BGS: A trawling simulation

Emil De Borger

8-2-2021

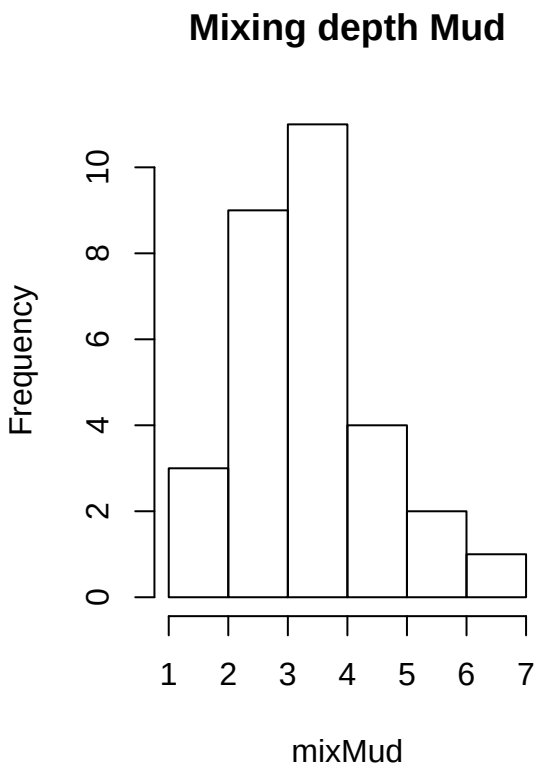
Introduction

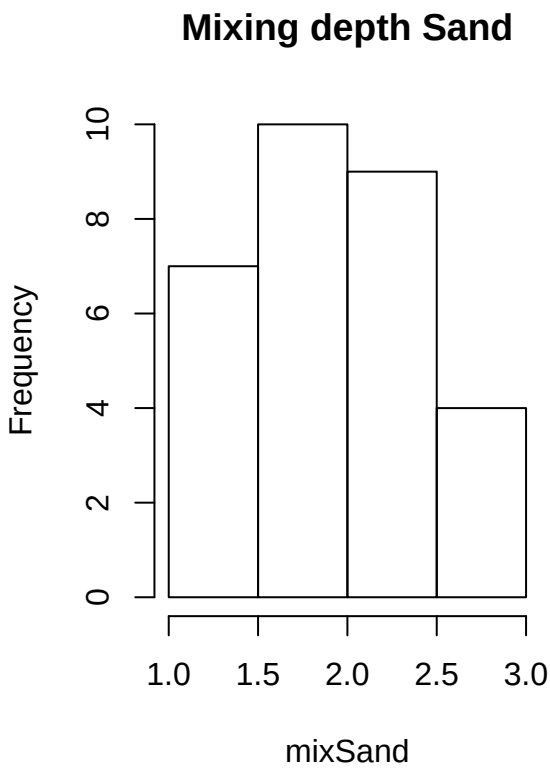
- CHECK IF TRUNCATED NORMAL DISTRIBUTION

Setting trawling depth distribution

- Simulating two different penetration depth distributions, as summarized by Putcher et al. (pers.comm).

```
set.seed(8022021)

# Log normal distribution.
##  mud
BT.mud  <- c(3.2,   1.2,    1.5,    6.7) # pd.mean, pd.sdev, pd.lower, pd.upper
pd      <- (log(6.7)+log(1.5))/2
sd      <- (log(6.7)-log(1.5))/(2*1.96)
mud     <- c(pd,sd,log(3.2))

### sand
BT.sand <- c(1.9,   0.6,    1.0,    3.7) # pd.mean, pd.sdev, pd.lower, pd.upper
pd      <- (log(3.7)+log(1.0))/2
sd      <- (log(3.7)-log(1.0))/(2*1.96)
sand    <- c(pd,sd,log(1.9))

### combine
pds <- rbind(mud,sand)
colnames(pds) <- c("mu","sd","mu2")

# Mixing depths --> generate 30 mixing depths based on given mu and sd.
mixMud   <- exp(rnorm(30, mean=pds[1,3]          , sd=pds[1,2]))
mixSand  <- exp(rnorm(30, mean=pds[2,3]          , sd=pds[2,2]))

# Erosion depth (set to 22 % of mixing depth)
erodMud  <- mixMud*0.22
erodSand <- mixSand*0.22

# Visual
par(mfrow = c(1,2))
hist(mixMud , main = "Mixing depth Mud")
hist(mixSand, main = "Mixing depth Sand")
```





Sediment parameters needed for CNPDIA

- CNPDIA has default parameters to run a simulation. Parameters may be changed to fit biogeochemical observations.
- Base parameter values and explanations can be retrieved with CNPDIAparms().

```
# Set custom parameter values to overwrite base values seen in CNPDIAparms().
sedparms <- list(Cflux = 3000, rFast = 0.05, rSlow = 0.00019,
                 biot = 0.005, por0 = 0.8, pordeep = 0.4)
```

Running simulation

By default

Additional parameters that can be specified are:

- The frequency of disturbance. By default, frequencies of 0 to 5 trawling events /year are simulated.
- The amount of years after which output needs to be shown (yrs). The default is 15
- For the species depletion / recovery : the mudcontent of the sediment, and the longevity of the species. Defaults are 10 % mud and longevity of 2.5 y.
- Daily bottom water temperatures. CHECK DEFAULT
- Daily water nutrient concentrations (oxygen, nitrate, ammonium, phosphate) when available. CHECK DEFAULT

- This can be done for carbon as well, otherwise the mean carbon flux is given (Cflux), which is then extended throughout the year according to the function specified in the "carb" argument.

So,

Loading nutrients and temperature from a fieldsite (optional)

```
load("./nutrients/tempcmems.rda")
load("./nutrients/phosphatecmems.rda")
load("./nutrients/oxygencmems.rda")
load("./nutrients/nitratecmems.rda")
load("./nutrients/ammoniumcmems.rda")
```

Running model

```
# Set frequency of disturbances to test ( /year), from 0 to 6 in this example.
frequency <- seq(0, 5, by = 1)

sandTrawl <- perturbRange(mixdist      = mixSand,
                          erosdist     = erodSand,
                          freq         = frequency,
                          yrs          = 10,
                          mudContent   = 0,
                          longevity    = 2.5,
                          sedimentpars = sedparms,
                          ox           = oxy,
                          nit          = nit,
                          amm          = amm,
                          pho          = pho,
                          temp         = tem,
                          carb         = list(data = NULL, amp = 1,
                                              period = 365, phase = 0,
                                              pow = 1, min = 0.3))

mudTrawl <- perturbRange(mixdist      = mixMud,
                         erosdist     = erodMud,
                         freq         = frequency,
                         yrs          = 10,
                         mudContent   = 0,
                         longevity    = 2.5,
                         sedimentpars = sedparms,
                         ox           = oxy,
                         nit          = nit,
                         amm          = amm,
                         pho          = pho,
                         temp         = tem,
                         carb         = list(data = NULL, amp = 1,
                                             period = 365, phase = 0,
                                             pow = 1, min = 0.3))
```

```
# Save your outpu
save(sandTrawl, file = "./rda/sandTrawl.rda")
```

```
save(mudTrawl , file = "./rda/mudTrawl.rda")
```

```
# Load your outpu
load(file = "./rda/sandTrawl.rda")
load(file = "./rda/mudTrawl.rda")
```

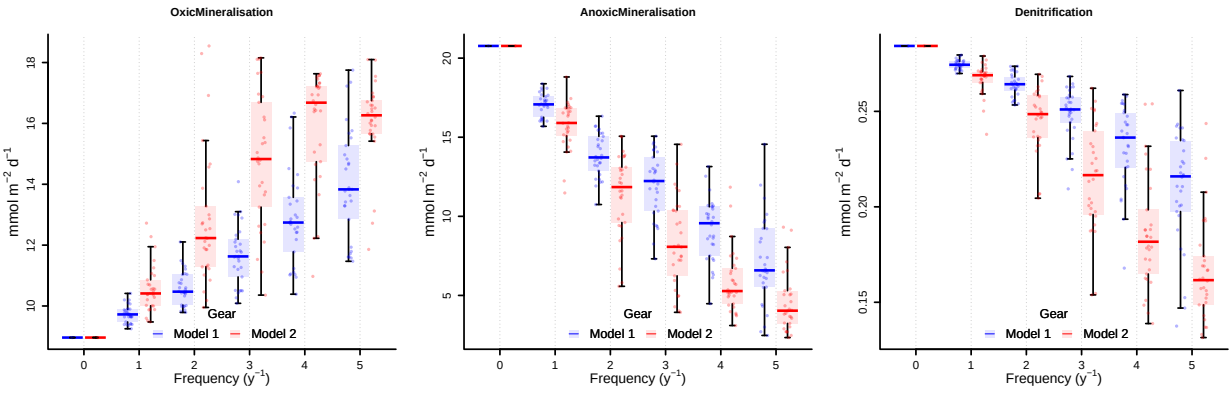
Plotting some output

The output of `perturbRange()` is a dataframe (str(outputname)) of which the named output can be manipulated with standard plotting procedures, such as `boxplot()` or `plot()`.

Boxplot

Using a wrapper around the boxplot function.

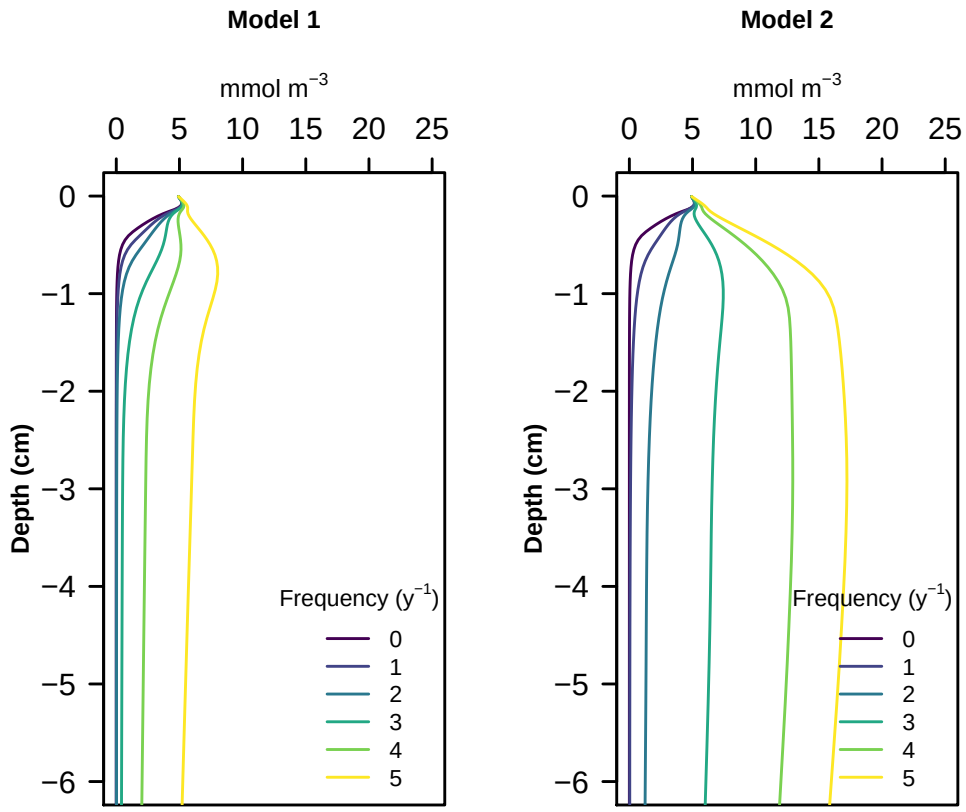
```
par(mfrow = c(1,3))
boxCompare(model1     = sandTrawl,
           model2     = mudTrawl,
           variable   = "OxicMineralisation",
           plotpoints = 1,
           ylabz      = expression(paste("mmol ", "m"^"-2", " d"^"-1")),
           leg        = TRUE)
boxCompare(model1     = sandTrawl,
           model2     = mudTrawl,
           variable   = "AnoxicMineralisation",
           plotpoints = 1,
           ylabz      = expression(paste("mmol ", "m"^"-2", " d"^"-1")),
           leg        = TRUE)
boxCompare(model1     = sandTrawl,
           model2     = mudTrawl,
           variable   = "Denitrification",
           plotpoints = 1,
           ylabz      = expression(paste("mmol ", "m"^"-2", " d"^"-1")),
           leg        = TRUE)
```



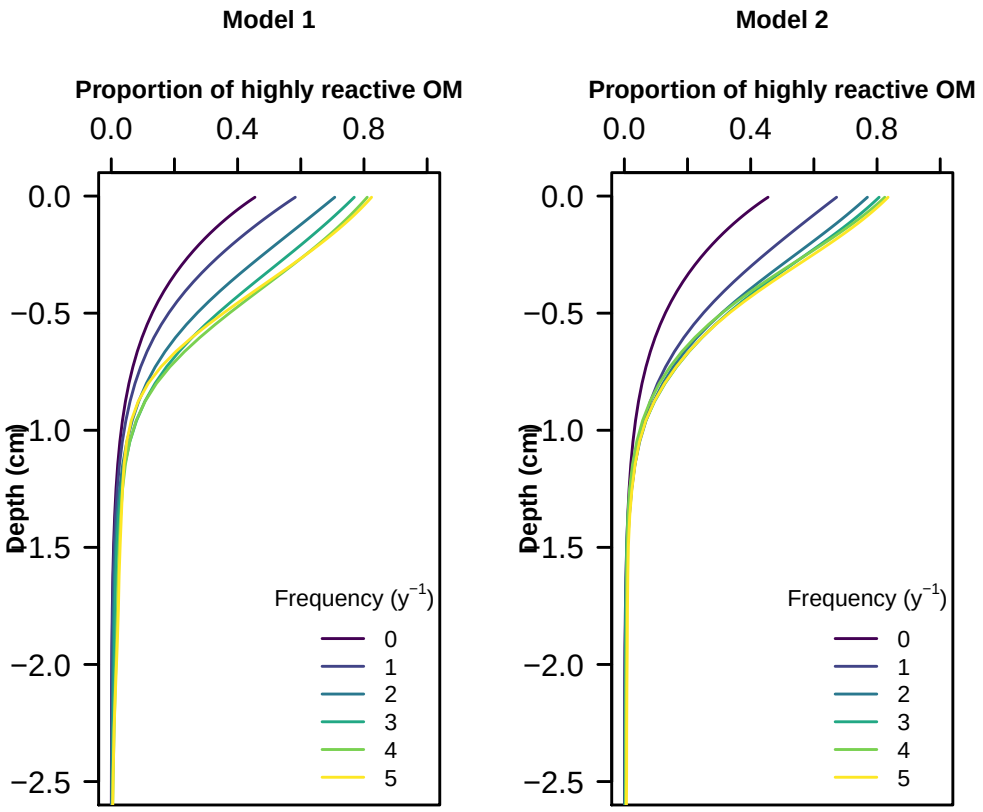
Profiles

Using `plot()`, also in a wrapper.

```
# Some required arguments
grid   <- setup.grid.1D(x.up = 0, dx.1 = 0.01, N = 100, L = 350)
depth  <- grid$x.mid
colors <- viridis(6)

# Plotting profiles
## Of Nitrate
par(mfrow = c(1,2), oma = c(4,4,4,4))
plotProfile(model     = sandTrawl,
            fraction  = "NO3",
            xlimz     = c(0, 25),
            ylimz     = c(-6, 0),
            depth,
            colors,
            plottitle = "Model 1")

plotProfile(model     = mudTrawl,
            fraction  = "NO3",
            xlimz     = c(0, 25),
            ylimz     = c(-6, 0),
            depth,
            colors,
            plottitle = "Model 2")
```




```
## Or organic matter quality
par(mfrow = c(1,2), oma = c(4,4,4,4))
plotProfile(model     = sandTrawl,
            fraction  = "QUAL",
            xlimz     = c(0, 1),
            ylimz     = c(-2.5, 0),
            depth,
            colors,
            plottitle = "Model 1")

plotProfile(model     = mudTrawl,
            fraction  = "QUAL",
            xlimz     = c(0, 1),
            ylimz     = c(-2.5, 0),
            depth,
            colors,
            plottitle = "Model 2")
```


Model 1
Proportion of highly reactive OM
0.0
0.4
0.8
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
Depth (cm)
Frequency (y^{-1})
0
1
2
3
4
5
Model 2
Proportion of highly reactive OM
0.0
0.4
0.8
0.0
−0.5
−1.0
−1.5
−2.0
−2.5
Depth (cm)
Frequency (y^{-1})
0
1
2
3
4
5