SS3: User Guide: ICES fishing opportunites

Henning Winker (GFCM - ICES WKBSS3)

Contents

Load `r4ss`, `FLRef` and `ss3om`

```
library(r4ss)
library(FLRef)
library(ss3om)
library(ss3diags)
```

1 SS3 assessment summary for anf.27.3a4

1.1 Step 1: Set up file paths and folders structure for loading and saving the SS3 model

Set up the file path to the folder where the SS3 model folder with run is located.

Define name of reference model folder with the SS3 model outputs

```
model = "ref"
```

Create `.rds` stock file name

```
stock.file = paste0("anf.27.3a4_", model, ".rds")
```

Load reference model

```
ss3rep = SS_output(file.path(model))
```

Create an `rdata` folder in the assessment model subdirectory.

```
rdata = file.path("rdata")
dir.create(rdata, showWarnings = FALSE)
```

Save the model as `rdata` file

```
saveRDS(ss3rep, file = file.path("rdata", stock.file))
```

...or load directly as `.rdata` if these had been saved already

```
ss3rep = readRDS(file.path(rdata, stock.file))
```

1.2 Specify benchmarks

```
Fmsy = 0.137
Fpa = 0.215
Fupper = 0.174
Flower = 0.105
Fp0.5 = 0.215
Btrigger = 38604
Blim = 25686
Bpa = 35692
TAC = 11293  # 2023
```

Make `data.frame` that is compatible with `FLPar` in `FLR`

```
benchmarks = data.frame(params = c("Fmsy", "Fpa", "Fupper", "Flower",
    "Fp0.5", "Btrigger", "Blim", "Bpa"), data = c(Fmsy, Fpa,
    Fupper, Flower, Fp0.5, Btrigger, Blim, Bpa))
```

Convert to `FLPar`

```
bms = as(benchmarks, "FLPar")
bms
   An object of class "FLPar"
   params
       Fmsy      Fpa   Fupper   Flower    Fp0.5 Btrigger     Blim      Bpa
   1.37e-01 2.15e-01 1.74e-01 1.05e-01 2.15e-01 3.86e+04 2.57e+04 3.57e+04
   units:  NA
```

1.3 Step 2: Convert SS3 to FLStockR

First, the `ssmvln()` from `FLRef` is used to generate the stock trajectories with uncertainty using a Monte-Carlo to generate a large number of iterations from multivariate log-normal approximation of the variance-covariance estimates.

For the conventional ICES advice, it is important to extend the assessment horizon to the reference year+1 to plot SSB and recruitment one-step-ahead $(y+1)$ of F and $Catch$.

```
years = ss3rep$startyr:(ss3rep$endyr + 1)

mvn = FLRef::ssmvln(ss3rep, Fref = "Btgt", verbose = F, years = years)
```

The option `Fref=Btgt`, and not `Fref=MSY` is chosen because the reference points were based on $B_{tgt} = B_{40}$, with a corresponding $F_{tgt} = F_{40}$.

This can be checked by

```
mvn$Btgtref
   [1] 0.4
```

Next the `mvn` object can easily converted into the `FLStockR` object

```
stk = ss2FLStockR(mvn)
```

By default, the reference points for F_{tgt} and B_{tgt} are extracted together with `MSY`, `B0` and `R0`.

```
stk@refpts
   An object of class "FLPar"
   params
       Ftgt     Btgt      MSY       B0       R0
        0.1  45738.6  33637.2 254698.0 312656.0
   units:  NA
```

However, for the final advice plot only the agreed benchmarks should be shown. This can be done by specifying `stk@refpts` as the `FLPar` object `bms`

Here, the reference points F_MSY, $F_{pa} = F_{p0.5}$, $MSYB_{trigger}$ ($B_{trigger}$),B_{pa} and B_{lim} are selected for plotting

```
stk@refpts = bms[c("Fmsy", "Fpa", "Btrigger", "Bpa", "Blim")]
```

The option `osa=TRUE` allows to plot the one-step-ahead forecast for SSB and recruitment through 2023, while omitting 2023 for F and Catch.

```
plotAdvice(stk, osa = T)
```

Next make a `FLStockR` with iterations generated MVLN Monte-Carlo (default nsim = 1000) to depict uncertainty

```
# with uncertainty
stki = ss2FLStockR(mvn, output = "iters", thin = 1)
# assign benchmark reference points
stki@refpts = stk@refpts
```

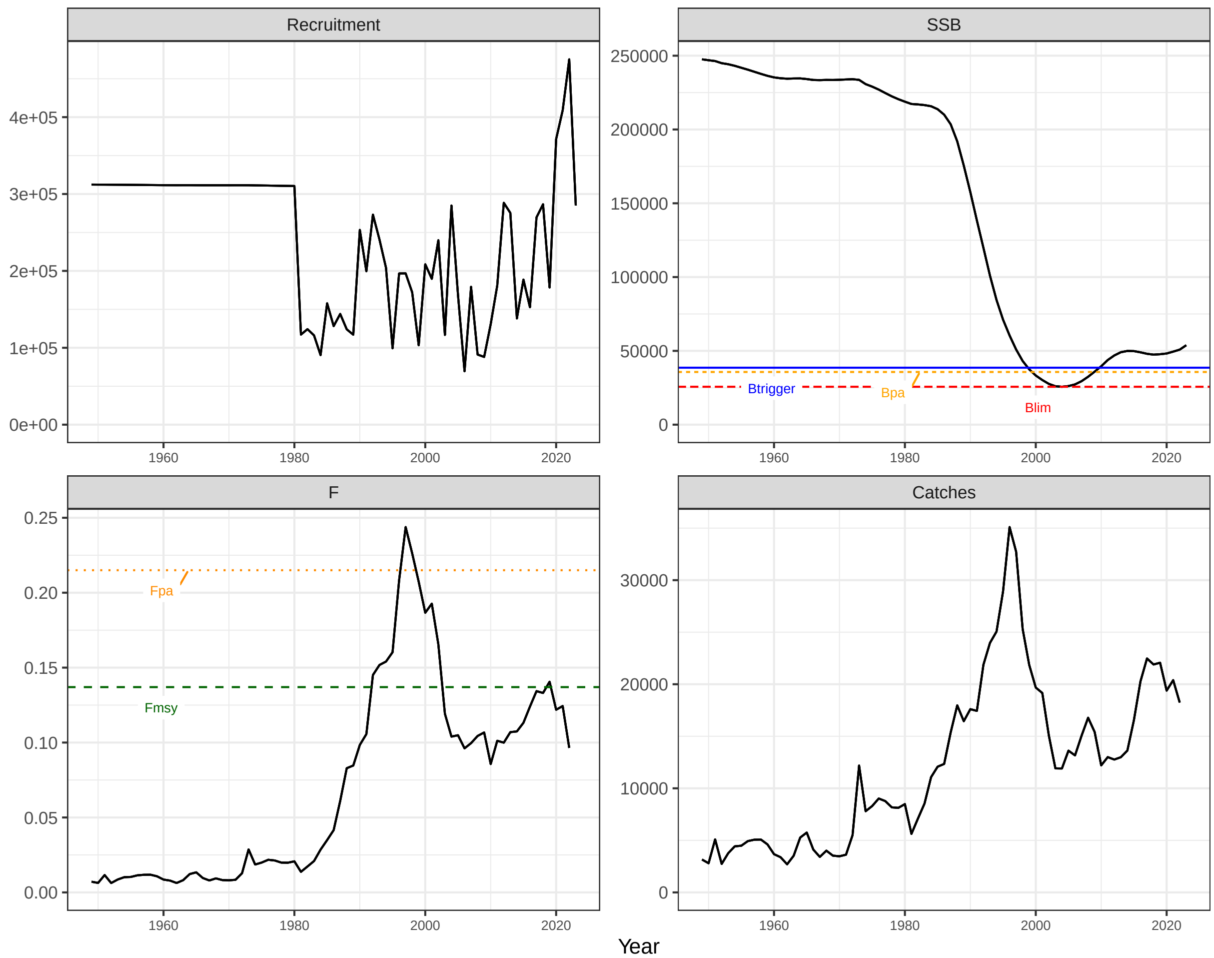

Figure 1: Comparison of estimated stock status trajectories.

Check

```
plotAdvice(stki, osa = TRUE)
```

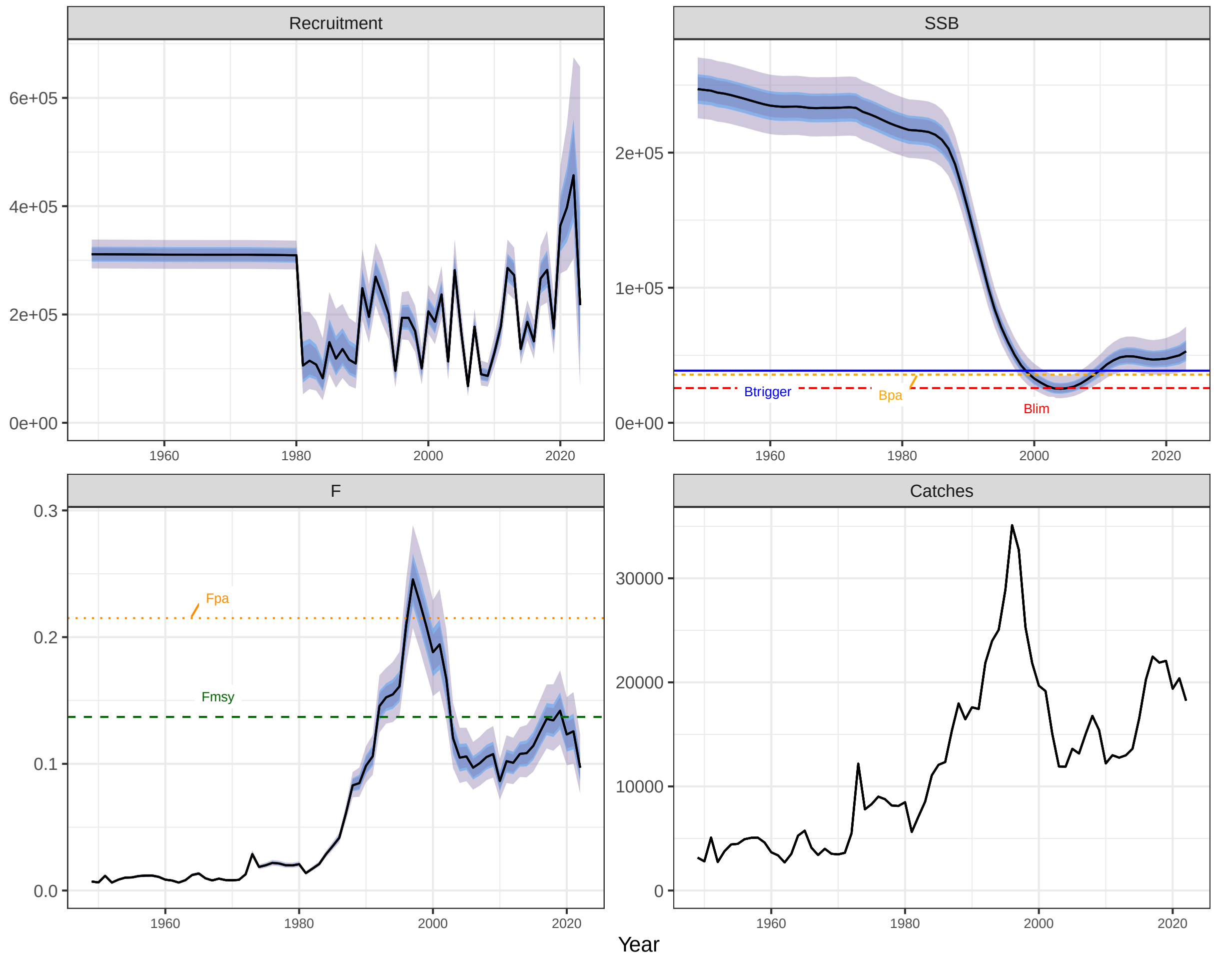

Figure 2: Uncertainty of estimated stock status trajectories with associated reference points, with solid line depicting the median

1.4 Step 3: Make Advice plot of stock status indicators with uncertainty

The final advice plot seeks to provide a standard format for presenting stock status indicators that shows the exact maximum likelihood estimates from the model (`stk`) and depicts the uncertainty around those from the Monte-Carlo approach (stki).

The plotting code allows to specify the years shown along the x-axis by adjusting the option `break=c(seq(years[1],tail(years,1),5),tail(years,1))` depending on the length of the time series (here every 5 years and the last year)

```
# Name plot padv
padv = plotAdvice(FLStocks(CIs = stki, Fit = stk), osa = TRUE) +
    scale_fill_manual(values = c("salmon", "black")) + scale_color_manual(values = c(0,
    "black")) + theme(legend.position = "none") + scale_x_continuous(breaks = c(seq(years[1] +
```

```
    1, tail(years, 1), 5), tail(years, 1))) + theme(axis.text.x = element_text(size = 8,
    angle = 90, vjust = 0.5))
padv
```

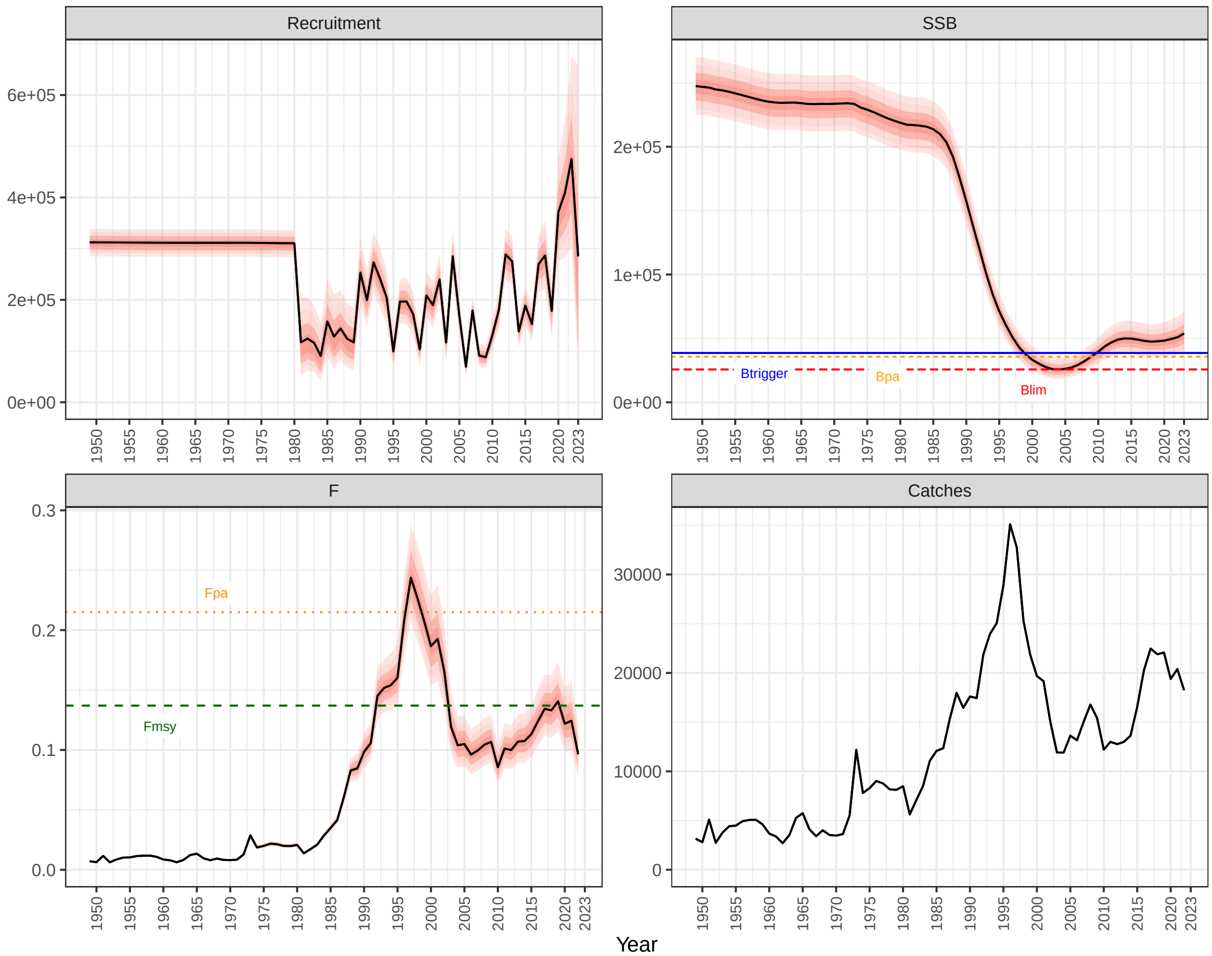

Figure 3: Estimated stock status trajectories with associated reference points for the 2024 benchmark update of European hake in GSAs 17-18

1.4.1 Save `FLStockR` objects in `.rds` format to `rdata`

It is adviced to specify additional information in the `FLStockR` object before saving it.

Label the `FLStockR` object properly

```
stk2 = stk
stk2@name = "anf.27.3a4"
stk2@desc = "2023, SS3, WKBSS3"
```

Note that `stk@name` will be used through this script to label file names!

```r
saveRDS(stk2, file = file.path(rdata, paste0(stk2@name, "_stk2.rds")))
saveRDS(stki, file = file.path(rdata, paste0(stk2@name, "_stki.rds")))
```

1.5 Step 5: Generate output trajectories for the ICES advice template

First a folder òutput is created to save the outputs (Tables, Figures).

```r
output.dir = file.path("output")
dir.create(output.dir, showWarnings = FALSE)
```

The output from ssmvln can now be directly converted in the ICES time series compatible format from SS3 model by

```r
icests = ss2ices(mvn)
```

```r
knitr::kable(icests, "pipe", align = "lccccc", caption = " Assessment summary. High and Low refer to 95%
```

Table 1: Assessment summary. High and Low refer to 95% confidence intervals.

year	Rec_low	Rec	Rec_high	SSB_low	SSB	SSB_high	Landings	Discards	F_low	F	F_upp
1949	285108.26	312257.0	338365.92	225450.30	247534.0	270464.14	3158.00	NA	0.006	0.007	0.008
1950	285072.40	312222.0	338331.94	224866.22	246919.0	269818.51	2800.00	NA	0.006	0.006	0.007
1951	285040.17	312190.0	338300.32	224354.77	246363.0	269216.56	5094.00	NA	0.010	0.012	0.013
1952	284954.82	312106.0	338218.08	222951.02	244928.0	267755.14	2736.00	NA	0.006	0.006	0.007
1953	284915.19	312065.0	338175.86	222309.21	244218.0	266974.03	3767.00	NA	0.008	0.009	0.010
1954	284852.16	312001.0	338111.17	221288.66	243139.0	265836.70	4426.00	NA	0.009	0.010	0.011
1955	284775.17	311923.0	338032.51	220031.83	241829.0	264475.50	4485.00	NA	0.009	0.010	0.011
1956	284697.49	311844.0	337952.52	218770.71	240515.0	263110.66	4929.00	NA	0.010	0.011	0.013
1957	284612.68	311758.0	337865.71	217395.14	239092.0	261643.49	5065.00	NA	0.011	0.012	0.013
1958	284527.18	311671.0	337777.58	216031.20	237683.0	260192.89	5076.00	NA	0.011	0.012	0.013
1959	284446.83	311589.0	337694.27	214744.94	236353.0	258822.32	4613.00	NA	0.010	0.011	0.012
1960	284382.08	311522.0	337625.30	213730.77	235294.0	257720.01	3672.00	NA	0.008	0.009	0.009
1961	284347.87	311485.0	337585.64	213191.98	234709.0	257087.42	3381.00	NA	0.007	0.008	0.009
1962	284332.45	311467.0	337565.10	212934.55	234415.0	256754.77	2700.00	NA	0.006	0.006	0.007
1963	284343.81	311476.0	337571.67	213104.76	234557.0	256864.81	3521.00	NA	0.007	0.008	0.009
1964	284347.59	311479.0	337573.87	213161.19	234607.0	256907.44	5270.00	NA	0.011	0.012	0.014
1965	284317.40	311450.0	337546.20	212700.25	234161.0	256480.51	5749.00	NA	0.012	0.013	0.015
1966	284276.77	311411.0	337509.04	212066.92	233543.0	255883.05	4117.00	NA	0.009	0.010	0.011
1967	284271.10	311405.0	337502.73	211971.76	233443.0	255778.30	3410.00	NA	0.007	0.008	0.009
1968	284280.87	311414.0	337510.91	212123.59	233584.0	255906.51	4012.00	NA	0.008	0.009	0.010
1969	284278.19	311411.0	337507.59	212082.70	233540.0	255859.33	3529.00	NA	0.007	0.008	0.009
1970	284287.80	311420.0	337515.93	212229.69	233681.0	255992.90	3470.00	NA	0.007	0.008	0.009
1971	284301.24	311433.0	337528.43	212434.52	233883.0	256190.59	3623.00	NA	0.008	0.008	0.009
1972	284313.21	311445.0	337540.41	212625.37	234076.0	256384.84	5497.00	NA	0.012	0.013	0.014
1973	284283.54	311417.0	337514.22	212162.16	233631.0	255962.54	12189.00	NA	0.026	0.029	0.032
1974	284089.61	311230.0	337335.10	209207.45	230736.0	253150.77	7801.00	NA	0.017	0.019	0.021
1975	283976.65	311118.0	337224.59	207530.64	229044.0	251452.28	8299.00	NA	0.018	0.020	0.022
1976	283842.54	310984.0	337091.30	205553.13	227036.0	249422.94	9021.00	NA	0.020	0.022	0.024
1977	283683.98	310825.0	336932.59	203262.39	224703.0	247057.78	8774.00	NA	0.019	0.021	0.024

year	Rec_low	Rec	Rec_high	SSB_low	SSB	SSB_high	Landings	Discards	F_low	F	F_upp
1978	283529.67	310669.0	336775.58	201076.38	222459.0	244763.80	8172.00	NA	0.018	0.020	0.022
1979	283396.58	310533.0	336637.25	199220.02	220536.0	242779.26	8123.00	NA	0.018	0.020	0.022
1980	283278.77	310412.0	336513.57	197587.27	218842.0	241028.31	8485.00	NA	0.019	0.021	0.023
1981	52704.35	117171.0	204994.30	196077.62	217284.0	239426.87	5623.00	NA	0.012	0.014	0.015
1982	62322.94	124233.0	204702.22	195829.68	216983.0	239068.84	7104.00	NA	0.016	0.017	0.019
1983	59413.94	116114.0	189167.76	195394.26	216527.0	238592.99	8542.00	NA	0.019	0.021	0.023
1984	41810.55	90512.3	156117.39	194621.82	215750.0	237816.11	11075.00	NA	0.026	0.029	0.031
1985	89760.59	157813.0	241365.59	192643.19	213758.0	235822.72	12078.00	NA	0.032	0.035	0.038
1986	64818.30	128221.0	210351.62	189014.84	210029.0	232007.34	12343.00	NA	0.038	0.042	0.046
1987	82489.70	144013.0	219301.48	182805.86	203532.0	225235.01	15377.00	NA	0.055	0.061	0.068
1988	68281.71	124042.0	193548.04	171760.86	191954.0	213146.83	17973.00	NA	0.074	0.083	0.093
1989	63164.96	116976.0	184645.95	156108.28	175514.0	195954.34	16451.00	NA	0.074	0.085	0.097
1990	190189.54	253315.0	320862.23	139153.55	157560.0	177030.18	17605.00	NA	0.085	0.098	0.114
1991	147470.09	199549.0	255738.69	121074.12	138380.0	156789.04	17441.00	NA	0.091	0.106	0.123
1992	216718.04	273187.0	331872.80	103673.33	119818.0	137106.20	21872.00	NA	0.125	0.145	0.170
1993	182751.97	240886.0	302752.50	85986.03	100981.0	117191.46	23971.00	NA	0.132	0.152	0.176
1994	153754.90	204018.0	257695.75	70654.38	84505.7	99642.02	25057.00	NA	0.133	0.154	0.180
1995	64894.33	99352.1	138936.17	58445.69	71165.6	85211.97	28913.00	NA	0.138	0.160	0.188
1996	154082.25	196656.0	241195.79	48724.97	60429.7	73493.95	35100.00	NA	0.179	0.208	0.245
1997	152566.74	196760.0	243251.15	40088.71	50927.4	63193.84	32728.00	NA	0.207	0.244	0.288
1998	130543.74	172183.0	216510.98	33049.36	43130.9	54717.40	25293.00	NA	0.191	0.226	0.271
1999	70811.50	103308.0	139749.76	28049.84	37486.7	48479.15	21854.00	NA	0.172	0.207	0.252
2000	164670.60	208576.0	254322.43	24403.53	33294.3	43771.36	19682.00	NA	0.153	0.187	0.229
2001	145307.82	189704.0	236712.30	21825.05	30248.3	40261.76	19157.00	NA	0.158	0.193	0.238
2002	192608.69	239906.0	288744.75	19636.98	27639.0	37233.37	15067.00	NA	0.134	0.165	0.206
2003	79084.78	116755.0	159251.87	18350.12	26040.4	35302.77	11916.00	NA	0.097	0.119	0.148
2004	232130.75	285050.0	339275.52	18170.94	25685.6	34716.99	11906.00	NA	0.085	0.104	0.128
2005	132669.07	168738.0	206402.53	18696.74	26145.6	35044.79	13618.00	NA	0.086	0.105	0.128
2006	49050.68	69476.3	92040.18	19774.10	27277.6	36174.46	13163.00	NA	0.080	0.096	0.117
2007	148852.44	179284.0	210140.22	21717.71	29464.0	38563.73	14639.53	415.467	0.083	0.100	0.121
2008	68833.18	91175.1	115012.87	24275.04	32464.9	42008.59	16476.49	305.513	0.087	0.104	0.127
2009	66803.64	88114.3	110801.36	27138.13	35941.1	46142.86	15349.83	60.168	0.089	0.107	0.130
2010	102780.79	130752.0	159963.94	30038.85	39576.1	50596.29	12117.18	87.824	0.072	0.086	0.104
2011	143558.52	181007.0	219930.50	33424.48	43791.7	55733.07	12884.46	112.537	0.085	0.101	0.123
2012	239487.15	288547.0	338300.75	35780.10	46889.9	59688.46	12275.08	488.923	0.084	0.100	0.121
2013	228001.51	275403.0	323541.24	37409.73	49091.2	62558.55	12363.31	625.694	0.090	0.107	0.129
2014	107507.66	138201.0	170434.79	37930.32	49946.8	63827.57	13275.26	352.739	0.089	0.107	0.131
2015	152139.89	188727.0	226423.35	37722.19	49841.2	63867.27	16167.81	384.195	0.094	0.113	0.138
2016	118100.14	152727.0	189207.30	36995.03	49023.5	62967.82	19810.43	460.574	0.103	0.124	0.150
2017	215149.34	269661.0	326138.76	36159.51	48028.9	61807.06	21860.71	614.292	0.112	0.134	0.163
2018	222139.96	286629.0	354489.20	35631.64	47455.9	61202.63	21562.49	335.508	0.110	0.133	0.163
2019	127008.31	178450.0	235042.30	35713.40	47740.5	61752.14	21716.40	349.596	0.115	0.141	0.174
2020	275752.24	371448.0	474453.83	35771.41	48220.0	62791.11	19090.88	297.125	0.099	0.122	0.153
2021	282108.65	408902.0	550621.42	36451.90	49489.3	64810.82	20176.42	215.576	0.100	0.124	0.157
2022	303472.18	475189.0	674652.16	37116.50	50852.0	67076.19	17997.53	241.475	0.077	0.096	0.123
2023	67743.28	284961.0	657394.60	39155.74	53819.1	71170.42	NA	NA	NA	NA	NA

The `timeseries` can be saved as `.csv` files

```
write.csv(icests, file = file.path(output.dir, paste0(stk2@name,
    "_ts.csv")), row.names = F)
```

2 F-based forecasts with SS3 and FLR

This section introduces F-based forecasting with SS3 for multi-fleet models (also works for single fleets and seasonal models), which is based on the so-called apic F values (F_{apic}), whereas the choice of the reference F-basis and the associated reference points may differ from the F_{apic} scale. For instance, GFCM and ICES, F_{bar} (option 5) is the default option. It is therefore necessary to rescale F-basis to F_{apic} for generating forecasts that are consistent with, e.g., F_{tgt} or F_{cur}.

F_{apic} is used for good reason in forecasts in order to account for multi- fleet selectivity. Comparing the partial impacts selectivity pattern requires setting the instantaneous rate of fishing mortaly F at comparable constant levels. For this purpose, it is important to consider that the definition of selectivity differs across regions (e.g. $Fbar$ or exploitation rate). With regards to temporal compatibility of partial fleet selectivity effects, F_{bar} has the undesirable property that its scale depends on the pre-specified age range across which F_a is averaged. For example, if F_{bar} is set to ages 2-4 to represent the dominant age classes under the current selectivity regime, but the goal is to evaluate the effect of selecting fish only at age-5, a common F_{bar} would result in disproportionately high F_a on ages 5+. This is because F_{bar} is computed for age ranges that are hardly selected for the definition $S_a = F_a/max(F_a)$. For this reason, it is more straight forward to use F_{apical} as the standardized quantity F quantify to account for partial impacts of fleet selectivity.

In the following, step-by-step guidelines are provided to setup an F_{apic}, so that it correctly corresponds to the F_{bar} baseline for F_{tgt} across multiple fleets and seasons.

2.1 Step 1: Basic setup

In this a case, a folder with the reference model run is created and the model outputs are loaded with `r4ss::SS_output`

next set up the folder where the SS3 model folder with run is located

Load the assessment model

```
ss3rep = readRDS(file.path("rdata", stock.file))
```

To organise the forecast outputs, first create a subfolder `forecast`

```
forecast.dir = file.path(model, "forecast")

dir.create(forecast.dir, showWarnings = F)
```

A new helper function `SSnewrun` was added to `ss3diags` to easily create subfolders for the forecast scenarios. First a `Fmsy` reference folder is created

To this specify a new subfolder path, where to run the forecast for a “fixed” F_{MSY} scenarios

```
fmsydir = file.path(forecast.dir, "Fmsy")
```

Create new F forecast model folder. Note that the data and control file and ss.exe names need to be specified if these diverge from the defaults `data.ss`, `control.ss` and `ss3.exe`

```
dat = "ang_dat.ss"
ctl = "ang_ctl.ss"
ss.exe = "ss3.exe"

SSnewrun(model = file.path(model), dat = dat, ctl = ctl, newdir = fmsydir,
    ss.exe = "ss3.exe")
```

Now the forecast file can be read be read with `r4ss`

```
fc <- SS_readforecast(file.path(fmsydir, "forecast.ss"), verbose = F)
```

2.2 Step 2: Initial F exploitation calculations for Fapic forecast

Extract the `$exploitation` output from the report file

```
Fexp = ss3rep$exploitation
```

Importantly, the `annual_F` are scaled to the F-basis (here F_{bar}), whereas fleet specific F values are always given as F_{apic}

Next compute the combined F_{apic} generically across fleets

```
Fexp$Fapic = apply(as.matrix(ss3rep$exploitation[, -c(1:6)]),
    1, sum, na.rm = T)
```

and aggregate across seasons, by taking the `mean` and not the `sum`.

```
Fapic = aggregate(Fapic ~ Yr, Fexp, mean)
```

Next compute the corresponding annual F_{bar} values from the `annual_F`

```
Fbar = aggregate(annual_F ~ Yr, Fexp, mean)
```

To work out exact ratio between F_{apic} and F_{bar} so that it is consistent with the benchmark calculations with ss3, it is necessary to extract the reference years for selectivity from the `forecast.ss` file.

The information required for the average selectivity conditions can be found in the `forecast.ss` file under `$Bmark_years`. The third and fourth position define the time horizon for the average selectivity across fleet, a value of -999 (here) indicates that the whole time series is use, but more commonly averages are taken, e.g. over the last 3 years, which can be specified as -2 0 or 2020 2022. The following code attempts to compute this generically.

```
endyr = ss3rep$endyr
if (fc$Bmark_years[3] < -90) {
    nfc = length(min(ss3rep$exploitation$Yr + 1):endyr)  # excluded init year
} else {
    # if specified (e.g. -2, 0)
    nfc = fc$Bmark_years[4] - fc$Bmark_years[3] + 1
}
# Benchmark reference years
bmyrs = (endyr - nfc + 1):endyr
bmyrs
    [1] 2012 2013 2014 2015 2016 2017 2018 2019 2020 2021 2022
```

NOTE: Other than here, it recommended to set the Bmark_years in the `forecast.ss` so that all quantities represent the mean last 3 years (i.e. -2). The advantage is that this allows using be consistent with the default FLR settings.

```
Fratio = mean(Fapic$Fapic[Fapic$Yr %in% max(bmyrs)]/Fbar$annual_F[Fbar$Yr %in%
    max(bmyrs)])
Fratio
   [1] 1.581069
```

`Fratio` defines the ratio of F_{apic} to F_{bar} for the reference period

Set F_{MSY} to benchmark

```
Fmsy = bms["Fmsy"][[1]]
```

This value is given as F_{bar} and therefore needs to be transformed to F_{apic}

```
Fmsy.apic = Fmsy * Fratio
Fmsy  # Fbar
   [1] 0.137
Fmsy.apic
   [1] 0.2166064
```

2.3 Step 3: Setting up the manual F forecast input structure

First, do some basic house keeping for the model structure. This is designed to work generically for any multi-fleet or seasonal structure

```
nseas = length(unique(ss3rep$exploitation$Seas))  # number of seasons
fleets = unique(ss3rep$fatage$Fleet)  # fleets
nfleets = length(fleets)  # number of fleet
```

Next, the mean Fapic by fleet and season is calculated

```
# subset to benchmark years for selectivity
fexp = ss3rep$exploitation[ss3rep$exploitation$Yr %in% bmyrs,
    ]
fexp = cbind(fexp[, 1:2], fexp[, -c(1:5)])[, -3]  #><  single fleet trick
# flip
fexp = reshape2::melt(fexp, id.vars = c("Yr", "Seas"), variable.name = "Fleet",
    value.name = "Fapic")
tail(fexp)
        Yr Seas Fleet    Fapic
   6  2017    1 Fleet 0.216209
   7  2018    1 Fleet 0.213365
   8  2019    1 Fleet 0.224472
   9  2020    1 Fleet 0.194284
   10 2021    1 Fleet 0.191422
   11 2022    1 Fleet 0.152462
```

The forecast file requires Fleet IDs not names. In the next step these are extracted and fleet names are converted in to Fleet IDs

```
fleet = data.frame(Fleet = ss3rep$FleetNames, ID = ss3rep$fleet_ID)
fexp$Fleet = fleet[match(fexp$Fleet, fleet$Fleet), 2]
```

Then, the relative proportions of F_{apic} by fleet and season can be computed

```
Fap = aggregate(Fapic ~ Seas + Fleet, fexp, mean)
Fap$prop = Fap$Fapic/sum(Fap$Fapic) * nseas
Fap
     Seas Fleet     Fapic prop
   1    1     1 0.1934263    1
```

In the next step, status quo F_{sq} for forecasting over the intermediate year(s) is defined. This can be relatively easily changed to intermediate catch years. Here, the F_{sq} is taken as F_{2022} to account for the systematically decreasing trend, and the intermediate years are set to 1, account for 1 data lag year.

```
# F status q
nfsq = 1
nint = 1
```

Compute the F_{sq} as F_{apic} vector by season and fleet

```
fsq = ss3rep$exploitation[ss3rep$exploitation$Yr %in% ((endyr -
    nfsq + 1):endyr), ]
fsq = cbind(fsq[, 1:2], fsq[, -c(1:5)])[, -3]  #><  single fleet trick
fsq = reshape2::melt(fsq, id.vars = c("Yr", "Seas"), variable.name = "Fleet",
    value.name = "Fapic")
Fsq = aggregate(Fapic ~ Seas + Fleet, fsq, mean)
```

Now, the forecast horizon can be defined in the loaded `starter.ss` object `fc`. Summary statistics on catch opportunities require a three year forecast horizon:

(1) Intermediate year based on F_{sq} or TAC
(2) Implementation year with catch and F outcomes
(3) One-step-ahead forecast of the SSB response and recruitment, when spawning is set to 1st of January (default)

```
fc$Nforecastyrs = 3
nfyrs = fc$Nforecastyrs
fyrs = endyr + c(1:nfyrs)
```

The F-vector that is passed on the forecast file comprises the season/fleet structure replicates for `ninit` for F_{sq} and the forecast years under F_{tgt} that is scaled to F_{apic} by the `Fratio` and portioned by fleets.

```
fvec = c(rep(Fsq$Fapic, nint), rep(Fmsy * Fratio * Fap$prop,
    nfyrs - nint))
```

Given the fleet, season, intermediate year and forecast years structures, the forecast table for the `forecast.ss` file can finally be constructed.

```
fc$ForeCatch = data.frame(Year = rep(fyrs, each = nseas * nfleets),
    Seas = 1:nseas, Fleet = rep(fleets, each = nseas), catch_or_F = fvec,
    Basis = 99)
tail(fc$ForeCatch, 9)
     Year Seas Fleet catch_or_F Basis
   1 2023    1     1  0.1524620    99
   2 2024    1     1  0.2166064    99
   3 2025    1     1  0.2166064    99
```

Note that the `Basis` 99 specifies that Fs are inputted, including F_{sq} for the intermediate year.

However, it also possible to input the TAC for the intermediate year. In cases of multi-fleet models it is advice to apportion the TAC for each fleet based on the most recent catch proportions by fleet (e.g. last years). The catch proportions can be computed from the information in

```
ss3rep$catch[ss3rep$catch$Yr %in% tail(years, 4), ]
        Fleet Fleet_Name Area   Yr Seas   Time   Obs   Exp Mult Exp*Mult  se
   1550     1      Fleet    1 2020    1 2020.5 19388 19388    1    19388 0.1
   1551     1      Fleet    1 2021    1 2021.5 20392 20392    1    20392 0.1
   1552     1      Fleet    1 2022    1 2022.5 18239 18239    1    18239 0.1
               F        Like vuln_bio sel_bio dead_bio ret_bio vuln_num sel_num
   1550 0.194284 6.14534e-17   100340 19685.1  19685.1   19388  39111.5 7670.69
   1551 0.191422 5.37600e-17   106630 20607.6  20607.6   20392  37980.4 7338.35
   1552 0.152462 4.46790e-17   120144 18480.5  18480.5   18239  46744.7 7190.08
        dead_num ret_num
   1550  7670.69 7116.45
   1551  7338.35 6949.55
   1552  7190.08 6670.45
```

In this case there is only one fleet so the TAC can be assigned directly to 2024 by declaring the `Basis` as 2 for catch

First, the relative proportions of $Catches$ by fleet and season is computed

```
Cexp = ss3rep$catch[ss3rep$catch$Yr %in% bmyrs, ]
```

```
Cap = aggregate(Exp ~ Seas + Fleet, Cexp, mean)
Cap$prop = Cap$Exp/sum(Cap$Exp) * nseas
Cap
     Seas Fleet   Exp prop
   1    1     1 18242    1
```

```
fc$ForeCatch[fc$ForeCatch$Year == 2023, "catch_or_F"] = TAC *
    Cap$prop
fc$ForeCatch[fc$ForeCatch$Year == 2023, "Basis"] = 2
fc$ForeCatch
     Year Seas Fleet   catch_or_F Basis
   1 2023    1     1 1.129300e+04     2
   2 2024    1     1 2.166064e-01    99
   3 2025    1     1 2.166064e-01    99
```

Finally, the forecast options need to be adjusted for manual input

```
fc$eof = TRUE
fc$InputBasis = -1
```

and then the modified `starter.ss` file can be saved

```
SS_writeforecast(fc, file = file.path(fmsydir, "forecast.ss"),
    overwrite = T, verbose = F)
```

2.4 Step 4: Running Fmsy forecasts with checks

In principle, the `Ftgt` can serve as a reference and the model does not have to be run if the goal is set up a number forecasts relative to F_{tgt}.

However, for illustration, the `Ftgt` forecast is run to check that the F_{apic} will produce F_{bar} estimates that are consistent with F_{tgt}.

To run

```
r4ss::run(fmsydir, skipfinished = F, show_in_console = T, exe = ss.exe)
```

After the run is finished (here under 2.5 min) the output can be loaded again.

```
fmsyrep = SS_output(fmsydir)
# safe as rdata
save(fmsyrep, file = file.path(rdata, "fwd_fmsy.rdata"))
```

For a quick check the `plotAdvice()` from `FLRef` can be used, but first the forecast needs to be converted into a "simplified" `FLStock` object, using the function `ssmvln`.

Switch `addprj=TRUE` on to add the forecast horizon.

```
mvn = FLRef::ssmvln(fmsyrep, Fref = "Btgt", addprj = T, verbose = F)
stkf = ss2FLStockR(mvn)
```

Again, assign benchmarks

```
stkf@refpts = stk@refpts
```

It can be readily seen that the F_{apic} based F_{MSY} forecast corresponds indeed to the F_{MSY} estimate on F_{bar} scale.

```
plotAdvice(window(stkf, start = 1991), osa = T) + geom_vline(xintercept = 2022.5,
    linetype = 2)
```

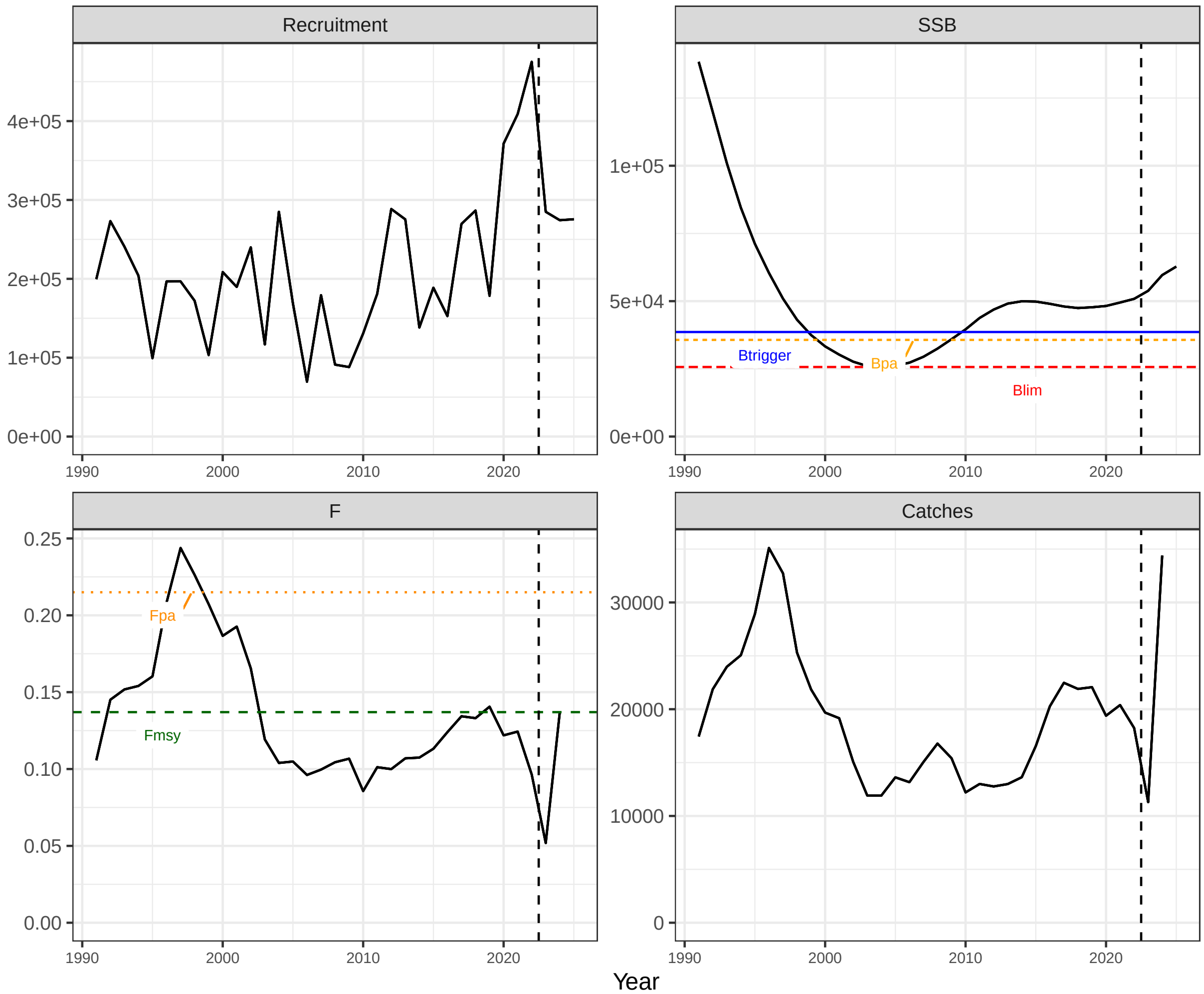

Figure 4: Stock trajectories for basecase run and a F_{tgt} forecast, relative to reference points

2.5 Step 5: Looping through fixed forecast scenarios

Set up the F-based forecasts for $F = 0$, $F_{pa} = F_{p0.5}$, $FMSY_{low}$, $FMSY_{high}$

```
Fs = c("Fsq", "F0", "Fpa", "Flower", "Fupper")
```

Add Fadv according ICES MSY approach of applying the advice rule. First extract the SSB for the intermediate year + 1.

```
fwdyr = max(years + 1)
fwdyr
   [1] 2024
b = an(ssb(stkf)[, ac(fwdyr)])
b
   [1] 59629.6
```

Apply Advice rule

```
Fadv = applyAR(b, btrigger = an(bms["Btrigger"]), fmsy = an(bms["Fmsy"]))
Fadv
   [1] 0.137
```

Here, $F_{adv} = F_{MSY}$ given the SSB is above the MSY $B_{trigger}$

Add Fadv scenrario to runs

```
Fs = c(Fs, "Fadv")
```

Specify F scenario values for forecasts

```
frefs = rbind(bms, FLPar(F0 = 1e-04, Fadv = Fadv))[Fs[Fs != "Fsq"]]

frefs
   An object of class "FLPar"
   params
       F0    Fpa Flower Fupper   Fadv
   0.0001 0.2150 0.1050 0.1740 0.1370
   units:  NA
```

Specify forecast folders

```
fdirs = file.path(forecast.dir, paste0(Fs))
```

Loop through the process of modifying the `forecast.ss` file iteratively.

```
for (i in 1:length(Fs)) {
    # create model folder by copying from the fmsydir with
    # forecast setup
    SSnewrun(model = fmsydir, dat = dat, ctl = ctl, newdir = fdirs[i],
        ss.exe = ss.exe)
```

```
    # Read Forecast file
    fc <- SS_readforecast(file.path(fdirs[i], "forecast.ss"))
    # Apply Ffrac Create F forecast vector (generic) F
    # target
    if (Fs[i] != "Fsq") {
        ftgt = an(frefs[Fs[i]])
        # scale to Fapic and apportion to fleets
        fvec = c(rep(ftgt * Fratio * Fap$prop, nfyrs - nint))
    } else {
        # Use Fsq
        fvec = c(rep(Fsq$Fapic, nfyrs - nint))
    }
    # Creat F forecast table in forecast.ss
    fc$ForeCatch[fc$ForeCatch$year %in% fc$ForeCatch$year[-c(1:nint)],
        "catch_or_F"] = fvec
    SS_writeforecast(fc, file = file.path(fdirs[i], "forecast.ss"),
        overwrite = T)
    r4ss::run(fdirs[i], skipfinished = F, show_in_console = TRUE,
        exe = ss.exe)
}
```

Load all runs in one go with `SSgetoutput`, including the Fmsy run

```
Ffwd = SSgetoutput(dirvec = c(fdirs, fmsydir))
save(Ffwd, file = file.path(rdata, "fwdFs.rdata"))
```

Check that these can be loaded

```
load(file = file.path(rdata, "fwdFs.rdata"), verbose = T)
   Loading objects:
     Ffwd
```

Quick check and plot `FLRef`

```
fstks = FLStocks(Map(function(x, y) {
    out = FLRef::ssmvln(x, Fref = "Btgt", verbose = F, run = y,
        addprj = T)
    out = ss2FLStockR(out)
    out@refpts = stk@refpts  # replace with benchmarks
    return(out)
}, x = Ffwd, y = as.list(c(paste0(Fs), "Fmsy"))))
names(fstks) = c(paste0(Fs), "Fmsy")
```

Make final pretty forecast plot for F scenarios

```
pstks = FLStocks(c(FLStocks(Assessment = window(stk, end = ss3rep$endyr)),
    window(fstks, start = ss3rep$endyr)))

# Name plot pffwd (pffwd for F-based and pcfwd for
# catch-based)
pffwd = plotAdvice(window(pstks, start = ss3rep$endyr - 10),
```

```
    osa = T) + geom_vline(xintercept = c(ss3rep$endyr:(ss3rep$endyr +
    2)), linetype = 2, linewidth = 0.3) + scale_color_manual(values = c("black",
    ss3col(length(fstks)))) + scale_x_continuous(breaks = seq(1900,
    2200, 2)) + theme(axis.text.x = element_text(size = 8, angle = 90,
    vjust = 0.5))

pffwd
```

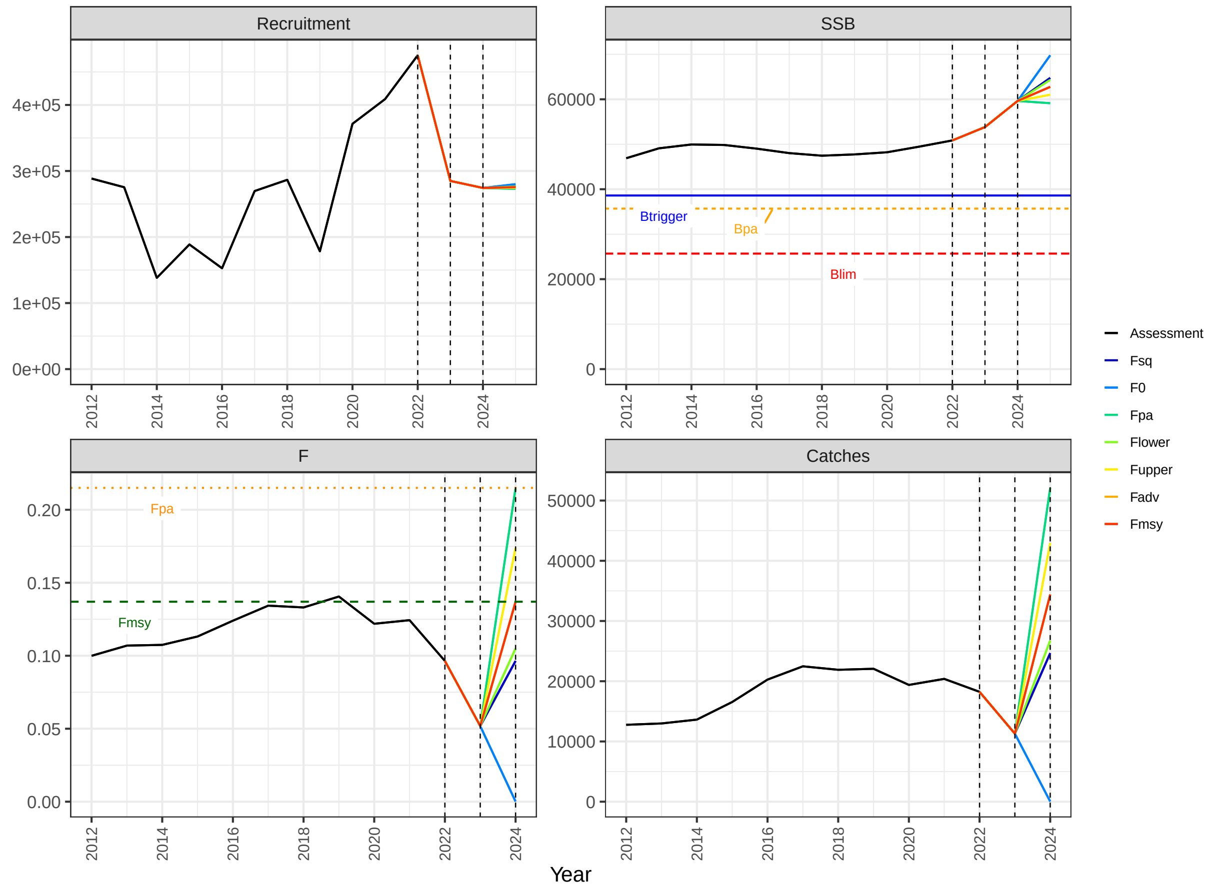

Figure 5: Trajectories for the forecast scenarios under different fishing mortalities. The vertical dashed lines denote from left to right: reference year, implementation year for short-term forecasts and the catch advice year for the short-term forecasts

Add uncertainties

```
fstksi = FLStocks(Map(function(x, y) {
    out = FLRef::ssmvln(x, Fref = "Btgt", verbose = F, run = y,
        addprj = T)
    out = ss2FLStockR(out, output = "iters")
    out@refpts = stk@refpts  # replace with benchmarks
    return(out)
}, x = Ffwd, y = as.list(c(paste0(Fs), "Fmsy"))))
names(fstksi) = paste0(c(paste0(Fs), "Fmsy"), ".CI")
```

```r
pstksi = FLStocks(c(FLStocks(Assessment = window(stki, end = ss3rep$endyr)),
    window(fstksi, start = ss3rep$endyr)))

pffwdi = plotAdvice(window(pstksi, start = ss3rep$endyr - 10),
    osa = T) + geom_vline(xintercept = c(ss3rep$endyr:(ss3rep$endyr +
    2)), linetype = 2, linewidth = 0.3) + scale_color_manual(values = c("black",
    ss3col(length(fstks)))) + scale_fill_manual(values = c("darkgrey",
    ss3col(length(fstks)))) + scale_x_continuous(breaks = seq(1900,
    2200, 2)) + theme(axis.text.x = element_text(size = 8, angle = 90,
    vjust = 0.5))

pffwdi
```

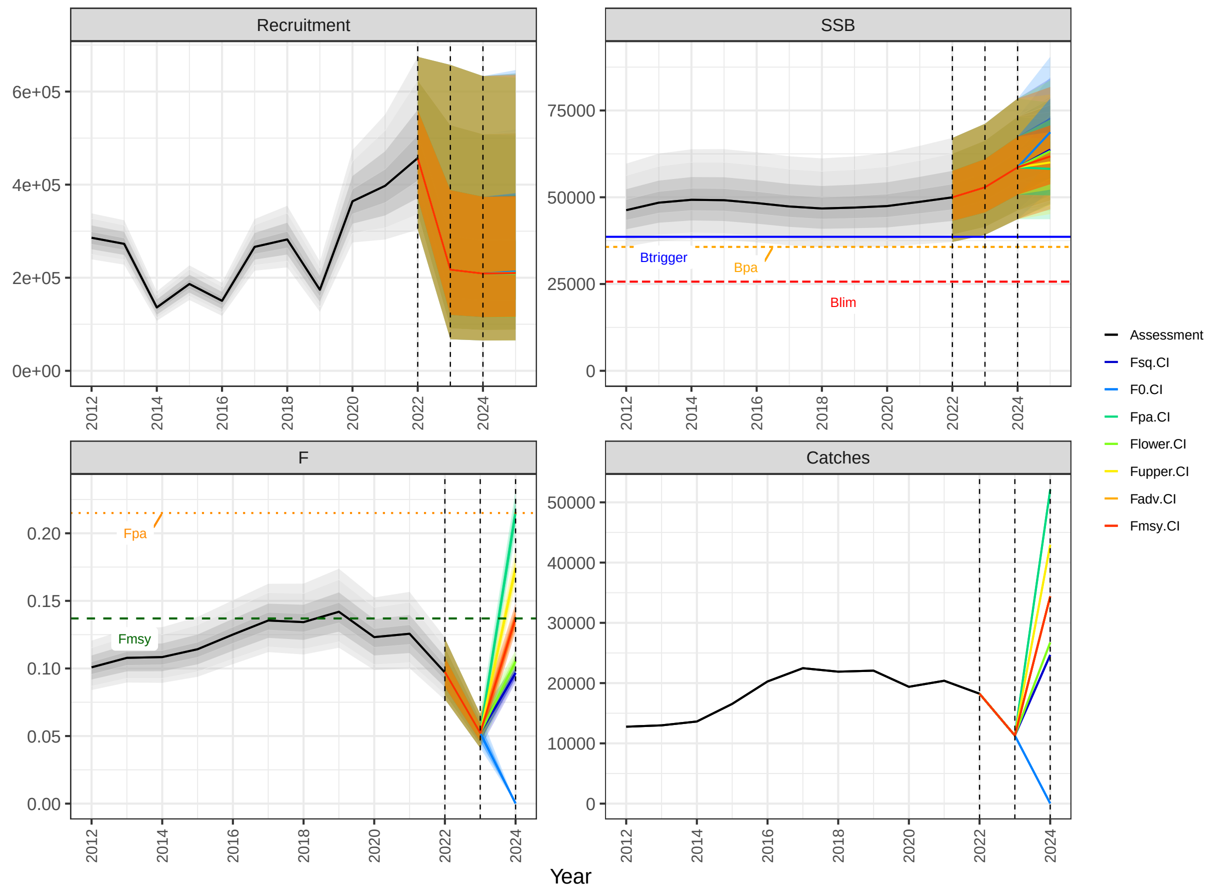

Figure 6: Trajectories of medians and 90% CIs for the forecast scenarios under different fishing mortalities. The vertical dashed lines denote from left to right: reference year, implementation year for short-term forecasts and the catch advice year for the short-term forecasts

2.6 Step 6: Search for limit MSY Btrigger and Blim

Here, an approach is introduced to use `ss3om` to translate SS3 to a full `FLR` stock object to then apply `FLasher` for solving for the $F's$ that lead to $MSYB_{trigger}$ and B_{lim}

To do this the SS3 model is loaded to `FLR` with `ss3om`

```
run = "ref"
ss3stk = window(ss3om::readFLSss3(run, wtatage = TRUE))
sr = ss3om::readFLSRss3(run)
stk@name = "anf.27.3a4"
stk@desc = "2022,  ICES, SS3"
```

Convert to `FLStockR` and assign refpts

```
ss3stk = FLStockR(ss3stk)
ss3stk@refpts = stk@refpts
```

Compare with ss3

```
plotAdvice(FLStocks(SS3 = window(stk, end = ss3rep$endyr), ss3om = ss3stk))
```

Find F for $MSYB_{trigger}$ in 2025 using the new `FLRef` function `fwdF4B()`

```
fwdBtri = fwdF4B(ss3stk, sr = sr, btgt = bms["Btrigger"], niy = 1,
    nfy = 3, ival = TAC, imet = "TAC", verbose = F)

F.Btri = an(tail(fbar(fwdBtri), 1))
F.Btri
   [1] 0.7966919
```

Find F for B_{lim}

```
fwdBlim = fwdF4B(ss3stk, sr = sr, btgt = bms["Blim"], niy = 1,
    nfy = 3, ival = TAC, imet = "TAC", verbose = F)

F.Blim = an(tail(fbar(fwdBlim), 1))
F.Blim
   [1] 1.394897
```

Check

```
plotAdvice(FLStocks(`F(Blim)` = window(fwdBlim, start = 2000),
    `F(Btri)` = window(fwdBtri, start = 2000)))
```

Run the F.Btri and F.Blim forecasts in SS3

```
Flims = c("F.Btri", "F.Blim")
frefs = rbind(frefs, FLPar(F.Btri = F.Btri, F.Blim = F.Blim))
```

Specify forecast folders

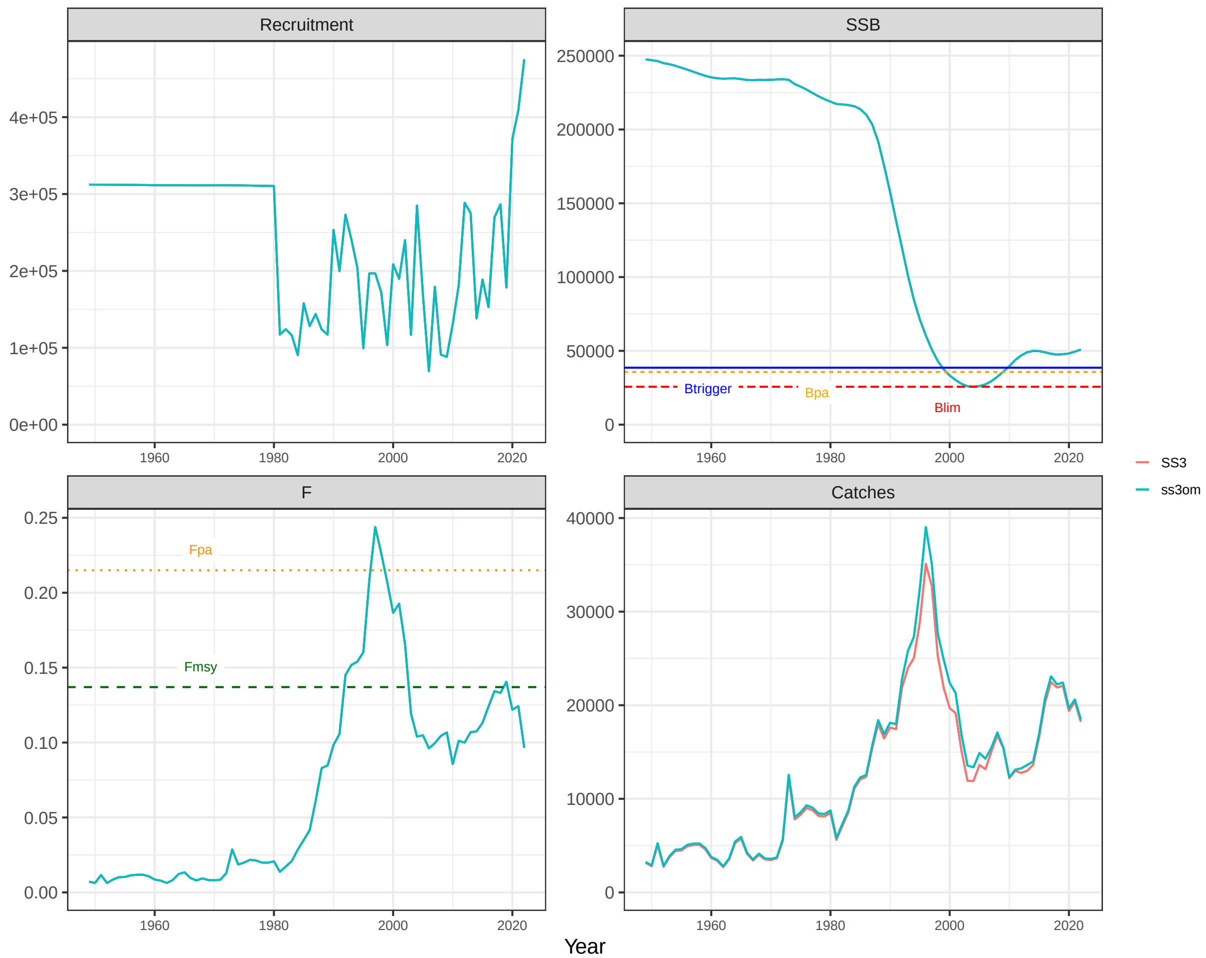

Figure 7: Comparison of estimated stock status trajectories from SS3 and ss3om

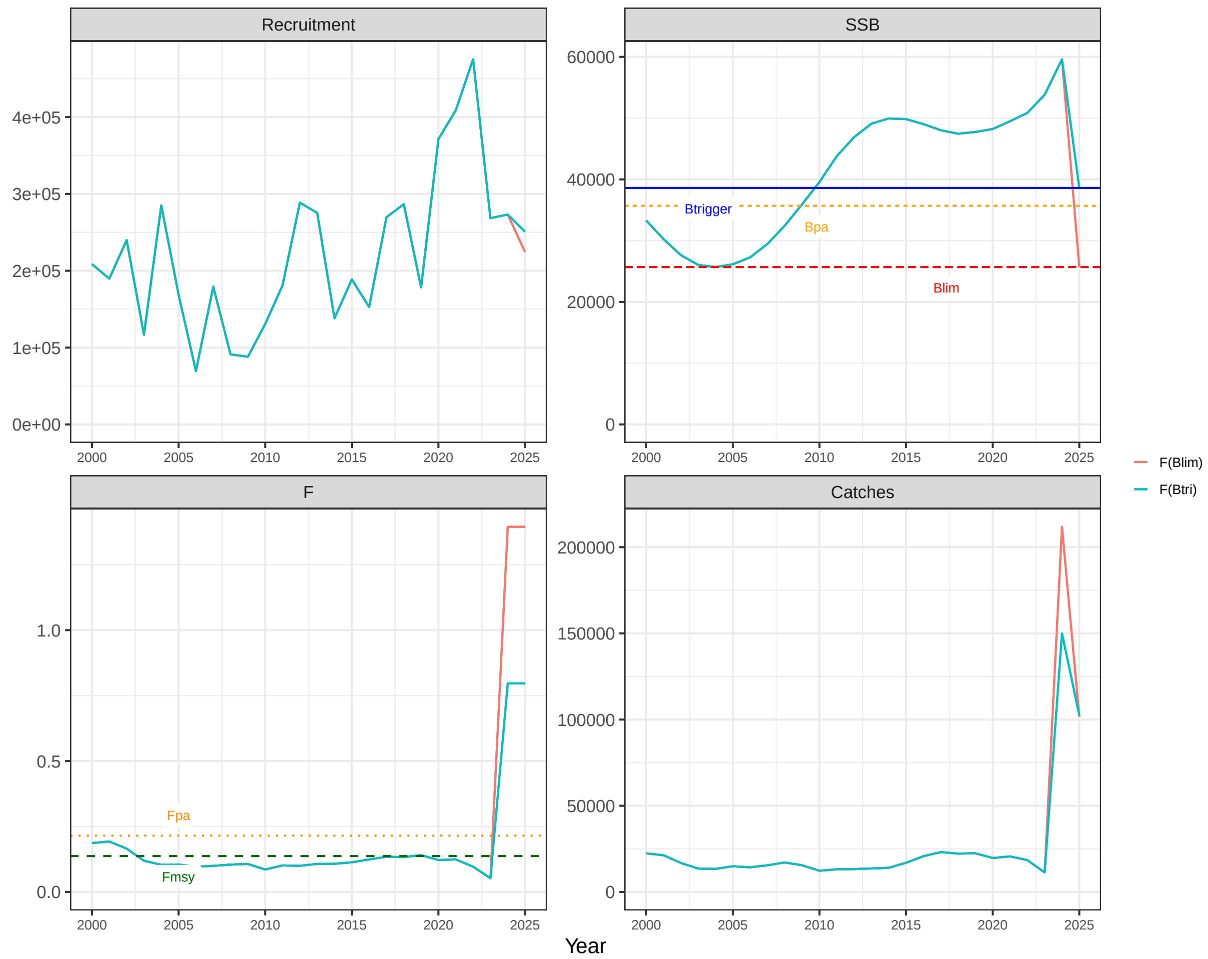

Figure 8: Forecast of F leading to Blim and Btrigger in 2025

```
fldirs = file.path(forecast.dir, paste0(paste0(Flims)))
```

Loop through the process of modifying the `forecast.ss` file iteratively.

```
for (i in 1:length(Flims)) {
    # create model folder by copying from the fmsydir with
    # forecast setup
    SSnewrun(model = fmsydir, dat = dat, ctl = ctl, newdir = fldirs[i],
        ss.exe = ss.exe)
    # Read Forecast file
    fc <- SS_readforecast(file.path(fldirs[i], "forecast.ss"))
    # Apply Ffrac Create F forecast vector (generic) F
    # target from fref FLPar
    ftgt = an(frefs[Flims[i]])

    # scale to Fapic and apportion to fleets
    fvec = c(rep(ftgt * Fratio * Fap$prop, nfyrs - nint))

    # Creat F forecast table in forecast.ss
    fc$ForeCatch[fc$ForeCatch$year %in% fc$ForeCatch$year[-c(1:nint)],
        "catch_or_F"] = fvec
    SS_writeforecast(fc, file = file.path(fldirs[i], "forecast.ss"),
        overwrite = T)
    r4ss::run(fldirs[i], skipfinished = F, show_in_console = TRUE,
        exe = ss.exe)
}
```

2.7 Step 7: Add fixed TAC forecast

```
fldirs = file.path(forecast.dir, paste0(paste0(Flims)))
```

Loop through the process of modifying the `forecast.ss` file iteratively.

```
tacdir = file.path(forecast.dir, "TACsq")

# create model folder by copying from the fmsydir with
# forecast setup
SSnewrun(model = fmsydir, dat = dat, ctl = ctl, newdir = tacdir,
    ss.exe = ss.exe)
# Read Forecast file
fc <- SS_readforecast(file.path(tacdir, "forecast.ss"))
# Creat Catch forecast table in forecast.ss
fc$ForeCatch[, "catch_or_F"] = rep(TAC * Cap$prop, nfyrs)
fc$ForeCatch[, "basis"] = rep(2, nrow(fc$ForeCatch))
SS_writeforecast(fc, file = file.path(tacdir, "forecast.ss"),
    overwrite = T)
r4ss::run(tacdir, skipfinished = F, show_in_console = TRUE, exe = ss.exe)
```

2.8 Step 8 : Summarize ICES Fishing Opportunities

Load all runs in one go with `SSgetoutput`, including the Fmsy run

```
Ffwd2 = SSgetoutput(dirvec = c(fdirs, fmsydir, fldirs, tacdir))
save(Ffwd2, file = file.path(rdata, "fwdFs2.rdata"))
```

Check that these can be loaded

```
load(file = file.path(rdata, "fwdFs2.rdata"), verbose = T)
   Loading objects:
     Ffwd2
```

Convert to `FLR`

```
fstks = FLStocks(Map(function(x, y) {
    out = FLRef::ssmvln(x, Fref = "Btgt", verbose = F, run = y,
        addprj = T)
    out = ss2FLStockR(out)
    out@refpts = stk@refpts  # replace with benchmarks
    return(out)
}, x = Ffwd2, y = as.list(c(Fs, "Fmsy", Flims, "TACsq"))))
names(fstks) = c(paste0(Fs), "Fmsy", Flims, "TACsq")
```

Rearrange

```
icesorder = c("Fadv", "Flower", "Fupper", "F0", "Fpa", "Fmsy",
    "F.Blim", "F.Btri", "Fsq", "TACsq")

fstks = fstks[icesorder]
```

Make final pretty forecast plot for F scenarios

```
pstks = FLStocks(c(FLStocks(Assessment = window(stk, end = ss3rep$endyr)),
    window(fstks, start = ss3rep$endyr)))

# Name plot pffwd (pffwd for F-based and pcfwd for
# catch-based)
pffwd = plotAdvice(window(pstks, start = ss3rep$endyr - 10),
    osa = T) + geom_vline(xintercept = c(ss3rep$endyr:(ss3rep$endyr +
    2)), linetype = 2, linewidth = 0.3) + scale_color_manual(values = c("black",
    ss3col(length(fstks)))) + scale_x_continuous(breaks = seq(1900,
    2200, 2)) + theme(axis.text.x = element_text(size = 8, angle = 90,
    vjust = 0.5))

pffwd
```

Create FLStocks with uncertainties

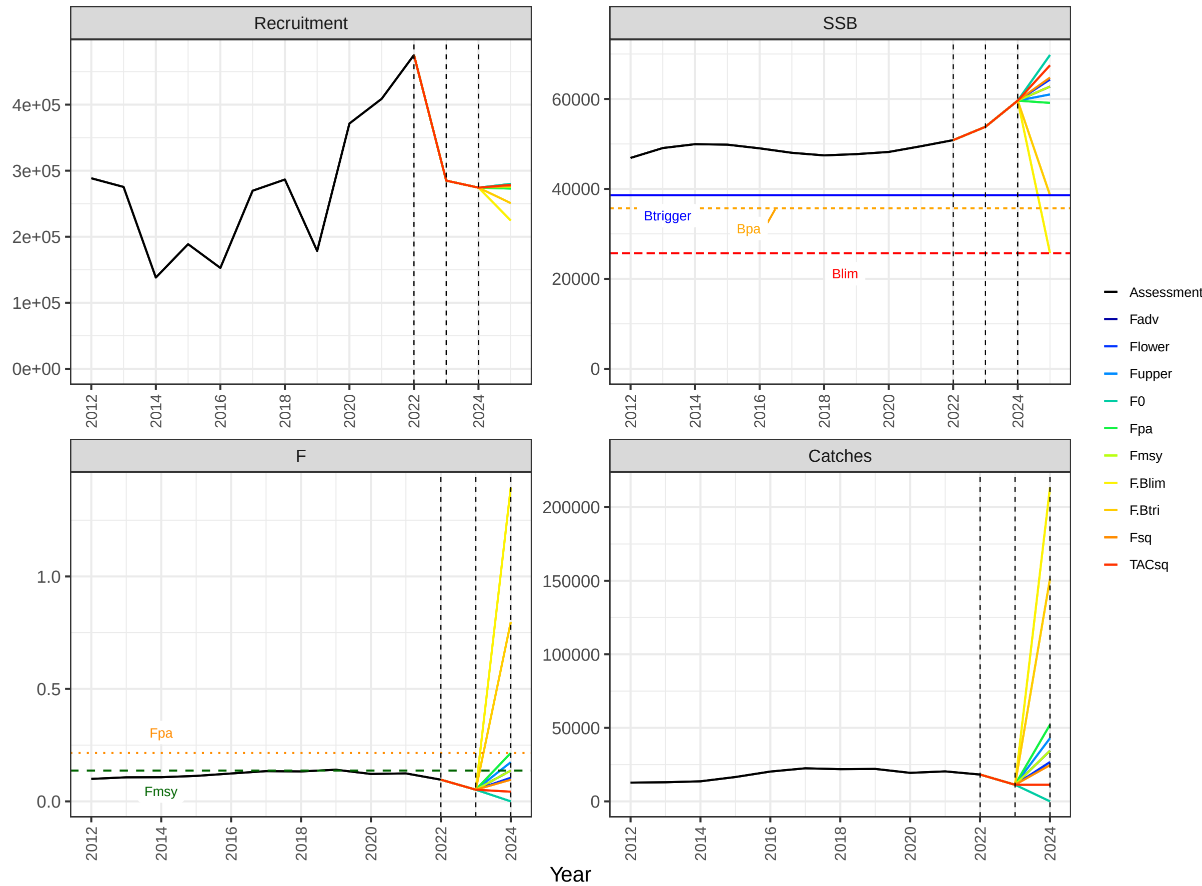

Figure 9: Trajectories for the forecast scenarios under different fishing mortalities. The vertical dashed lines denote from left to right: reference year, implementation year for short-term forecasts and the catch advice year for the short-term forecasts

```r
fstksi = FLStocks(Map(function(x, y) {
    out = FLRef::ssmvln(x, Fref = "Btgt", verbose = F, run = y,
        addprj = T)
    out = ss2FLStockR(out, output = "iters")
    out@refpts = stk@refpts  # replace with benchmarks
    return(out)
}, x = Ffwd2, y = as.list(c(Fs, "Fmsy", Flims, "TACsq"))))
names(fstksi) = c(paste0(Fs), "Fmsy", Flims, "TACsq")
fstksi = fstksi[icesorder]
```

Make ICES Fishing Opportunity Table input

```r
fish.ops = fwd2ices(stock = fstks, uncertainty = fstksi)
```

```r
knitr::kable(fish.ops, "pipe", align = "lccccc", caption = " Summary of Fishing Opportunities.")
```

Table 2: Summary of Fishing Opportunities.

Basis	C_2024	F_2024	SSB_2025	SSB_change	P(SSB<Blim)
Fadv	34395.4	0.137	62760.5	5.250580	0.0
Flower	26752.0	0.105	64324.3	7.873103	0.0
Fupper	42954.2	0.174	61007.6	2.310933	0.0
F0	26.8	0.000	69779.8	17.022083	0.0
Fpa	52102.3	0.215	59131.7	-0.834988	0.0
Fmsy	34395.4	0.137	62760.5	5.250580	0.0
F.Blim	213336.0	1.395	25715.8	-56.874103	53.6
F.Btri	151339.0	0.797	38631.1	-35.214893	2.1
Fsq	24665.9	0.096	64750.8	8.588352	0.0
TACsq	11293.0	0.043	67482.3	13.169131	0.0

The output can also be saved as `.csv` files

```r
write.csv(fish.ops, file = file.path(output.dir, paste0(stk2@name,
    "_FishOps.csv")), row.names = F)
```