

Soils & Discharge Merged

PAZ

17 novembre 2016

Purpose

This file merges outlet data with soil data to plot cumulative exported and remaining S-metolachlor mass. The nearest soil sample date for each transect is used to match the initial time ("ti") of the sampling discharge period. This is most adequate merging location given that samples took place shortly before relaunching the automatic sampler.

Note that week numbers for water and soils are offset by one. I.e. Week 1 soils influence/regulate Week 2's water sample results.

Imports:

- **WeeklyHydroContam_R.csv**
- **WeeklySoils_Rng.csv**

Generates:

- **WeekSoilHydroCont_R.csv**
- **MassBalance_R.csv**

Required R-packages:

```
library("plyr")
library("dplyr")
```

Working directory

```
# setwd("D:/Documents/these_pablo/Alteckendorf2016/R")
# setwd("/Users/DayTightChunks/Documents/PhD/Routput/Alteck/R")
# setwd("D:/Documents/these_pablo/Alteckendorf2016/00_TransparencyFolder")
getwd()
```

```
## [1] "D:/Documents/these_pablo/Alteckendorf2016/HydrologicalMonitoring"
```

Lab parameters

```
# Initial signature measured in tank
initialDelta = d13Co = -32.253
```

Import files

```
outlet = read.csv2("Data/WeeklyHydroContam_R.csv", header = T)
outlet$ti <- as.POSIXct(outlet$ti, "%Y-%m-%d %H:%M", tz = "EST")
sum(is.na(outlet$ti))
```

```
## [1] 0
```

```
# Select variables from Water dataset
```

```
outlet <- outlet[, c("ti", "WeekSubWeek", "Event",
                    "timeSinceApp", "timeSinceApp.NoSo",
                    "timeSinceApp.N", "timeSinceApp.T", "timeSinceApp.S",
                    "diss.d13C", "SD.d13C", "CumOutDiss.g", "CumOutFilt.g",
                    "CumAppMass.g", "CumAppMass.g.N", "CumAppMass.g.T", "CumAppMass.g.S",
                    "iniCo.ug.g.N", "iniCo.ug.g.T", "iniCo.ug.g.S",
                    "CumOutMELsm.g")]

print("Water")
```

```
## [1] "Water"
```

```
names(outlet)
```

```
## [1] "ti"                "WeekSubWeek"      "Event"
## [4] "timeSinceApp"      "timeSinceApp.NoSo" "timeSinceApp.N"
## [7] "timeSinceApp.T"    "timeSinceApp.S"   "diss.d13C"
## [10] "SD.d13C"           "CumOutDiss.g"     "CumOutFilt.g"
## [13] "CumAppMass.g"      "CumAppMass.g.N"   "CumAppMass.g.T"
## [16] "CumAppMass.g.S"    "iniCo.ug.g.N"     "iniCo.ug.g.T"
## [19] "iniCo.ug.g.S"      "CumOutMELsm.g"
```

```
soils = read.csv2("Data/WeeklySoils_Rng.csv", header = T) # Corrected with only ngC > 2.0
names(soils)
```

```
## [1] "ID"                "Transect"         "Wnum"
## [4] "Date.Soil"         "Date.ti"          "Conc.mug.g.dry.soil"
## [7] "Conc.ComSoil.SD"   "N_compsoil"       "comp.d13C"
## [10] "comp.d13C.SD"      "N_isoComp"        "prctError"
## [13] "DD13C.comp"        "comp.IMP.d13C"    "MassSoil.g"
## [16] "MassSoil.g.SD"     "Area.N"           "Area.T"
## [19] "Area.S"
```

```
soils$Date.ti <- as.POSIXct(soils$Date.ti, "%Y-%m-%d %H:%M", tz = "EST")
#soils$Date.ti <- as.POSIXct(soils$Date.ti, "%d/%m/%Y %H:%M", tz = "EST")
sum(is.na(soils$Date.ti))
```

```
## [1] 0
```

```
print("Soils")
```

```
## [1] "Soils"
```

```
str(soils)
```

```
## 'data.frame':   51 obs. of  19 variables:
## $ ID                : Factor w/ 51 levels "AW-N-0","AW-N-0x",...: 2 19 36 1 18 35 3 20 37 10 ...
## $ Transect          : Factor w/ 3 levels "N","S","T": 1 2 3 1 2 3 1 2 3 1 ...
## $ Wnum              : int  -1 -1 -1 0 0 0 1 1 1 2 ...
## $ Date.Soil         : Factor w/ 17 levels "03/05/2016 13:10",...: 13 13 13 16 16 16 3 3 3 7 ...
## $ Date.ti           : POSIXct, format: "2016-03-25 00:04:00" "2016-03-25 00:04:00" ...
## $ Conc.mug.g.dry.soil: num  0.0686 0.076 0.0438 1.0376 3.738 ...
## $ Conc.ComSoil.SD   : num  0.01029 0.0114 0.00657 0.15563 0.5607 ...
```

```
## $ N_compsoil      : int  NA NA NA NA NA NA 3 3 3 3 ...
## $ comp.d13C       : num  NA NA NA NA NA ...
## $ comp.d13C.SD    : num  NA NA NA NA NA ...
## $ N_isoComp       : int  NA NA NA NA NA NA 3 3 3 3 ...
## $ prctError       : num  NA NA NA NA NA ...
## $ DD13C.comp      : num  NA NA NA NA NA ...
## $ comp.IMP.d13C   : num  NA NA NA NA NA ...
## $ MassSoil.g       : num  94.6 100.2 19 1430.5 4928.3 ...
## $ MassSoil.g.SD    : num  14.19 15.03 2.84 214.58 739.24 ...
## $ Area.N          : num  139266 139266 139266 139266 139266 ...
## $ Area.T          : num  43713 43713 43713 43713 43713 ...
## $ Area.S          : num  133175 133175 133175 133175 133175 ...
```

Get soil concentrations for each transect for merging horizontally

```
# North
soils.N <- subset(soils, soils$Transect == "N")
soils.N <- soils.N[, c("Date.ti",
  "MassSoil.g", "MassSoil.g.SD",
  "Conc.mug.g.dry.soil",
  "comp.d13C", "comp.d13C.SD", # "comp.d13C.SE",
  "ID",
  "Area.N", "Area.T", "Area.S" #,
  # "Area_Nt", "Area_Tt", "Area_St"
)]

colnames(soils.N) <- c("ti",
  "MassSoil.g.North", "MassSoil.g.SD.North",
  "Conc.mug.g.dry.soil.N",
  "comp.d13C.North", "comp.d13C.SD.North", # "comp.d13C.SE.North",
  "ID.N",
  "Area.N", "Area.T", "Area.S" #,
  # "Area.Nt", "Area.Tt", "Area.St"
)

# Talweg
soils.T <- subset(soils, soils$Transect == "T")
soils.T <- soils.T[, c("Date.ti",
  "MassSoil.g", "MassSoil.g.SD",
  "Conc.mug.g.dry.soil",
  "comp.d13C", "comp.d13C.SD", # "comp.d13C.SE",
  "ID")]

colnames(soils.T) <- c("ti",
  "MassSoil.g.Talweg", "MassSoil.g.SD.Talweg",
  "Conc.mug.g.dry.soil.T",
  "comp.d13C.Talweg", "comp.d13C.SD.Talweg", # "comp.d13C.SE.Talweg",
  "ID.T" )

# South
soils.S <- subset(soils, soils$Transect == "S")
soils.S <- soils.S[, c("Date.ti",
```

```

      "MassSoil.g", "MassSoil.g.SD",
      "Conc.mug.g.dry.soil",
      "comp.d13C", "comp.d13C.SD", # "comp.d13C.SE",
      "ID" )]
colnames(soils.S) <- c("ti",
      "MassSoil.g.South", "MassSoil.g.SD.South",
      "Conc.mug.g.dry.soil.S",
      "comp.d13C.South", "comp.d13C.SD.South", # "comp.d13C.SE.South",
      "ID.S" )

```

Total Catchment Mass, Bulk Mass and Bulk Isotopes

Bulk isotopes are calculated based on the following:

$$\delta_{bulk} = \frac{M_{North}}{M_{tot}}\delta_{North} + \frac{M_{Talweg}}{M_{tot}}\delta_{Talweg} + \frac{M_{South}}{M_{tot}}\delta_{South}$$

Bulk mass (not currently used) is a proxy for the potential mass that can be discharged at a give time. It is calculated based on the pondered mass from each transect area and the proportion of that area in relation to the entire catchment such that:

$$M_{bulk} = \frac{A_{North}}{A_{tot}}M_{North} + \frac{A_{Talweg}}{A_{tot}}M_{Talweg} + \frac{A_{South}}{A_{tot}}M_{South}$$

$$C_{bulk} = \frac{A_{North}}{A_{tot}}C_{North} + \frac{A_{Talweg}}{A_{tot}}C_{Talweg} + \frac{A_{South}}{A_{tot}}C_{South}$$

```

library(zoo)

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##      as.Date, as.Date.numeric

class(outlet$ti)

## [1] "POSIXct" "POSIXt"

class(soils.T$ti)

## [1] "POSIXct" "POSIXt"

soilsOut <- merge(outlet, soils.N, by = "ti", all = T)
soilsOut <- merge(soilsOut, soils.T, by = "ti", all = T)
soilsOut <- merge(soilsOut, soils.S, by = "ti", all = T)

soilsOut$DD13C.North <- soilsOut$comp.d13C.North - initialDelta
soilsOut$DD13C.Talweg <- soilsOut$comp.d13C.Talweg - initialDelta
soilsOut$DD13C.South <- soilsOut$comp.d13C.South - initialDelta

# Total mass in catchment
soilsOut$CatchMassSoil.g <-

```

```

soilsOut$MassSoil.g.North +
soilsOut$MassSoil.g.Talweg +
soilsOut$MassSoil.g.South

soilsOut$CatchMassSoil.g.SD <- (
  ( soilsOut$MassSoil.g.SD.North^2 +
    soilsOut$MassSoil.g.SD.South^2 +
    soilsOut$MassSoil.g.SD.Talweg^2
  )/3
)^0.5

# Approximate initial mass for first 6 observation (before recording composite soils)
#soilsOut$CatchMassSoil.g[1:6] <- ifelse(is.na(soilsOut$CatchMassSoil.g),
#                                     soilsOut$CumAppMass.g-soilsOut$CumOutMELsm.g,
#                                     soilsOut$CatchMassSoil.g)

#soilsOut$BulkMass.g <-
# soilsOut$MassSoil.g.North*(soilsOut$Area.N/(soilsOut$Area.N+soilsOut$Area.T+soilsOut$Area.S)) +
# soilsOut$MassSoil.g.Talweg*(soilsOut$Area.T/(soilsOut$Area.N+soilsOut$Area.T+soilsOut$Area.S)) +
# soilsOut$MassSoil.g.South*(soilsOut$Area.S/(soilsOut$Area.N+soilsOut$Area.T+soilsOut$Area.S))

# Bulk catchment isotopes
soilsOut$BulkCatch.d13 <-
  (soilsOut$MassSoil.g.North/soilsOut$CatchMassSoil.g)*soilsOut$comp.d13C.North +
  (soilsOut$MassSoil.g.Talweg/soilsOut$CatchMassSoil.g)*soilsOut$comp.d13C.Talweg +
  (soilsOut$MassSoil.g.South/soilsOut$CatchMassSoil.g)*soilsOut$comp.d13C.South

soilsOut$BulkCatch.d13.SD <-
  (soilsOut$MassSoil.g.North/soilsOut$CatchMassSoil.g)*soilsOut$comp.d13C.SD.North +
  (soilsOut$MassSoil.g.Talweg/soilsOut$CatchMassSoil.g)*soilsOut$comp.d13C.SD.Talweg +
  (soilsOut$MassSoil.g.South/soilsOut$CatchMassSoil.g)*soilsOut$comp.d13C.SD.South

soilsOut$DD13.Bulk <- (soilsOut$BulkCatch.d13 - (d13Co))

# Bulk catchment concentrations
soilsOut$Area.Catchment <- soilsOut$Area.N + soilsOut$Area.T + soilsOut$Area.S

soilsOut$BulkCatch.Conc <-
  (soilsOut$Area.N/soilsOut$Area.Catchment)*soilsOut$Conc.mug.g.dry.soil.N +
  (soilsOut$Area.T/soilsOut$Area.Catchment)*soilsOut$Conc.mug.g.dry.soil.T +
  (soilsOut$Area.S/soilsOut$Area.Catchment)*soilsOut$Conc.mug.g.dry.soil.S

soilsOut$iniCo.Bulk <-
  soilsOut$iniCo.ug.g.N*(soilsOut$Area.N/soilsOut$Area.Catchment) +
  soilsOut$iniCo.ug.g.T*(soilsOut$Area.T/soilsOut$Area.Catchment) +
  soilsOut$iniCo.ug.g.S*(soilsOut$Area.S/soilsOut$Area.Catchment)

print("Merged Soils and Outlet by ti")

## [1] "Merged Soils and Outlet by ti"

str(soilsOut)

## 'data.frame': 52 obs. of 52 variables:

```

```

## $ ti : POSIXct, format: "2016-03-25 00:04:00" "2016-03-25 12:04:00" ...
## $ WeekSubWeek : Factor w/ 51 levels "W0-0x","W0-1",...: 1 2 3 4 5 6 26 27 28 29 ...
## $ Event : int 0 0 0 1 1 1 2 2 2 3 ...
## $ timeSinceApp : num 0.5 3.9 5.5 6.6 7.6 11.6 12.6 14 20.6 2.2 ...
## $ timeSinceApp.NoSo : num 0.5 3.9 5.5 6.6 7.6 11.6 12.6 14 20.6 2.2 ...
## $ timeSinceApp.N : num 0.5 3.9 5.5 6.6 7.6 11.6 12.6 14 20.6 2.2 ...
## $ timeSinceApp.T : num 0.5 3.9 5.5 6.6 7.6 11.6 12.6 14 20.6 2.2 ...
## $ timeSinceApp.S : num 0.5 3.9 5.5 6.6 7.6 11.6 12.6 14 20.6 2.2 ...
## $ diss.d13C : num NA NA NA -31.5 -31.7 ...
## $ SD.d13C : num NA NA NA 0.104 0.152 ...
## $ CumOutDiss.g : num 0.00354 0.02815 0.19818 2.84809 5.2051 ...
## $ CumOutFilt.g : num 0.00345 0.0069 0.01263 0.01571 0.01923 ...
## $ CumAppMass.g : num 31670 31670 31670 31670 31670 ...
## $ CumAppMass.g.N : num 8429 8429 8429 8429 8429 ...
## $ CumAppMass.g.T : num 6904 6904 6904 6904 6904 ...
## $ CumAppMass.g.S : num 16337 16337 16337 16337 16337 ...
## $ iniCo.ug.g.N : num 8.46 8.46 8.46 8.46 8.46 ...
## $ iniCo.ug.g.T : num 7.1 7.1 7.1 7.1 7.1 ...
## $ iniCo.ug.g.S : num 17.7 17.7 17.7 17.7 17.7 ...
## $ CumOutMELsm.g : num 0.302 2.38 4.76 35.001 62.009 ...
## $ MassSoil.g.North : num 94.6 NA NA 1430.5 NA ...
## $ MassSoil.g.SD.North : num 14.2 NA NA 214.6 NA ...
## $ Conc.mug.g.dry.soil.N : num 0.0686 NA NA 1.0376 NA ...
## $ comp.d13C.North : num NA NA NA NA NA ...
## $ comp.d13C.SD.North : num NA NA NA NA NA ...
## $ ID.N : Factor w/ 51 levels "AW-N-0","AW-N-0x",...: 2 NA NA 1 NA NA 3 NA NA 10 ...
## $ Area.N : num 139266 NA NA 139266 NA ...
## $ Area.T : num 43713 NA NA 43713 NA ...
## $ Area.S : num 133175 NA NA 133175 NA ...
## $ MassSoil.g.Talweg : num 19 NA NA 404 NA ...
## $ MassSoil.g.SD.Talweg : num 2.84 NA NA 60.64 NA ...
## $ Conc.mug.g.dry.soil.T : num 0.0438 NA NA 0.9342 NA ...
## $ comp.d13C.Talweg : num NA NA NA NA NA ...
## $ comp.d13C.SD.Talweg : num NA NA NA NA NA ...
## $ ID.T : Factor w/ 51 levels "AW-N-0","AW-N-0x",...: 36 NA NA 35 NA NA 37 NA NA 44 .
## $ MassSoil.g.South : num 100 NA NA 4928 NA ...
## $ MassSoil.g.SD.South : num 15 NA NA 739 NA ...
## $ Conc.mug.g.dry.soil.S : num 0.076 NA NA 3.738 NA ...
## $ comp.d13C.South : num NA NA NA NA NA ...
## $ comp.d13C.SD.South : num NA NA NA NA NA ...
## $ ID.S : Factor w/ 51 levels "AW-N-0","AW-N-0x",...: 19 NA NA 18 NA NA 20 NA NA 27 .
## $ DD13C.North : num NA NA NA NA NA ...
## $ DD13C.Talweg : num NA NA NA NA NA ...
## $ DD13C.South : num NA NA NA NA NA ...
## $ CatchMassSoil.g : num 214 NA NA 6763 NA ...
## $ CatchMassSoil.g.SD : num 12 NA NA 446 NA ...
## $ BulkCatch.d13 : num NA NA NA NA NA ...
## $ BulkCatch.d13.SD : num NA NA NA NA NA ...
## $ DD13.Bulk : num NA NA NA NA NA ...
## $ Area.Catchment : num 316155 NA NA 316155 NA ...
## $ BulkCatch.Conc : num 0.0683 NA NA 2.1608 NA ...
## $ iniCo.Bulk : num 12.2 NA NA 12.2 NA ...

```

```
write.csv2(soilsOut,
           'Data/MassBalance_R.csv', row.names = F)
```

Plot

```
library("ggplot2")
library("scales")
library("reshape2")
library("cowplot")
```

```
##
```

```
## Attaching package: 'cowplot'
```

```
## The following object is masked from 'package:ggplot2':
```

```
##
```

```
## ggsave
```

```
# Melt data set
```

```
##Subset the necessary columns
```

```
soilsRemainMass <- soilsOut[, c("ti", "CumAppMass.g", "CumOutDiss.g", "CumOutFilt.g", "CumOutMELsm.g",
soilsRemainMass$CumAppMass.g[1:3] <- NA
```

```
# Replace Catchment Mass's NA with the most recent non-NA prior to it (assumes no degradation).
```

```
# Purpose: To match continuous outlet time array
```

```
soilsRemainMass$CatchMassSoil.g <- na.locf(soilsRemainMass$CatchMassSoil.g)
```

```
# soilsRemainMass <- na.omit(soilsRemainMass)
```

```
##Then rearrange your data frame
```

```
remainMassMolten = melt(soilsRemainMass, id=c("ti"))
```

```
# View(remainMassMolten)
```

```
pg <- remainMassMolten
```

```
# Change variable names:
```

```
levels(pg$variable)[levels(pg$variable)=="CumAppMass.g"] <- "Applied SM Cum. (Survey)"
```

```
levels(pg$variable)[levels(pg$variable)=="CumOutMELsm.g"] <- "MEL-SM Cum. (Outlet)"
```

```
levels(pg$variable)[levels(pg$variable)=="CatchMassSoil.g"] <- "Remaining SM (1cm Soil)"
```

```
levels(pg$variable)[levels(pg$variable)=="CumOutDiss.g"] <- "Dissolved SM Cum. (Outlet)"
```

```
levels(pg$variable)[levels(pg$variable)=="CumOutFilt.g"] <- "Sediment SM Cum. (Outlet)"
```

```
# Change the order:
```

```
levels(pg$variable)
```

```
## [1] "Applied SM Cum. (Survey)" "Dissolved SM Cum. (Outlet)"
```

```
## [3] "Sediment SM Cum. (Outlet)" "MEL-SM Cum. (Outlet)"
```

```
## [5] "Remaining SM (1cm Soil)"
```

```
pg$variable <- factor(pg$variable, levels = c("Applied SM Cum. (Survey)", "Remaining SM (1cm Soil)", "Dissolved SM Cum. (Outlet)", "Sediment SM Cum. (Outlet)", "MEL-SM Cum. (Outlet)"))
```

```
pgSimple <- pg[which(pg$variable != ("Dissolved SM Cum. (Outlet)") & pg$variable != ("Sediment SM Cum. (Outlet)"),
```

```
# names(pg)[names(pg)=="variable"] <- "Estimated Mass"
```

```

massBalTop <- ggplot(pg) +
  geom_line(aes(x=ti, y=value, group = variable, color=variable)) +

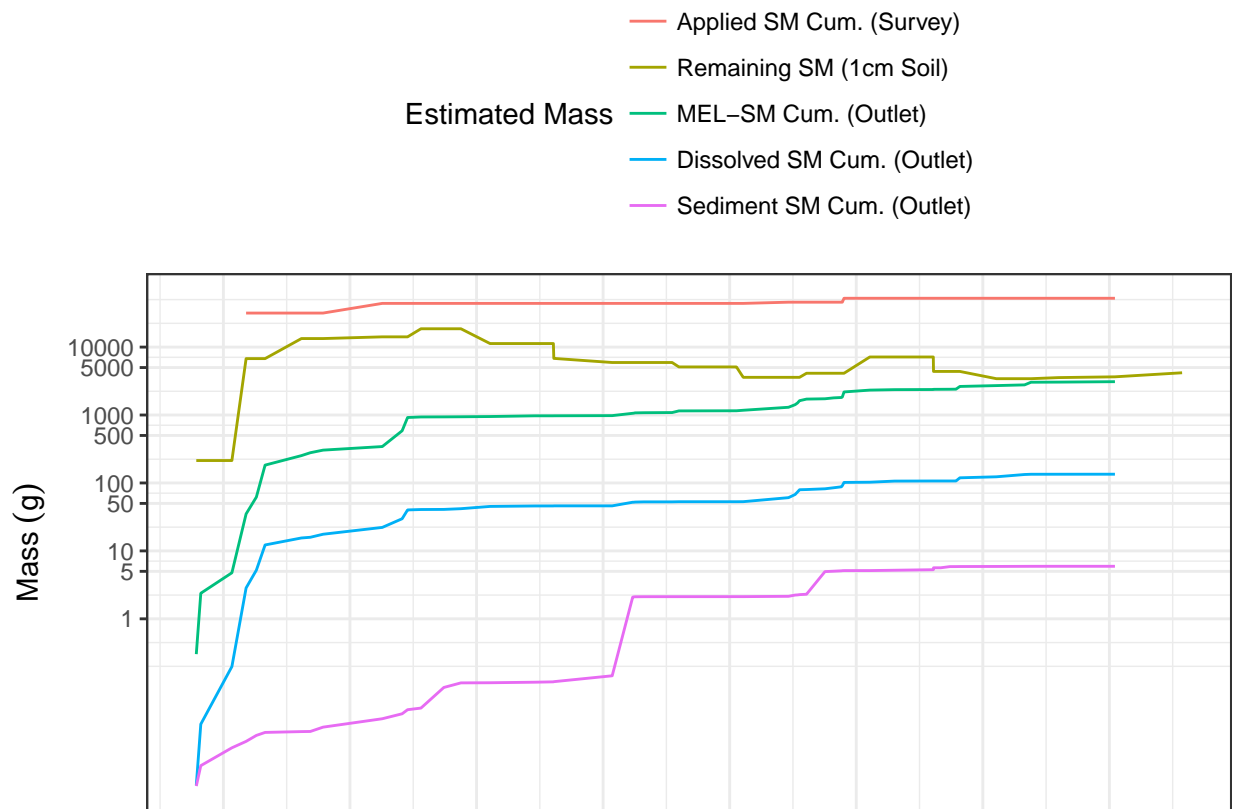
  # Themes and axes
  theme_bw() +
  theme(# axis.text.x=element_text(angle = 45, hjust = 1),
        axis.text.x=element_blank(),
        axis.title.x=element_blank(),
        legend.position="top"

        )+
  labs(color = "Estimated Mass") +
  guides(col = guide_legend(ncol = 1)) + # Sets legend parameters

  # xlab("Date") +
  scale_x_datetime(breaks = date_breaks("2 weeks"), labels = date_format("%b %d")) +
  ylab(expression(paste("Mass ", {(g)}))) +
  # scale_y_continuous(breaks = c(100, 5000, 10000, 20000), limits = c(100, 20000) )
  scale_y_continuous(trans=log_trans(), breaks=c(1,5,10,50,100,500,1000,5000, 10000))
massBalTop

```

Warning: Removed 7 rows containing missing values (geom_path).



```

massBalBottom <- ggplot(pg) +
  geom_line(aes(x=ti, y=value, color=variable)) +

```



```

# Themes and axes
theme_bw() +
theme(axis.text.x=element_text(angle = 45, hjust = 1),
      #axis.text.x=element_blank(),
      #axis.title.x=element_blank(),
      legend.position="none"
    )+
# guides(col = guide_legend(nrows = 2)) + # Sets legend parameters
xlab("Date") +
scale_x_datetime(breaks = date_breaks("2 weeks"), labels = date_format("%b %d")) +
ylab(expression(paste("Mass. S-Meto. ", {(g)}))) +
scale_y_continuous(breaks = c(1, 25, 50, 100), limits = c(0, 100) )

massBal = plot_grid(massBalTop, massBalBottom, ncol = 1, nrow = 2, align = "v")

## Warning: Removed 7 rows containing missing values (geom_path).
## Warning: Removed 63 rows containing missing values (geom_path).
massBal_MEL <- ggplot(pgSimple) +
  geom_line(aes(x=ti, y=value, group = variable, color=variable)) +

  # Themes and axes
  theme_bw() +
  theme(# axis.text.x=element_text(angle = 45, hjust = 1),
        #axis.text.x=element_blank(),
        #axis.title.x=element_blank(),
        legend.position="top"
      )+
  labs(color = "Estimated Mass") +
  guides(col = guide_legend(ncol = 3)) + # Sets legend parameters

  # xlab("Date") +
  scale_x_datetime(breaks = date_breaks("2 weeks"), labels = date_format("%b %d")) +
  ylab(expression(paste("Mass ", {(g)})))

massBal_MEL

## Warning: Removed 5 rows containing missing values (geom_path).

```

