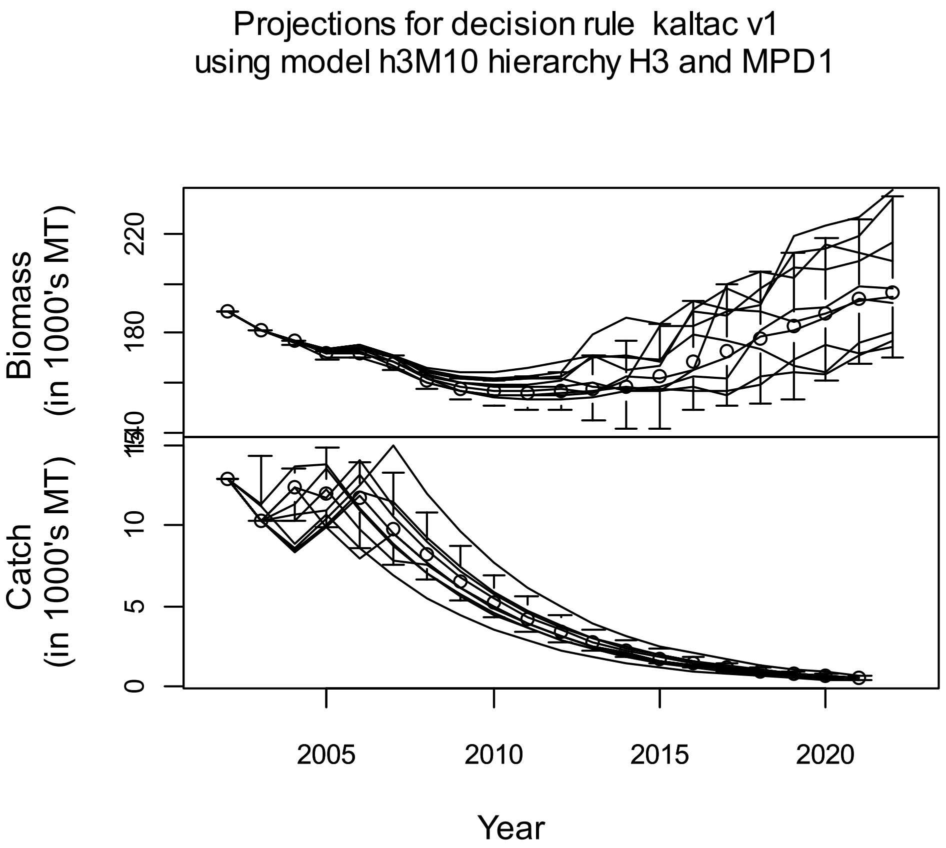


Figure A10-10. Kaltac rule, version 1, hierarchy 3, time trajectories of biomass and catch for model (a) h3m10, (b) h9m15 and (c) h6m05 (model is noted in figure heading). Note the different y-scales.



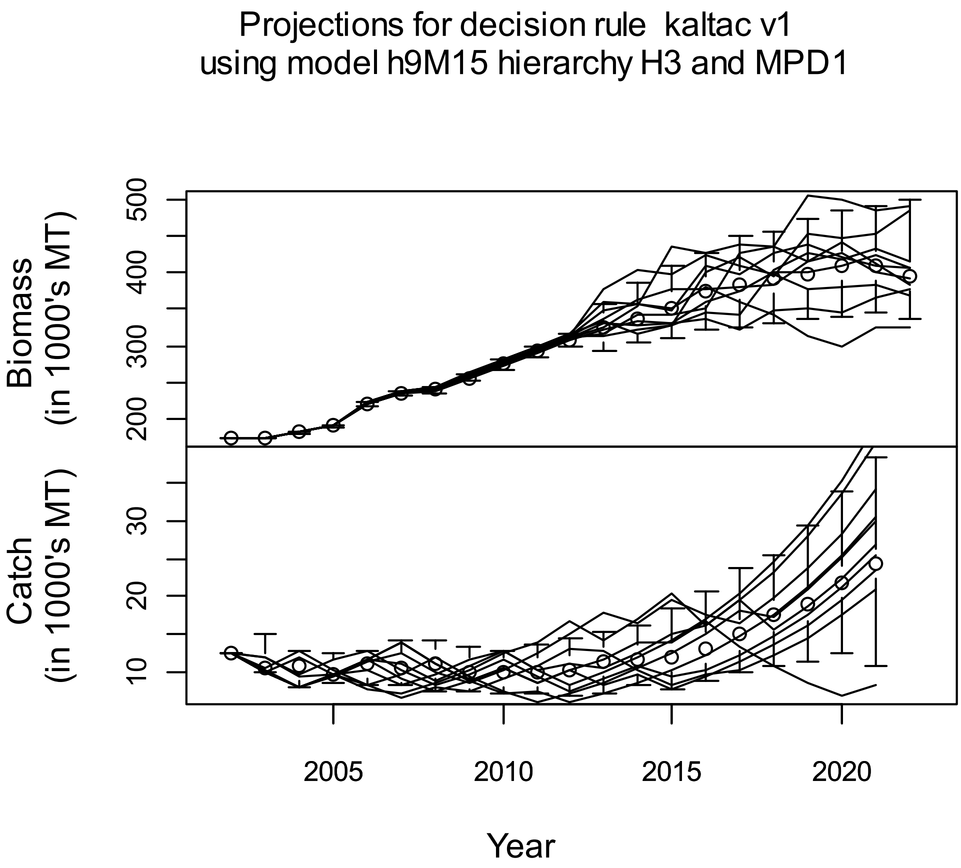
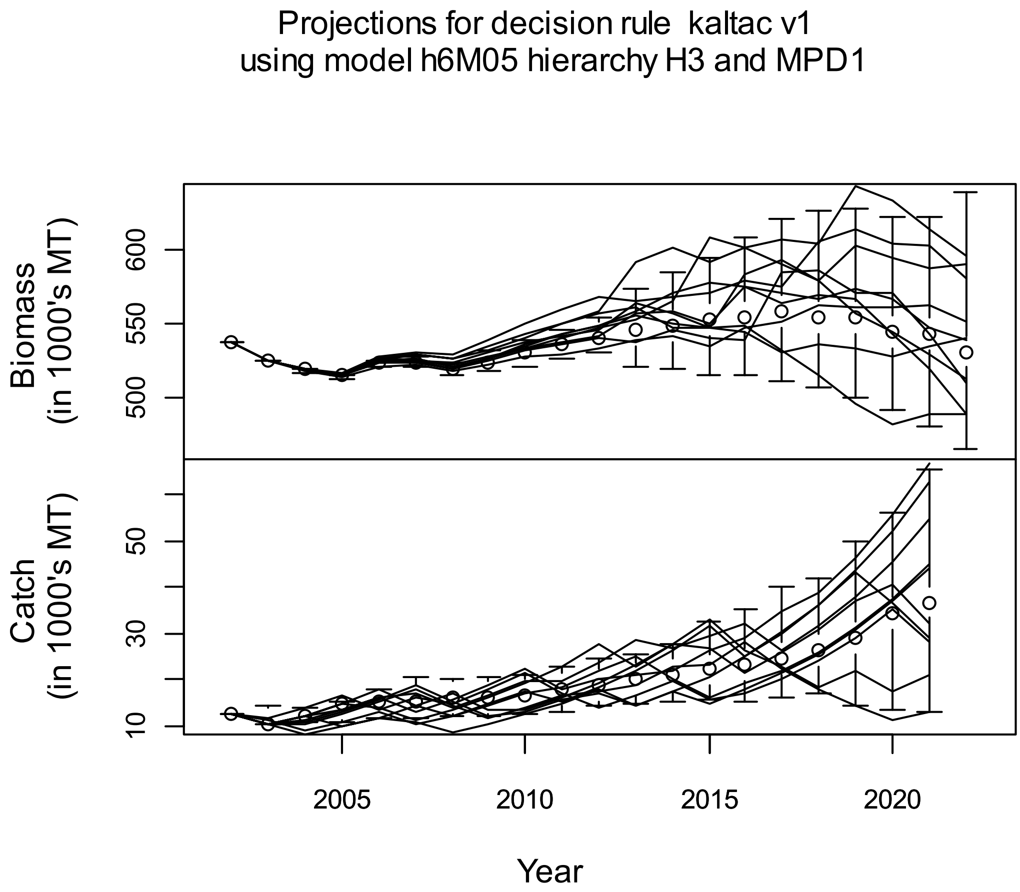


Figure A10-10 continued.



A11. Comp_FC ("Fox" and "CPUE" composite rule)

A11.1. Description of the rule

A11.1.1. Overview

This rule is a crude attempt at merging two of the previous decision rules, "Fox" and "CPUE". The TAC from both "Fox" and "CPUE" is calculated and a composite TAC is passed to the OM. We chose these two rules since they are very different in performance. "CPUE" tends to be overly aggressive while "Fox" is overly conservative.

A11.1.2. Mathematical description

$$TAC_{y+1} = \omega TAC_y + (1-\omega)\kappa TAC_{Fox} + (1-\omega)(1-\kappa)TAC_{CPUE}$$

A11.1.3. Versions (tuning parameter values)

For all versions, $\omega = 0.8$, CPUE (regression over 5 years, $\kappa = 1.0$), Fox ($\beta = 0.9$, $\eta = 0.9$).

Rule version	Details
v1	$\kappa = 0.05$
v2	$\kappa = 0.10$
v3	$\kappa = 0.15$
v4	$\kappa = 0.25$
v5	$\kappa = 0.50$
v6	$\kappa = 0.75$

A11.2. Performance of rule

A11.2.1. Overview

Merging the two rules yields results that fall between those of the two individual rules. The star plots do not flatten as one changes the value of κ. That is, merging rules "CPUE" and "Fox" results in intermediate performance levels that are not better than those of each separate rule. "CPUE" has smaller catch variability than "Fox" and one can see how the star plots of "comp_FC" get narrower as κ decreases.

Note that the composite rule evaluated here is very simple and could be formulated to go beyond a simple additive rule.

A11.2.2. Graphics

Figure A11-1a.

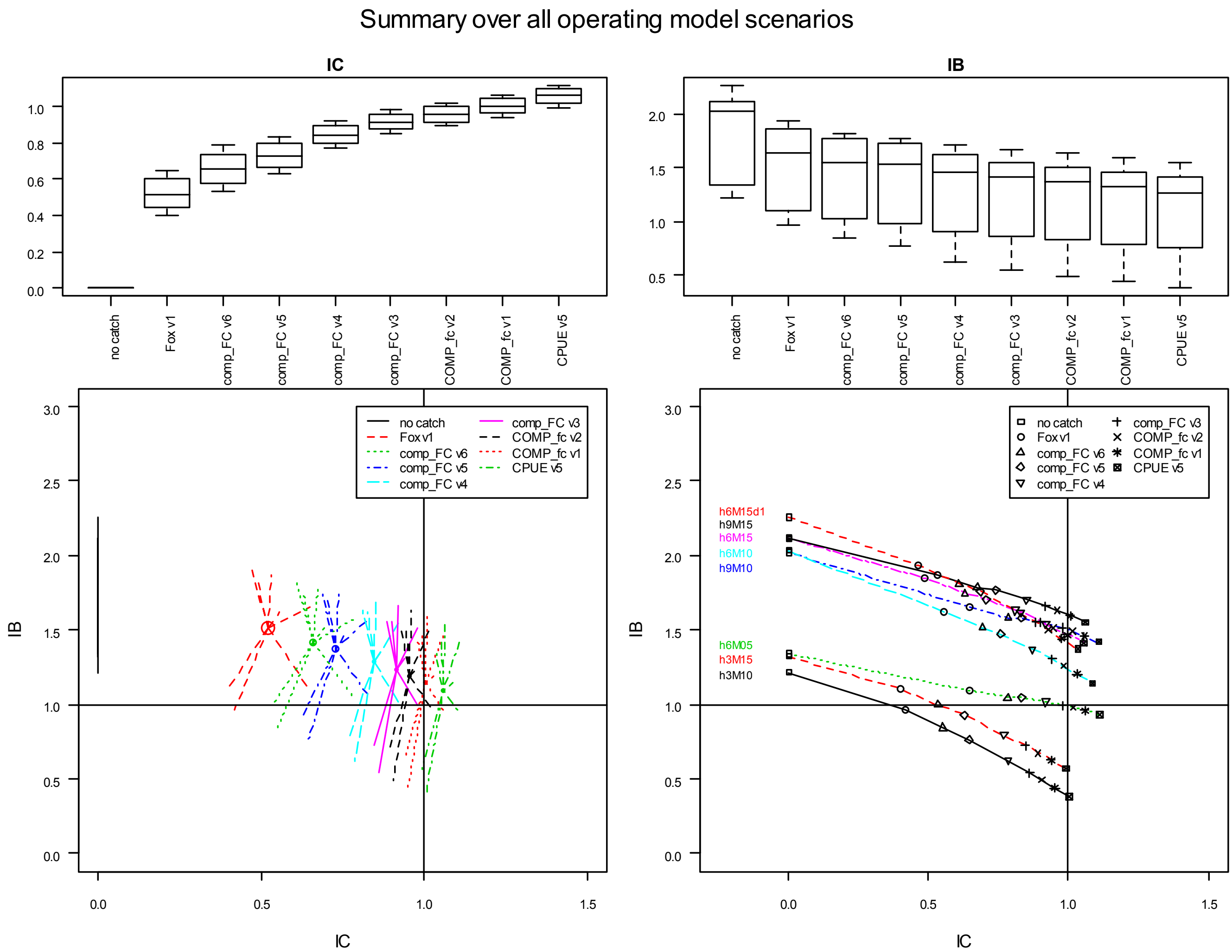


Figure A11-1b.

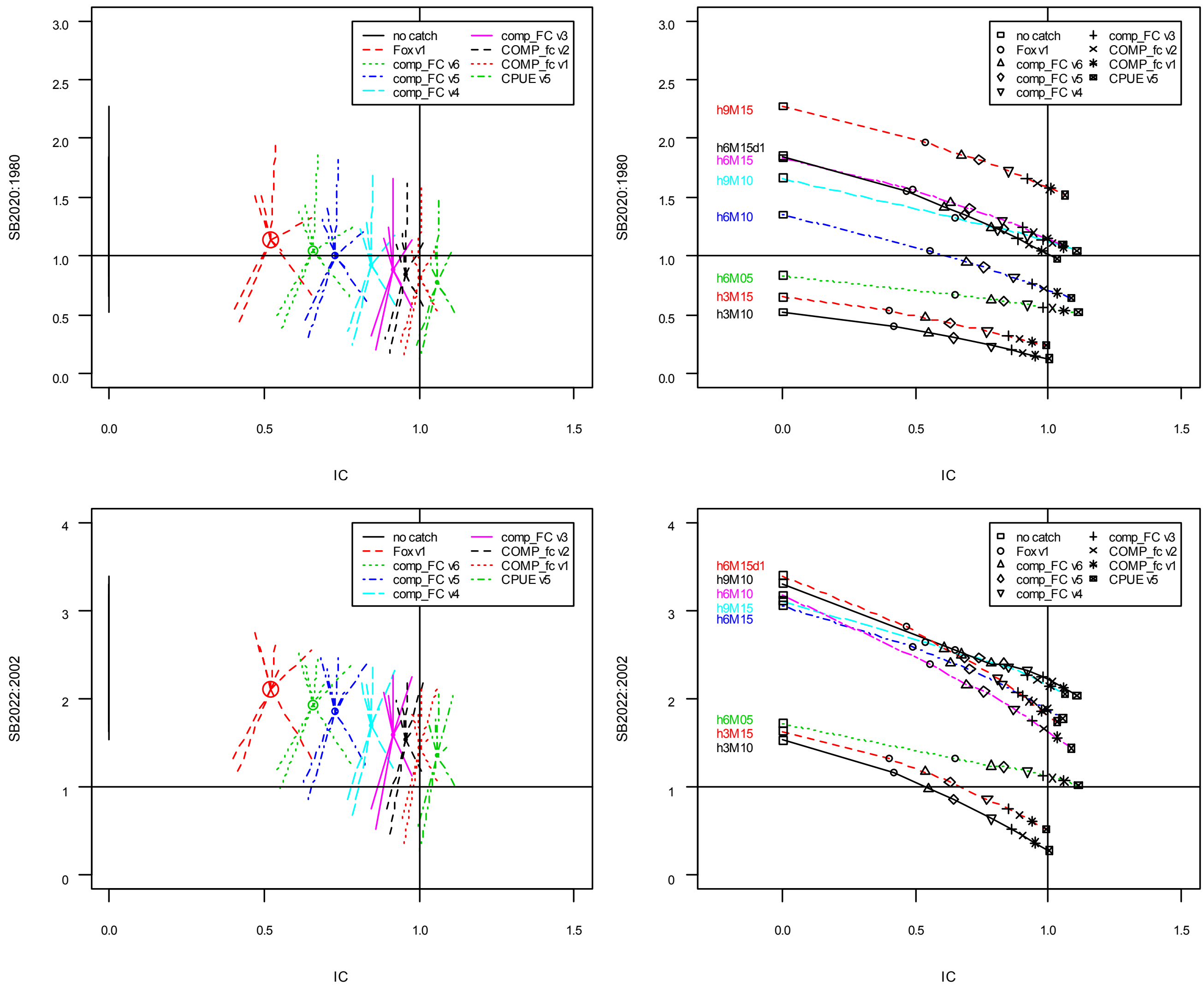


Figure A11-2.

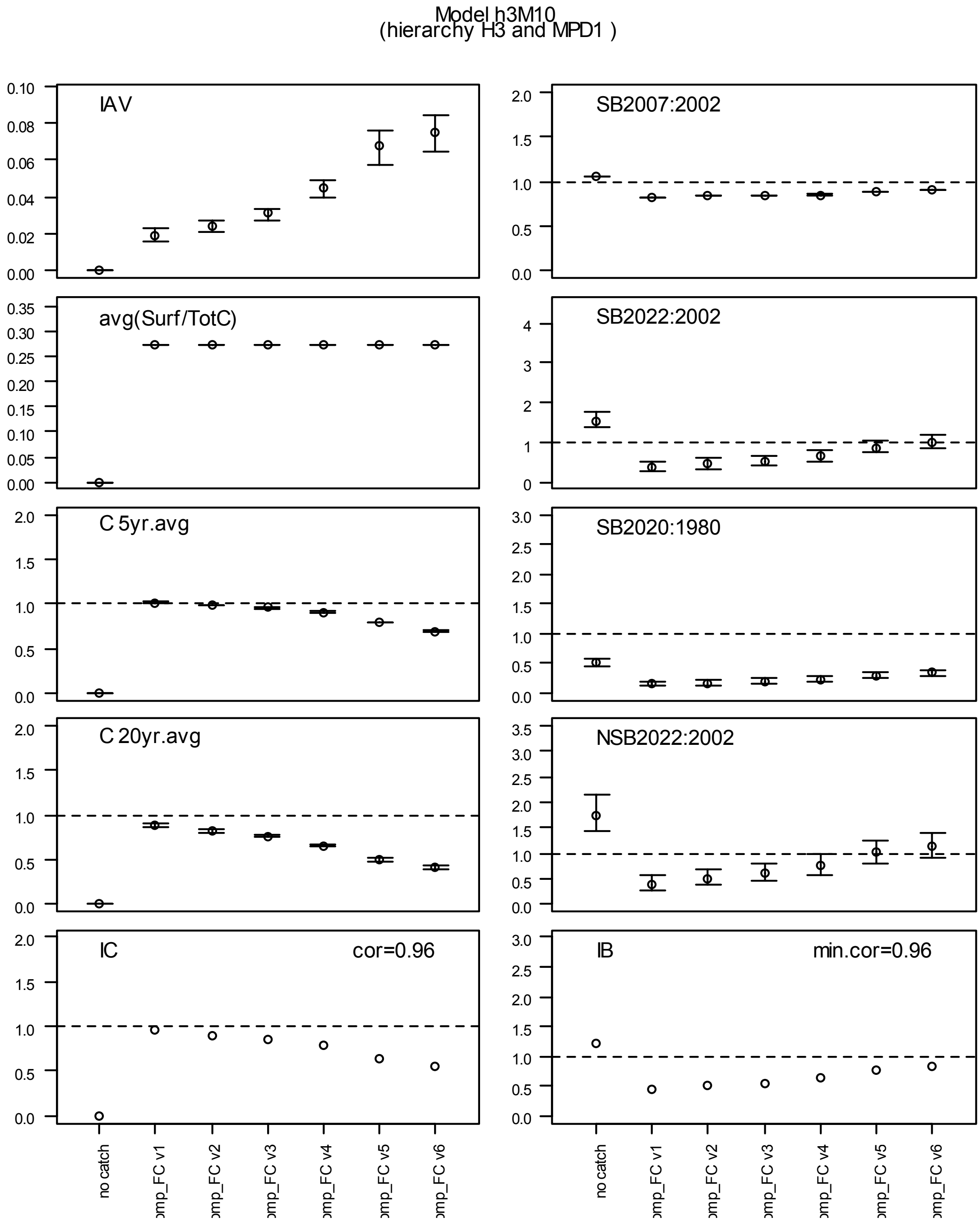

Model h3M10
(hierarchy H3 and MPD1)
IAV
SB2007:2002
avg(Surf/TotC)
SB2022:2002
C 5yr.avg
SB2020:1980
C 20yr.avg
NSB2022:2002
IC
cor=0.96
IB
min.cor=0.96
no catch
omp_FC v1
omp_FC v2
omp_FC v3
omp_FC v4
omp_FC v5
omp_FC v6

Figure A11-3.

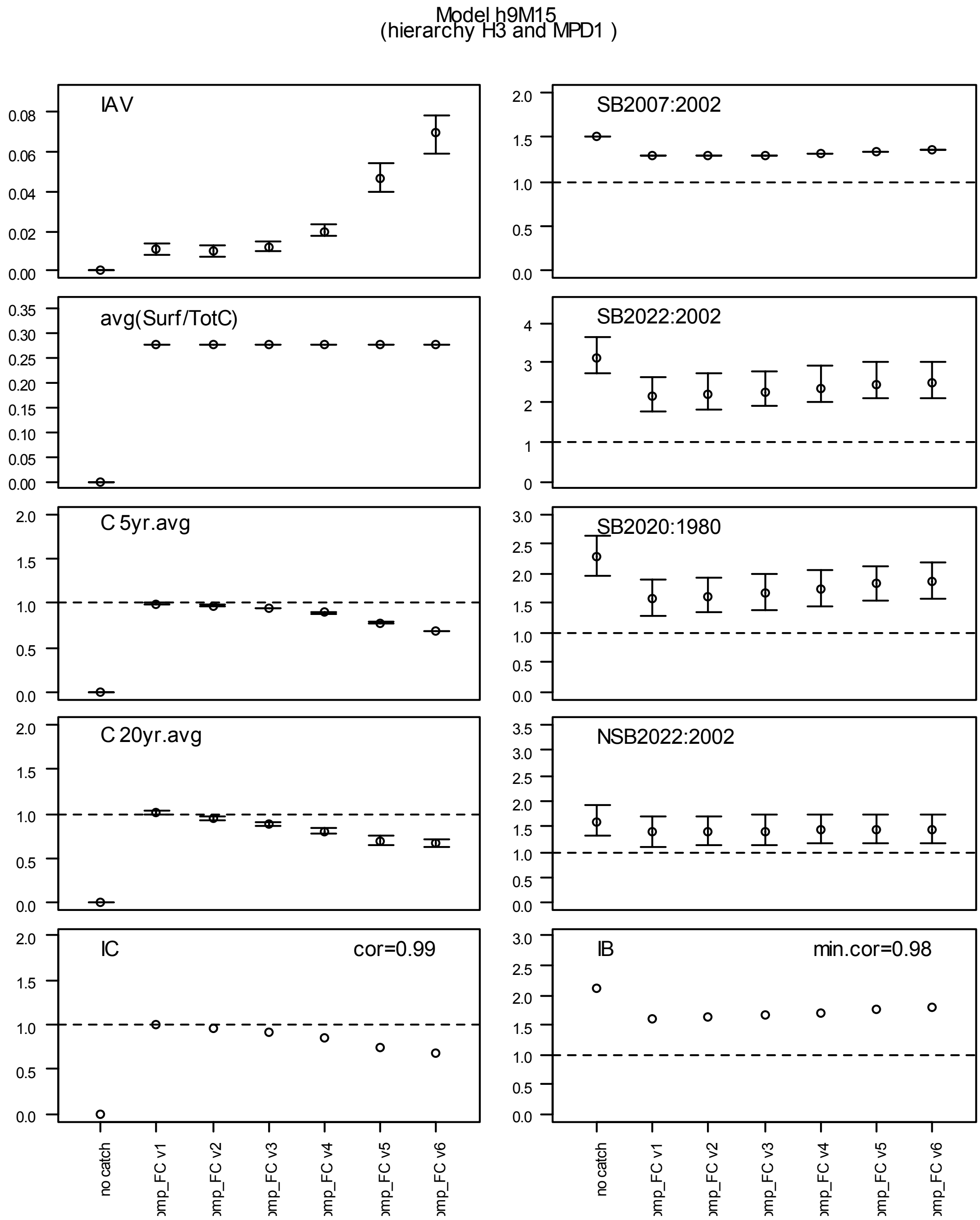

Model h9M15
(hierarchy H3 and MPD1)
IAV
SB2007:2002
avg(Surf/TotC)
SB2022:2002
C 5yr.avg
SB2020:1980
C 20yr.avg
NSB2022:2002
IC
cor=0.99
IB
min.cor=0.98
no catch
omp_FC v1
omp_FC v2
omp_FC v3
omp_FC v4
omp_FC v5
omp_FC v6

Figure A11-4.

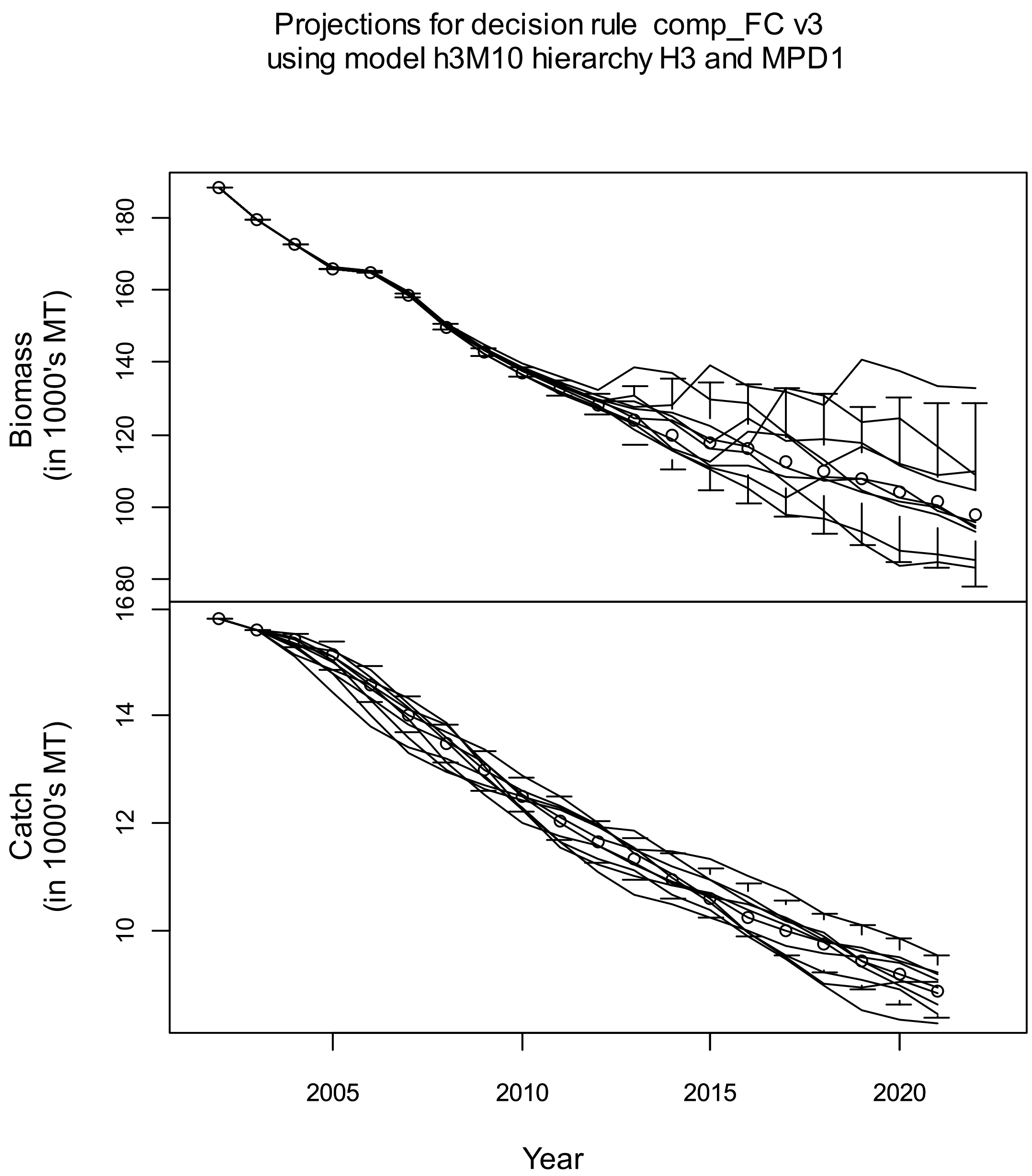
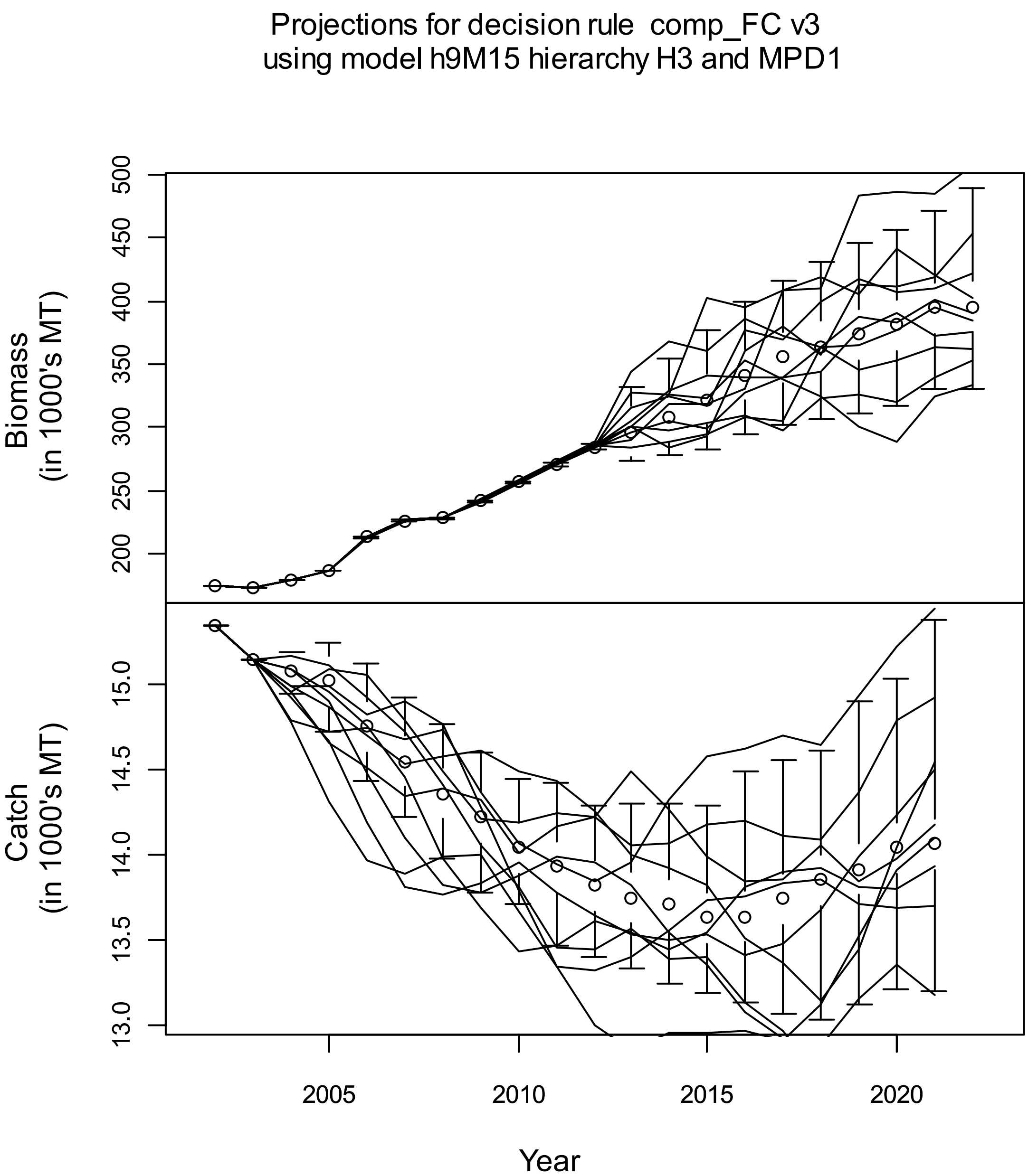

Projections for decision rule comp_FC v3
using model h3M10 hierarchy H3 and MPD1
Biomass
(in 1000's MT)
180
160
140
120
100
80
16
14
12
10
Catch
(in 1000's MT)
2005
2010
2015
2020
Year

Figure A11-5.


Projections for decision rule comp_FC v3
using model h9M15 hierarchy H3 and MPD1
Biomass
(in 1000's MT)
200
250
300
350
400
450
500
Catch
(in 1000's MT)
13.0
13.5
14.0
14.5
15.0
2005
2010
2015
2020
Year