

Getting Functionnal-structural plant model into a pipeline:

MARSHAL pipeline

Adrien Heymans

November 2018

1 Introduction

Coupling model with each other can be use to find releavant sequence of information. Depending of the models, the complexity of the system is significantly increase.

The system on which we are going to work is a below ground system. Where, we focuses on observing hydraulic architecture of a root system.

In our case, the bondary condition of the system are :

- The water potential of the soil
- The root system architecture
- The hydraulic properties of the root system
- The initial pressure strength that pull water out of the collar

In this case, we illustrate the fact that models can be linked with one another. Different element that are modeled such as root growth or the evolution of the water potential in the soil can be complementary. Dealing with models as an input for another model will increase the dynamic approach of the overall. To summarise this concept, the following illustration use the coupling of models that build root system architecture (CRootBox), the infomation of the soil water potential (Hydrus1D) with another that use this explicit root system architecture and the soil water potential to compute the hydraulic architecture of the root system (MARSHAL). Other kind of pipeline such as this can be achieved as well.

To run this example, the following package and function are require. The function are strored in a GitHub depository. link :

- <https://github.com/MARSHAL-ROOT/marshall>

```
install.packages("devtools")
library(devtools)
install_github("MARSHAL-ROOT/marshall")
```

- [https://github.com/MARSHAL-ROOT/... /!](https://github.com/MARSHAL-ROOT/...) to be completed

```
library(data.table)
library(tidyverse)
library(Matrix)
library(readxl)
library(cowplot)
library(xml2)
library(knitr)
#library(marshall) # ! not working
`%in%` <- compose(`!`, `%in%`)

source("R/io_function.R")
source("R/getSUF.R")
```

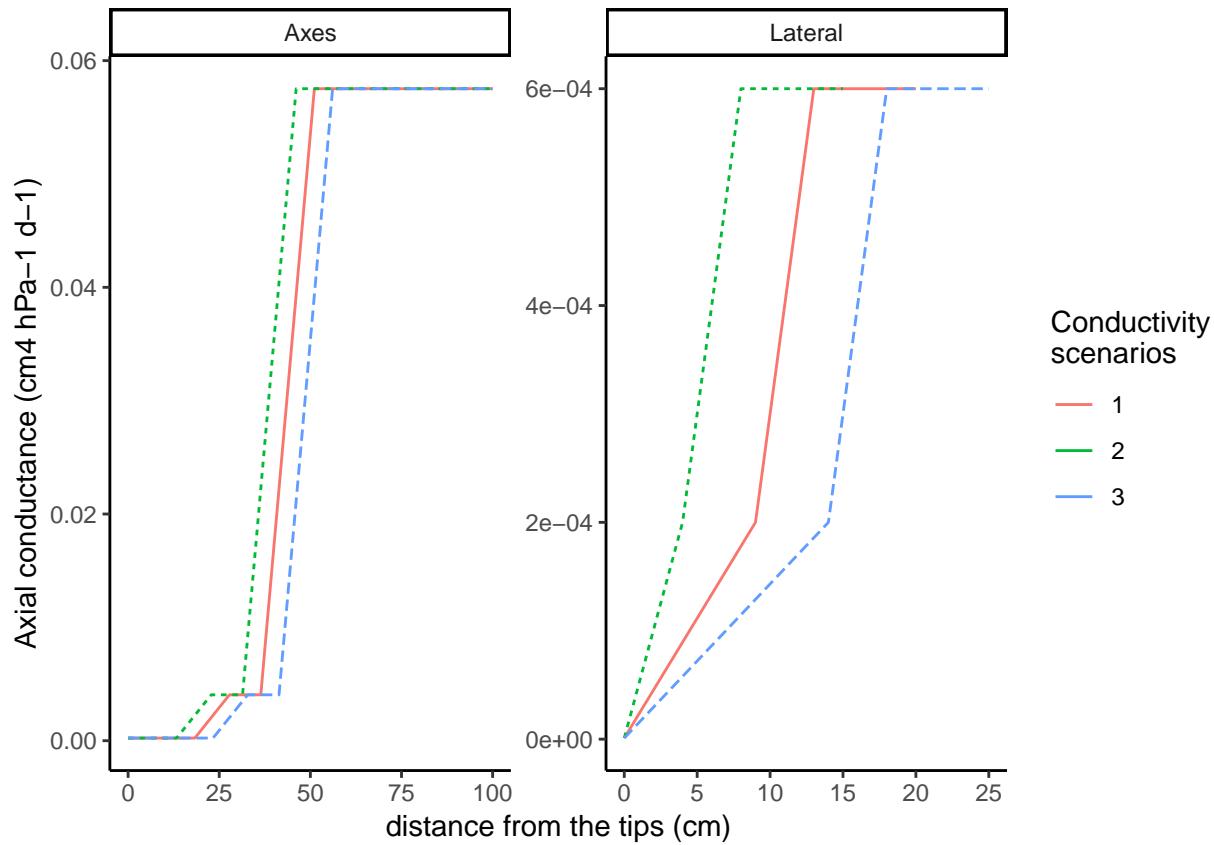
The following lines of code give the boundary conditions of the models.

The hydraulic properties of the root system can be found in the literature or by measuring it on root samples. Here, the conductivities of the root come from the Doussan et al. 1998 paper. To show some possibilities, two scenarios were built from this Doussan et al. table of conductivities. The first, which is the original values placed along the root axes from the apex. The second, is based on the first but the maturity of the root anatomy is advanced of 5 cm towards the apex.

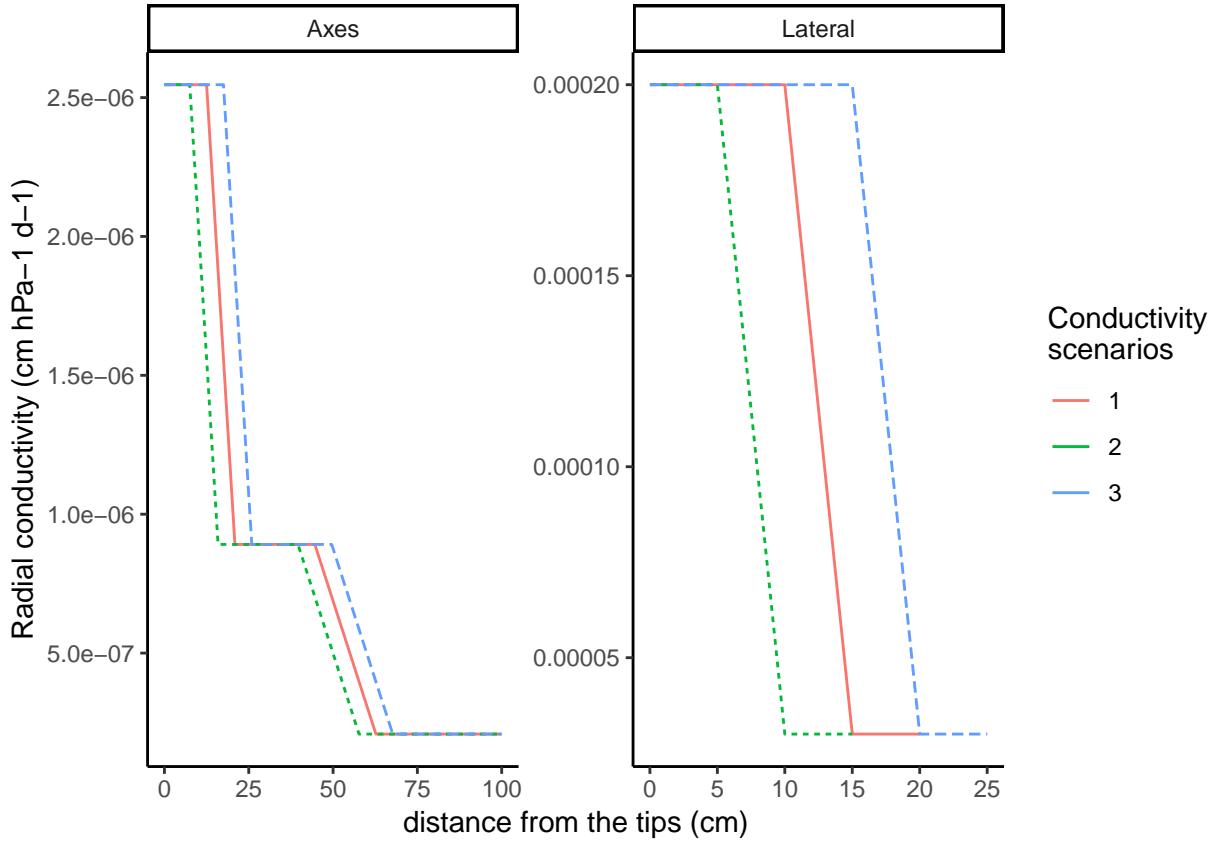
```
# Conductivities from Doussan et al. 1998
conductivities <- read_excel("www/conductivities.xlsx")

#Creating scenarios of Conductivities similar to the Doussan et al. 1998
#at the exception of all changes of conductivities have been move forward or backward to the tip.
conduct_1 <- conductivities%>%
  mutate(scenario = 1)
conduct_2 <- conductivities%>%
  mutate(scenario = 2)
conduct_2$x[which(conduct_2$x %!in% c(0, 100))] <- conduct_2$x[which(conduct_2$x %!in%
  c(0, 100))]-5
conduct_3 <- conductivities%>%
  mutate(scenario = 3)
conduct_3$x[which(conduct_3$x %!in% c(0, 100))] <- conduct_3$x[which(conduct_3$x %!in%
  c(0, 100))]+5
conductivities <- rbind(conduct_1, conduct_2, conduct_3)

#Evolution of Kx along the root types
conductivities %>%
  filter(type == "kx",
    order_id == 1 | order_id == 2) %>%
  ggplot(aes(x,y, colour = factor(scenario)))+
  geom_line(aes(color = factor(scenario), linetype = factor(scenario)))+
  theme_classic()+
  xlab("distance from the tips (cm)") +
  ylab("Axial conductance (cm4 hPa-1 d-1)")+
  labs(colour = "Conductivity \nscenarios")+
  guides(linetype = F)+
  facet_wrap(~order, scales = "free", labeller =as_labeller(babbel))
```



```
#Evolution of Kr along the root types
conductivities %>%
  filter(type == "kr",
         order_id == 1 | order_id == 2) %>%
  ggplot(aes(x,y, colour = factor(scenario)))+
  geom_line(aes(color = factor(scenario), linetype = factor(scenario)))+
  theme_classic()+
  xlab("distance from the tips (cm)")+
  ylab("Radial conductivity (cm hPa-1 d-1)")+
  labs(colour = "Conductivity \nscenarios")+
  guides(linetype = F)+
  facet_wrap(~order, scales = "free", labeller =as_labeller(babbel))
```



2 Root architecture with CRootBox

Modelling your root architecture can be achieved with CRootBox for instance. Some examples of the parameter files for a couple of species are details in the “modelparameter” folder, which is linked to the CRootBox depository. For the purpose of the example we will choose to work with the “Zea_mays_3_Postma_2011” parameter.

```
#Number of repetition per root system architecture example
nrep <- 5

#List of three CrootBox parameter for corn
param_choise <- c("Zea_mays_1_Leitner_2010",
                  "Zea_mays_3_Postma_2011",
                  "Zea_mays_4_Leitner_2014")

all_roots <- NULL
for(param_name in param_choise){
  #Load param of the selected example
  rparam <- read_rparam(path = paste0("./www/", param_name, ".rparam"))
  pparam <- read_pparam(path = paste0("./www/", param_name, ".pparam"))
  #overwrite the param files that will be used
  write_rparam(rparam, "./www/param.rparam")
  write_pparam(pparam, "./www/param.pparam")

  for(i in 1:nrep){
    # Run CrootBox ----
  }
}
```

```

system("www/a.exe")

# Get all the files exported by CRootBox
fls <- list.files("./")
fls <- fls[grep1("rootsystem.txt", fls)]

for(f in fls){
  temp <- fread(f, header = T)
  temp$age <- strsplit(f, "_")[[1]][1]
  temp$rep <- i
  temp$ex <- param_name
  all_roots <- rbind(all_roots, temp)
}
}

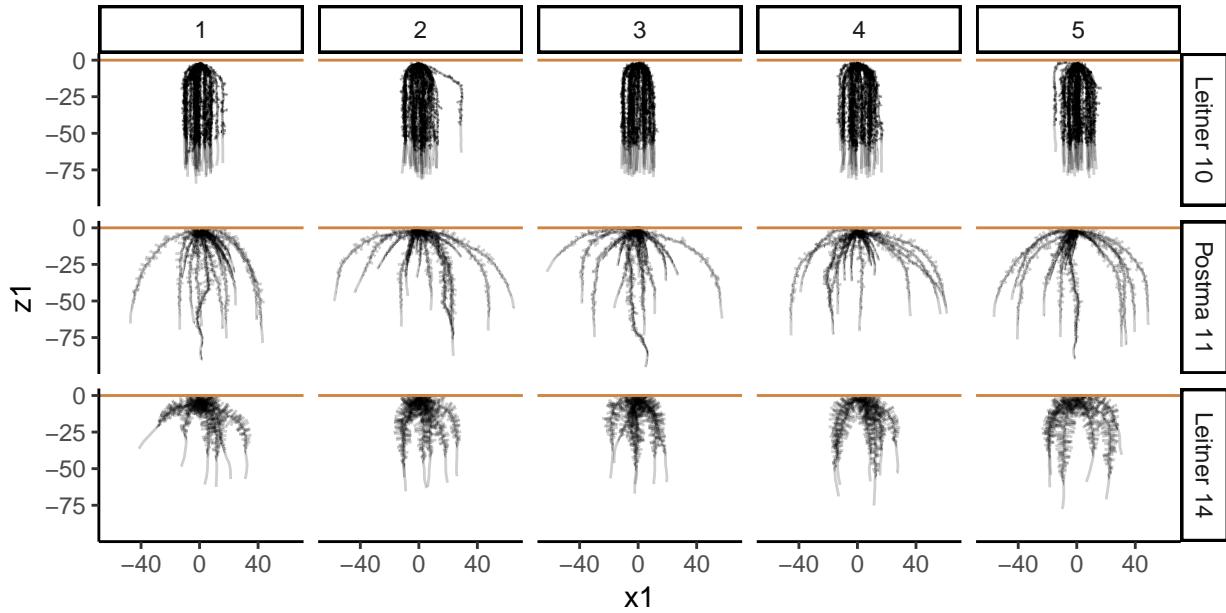
write_csv(all_roots, "all_roots.csv")
#all_roots <- read.csv("all_roots.csv")

full_time <- sort(unique(as.numeric(all_roots$age)))

time_step = full_time[2]-full_time[1]
endTime = max(full_time)
ntimesteps = endTime/time_step

```

The following figure shows the root system architecture that comes out CRootBox at the half of the simulation time.



We can get access to the convex hull

```

root_20 <- all_roots%>%
  filter(age %in% c("20", "40", "60"))
pol_area = poly_area<- NULL
for(i in unique(all_roots$ex)){
  for(j in unique(all_roots$rep)){
    for(ag in unique(root_20$age)){
      for(xy in c("x", "y")){#get the convex hull from two side
        tmp <- root_20%>%
          filter(ex == i, rep == j, age == ag)%>%
          select(paste0(xy,"2"),z2, ex, rep, age)
          #select(x2,z2, ex, rep, age)
        first_col <- tmp[,1]
        x <- matrix(c(first_col, tmp$z2), nc = 2)
        ch <- chull(x= first_col, y = tmp$z2)
        ch <- c(ch, ch[1])

        #Area of the convex hull
        xy.coords <- cbind(first_col, tmp$z2)
        chull.coords <- xy.coords[ch,]
        chull.poly <- sp::Polygon(chull.coords, hole = F)
        chull.area <- chull.poly@area

        polyg <- cbind(as.data.frame(x[ch,]), ex = i, rep = paste0(j,xy), age = ag)
        pol_area <- c(chull.area/10000)%>%
          as.tibble()%>%
      }
    }
  }
}

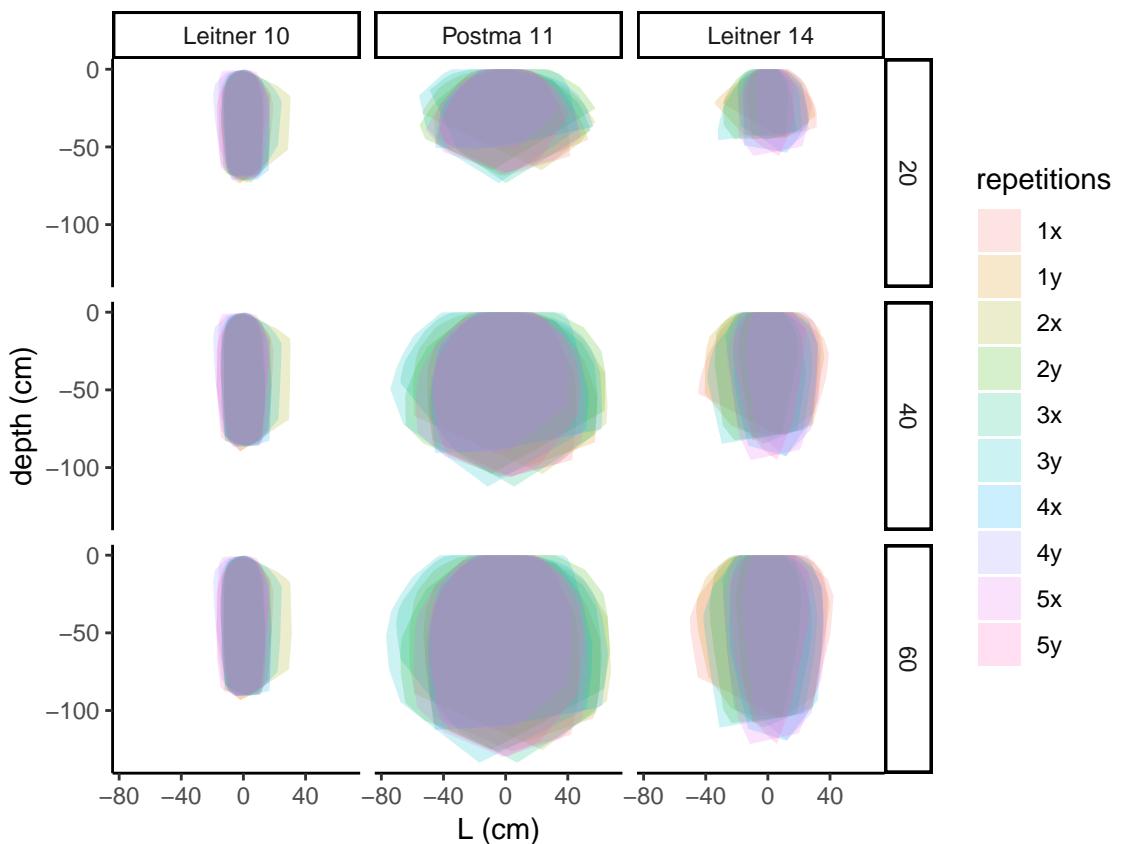
```

```

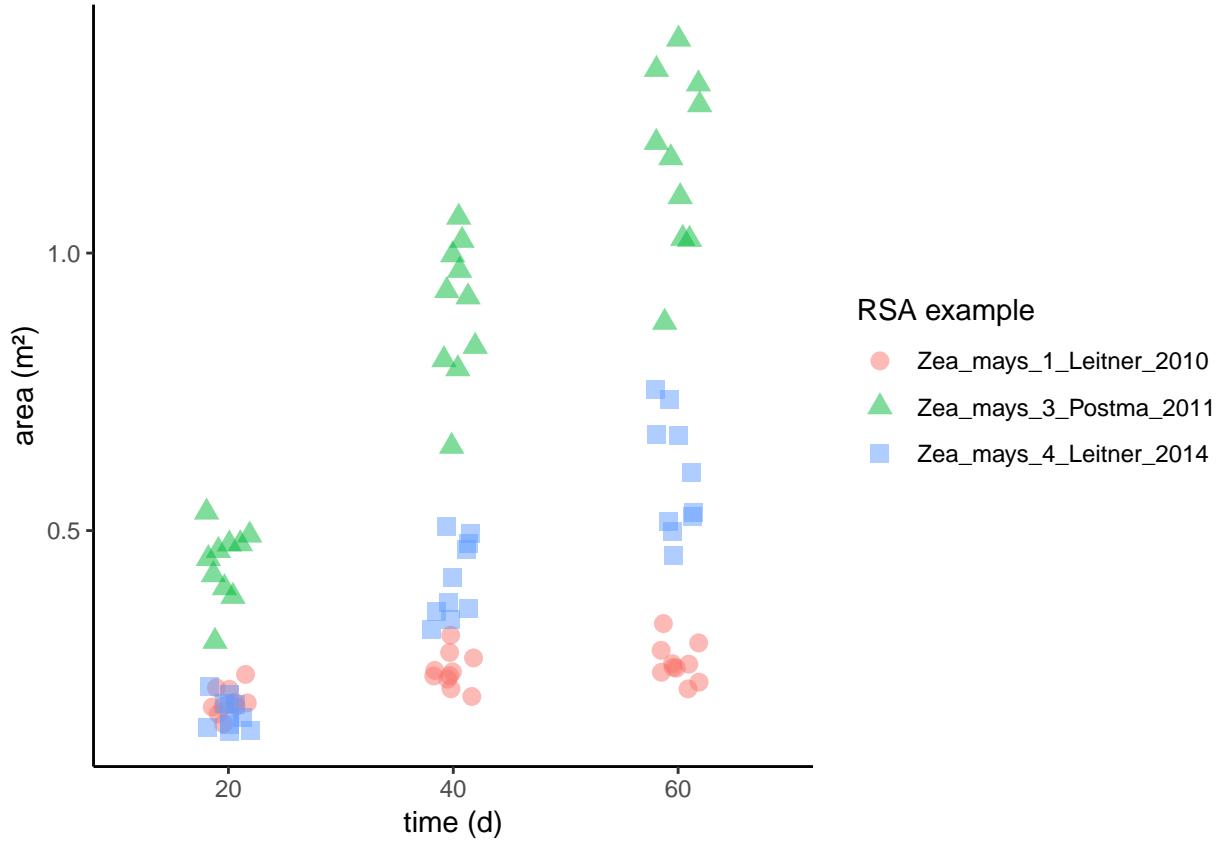
        transmute(area = value, ex = i, rep= paste0(j,xy), age = as.numeric(ag))
poly_area <- rbind(poly_area, pol_area)
pol_ch <- rbind(pol_ch, polyg)
}
}
}
}

pol_ch%>%
ggplot()+
theme_classic()+
coord_fixed() +
geom_polygon(aes(x = V1, y = V2, fill = factor(rep)), alpha = 0.2)+
ylab("depth (cm)")+
xlab("L (cm)")+
facet_grid(age ~ ex, labeller = as_labeller(babbel))+
labs(fill = "repetitions")

```

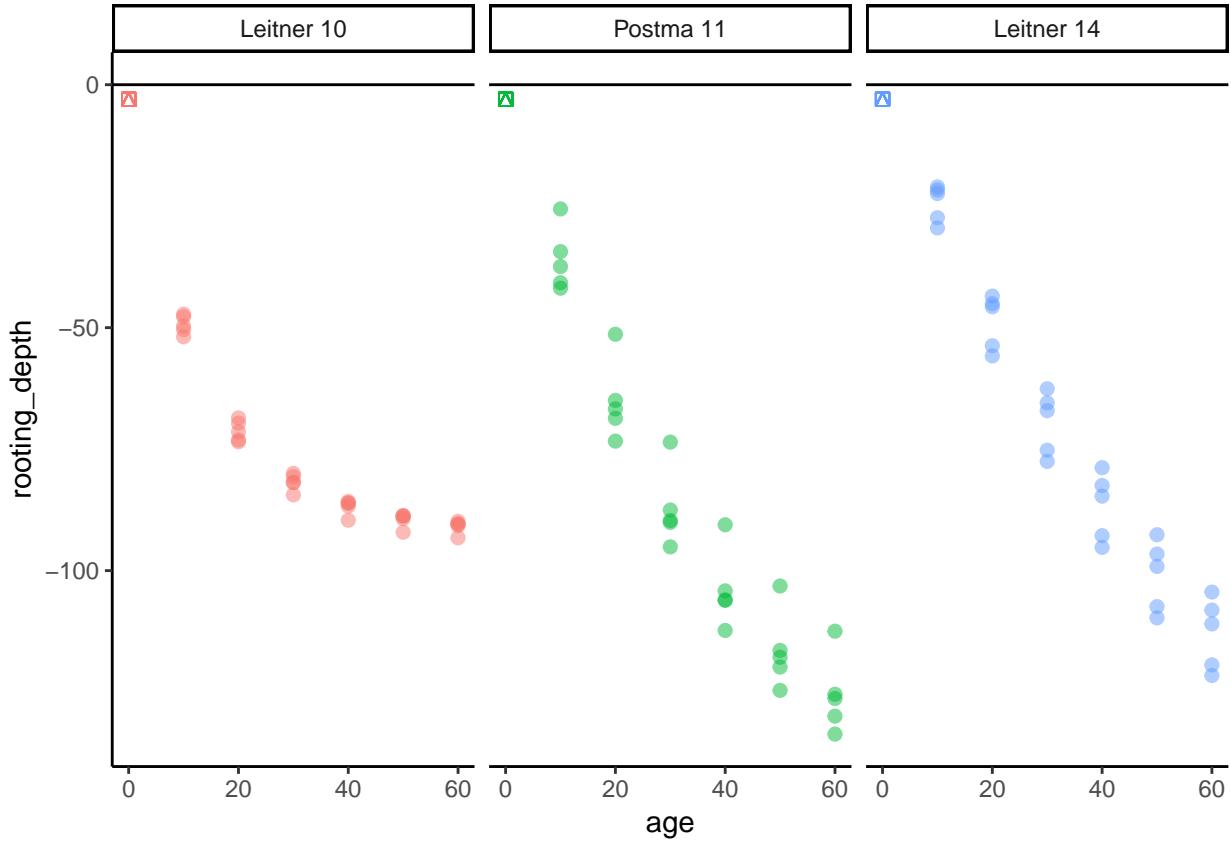


as well as the area of the convex hull.



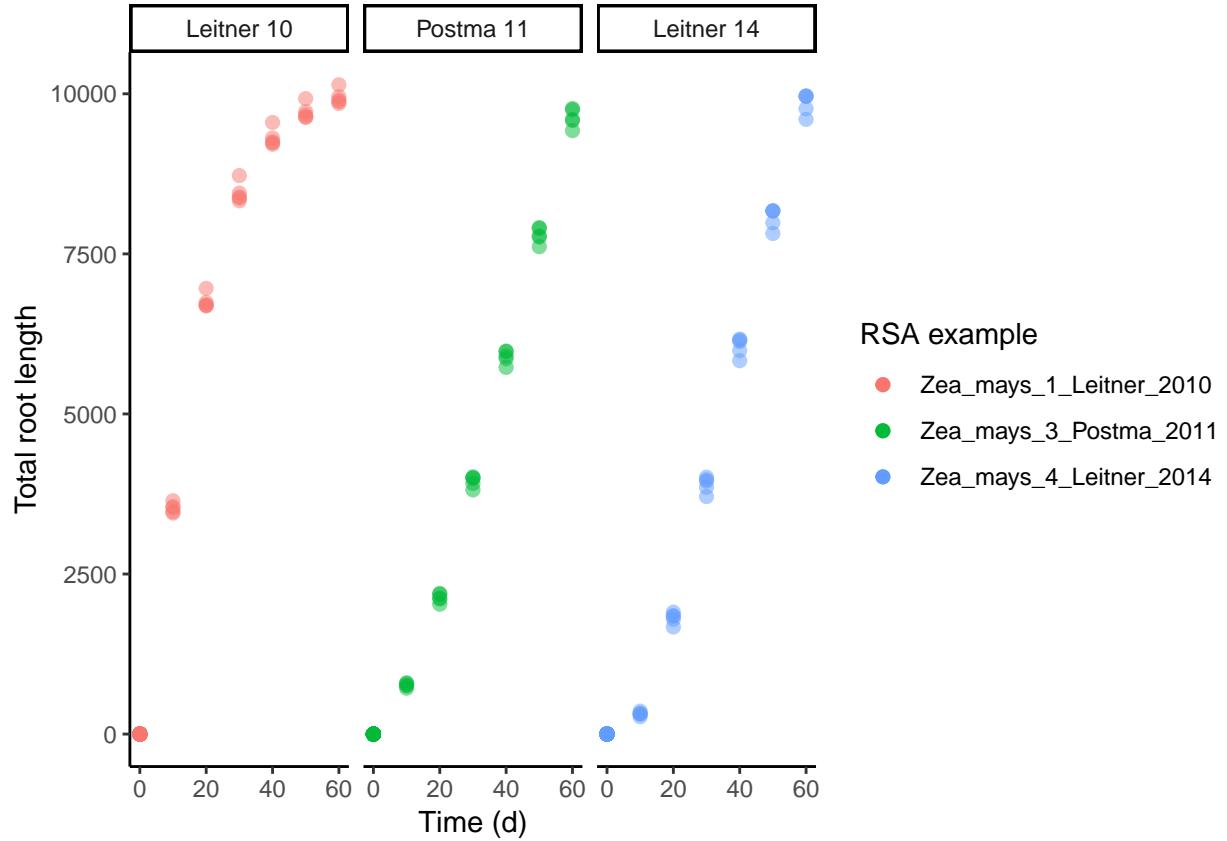
```
RLD <- all_roots %>%
  mutate(age = as.numeric(age)) %>%
  group_by(age, rep, ex) %>%
  summarise(root = sum(length),
            rooting_depth = min(z2))
RD <- all_roots %>%
  mutate(age = as.numeric(age)) %>%
  group_by(age, ex) %>%
  summarise(rooting_depth = min(z2),
            rDepth = mean(rooting_depth)) %>%
  as.data.frame()

RLD %>%
  ggplot(aes(age, rooting_depth, colour = factor(ex))) +
  geom_point(size = 2, alpha = 0.5) +
  geom_point(aes(x = 0, y = -3), size = 2, alpha = 0.5, shape = 14) +
  geom_hline(yintercept = 0) +
  theme_classic() +
  facet_wrap(~ex, labeller = as_labeller(babbel)) +
  guides(colour = F)
```

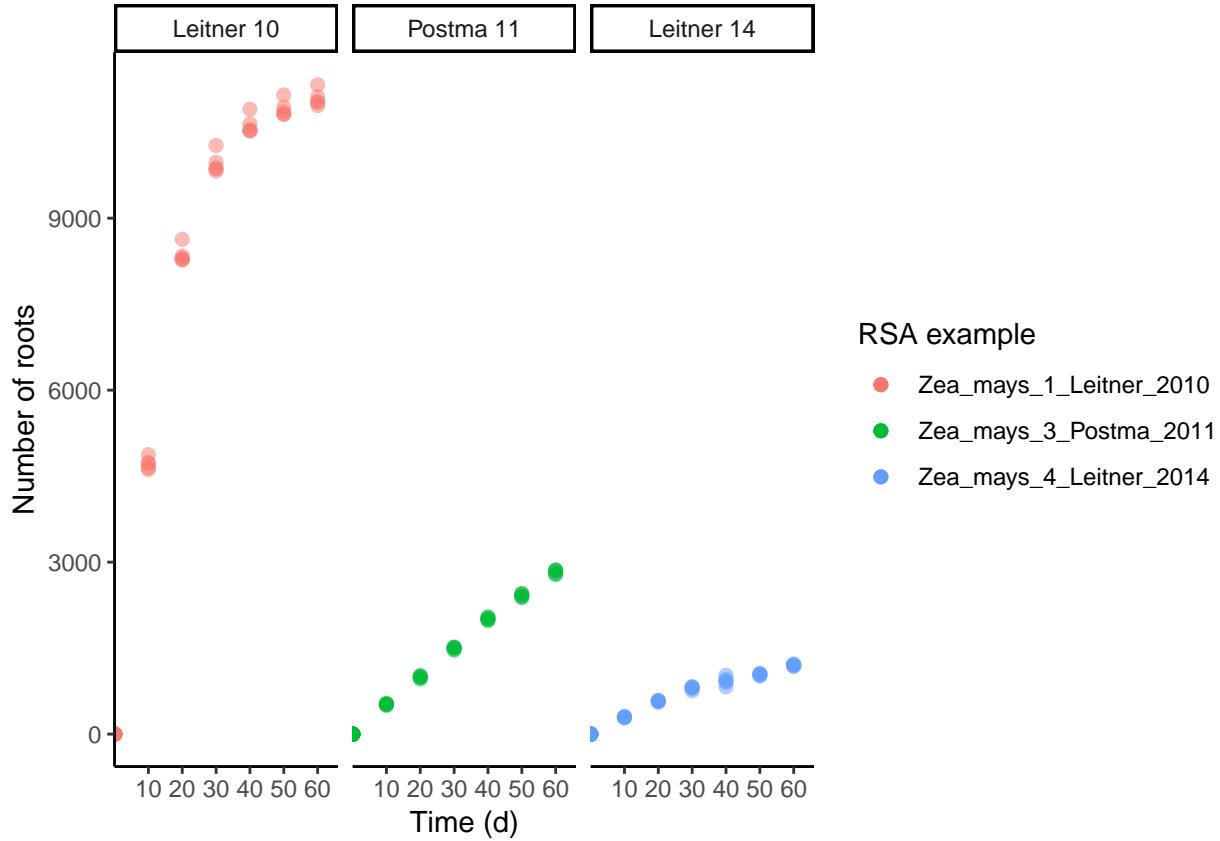


The total root length

```
RLD %>%
  ggplot(aes(age, root, colour = factor(ex))) +
  geom_point(size = 2, alpha = 0.5) +
  geom_point(aes(x = 0, y = 0), size = 2) +
  theme_classic() +
  facet_wrap(~ex, labeller = as_labeller(babbel)) +
  labs(colour = "RSA example") +
  ylab("Total root length") +
  xlab("Time (d)")
```



```
all_roots%>%
  group_by(age, ex, rep)%>%
  summarise(tip = max(sort(unique(branchID))))%>%
  ggplot(aes(age, tip, colour = factor(ex)))+
  geom_point(size = 2, alpha = 0.5)+
  geom_point(aes(x = 0, y = 0), size = 2)+
  theme_classic()+
  facet_wrap(~ex, labeller = as_labeller(babbel))+
  labs(colour = "RSA example")+
  ylab("Number of roots")+
  xlab("Time (d)")
```



The pressure head of the water potential within the soil is created from the following line of code. Here, there are three soil scenarios with three different coefficient in the drying function.

```
profile_depth <- 150
z <- seq(-profile_depth, 0, by = 5)
value <- sort(rep(full_time, length(z)))
psi <- rep(-230, length(value))
z <- rep(z, ntimesteps)

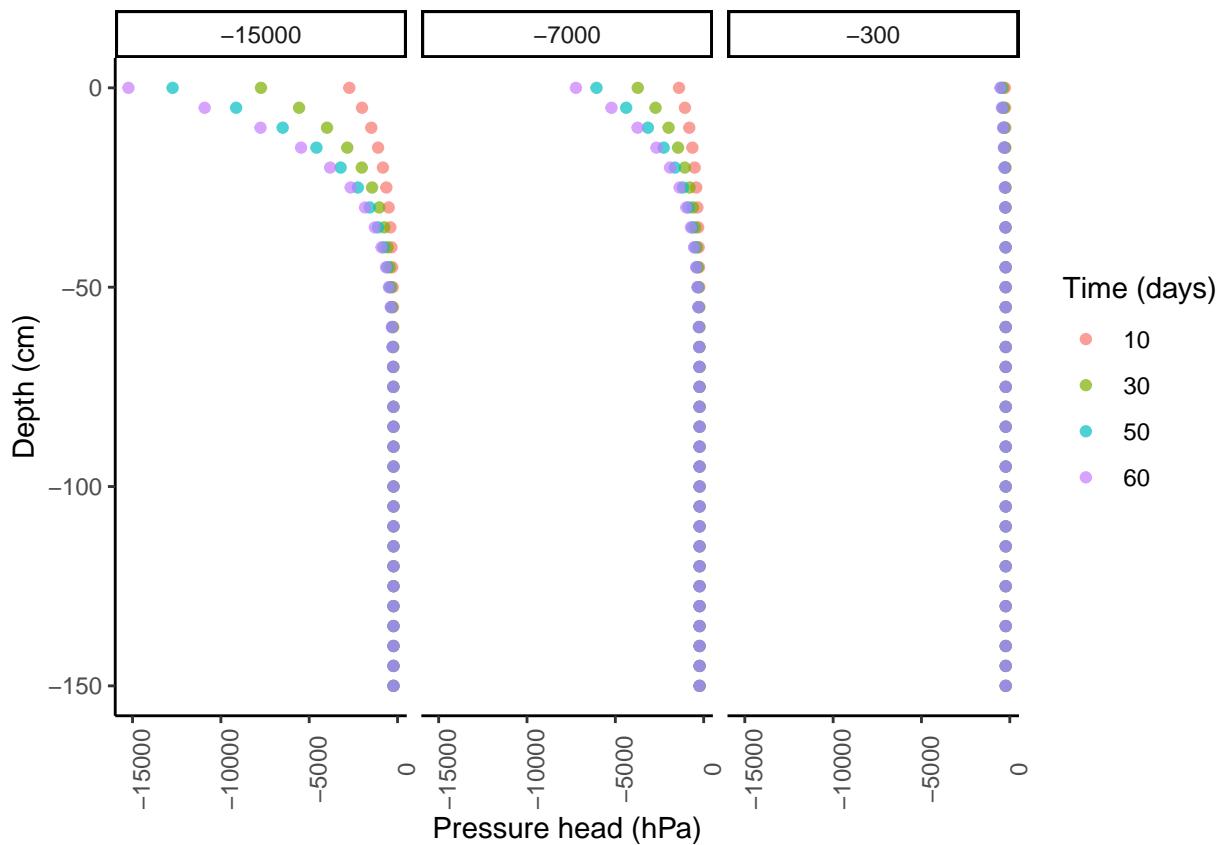
soil <- NULL
for(i in c(-300, -7000, -15000)){
  soil_tmp <- cbind(z, value, psi) %>%
    as.tibble() %>%
    mutate(psi = i*((z+profile_depth)^(10))*value) /
      ((profile_depth^10)*endTime)-230,
    humid = i)
  soil_tmp <- soil_tmp %>%
    mutate(id = 1:nrow(soil_tmp))
  soil<- rbind(soil, soil_tmp)
}

soil %>%
  filter(value %in% c(10,30,50,60)) %>%
```

```

ggplot(aes(psi, z, colour = factor(value)))+
  geom_point(alpha = 0.7)+
  theme_classic()+
  xlab("Pressure head (hPa)") +
  ylab("Depth (cm)")+
  labs(colour = "Time (days)")+
  facet_grid(~humid)+
  theme(axis.text.x=element_text(angle=90))

```



```
tputs <- -15000
```

3 MARSHAL

MARSHAL provide an estimation of the conductance of the whole root system, the actual and the potential transpiration. The input needed in order to provide those relevant informations are :

1. The root architecture description.
2. The different conductivities of the different root type.
3. The water soil potential.
4. The initial transpiration potential.

```

new_all_roots <- NULL
results<- NULL
#Computing the hydraulic of the root system
k <- 1

```

```

#Total of simulation that will be launch
all <- length(unique(all_roots$rep)) * length(unique(conductivities$scenario)) *
      length(unique(all_roots$age)) * length(unique(all_roots$ex)) *
      length(unique(soil$humid))
pb = txtProgressBar(min = 0, max = all, initial = 0, style = 3)

##  

|  

|          |  0%  

for(exa in unique(all_roots$ex)){# Example of RSA
  for(re in unique(all_roots$rep)){ # repetition
    for(ag in sort(as.numeric(unique(all_roots$age)))){# age of the RS
      for(sen in unique(conductivities$scenario)){ #hydraulic properties
        for(hum in unique(soil$humid)){ #Scenario of water potential in the soil

          # Progression bar
          setTxtProgressBar(pb,k)

          #Select specific Soil for the simulation
          temp_soil <- soil %>%
            filter(value == ag,
                  humid == hum)%>%
            select(-value, -humid)
          #select specific root system for the simulation
          temp_root <- all_roots %>%
            filter(age == ag,
                  rep == re,
                  ex == exa)%>%
            as.tibble()%>%
            select(-ex)
          #select specific conductivities
          temp_conduct <- conductivities%>%
            filter(scenario == sen)%>%
            select(-scenario)

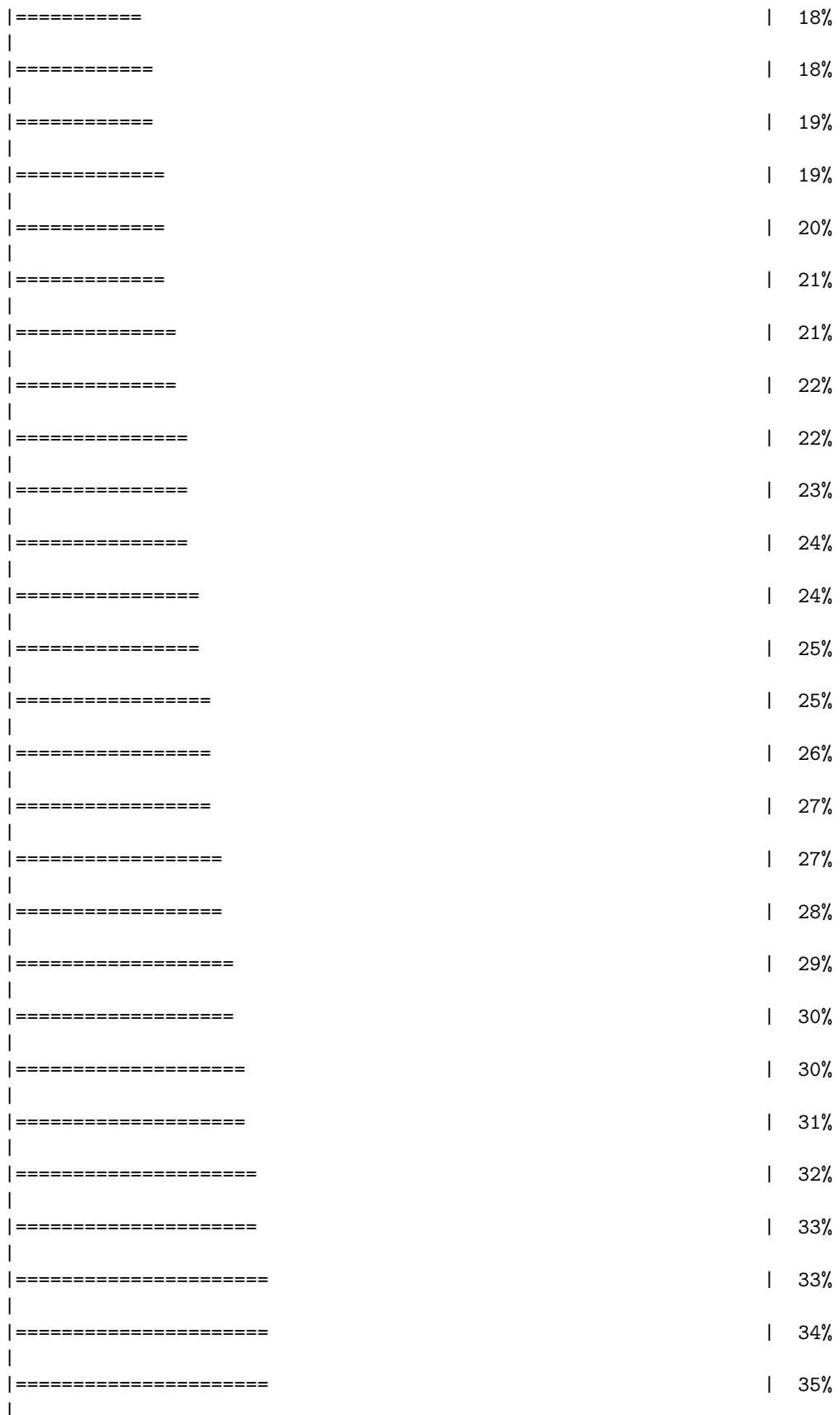
          # Run MARSHAL ----
          hydraulics <- getSUF(temp_root,
                                temp_conduct,
                                temp_soil,
                                hetero = T,
                                Psi_collar = tpots)

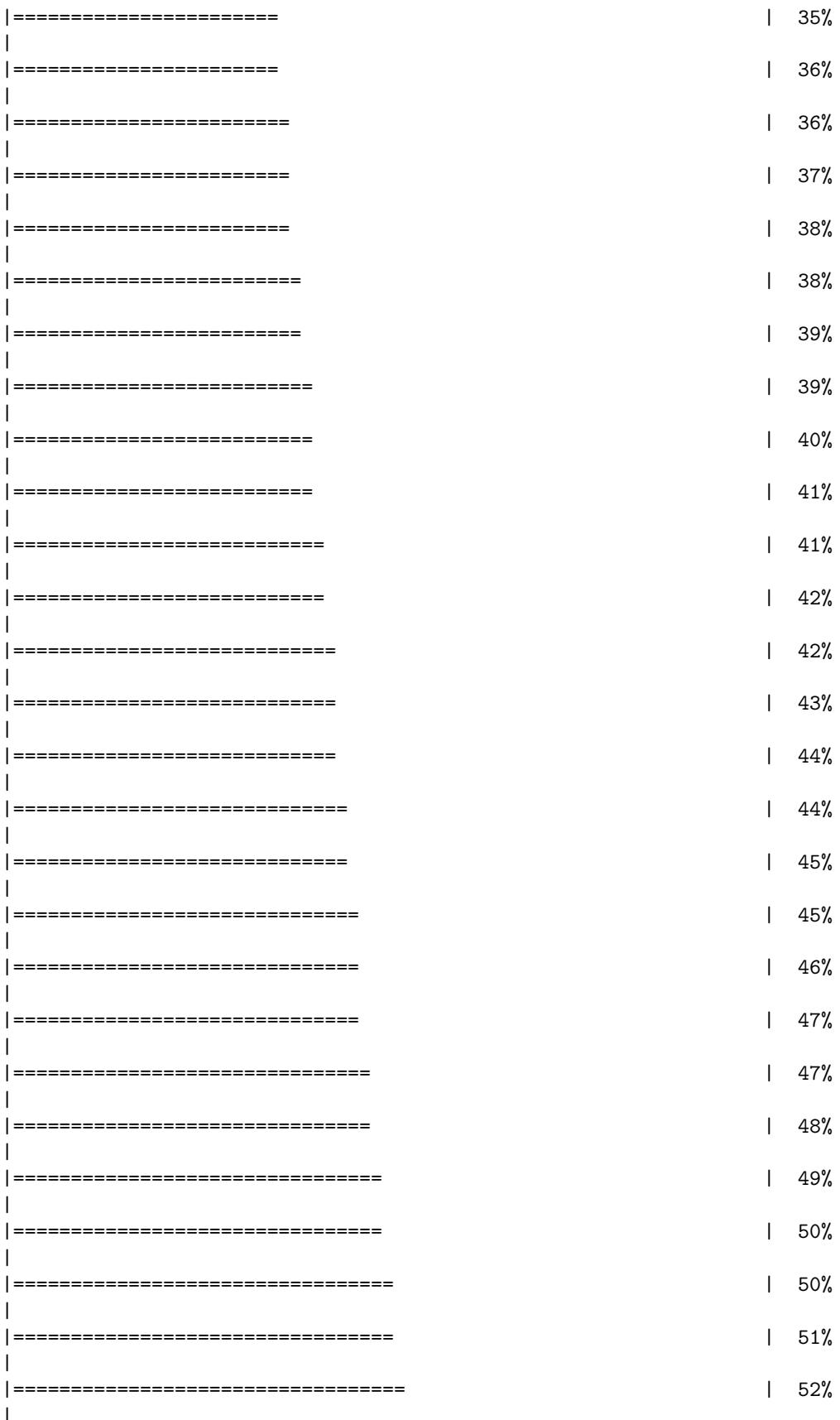
          #Aggregate output from MARSHAL
          results <- rbind(results, data.frame(krs = hydraulics$krs,
                                                tact = hydraulics$tact,
                                                tpot = hydraulics$tpot,
                                                tp = tpots,
                                                scenario = sen,
                                                age = ag,
                                                rep = re,
                                                ex = exa,
                                                humid = hum,
                                                simulation = k))
        }
      }
    }
  }
}

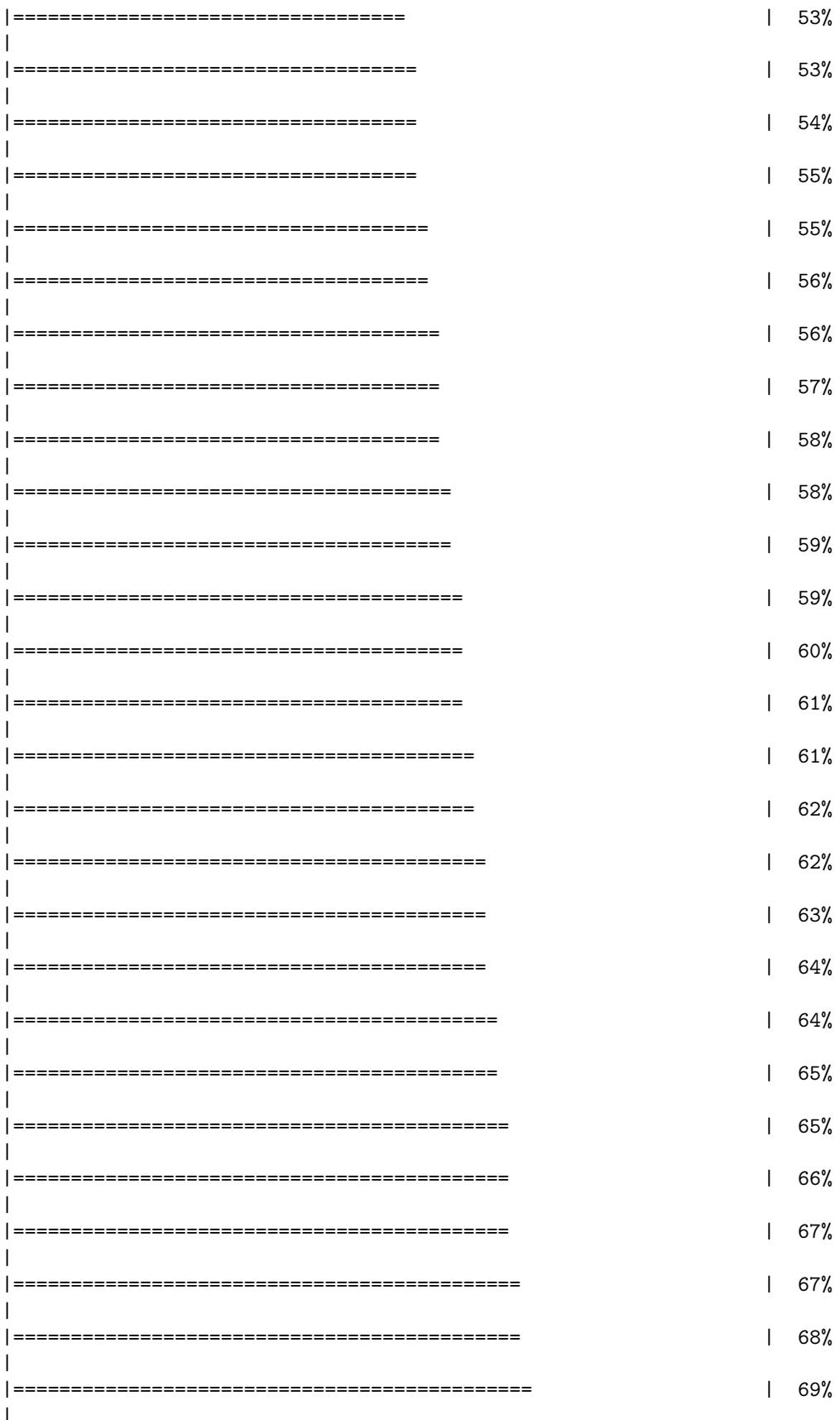
```



```
##  
|  
|  
|  
|= | 1%  
|  
|= | 1%  
|  
|= | 2%  
|  
|== | 2%  
|  
|== | 3%  
|  
|== | 4%  
|  
|== | 4%  
|  
|== | 5%  
|  
|==== | 5%  
|  
|==== | 6%  
|  
|==== | 7%  
|  
|==== | 7%  
|  
|==== | 8%  
|  
|==== | 9%  
|  
|==== | 10%  
|  
|==== | 10%  
|  
|==== | 11%  
|  
|==== | 12%  
|  
|==== | 13%  
|  
|==== | 13%  
|  
|==== | 14%  
|  
|==== | 15%  
|  
|==== | 15%  
|  
|==== | 16%  
|  
|==== | 16%  
|  
|==== | 17%
```







====	70%
====	70%
====	71%
====	72%
====	73%
====	73%
====	74%
====	75%
====	75%
====	76%
====	76%
====	77%
====	78%
====	78%
====	79%
====	79%
====	80%
====	81%
====	81%
====	82%
====	82%
====	83%
====	84%
====	84%
====	85%
====	85%
====	86%

```

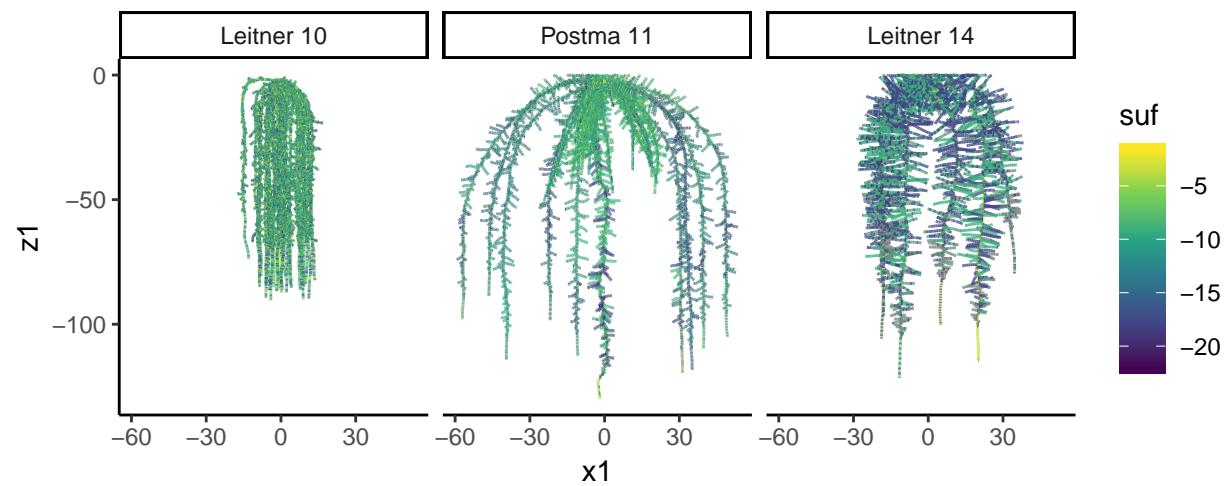
|=====| 87%
|
|=====| 87%
|
|=====| 88%
|
|=====| 89%
|
|=====| 90%
|
|=====| 90%
|
|=====| 91%
|
|=====| 92%
|
|=====| 93%
|
|=====| 93%
|
|=====| 94%
|
|=====| 95%
|
|=====| 95%
|
|=====| 96%
|
|=====| 96%
|
|=====| 97%
|
|=====| 98%
|
|=====| 98%
|
|=====| 99%
|
|=====| 99%
|
|=====| 100%
#Save
write_csv(new_all_roots, "new_all_roots.csv")
#new_all_roots <- read.csv("new_all_roots.csv")

```

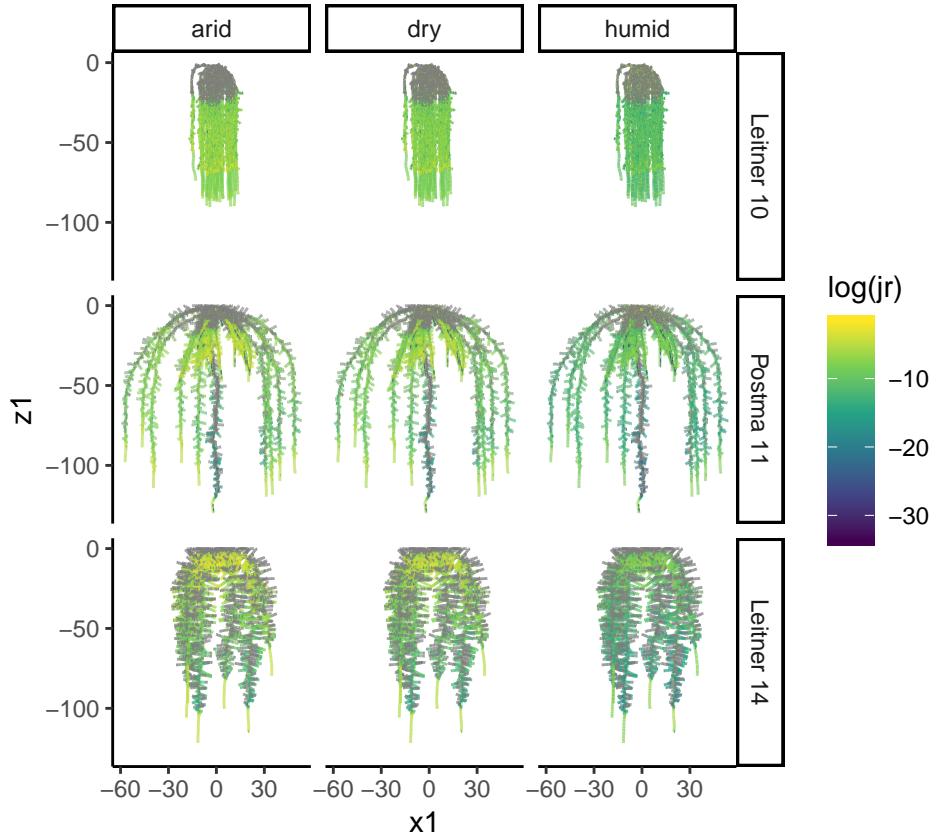
3.1 Results :

Once the simulation are computed, it is possible to visualize the output on a various form

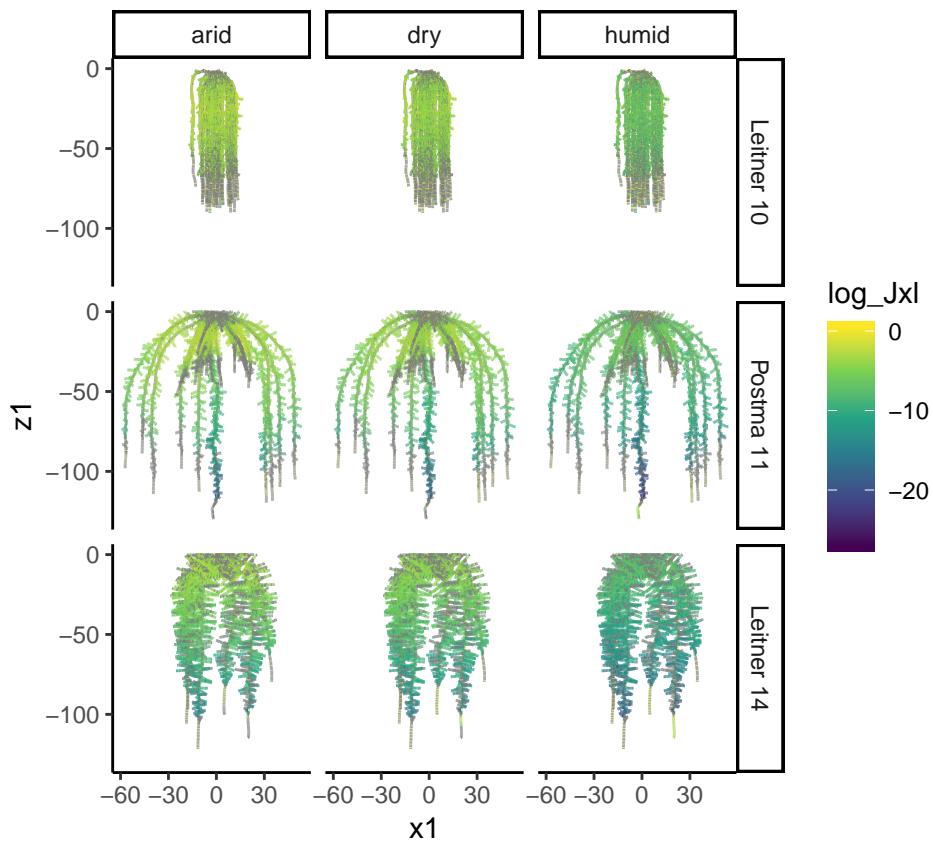
3.1.1 Standard uptake fraction



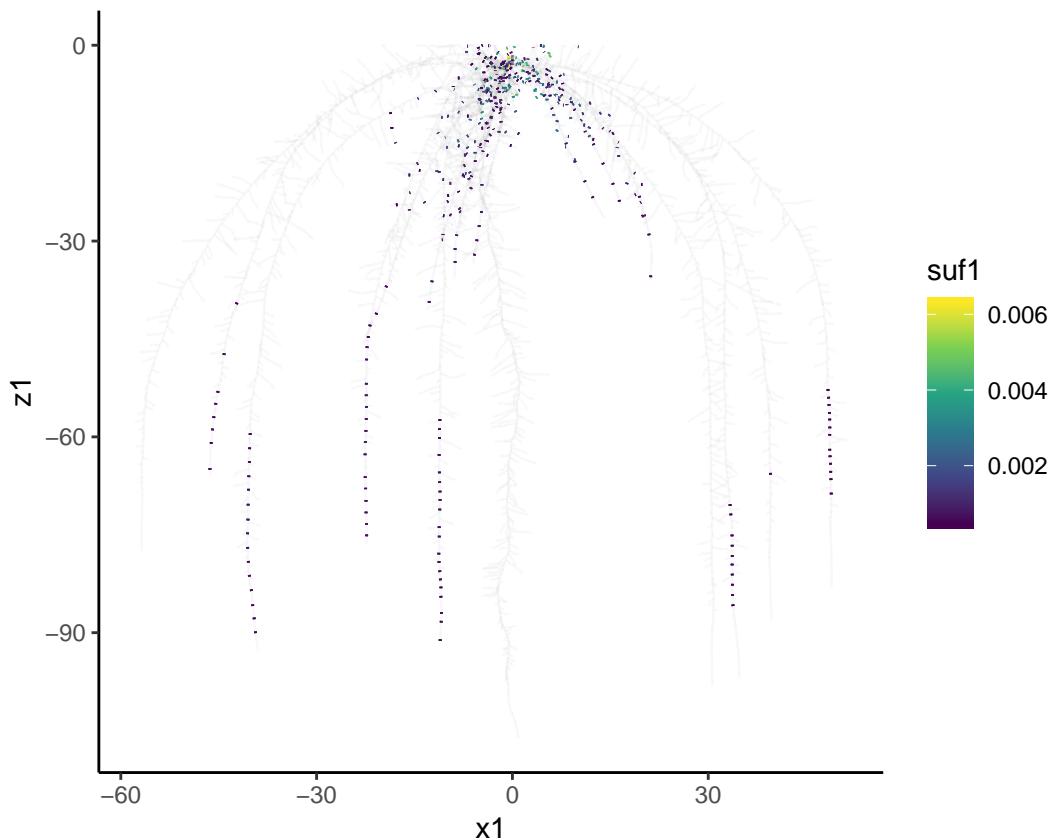
The absorption



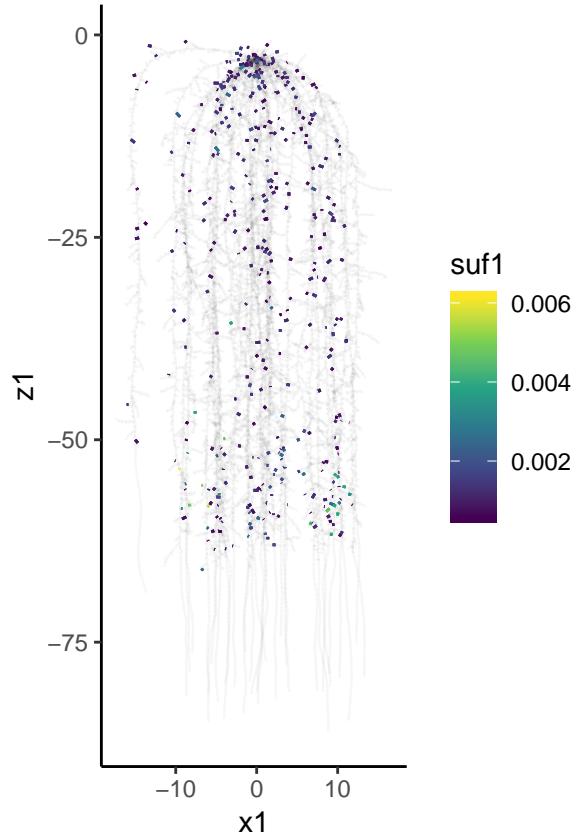
The axial fluxes



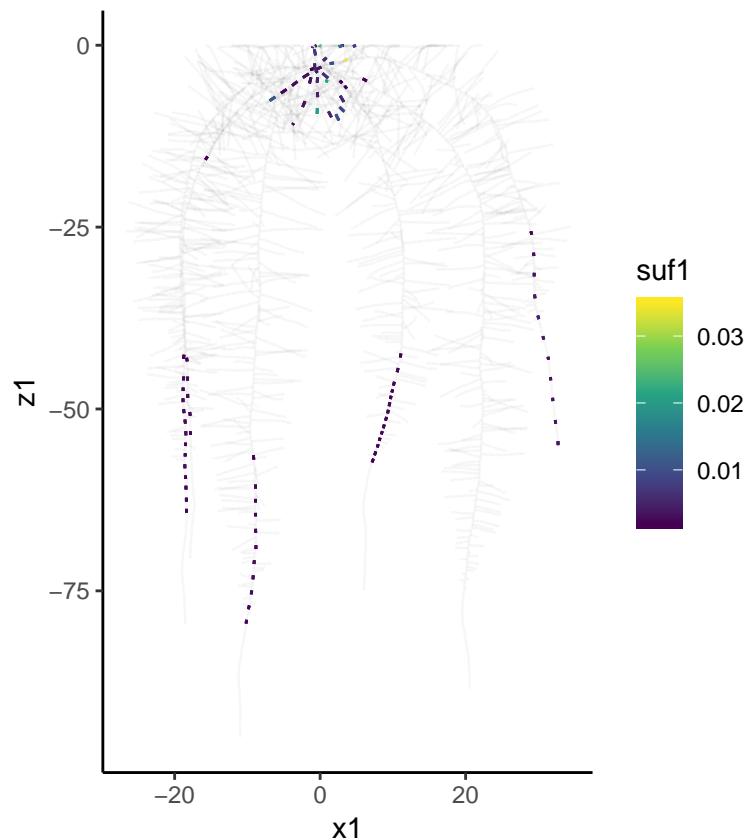
The smallest region on which more or less 50% of the water uptake occurs



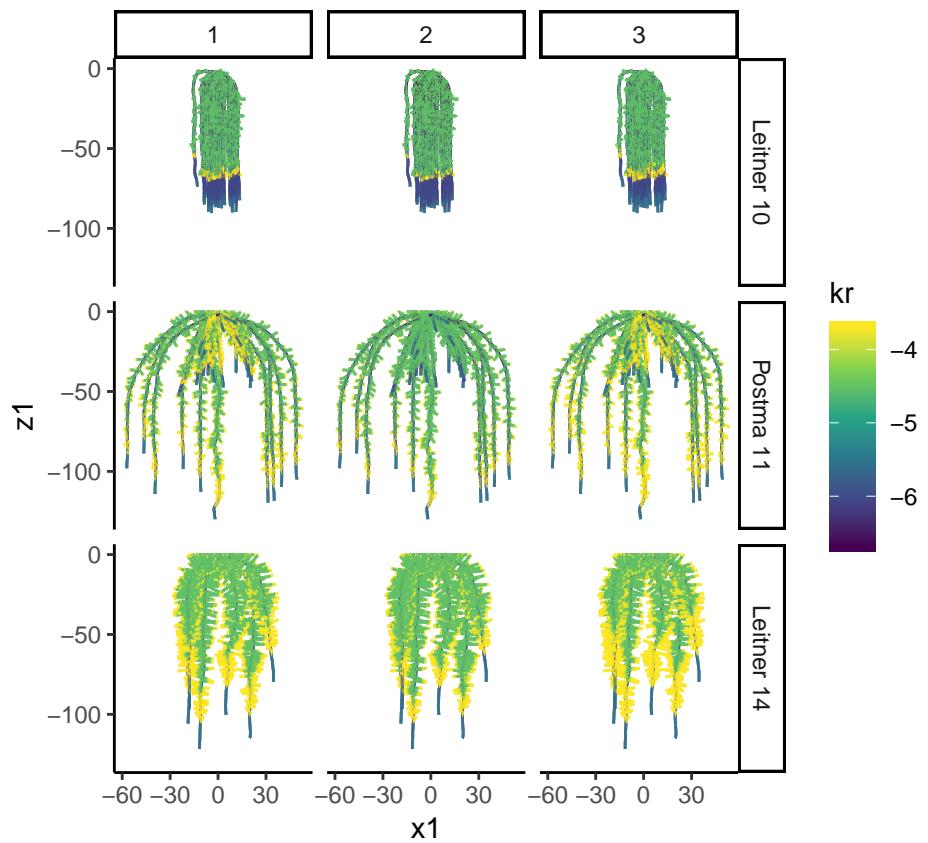
```
## # A tibble: 1 x 2
##   ex                  high
##   <chr>                <dbl>
## 1 Zea_mays_3_Postma_2011 0.545
```

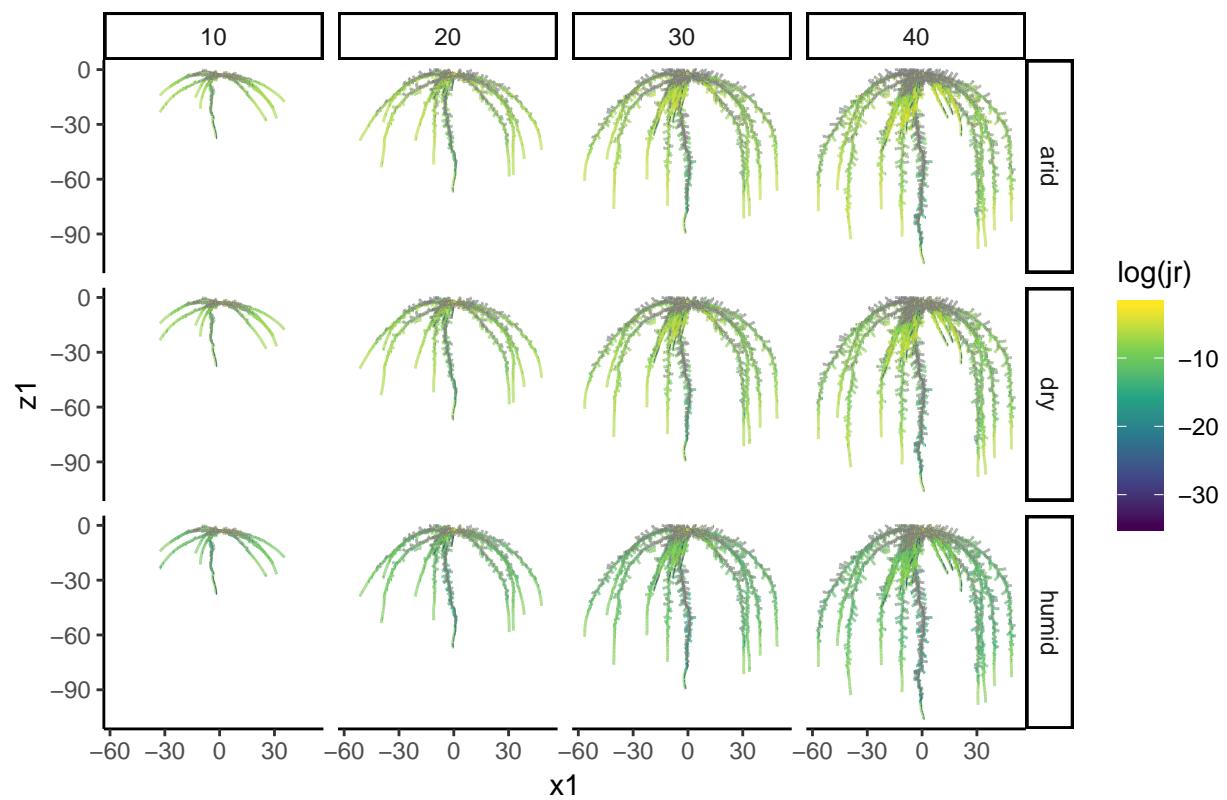


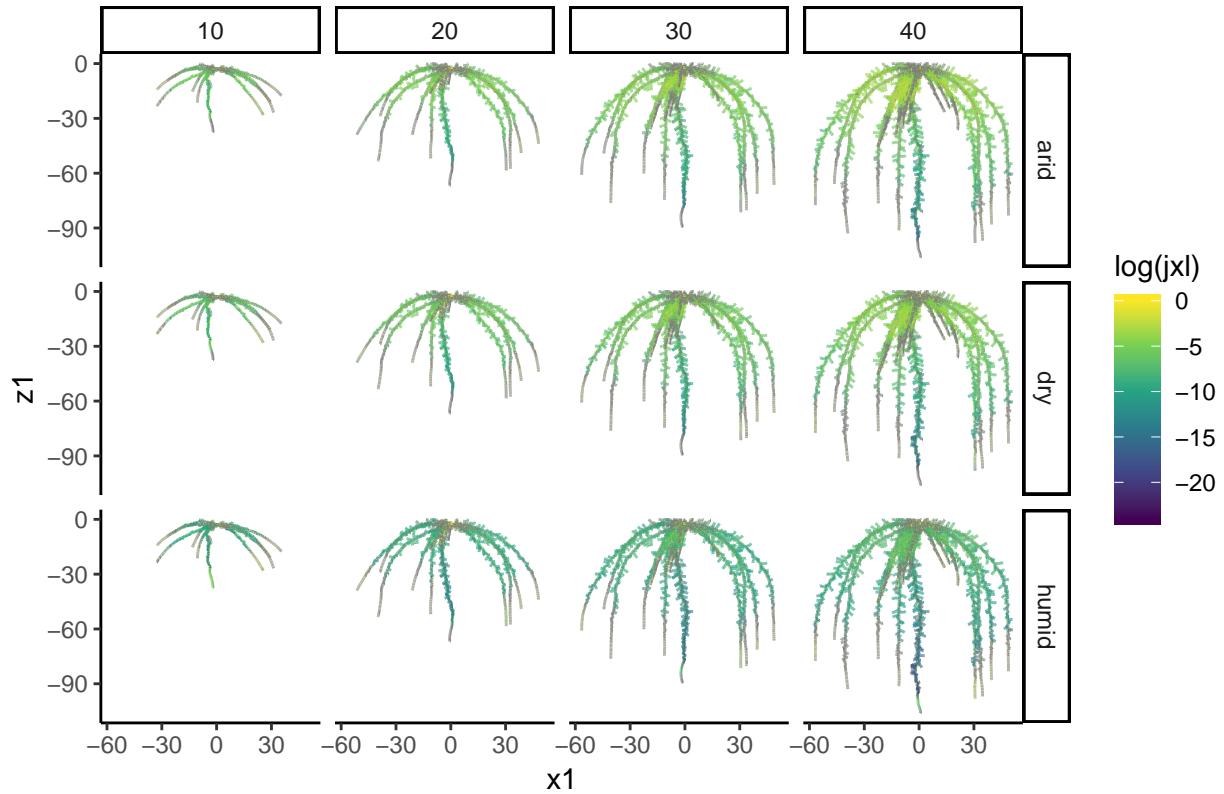
```
## # A tibble: 1 x 2
##   ex           high
##   <chr>        <dbl>
## 1 Zea_mays_1_Leitner_2010 0.515
```



```
## # A tibble: 1 x 2
##   ex           high
##   <chr>        <dbl>
## 1 Zea_mays_4_Leitner_2014 0.443
```







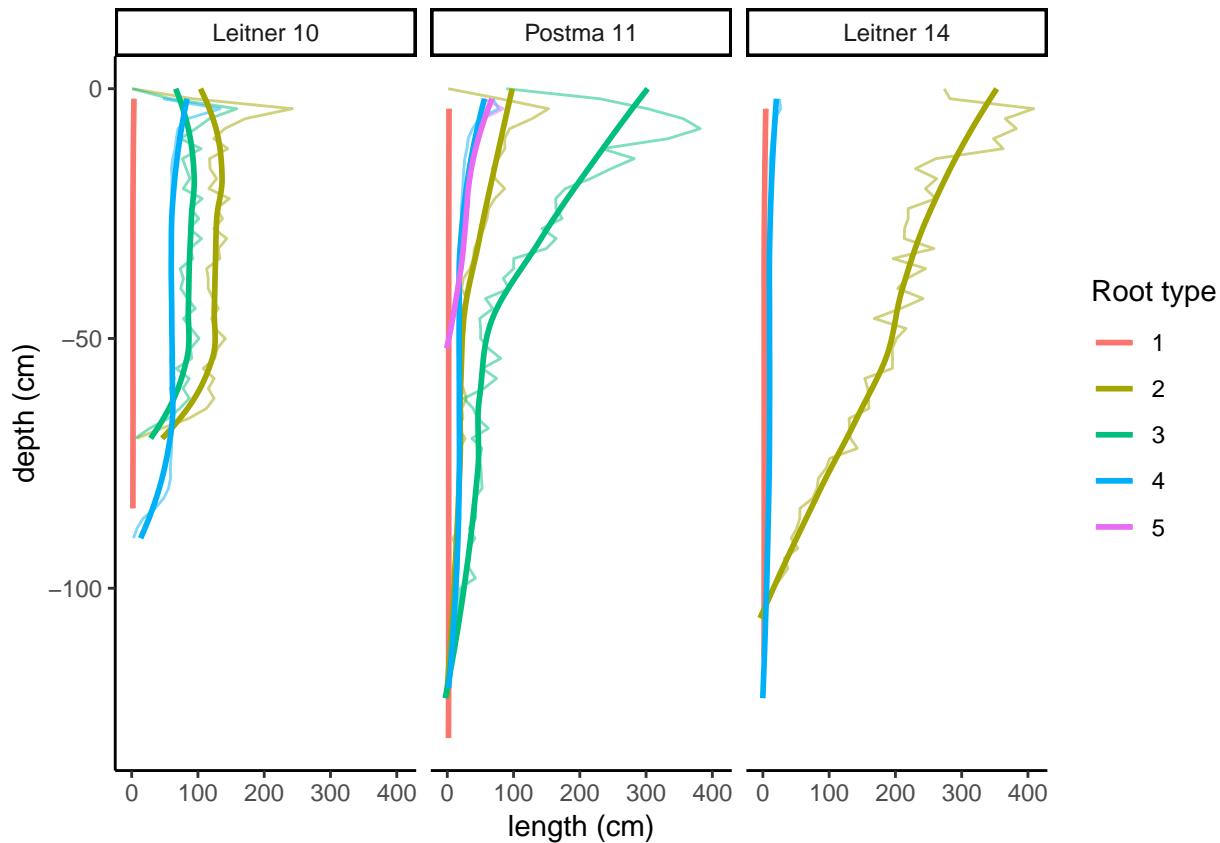
```

dens <- new_all_roots%>%
  mutate(age = as.numeric(age))%>%
  group_by(type, age, rz1 = round(z1/2)*2, scenario, ex, humid)%>%
  summarise(root = sum(length),
            su = sum(suf),
            su1 = sum(suf1),
            j = sum(jr),
            jx = sum(jxl),
            p = sum(psi),
            kr = sum(kr))

dens%>%
  filter(age == 60,
         scenario == min(scenario),
         humid == min(humid))%>%
  ggplot(aes(rz1, root, colour=factor(type))) +
  geom_line(alpha = 0.5) +
  geom_smooth(se=F) +
  theme_classic()+
  coord_flip() +
  xlab("depth (cm)") +
  ylab("length (cm)")+
  facet_wrap(~ex, labeller = as_labeller(babbel))+
  labs(colour = "Root type")

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'

```

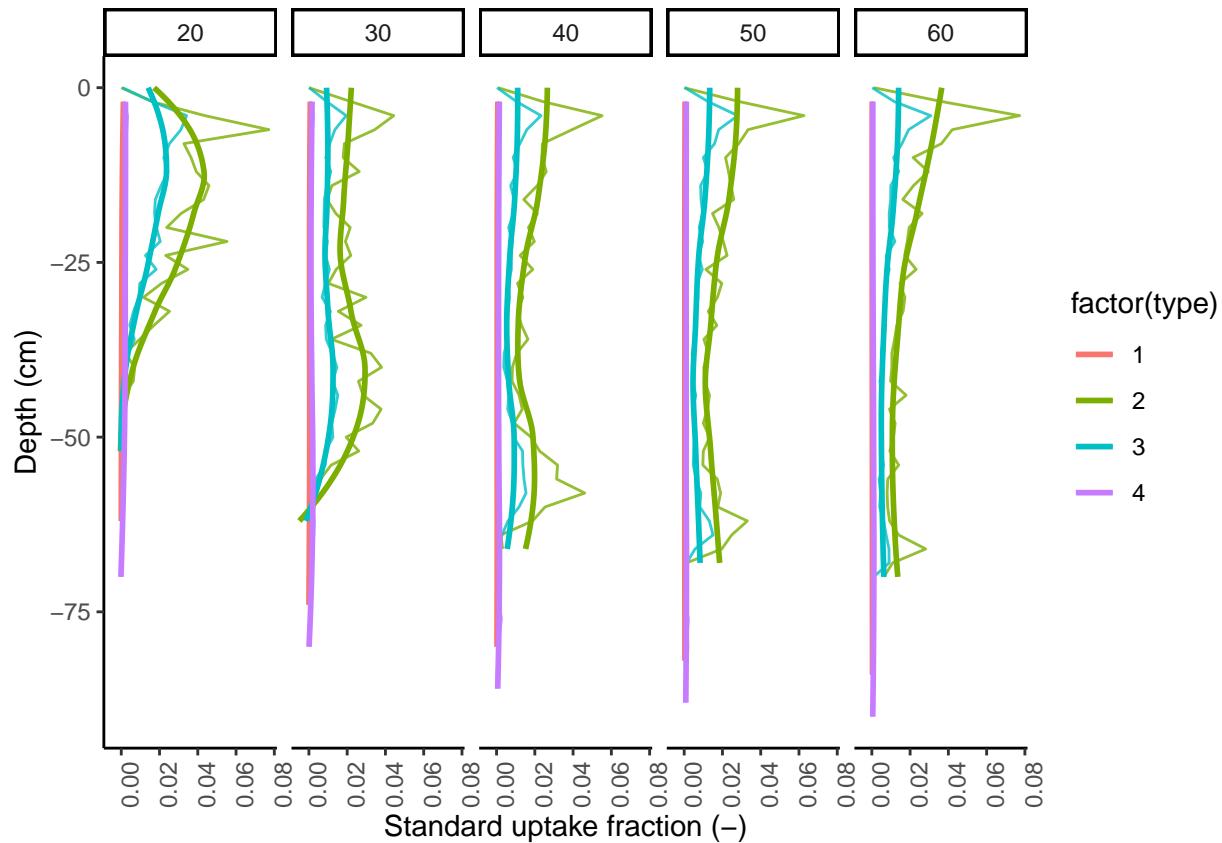


```

dens%>%
  filter(ex == unique(dens$ex)[1] ,
        scenario == 1,
        age > 10)%>%
  ggplot(aes(rz1, su1, colour=factor(type)))+
  geom_line(alpha = 0.8) +
  geom_smooth(se=F, alpha = 0.1) +
  coord_flip() +
  theme_classic()+
  xlab("Depth (cm)") +
  ylab("Standard uptake fraction (-)")+
  #labs(colour = "Root type")+
  theme(axis.text.x=element_text(angle=90)) +
  facet_grid(~age, labeller = as_labeller(babbel))

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'

```

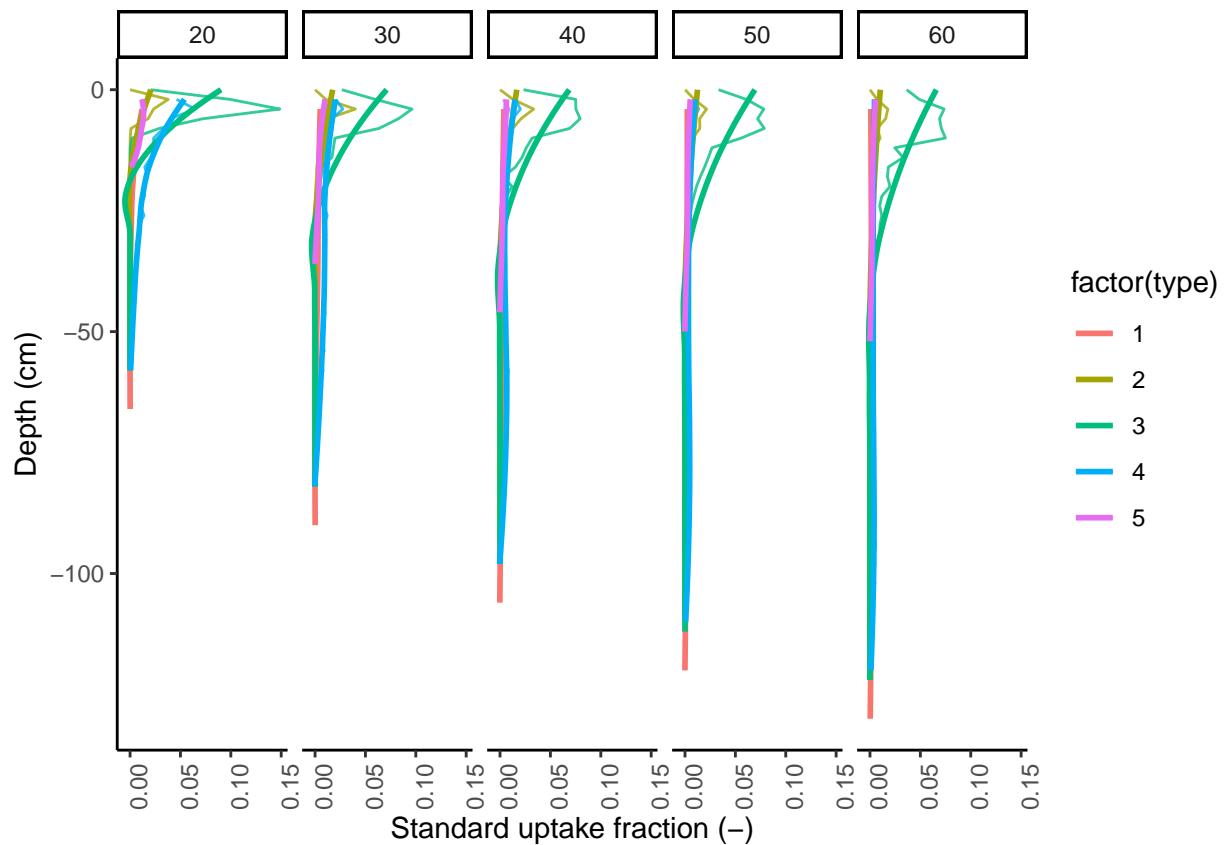


```

dens%>%
  filter(ex == unique(dens$ex)[2],
        scenario == 1,
        age > 10)%>%
  ggplot(aes(rz1, su1, colour=factor(type))) +
  geom_line(alpha = 0.8) +
  geom_smooth(se=F, alpha = 0.1) +
  coord_flip() +
  theme_classic()+
  xlab("Depth (cm)") +
  ylab("Standard uptake fraction (-)")+
  theme(axis.text.x=element_text(angle=90)) +
  facet_grid(~age)

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'

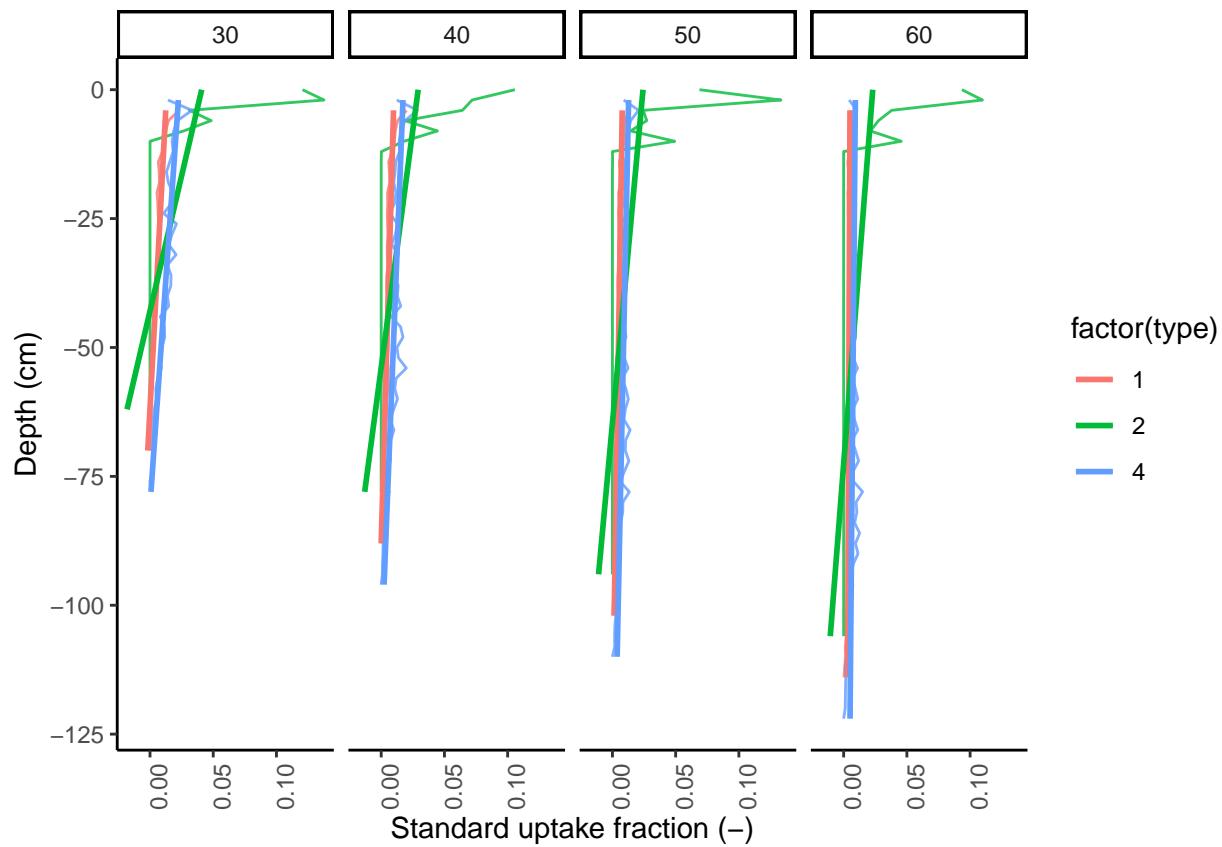
```



```

dens%>%
  filter(ex == unique(dens$ex)[3],
        scenario == 1,
        age > 20)%>%
  ggplot(aes(rz1, su1, colour=factor(type))) +
  geom_line(alpha = 0.8) +
  geom_smooth(se=F, alpha = 0.1, method = "gam") +
  coord_flip() +
  theme_classic()+
  xlab("Depth (cm)") +
  ylab("Standard uptake fraction (-)")+
  theme(axis.text.x=element_text(angle=90)) +
  facet_grid(~age)

```

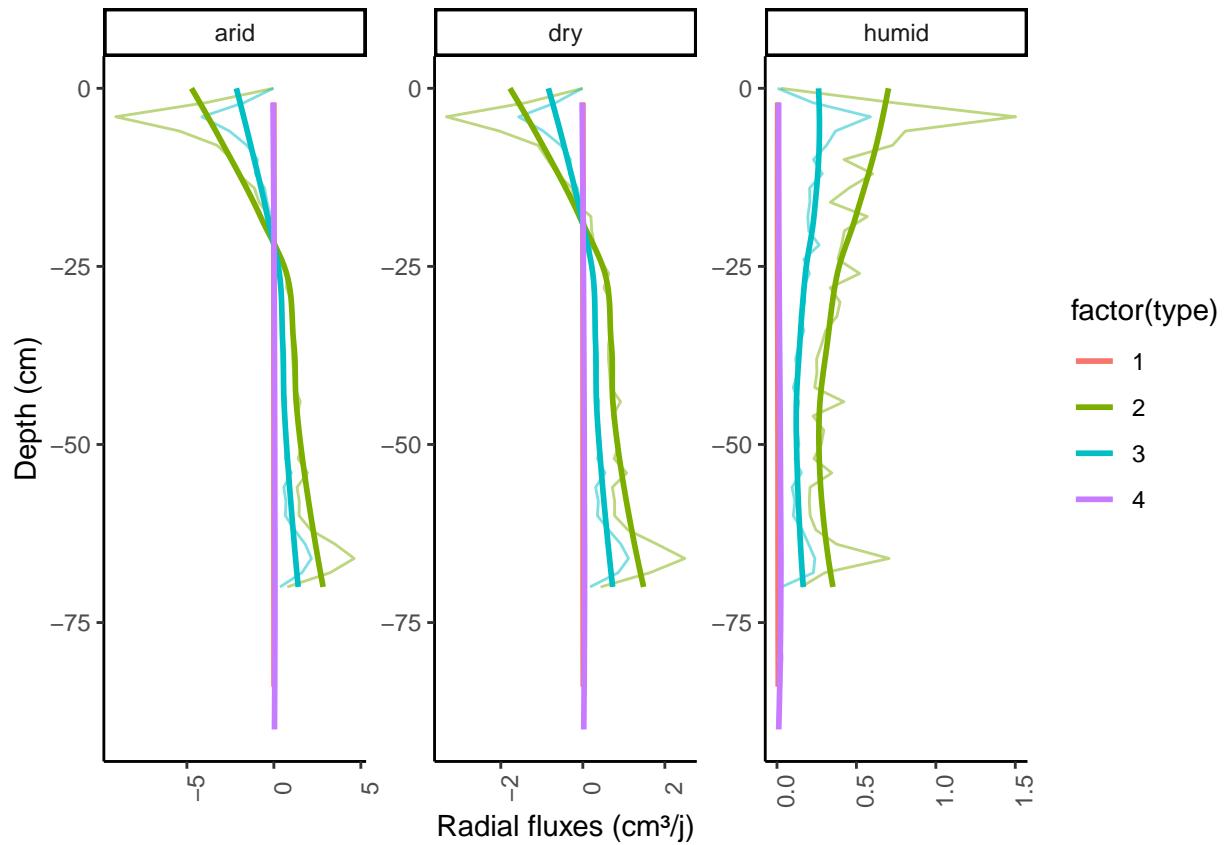


```

dens%>%
  filter(age == 60,
         scenario == 1,
         ex == unique(dens$ex)[1])%>%
  ggplot(aes(rz1, j, colour=factor(type))) +
  geom_line(alpha = 0.5) +
  geom_smooth(se=F) +
  coord_flip() +
  theme_classic()+
  xlab("Depth (cm)") +
  ylab("Radial fluxes (cm³/j)")+
  theme(axis.text.x=element_text(angle=90)) +
  facet_wrap(~humid, scales = "free", labeller = as_labeller(babbel))

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'

```

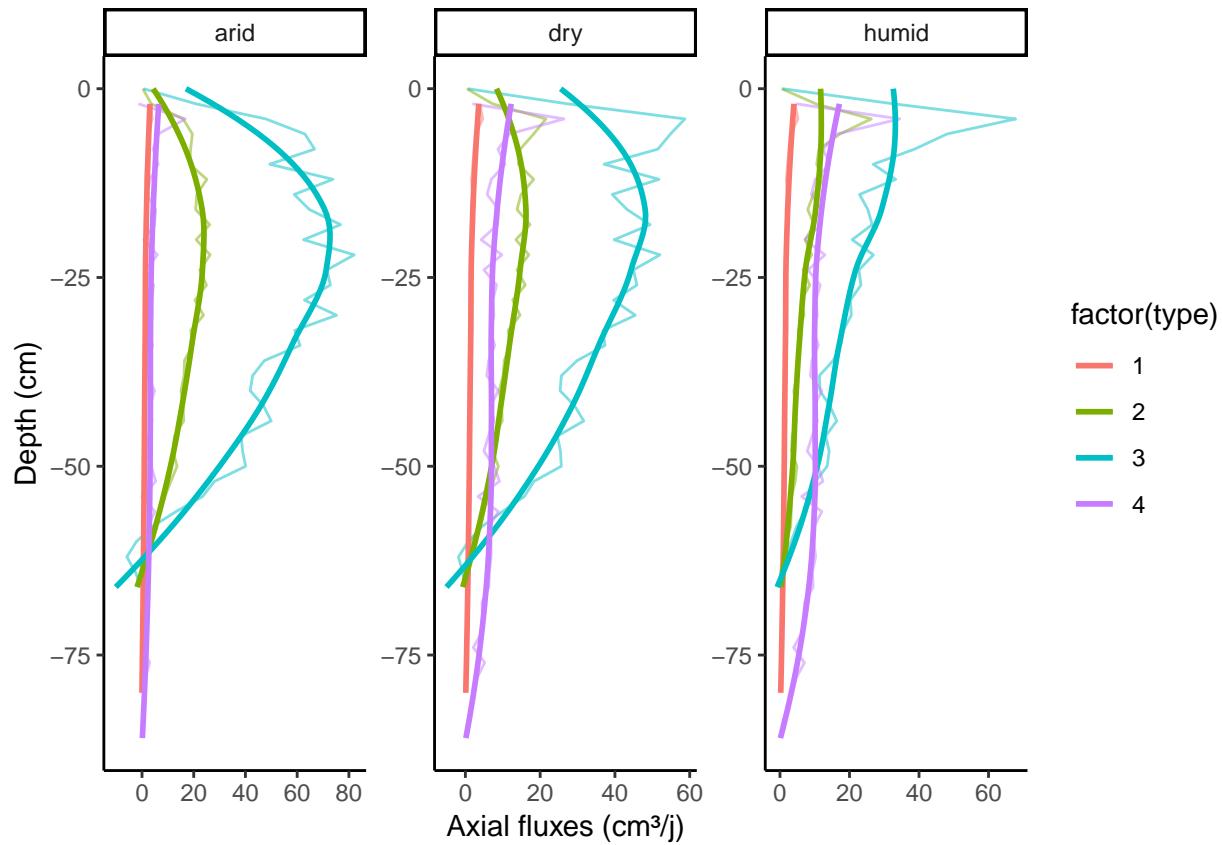


```

dens%>%
  filter(age == 40,
         scenario == 1,
         ex == unique(dens$ex)[1])%>%
  ggplot(aes(rz1, jx, colour=factor(type))) +
  geom_line(alpha = 0.5) +
  geom_smooth(se=F) +
  coord_flip() +
  theme_classic()+
  xlab("Depth (cm)") +
  ylab("Axial fluxes (cm³/j)")+
  facet_wrap(~humid, scales = "free", labeller = as_labeller(babbel))

```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

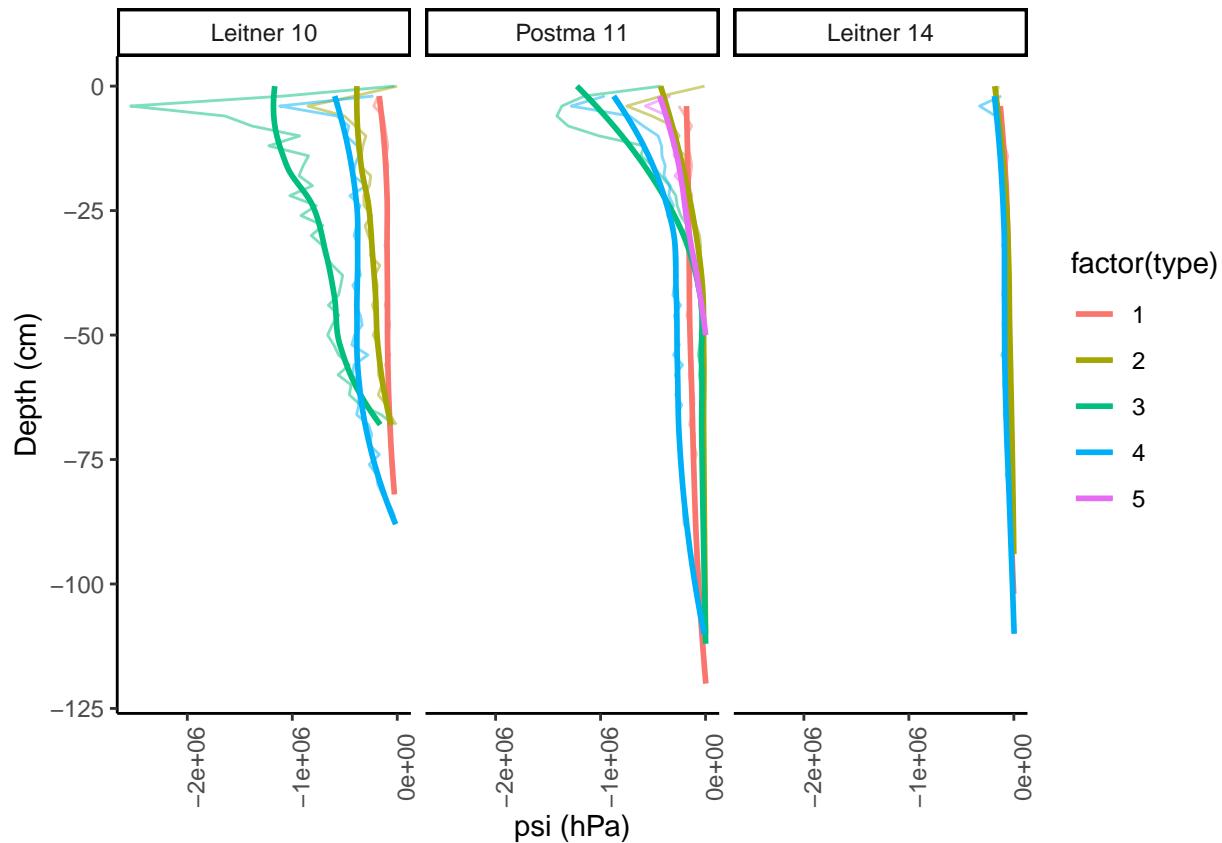


```

dens%>%
  filter(age == 50,
         humid == unique(dens$humid)[3],
         scenario == 2)%>%
  ggplot(aes(rz1, p, colour=factor(type))) +
  geom_line(alpha = 0.5) +
  geom_smooth(se=F) +
  coord_flip() +
  theme_classic()+
  xlab("Depth (cm)") +
  ylab("psi (hPa)")+
  theme(axis.text.x=element_text(angle=90)) +
  facet_wrap(~ex, labeller = as_labeller(babbel))

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'

```



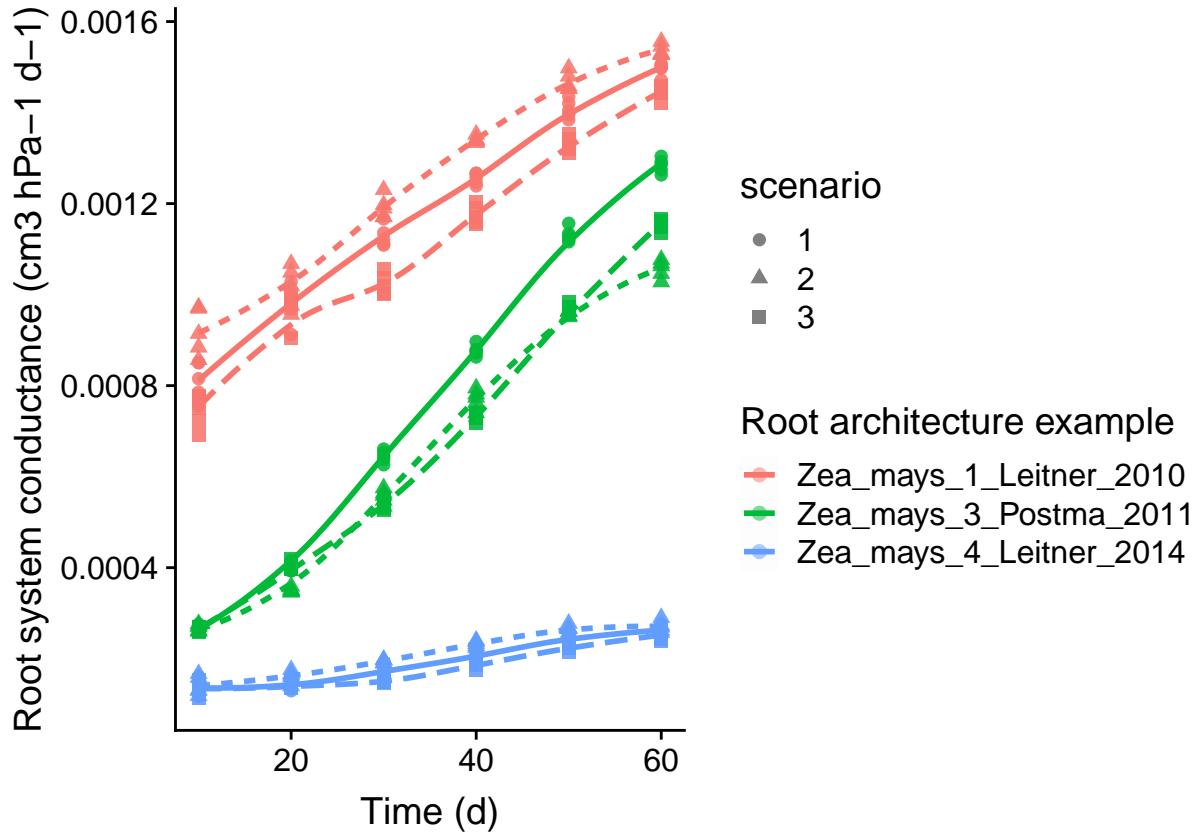
```
results%>%
  ggplot(aes(x= age, y = krs, colour = factor(ex)))+
  geom_point(aes(shape = factor(scenario)),size = 2, alpha = 0.5)+  

  geom_smooth(aes(linetype = factor(scenario)), alpha = 0.02)+  

  xlab("Time (d)") +  

  ylab("Root system conductance (cm3 hPa-1 d-1)")+
  labs(colour = "Root architecture example",
       shape = "scenario")+
  guides(linetype = F)

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

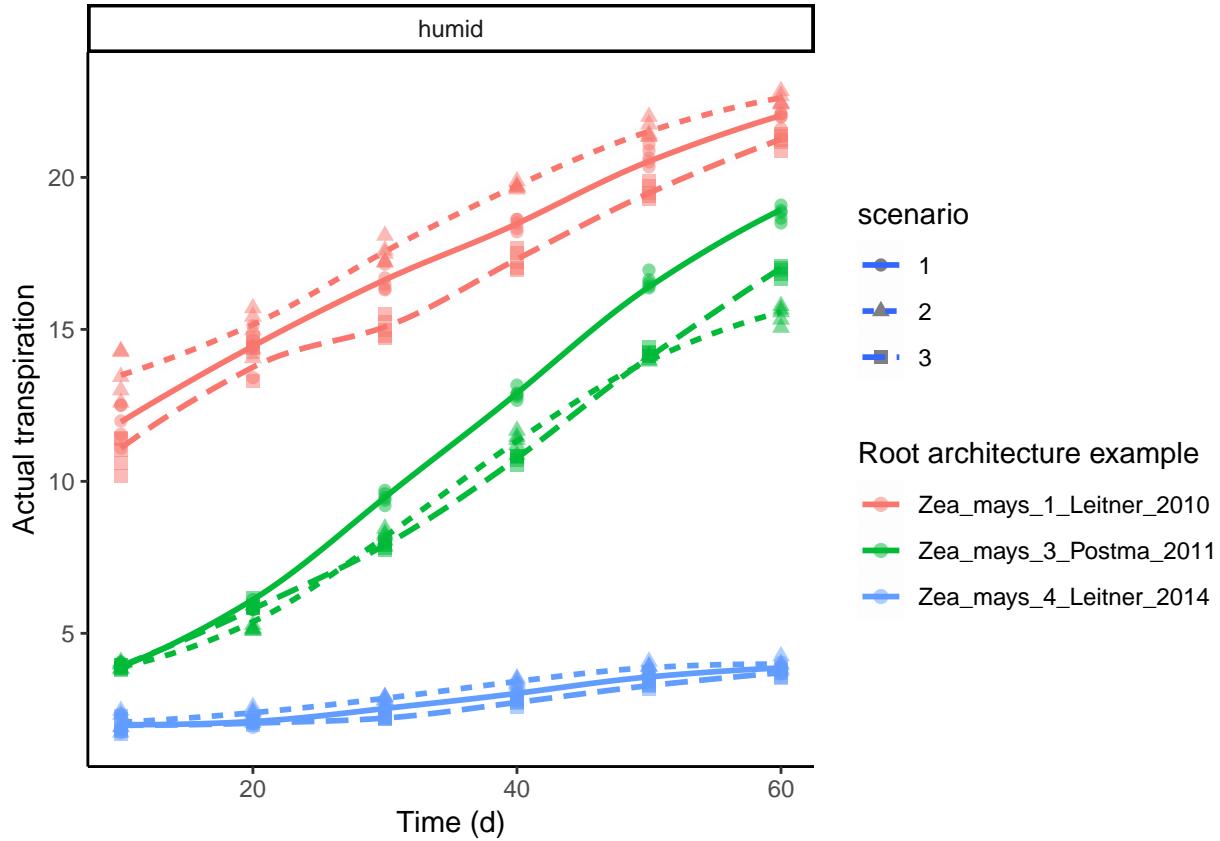


```
#facet_grid(~scenario)

Transpiration<- results%>%
  mutate(humid = as.character(humid))%>%
  group_by(age, rep, scenario, humid, ex)%>%
  summarise(Tpot = max(tpot, tact),
            Tact = min(tpot, tact))

Transpiration%>%
  filter(humid == "-300")%>%
  ggplot(aes(x= age, y = Tact, colour = factor(ex))) +
  geom_point(aes(shape = factor(scenario)), size = 2, alpha = 0.5) +
  geom_smooth(aes(x = age, y = Tpot, colour = factor(ex),
                  linetype = factor(scenario)), alpha = 0.01) +
  theme_classic() +
  xlab("Time (d)") +
  ylab("Actual transpiration") +
  labs(colour = "Root architecture example") +  labs(colour = "Root architecture example",
    shape = "scenario",
    linetype = "scenario") +
  #guides(linetype = F) +
  facet_wrap(~humid, scales = "free", labeller = as_labeller(babbel))

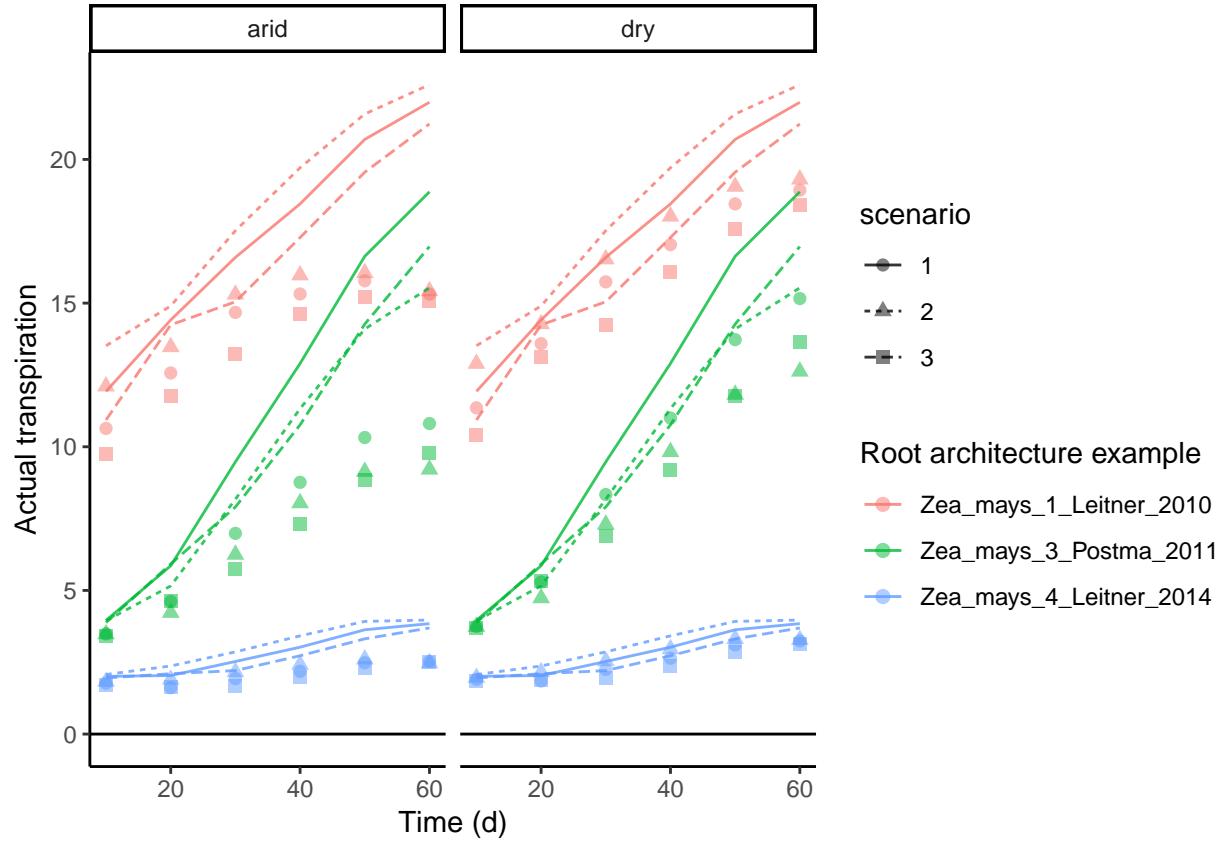
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



The actual transpiration under drought whith two scenarion of conductivities

```
Transpiration%>%
  filter(humid %in% c("-7000", "-15000"))%>%
  group_by(ex, humid, age, scenario)%>%
  summarise(Tact = mean(Tact),
            Tpot = mean(Tpot))%>%

ggplot(aes(x= age, y = Tact, colour = factor(ex)))+
  geom_point(aes(shape = factor(scenario)),size = 2, alpha = 0.5)+
  geom_line(aes(x = age, y = Tpot, colour = factor(ex),
                linetype = factor(scenario)), alpha = 0.8)+
  geom_hline(yintercept = 0)+
  theme_classic()+
  xlab("Time (d)") +
  ylab("Actual transpiration")+
  labs(colour = "Root architecture example")+
  labs(colour = "Root architecture example",
       shape = "scenario",
       linetype = "scenario")+
  facet_grid(~humid, labeller = as_labeller(babbel))
```



The different line represent a potential transpiration.

4 Reference