

Getting a Functionnal-Structural Plant Model into a comprehensive R pipeline:

MARSHAL-pipeline

Adrien Heymans, Felicien Meunier, and Guillaume Lobet

décembre, 2018

Introduction

MARSHAL, stand for MAize Root System Hydraulic Architecture soLver. It combines the root architecture model CRootBox (Schnepf et al. 2018) with the method for solving water flow in Root System Hydraulic Architecture (RSHA) of Meunier et al. (2017) with the method for computing macroscopic parameter of Couvreur et al. (2012). MARSHAL calculates root system conductance, 1 to 3-D uptake distribution and other upscaled variables (plant leaf water potential or transpiration) for any combination of structural and functional traits.

How to use MARSHAL in the R environment

To run this example, the marshal package is required. This package can be found on GitHub. Alternatively, by simply executing the following line in the R environment:

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

MARSHAL is built upon a handful of other packages that need to be loaded. One additional set of function has to be loaded manually.

```
library(data.table)
library(Matrix)
library(tidyverse)
library(readxl)
library(cowplot)
library(xml2)
library(knitr)
`%!in%` <- compose(`!`, `!in%`)
source("R/io_function.R")
```

Boundary conditions of MARSHAL.

The inputs of MARSHAL are:

- The hydraulic properties of the roots.
- The water potential of the soil.
- The root system architecture (RSA).
- The initial pressure strength that pulls water out of the collar.

To illustrate the potential of MARSHAL, a total of 27 scenarios were created. They were made from three RSA examples, three different hydraulics properties of the roots, and three vertical distributions of the soil water potential ($3 \text{ soil} \times 3 \text{ RSA} \times 3 \text{ hydraulic conductivities}$).

Hydraulic properties of the root system

The hydraulic conductivities of the root system can be found in the literature, by measuring them on root samples or with the help of modelling tools. Here, the hydraulic conductivities of the root come from the Doussan et al. paper (1998). To show some possibilities, three scenarios were built from this Doussan et al. table of conductivities (figure 1):

1. The first scenario comes from the Doussan et al. table of conductivities placed along the root axes.
2. The second scenario is based on the first, but the maturity of the root anatomy is moved 5 cm towards the apex.
3. The third is the opposite of the second, the maturity of the root anatomy is moved 5 cm backwards.

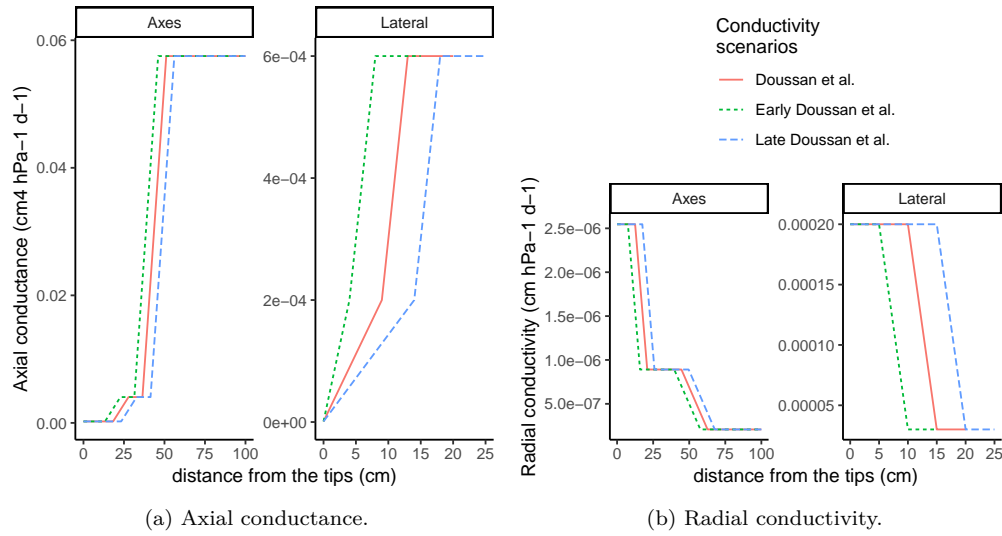


Figure 1: Conductivities properties on axes and lateral root type.

Root system architecture with CRootBox

Modelling RSA can be achieved with CRootBox (Schnepf et al. 2018). For a couple of species, some examples of the parameter files are details in the *modelparameter* folder, which is linked to the CRootBox repository.

For the purpose of the example, three parameter files of maize were chosen:

- *Zea_mays_3_Postma_2011*
- *Zea_mays_1_Leitner_2010*
- *Zea_mays_4_Leitner_2014*

The following lines of code allows creating five unique RSAs for each parameter file. It computes the root growth during 60 days. Every ten days, the code loads the architecture information in a data table. If needed, the time resolution can be increased inside the CRootBox implementation. Within the implementation of this example, the root growth has been restricted to the below ground level for a more realistic point of view. Furthermore, some segments were not build up to the whole root system because they were shorter than the spatial resolution of the root system.

```
# Number of repetition per RSA example
nrep <- 5
```

```

# List of three CRootBox parameter files 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 use
  write_rparam(rparam, "./www/param.rparam")
  write_pparam(pparam, "./www/param.pparam")

  for(i in c(1:nrep)){
    #####
    # Run CRootBox #
    #####
    if(Sys.info()['sysname'] == "Darwin"){
      system("chmod 777 www/a-mac.out")
      system("www/a-mac.out")
    }
    else if(Sys.info()['sysname'] == "Windows"){
      system("www/a.exe")
    }
    else if(Sys.info()['sysname'] == "Linux"){
      system("chmod 777 www/a.out")
      system("www/a.out")
    }
    else{
      message("An error occurred, which depended of your
operating system. If so, please compile CRootBox on
your device. The instruction can be found:
https://github.com/MARSHAL-ROOT/marshall-pipeline/tree/master/17\_06%20CRootBox")
    }

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

    for(f in fls){
      temp <- fread(f, header = T)
      temp$age <- strsplit(f, "_")[[1]][1]
      temp$rep <- i
      temp$ex <- param_name
      # Save the temporary data in a permanent file to use.
      all_roots <- rbind(all_roots, temp)
    }
  }
}

# Saving the root system information in a .csv
write_csv(all_roots, "all_roots.csv")

```

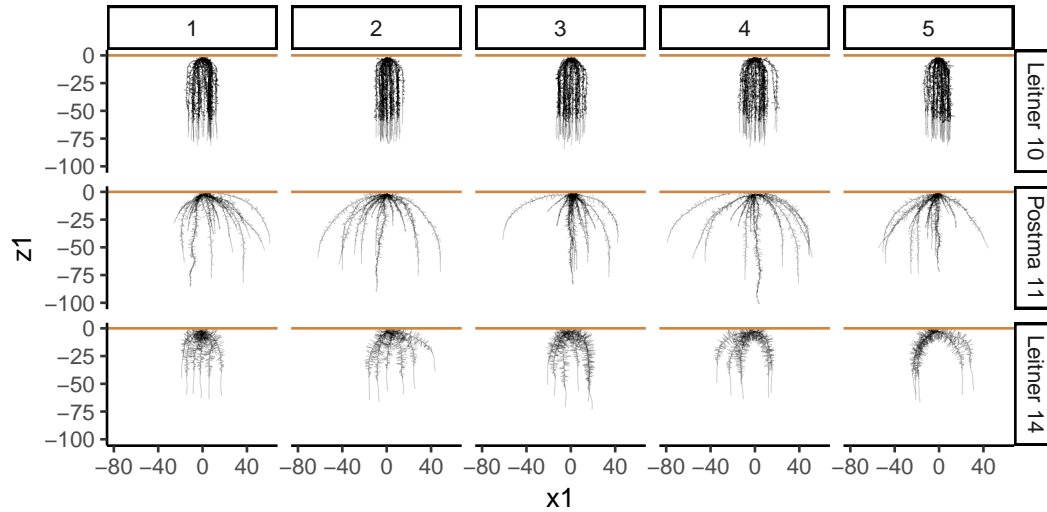


Figure 2: Projection in two dimensions of the different root systems computed with CRootBox at the age of 30 days

```
# To gain time once you have a all_roots.csv file created
# Loading the .csv file can be achieve with:
all_roots <- read.csv("all_roots.csv")
```

Architectural phenes

Once the complete RSA is computed, there are a number of architectural phenes that can be accessed, such as:

- the area of the convex hull
- the rooting depth
- the root length density along the soil profile
- the number of individual roots per root system
- the total length of the root system
- ...

The convex hull

The convex hull area is a classical root system descriptor. It gives qualitative information about root exploration.

```
root_20 <- all_roots%>%
  filter(age %in% c("20", "40", "60"))
pol_ch = 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 sides
        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]
```



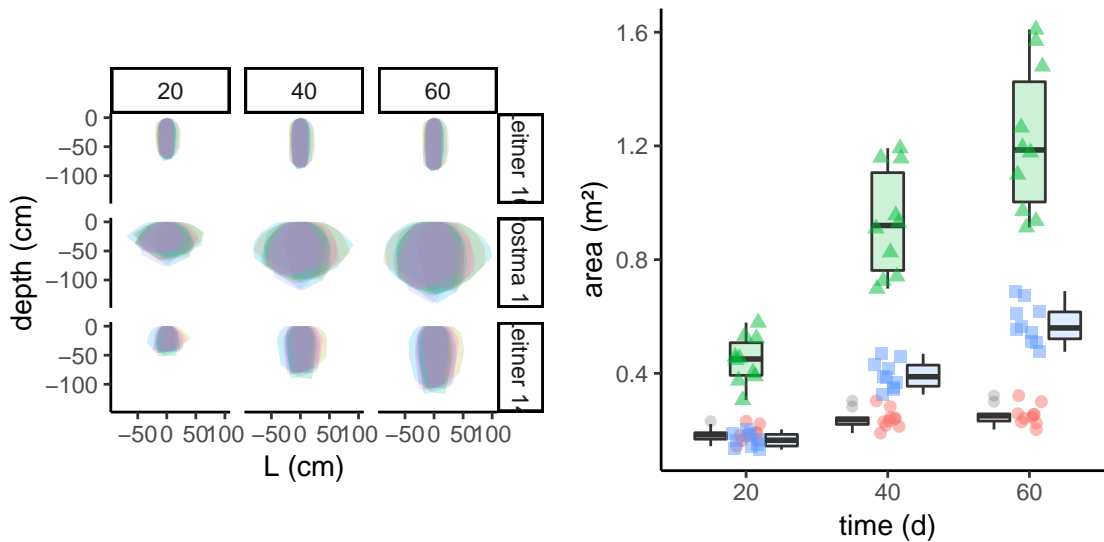
```

x <- matrix(c(first_col, tmp$z2), nc = 2)
# Get the indices of the unique points lying on the convex hull
ch <- chull(x= first_col, y = tmp$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)%>% # Transformation into square meter
  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)
}
}
}
}

```



(a) Superposition of all the convex hull repetitions.

(b) The area of each convex hull.

Figure 3: Area of the convex hull for the different RSA examples. \triangle = Postma 2011, \square = Leitner 2014, and \bigcirc = Leitner 2010 RSA example

The maximal rooting depth

Thanks to the detailed information coming from CRootBox, it is possible to measure some root phenes along the z axis. The maximal rooting depth is an interesting phenotypic value. This can be used in other models such as Feddes et al. (1978) to estimate root water uptake.

```

# Summarise the total root length and the rooting depth information.
RLD <- all_roots%>%

```

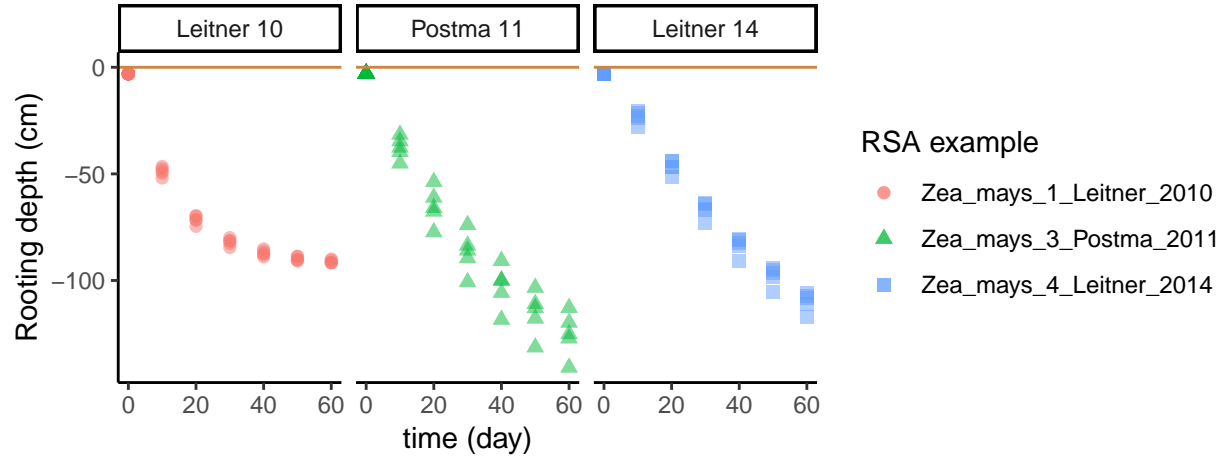


Figure 4: Rooting depth of the different root system examples

```
mutate(age = as.numeric(age))%>%
group_by(age, rep, ex)%>%
summarise(root = sum(length),
           rooting_depth = min(z2))

RLD2 <- all_roots%>%
mutate(age = as.numeric(age))%>%
group_by(age, rz1 = round(z1/2)*2, # on every two cm of the profile
ex)%>%
summarise(root = sum(length)/nrep)
```

Root length density

Root length density is a major phenotypic trait measured in fields when below ground information is needed. In this case, the total root length is computed every ten days.

The water soil potential

Within the soil, the pressure head of the water potential is created from the next lines of code. The three soil scenarios (figure 6) were built from the following equation, and they differ from one another by the three different coefficients.

The drying function is:

$$\psi_i = \alpha \times \frac{(Z_i + L)^{10} \times T_i}{L^{10} \times T_{end}} - 230$$

- ψ_i = water potential in the soil at i cm below the surface.
- α = drying coefficient of the soil (-300 = “humid”, -7000 = “dry” and -15.000 = “arid”).
- Z_i = the depth (cm)
- L = the maximal soil profile depth.
- T = the simulation time (day)
- -230 is the water potential in the soil at the field capacity for a silty soil.

```
# Take the same time variable as the one that were implemented in CRootBox.
full_time <- sort(unique(as.numeric(all_roots$age)))
time_step = full_time[2]-full_time[1]
endTime = max(full_time)
```

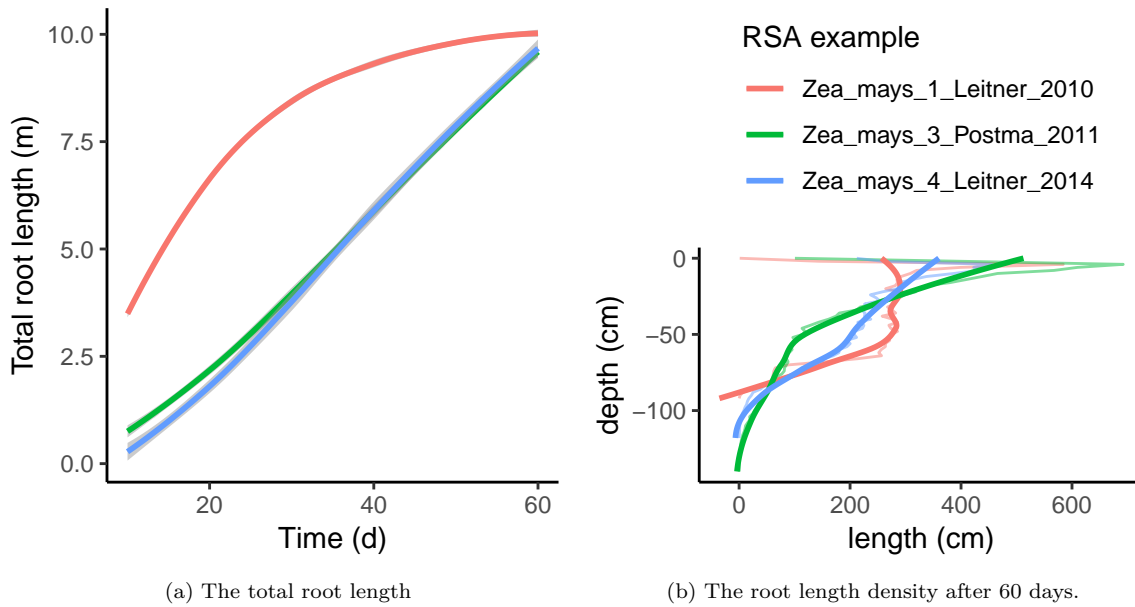


Figure 5: The root length

```
ntimesteps = endTime/time_step

# Creating some values for the water potential in the soil along the profile depth.
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
# Making three scenarios
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)
}
```

The water potential at the collar

Expressed in hecto-Pascal, the pressure head defines the transpiration strength that pulls the water out of the root system. The default value is -15.000 hPa

```
# Water potential at the collar (hPa)
tpots <- -15000
```

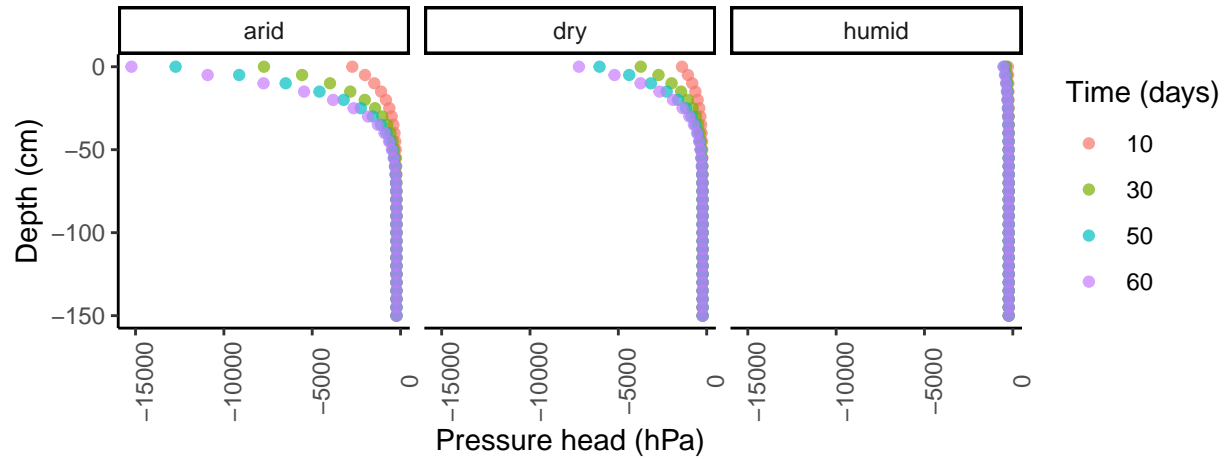


Figure 6: Water potential evolution during the simulation time along the soil profile

MARSHAL

MARSHAL provides an estimation of the conductance of the whole root system as well as the actual and the potential transpiration, based on the soil profile. The MARSHAL function *getSUF* is called to solve the hydraulic part of the root system for all the scenarios at each time step of the process.

```
new_all_roots <- NULL
results<- NULL
k <- 1
# Computing the root system hydraulic architecture
# Loop to run MARSHAL with the selected simulation parameters.
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

          # 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))

# Keep the information about the very detail hydrological parameter on
# all the root segment only for two plant repetitions per simulation.
if(re >= nrep-1){
  # Format dataset to be compatible with MARSHAL output
  first <- temp_root[temp_root$node1ID == 0,]
  nodals_ids <- unique(temp_root$branchID[temp_root$type == 4 |
                                          temp_root$type == 5])
  for(no in nodals_ids){
    temp <- temp_root[temp_root$branchID == no]
    temp <- temp[1,]
    connection <- data.frame(node1ID = 0,
                             node2ID = temp$node1ID,
                             branchID = temp$branchID,
                             x1 = first$x1, y1 = first$y1, z1 = first$z1,
                             x2 = temp$x1, y2 = temp$y1, z2 = temp$z1,
                             radius = temp$radius,
                             length = sqrt((first$x1-temp$x1)^2 +
                                             (first$y1-temp$y1)^2 +
                                             (first$z1-temp$z1)^2 ),
                             R = 0, G = 0, B = 0,
                             time = temp$time,
                             type = temp$type,
                             age = temp$age,
                             rep = temp$rep)
    new_table = rbind(temp_root, connection)
    temp_root = new_table
  }
  temp_root <- temp_root[order(temp_root$node2ID, decreasing = F),]

# Merge output of MARSHAL on specific root segment
temp_root$suf <- as.vector(hydraulics$suf)
temp_root$suf1 <- as.vector(hydraulics$suf1)
temp_root$kx <- as.vector(hydraulics$kx)

```

```

temp_root$kr <- as.vector(hydraulics$kr)
temp_root$j_r <- as.vector(hydraulics$j_r)
temp_root$psi <- as.vector(hydraulics$psi)
temp_root$j_xl <- as.vector(hydraulics$j_xl)
temp_root$psi_soil <- as.vector(hydraulics$psi_soil)
# Add the simulation specificity to the dataset
temp_root$scenario <- sen
temp_root$ex <- exa
temp_root$humid <- hum
temp_root$rep <- re
new_all_roots <- rbind(new_all_roots, temp_root)
}
k <- k + 1
}
}
}
}

# Save the results
write_csv(new_all_roots, "new_all_roots.csv")
# Load the saved results
new_all_roots <- read_csv("new_all_roots.csv")

```

Results:

Once the hydraulic part has been associated with each root segment and the macroscopic properties have been computed, they can be visualized in different ways (1 to 3-D).

Standard uptake fraction (SUF)

The Standard Uptake Fraction (SUF) is the relative distribution of root water uptake between root segments when water is equally available in space (units: %) (Couvreur, Vanderborgh, and Javaux 2012).

One of the major advantages of the SUF is to estimate where the segments that contribute to the water uptake of the plant should be. An interesting focus can be made on the smallest regions that contain 50% of the water uptake when the water soil potential is homogeneous (figure 8).

```

fif <- NULL
for (exa in 1:length(param_choise)) {
fi <- new_all_roots%>%
  filter(age == 60,
         rep == nrep,
         scenario == "Doussan et al.",
         humid == min(humid),
         ex == param_choise[exa])%>%
  mutate(or_suf = sort(suf1, decreasing = T))

i <- 1
k <- 0
while(k <= 0.5){
  k <- k + fi$or_suf[i]
  i <- i + 1
}

```

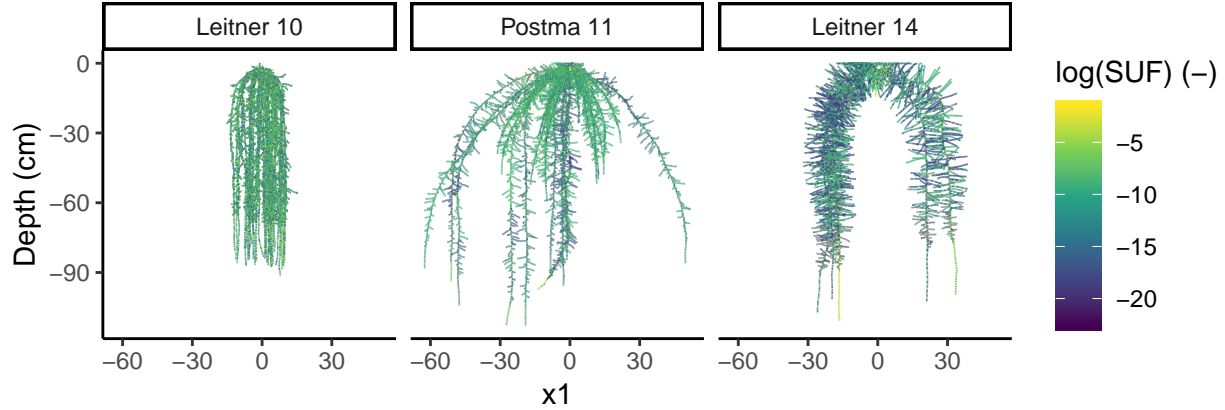


Figure 7: Standard uptake fraction of the different RSA example

```
fif <- rbind(fif, c(ex = param_choise[exa], suf = fi$or_suf[i]))%>%
  as.tibble()%>%
  mutate(suf = as.numeric(suf))
}
```

To look at the different hydraulic marco-properties along the z profile, the mean values of the variable total were taken every two centimeters.

```
# For the two repetitions on which we have values on each segment,
# we take the mean values of the sum of each variables.
dens <- new_all_roots%>%
  mutate(age = as.numeric(age))%>%
  group_by(type, age, rz1 = round(z1/2)*2, # on every two cm of the profile
    scenario, ex, humid)%>%
  summarise(root = sum(length)/2, # The mean of the sum on the two repetitions
    su = sum(suf)/2,
    sud = sum(suf/length)/2,
    su1 = sum(suf1)/2,
    j = sum(jr)/2,
    jx = sum(jxl)/2,
    p = sum(psi)/2,
    kr = sum(kr)/2)
```

```
dens%>%
  filter(age == 40,
    scenario == "Doussan et al.")%>%
  ggplot(aes(rz1, jx, colour=factor(type))) +
```

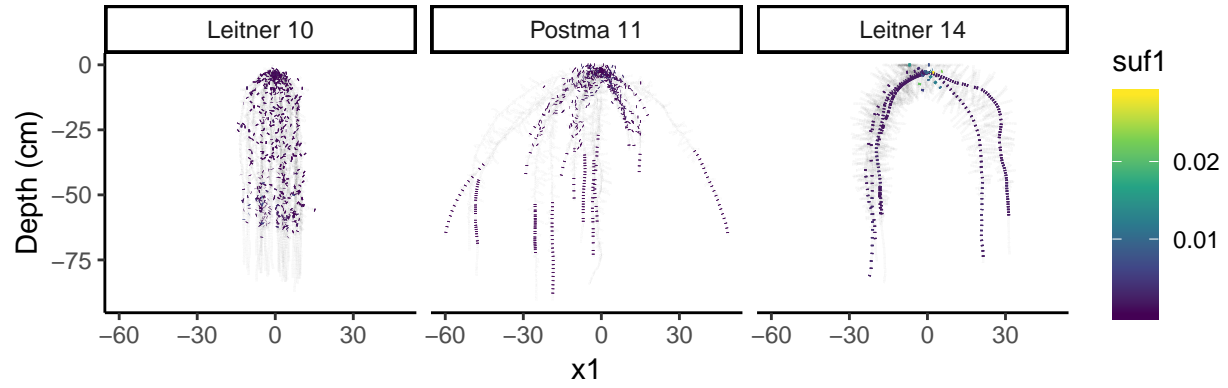


Figure 8: The smallest region on the root system hydraulic architecture that contains 50 % of the SUF

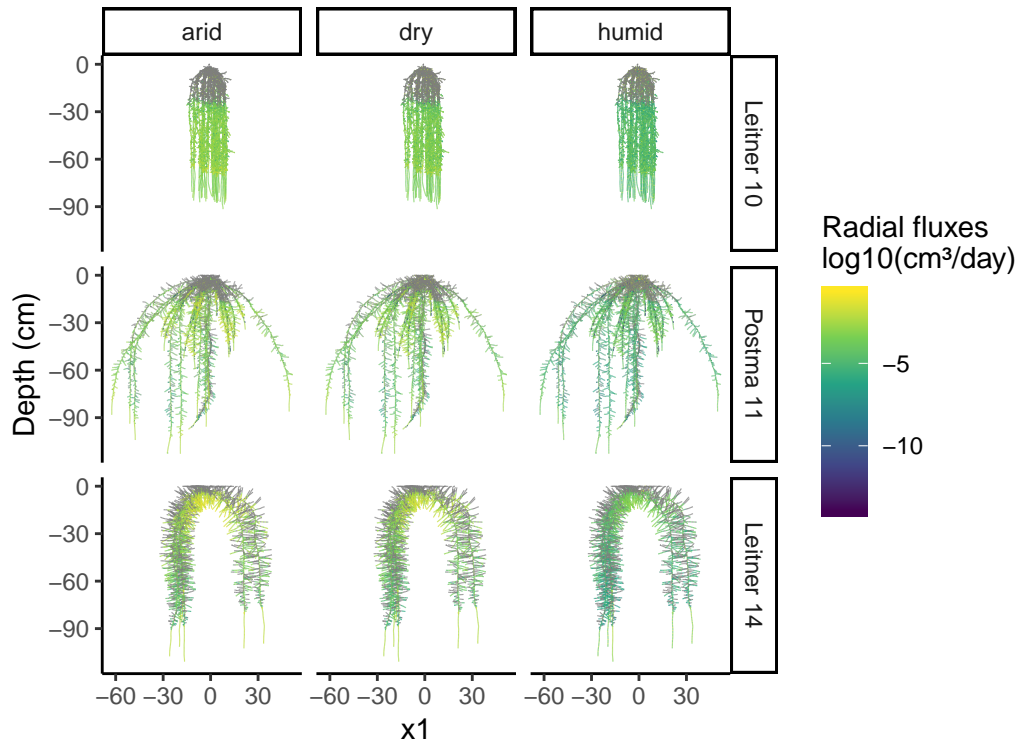


Figure 9: Radial fluxes, comparison between the RSA and the soil scenarios

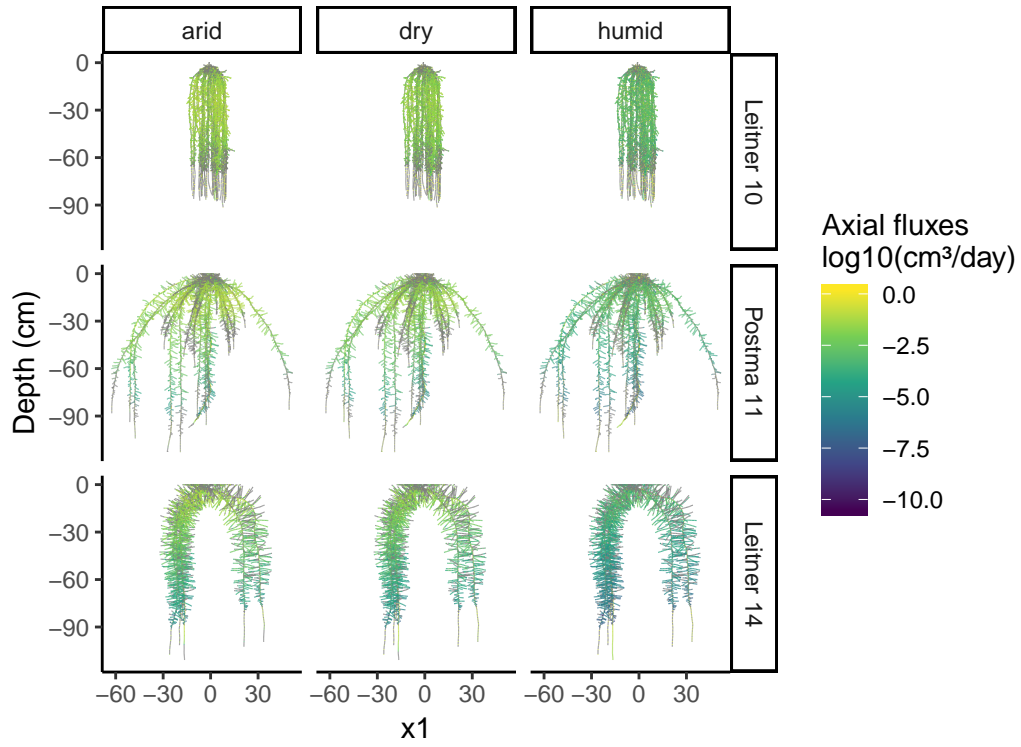


Figure 10: Axial fluxes, comparison between the RSA and the soil scenarios

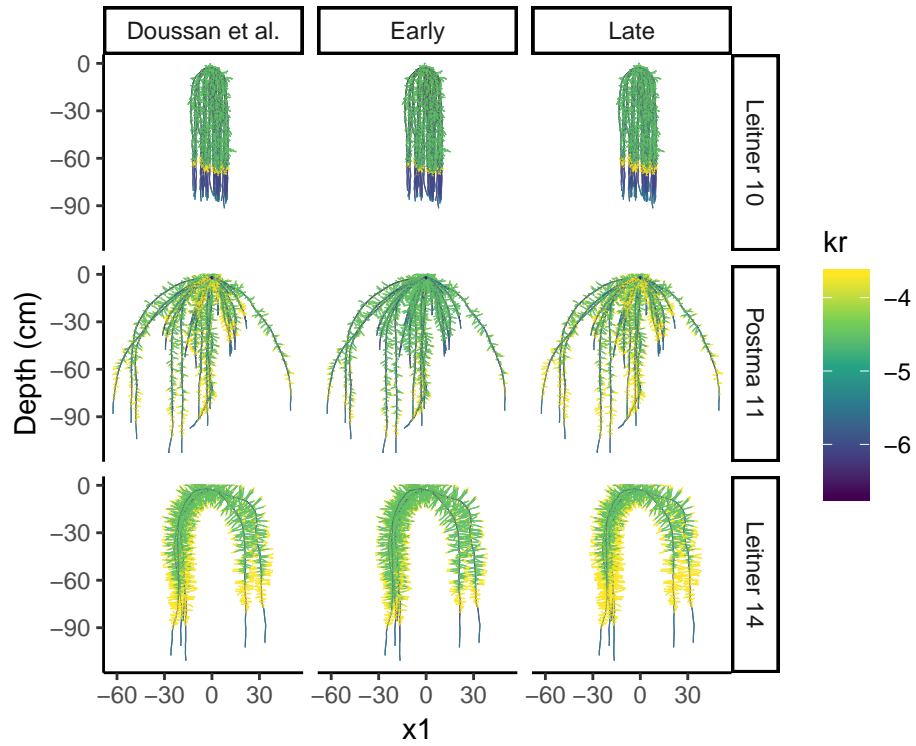


Figure 11: The radial conductivity (Kr) on root segment, comparison between the different conductivity scenarios and main RSA examples

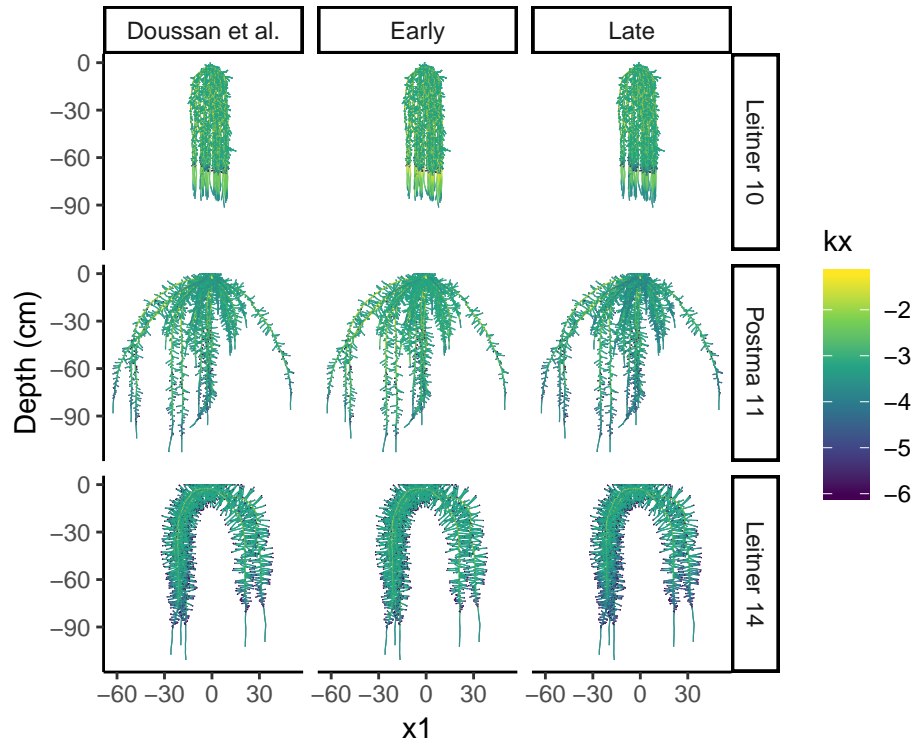


Figure 12: The axial conductance (K_x) on root segment, comparison between the different conductivity scenarios and main RSA examples

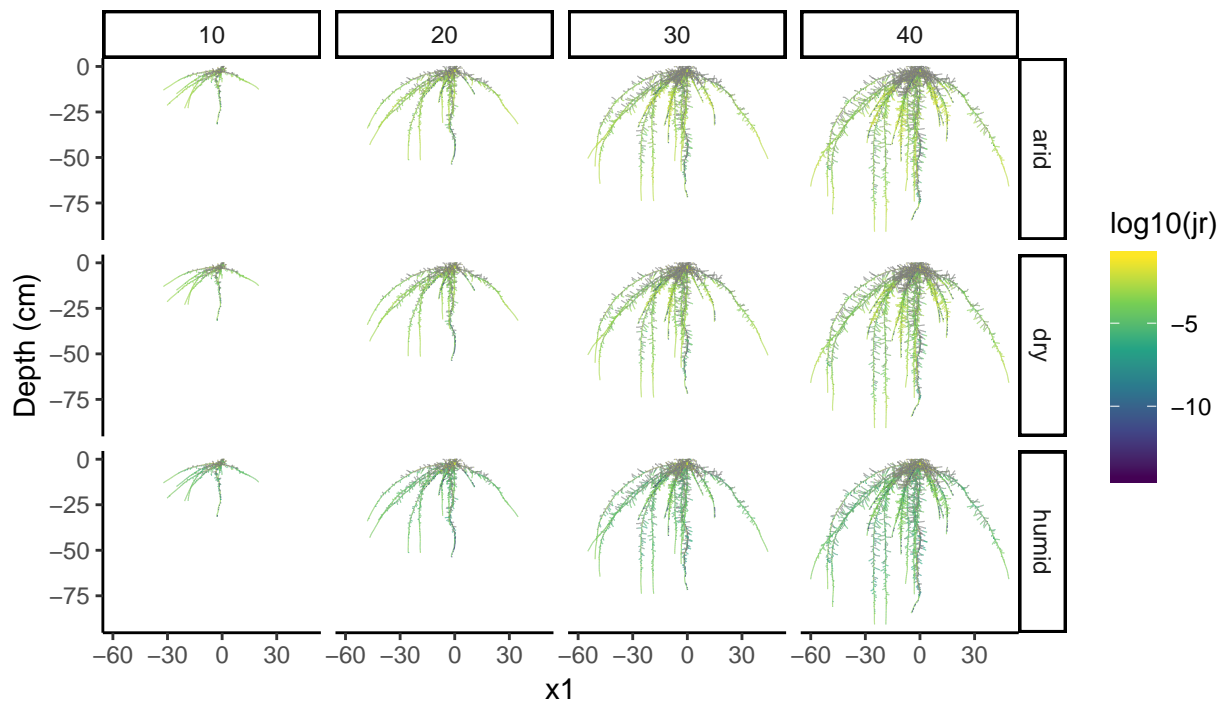


Figure 13: Absorption dynamics due to different soil scenarios on the Postma RSA example

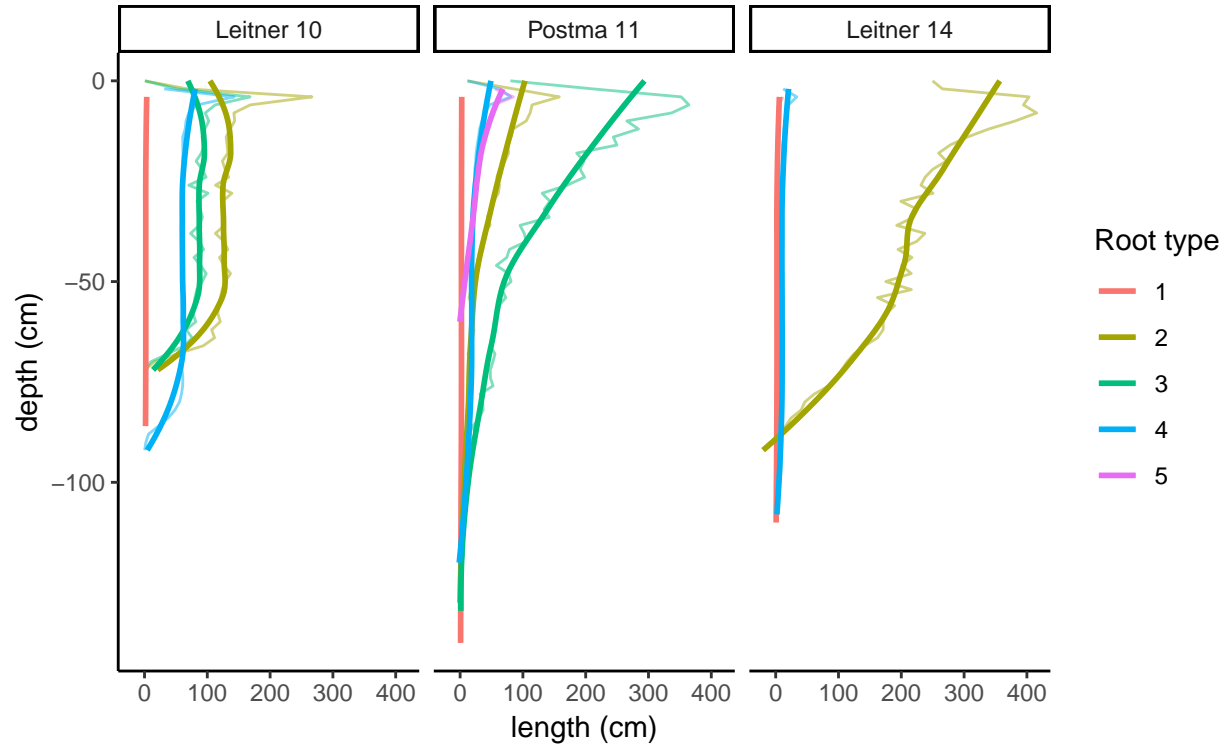


Figure 14: Root length density

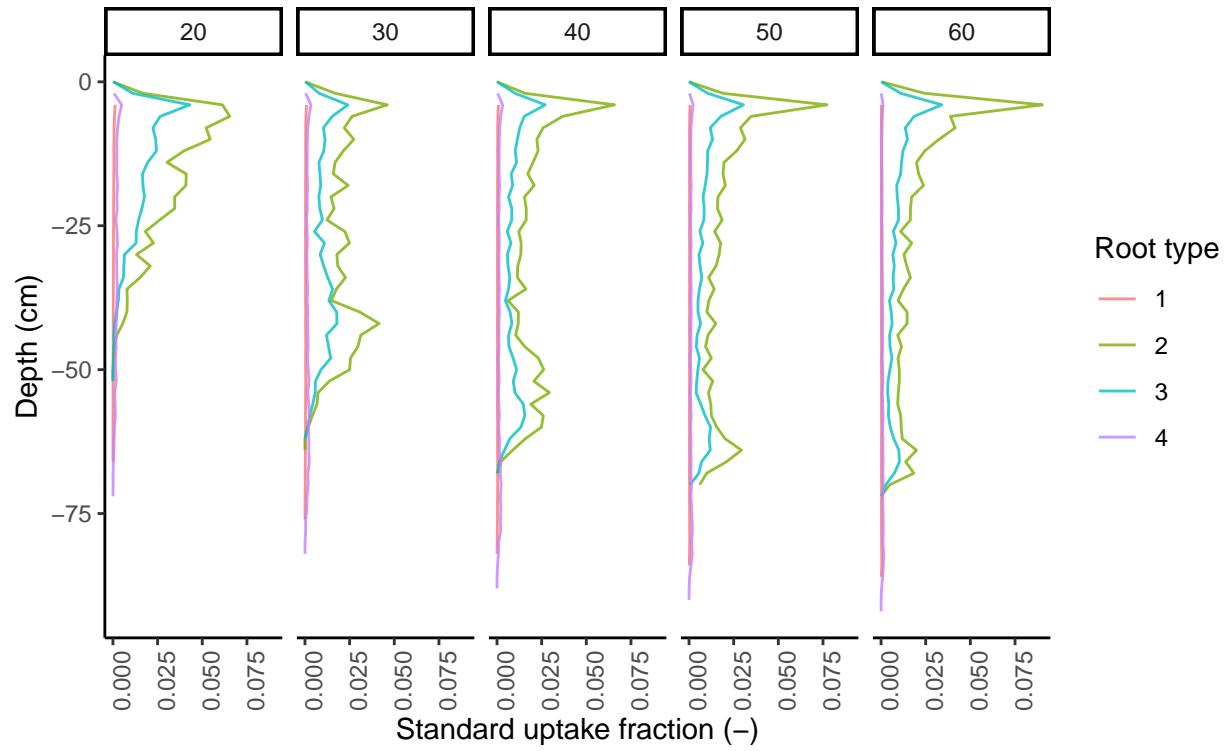


Figure 15: Evolution of the SUF with Leitner 2010 RSA parameter

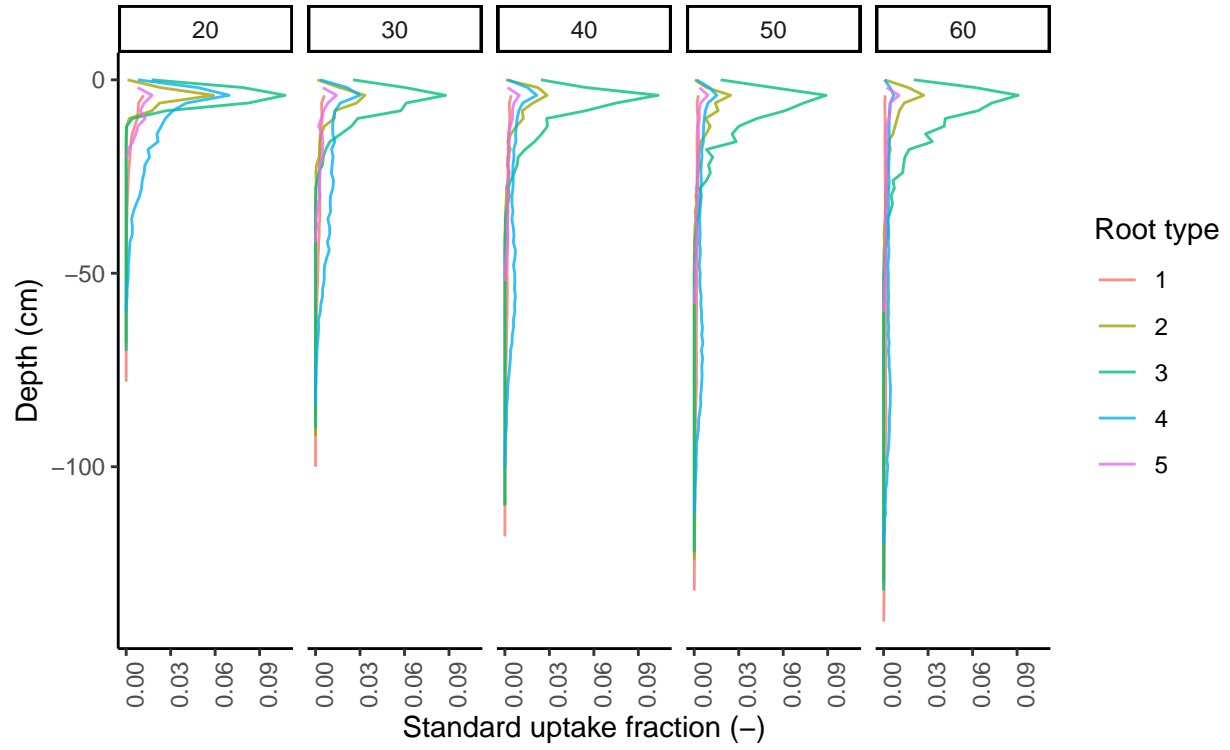


Figure 16: Evolution of the SUF with Postma 2011 RSA parameter

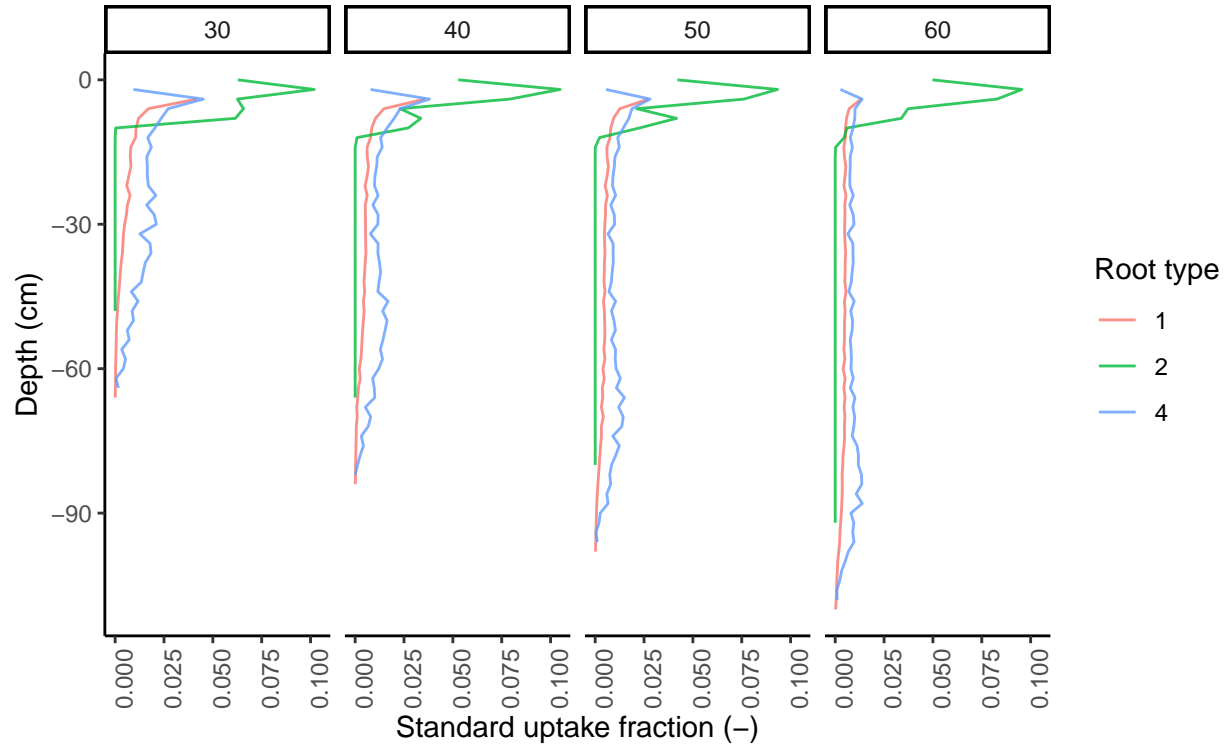


Figure 17: Evolution of the SUF with Leitner 2014 RSA parameter

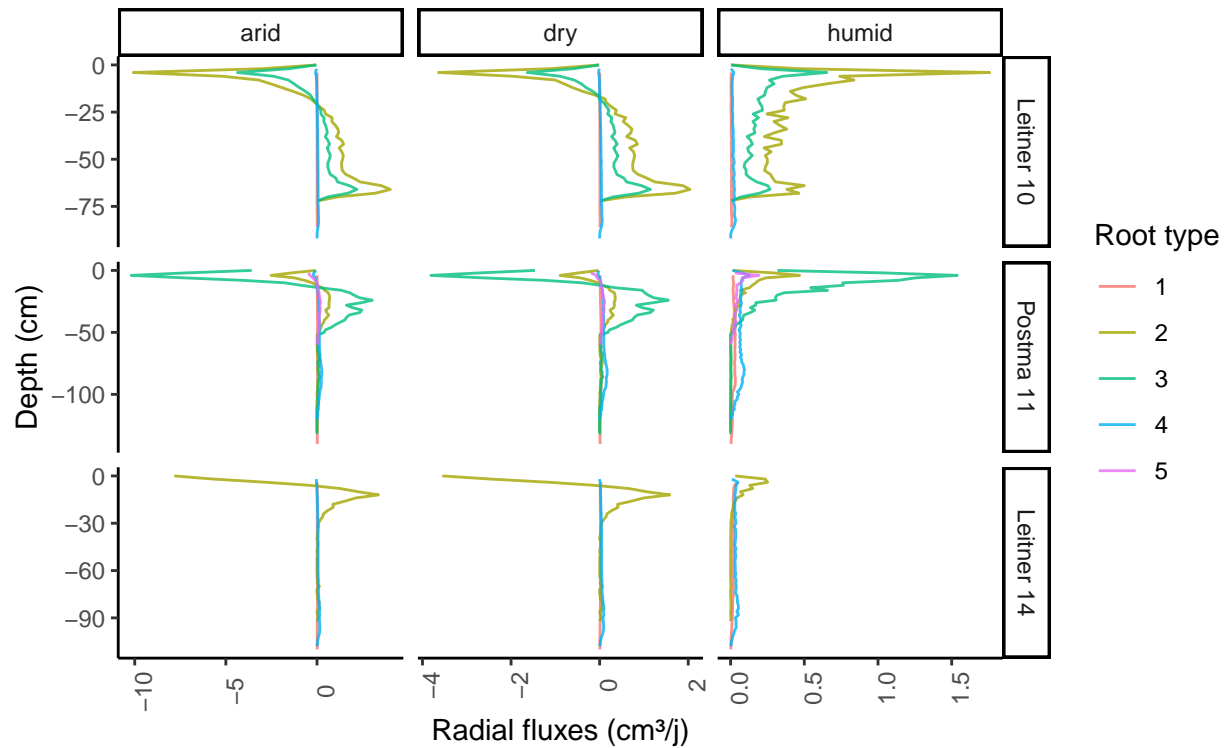


Figure 18: Comparison of the radial fluxes between the different soil scenarios and the RSAs

```
geom_line(alpha = 0.7) +
#geom_smooth(se=F) +
coord_flip() +
theme_classic()+
xlab("Depth (cm)") +
ylab("Axial fluxes (cm³/j)")+
labs(colour = "Root type")+
facet_grid(ex~humid, scales = "free", labeller = as_labeller(babel))
```

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

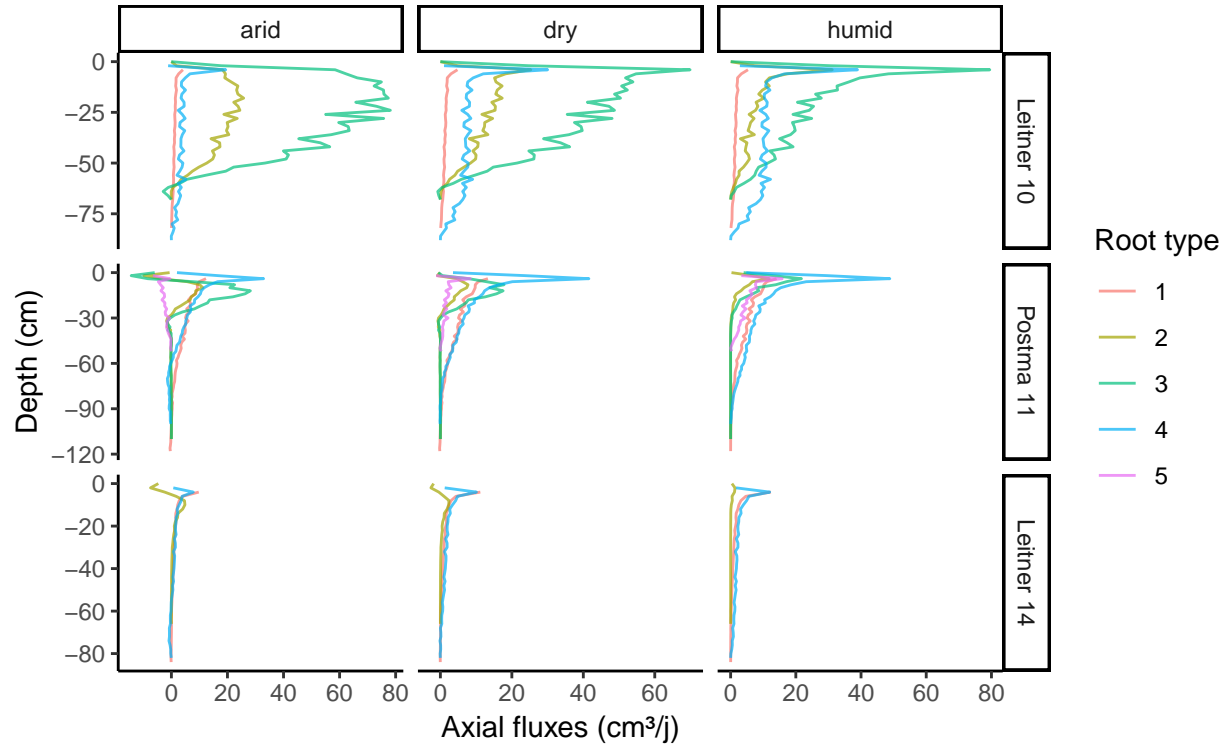


Figure 19: Comparison of the axial fluxes between the different soil scenarios and the RSAs

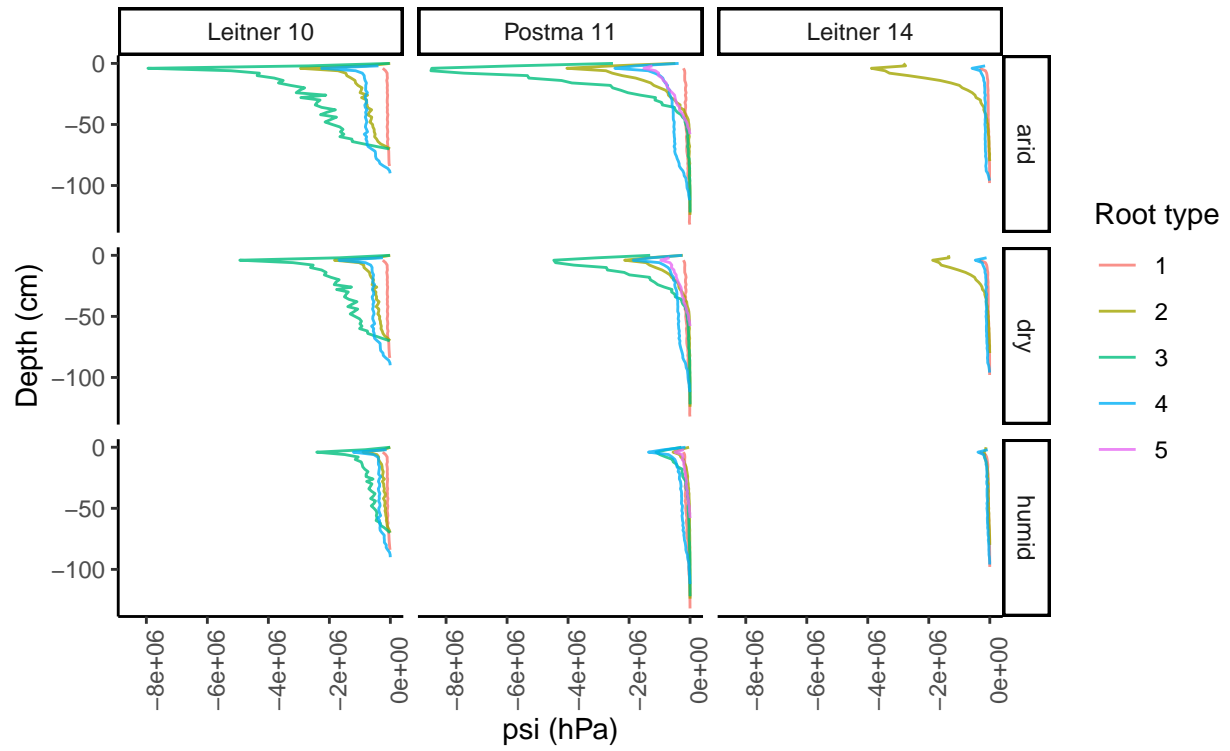


Figure 20: Water pressure head at the soil-root interface

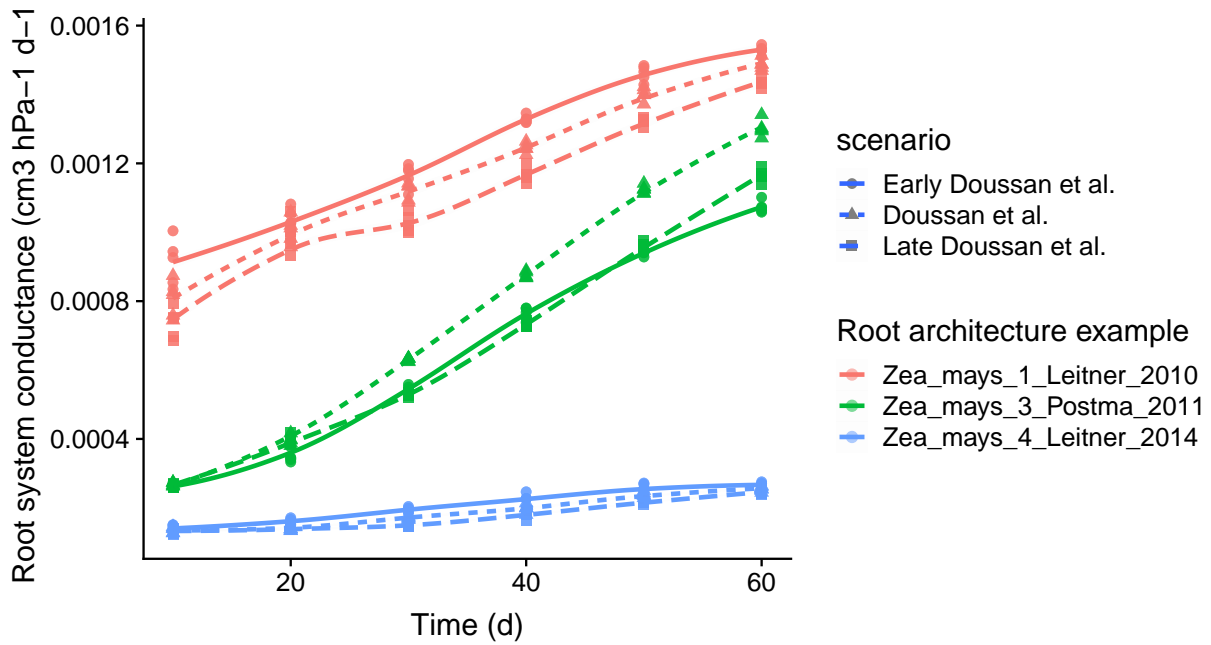


Figure 21: Evolution of the whole root system conductance. Effect comparison between RSA and root conductivity modification.

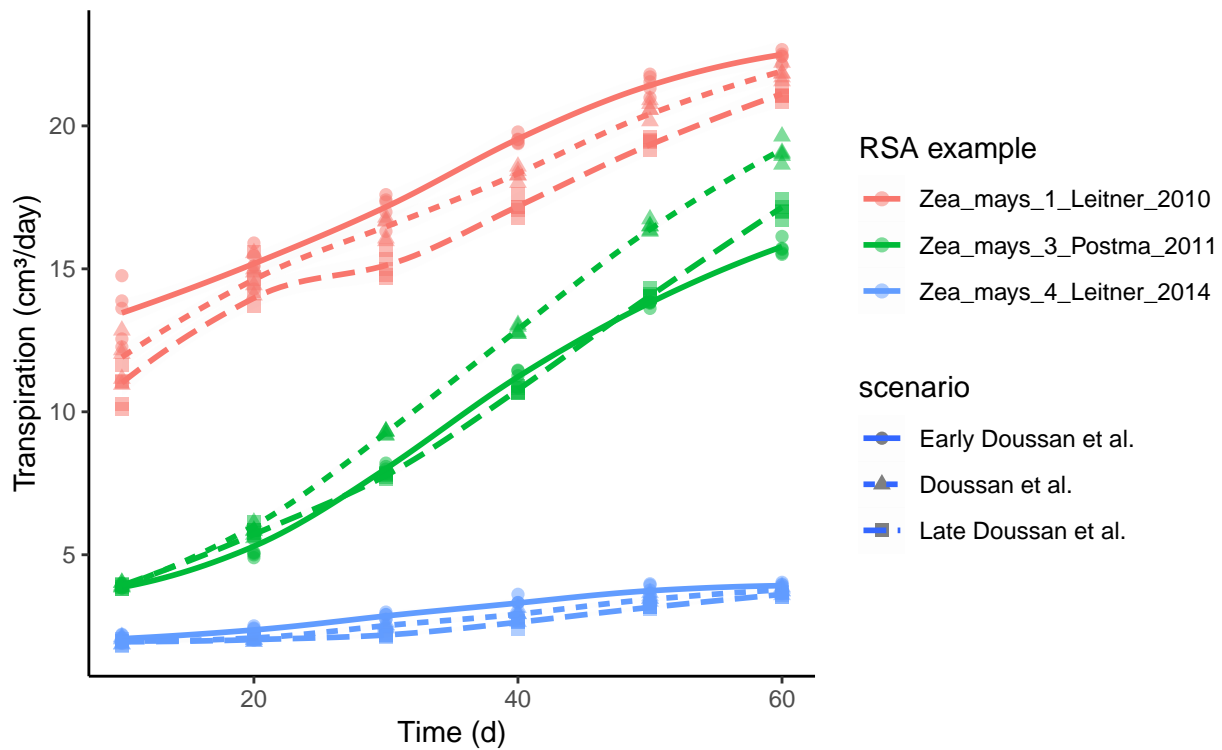


Figure 22: Evolution of the transpiration rate in a wet soil.

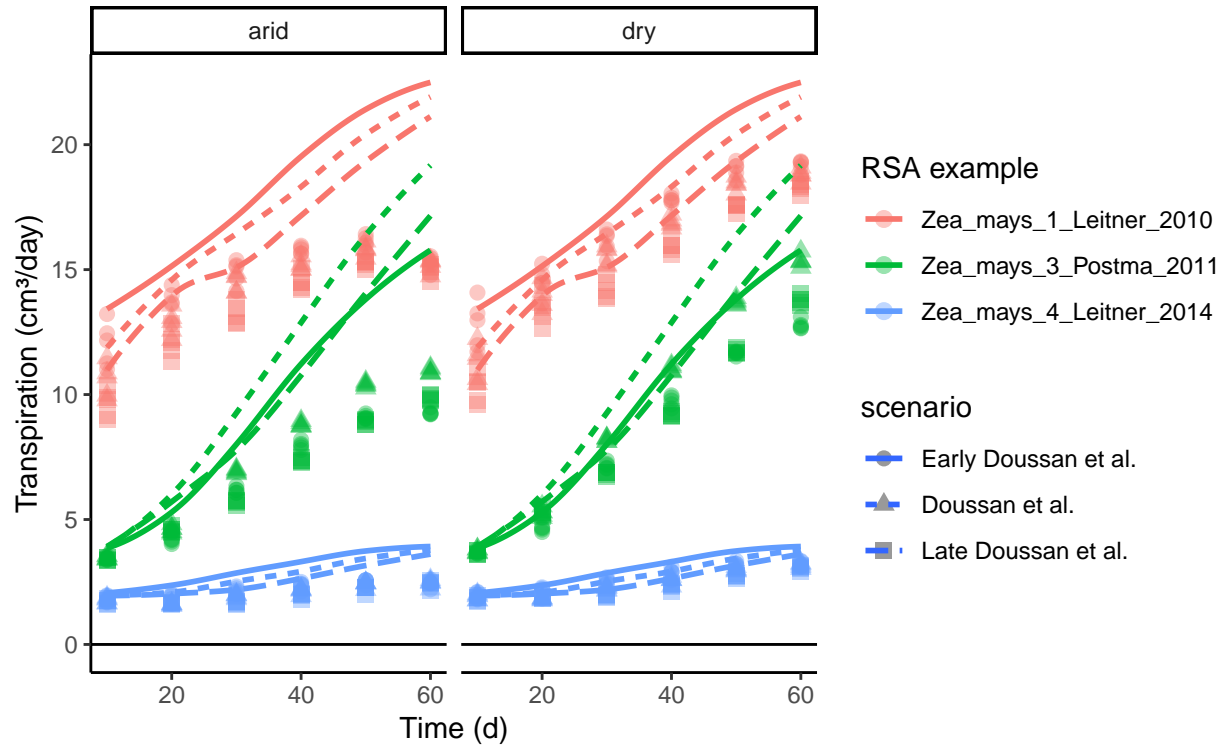


Figure 23: Evolution of the transpiration rate in two different soil scenarios where the soil is drying up.

References

- Couvreux, V, J Vanderborght, and M Javaux. 2012. "A Simple Three-Dimensional Macroscopic Root Water Uptake Model Based on the Hydraulic Architecture Approach." *Hydrol. Earth Syst. Sci.* 16 (8). Copernicus GmbH: 2957–71.
- Doussan, Claude, Gilles Vercambre, and Loïc Pagès. 1998. "Modelling of the Hydraulic Architecture of Root Systems: An Integrated Approach to Water Absorption–Distribution of Axial and Radial Conductances in Maize." *Ann. Bot.* 81: 225–32.
- Feddes, RA, PJ Kowalik, and H Zaradny. 1978. "Water Uptake by Plant Roots." *Simulation of Field Water Use and Crop Yield*. Wiley New York, 16–30.
- Meunier, F, V Couvreur, X Draye, J Vanderborght, and M Javaux. 2017. "Towards Quantitative Root Hydraulic Phenotyping: Novel Mathematical Functions to Calculate Plant-Scale Hydraulic Parameters from Root System Functional and Structural Traits." *J. Math. Biol.* 75 (5): 1133–70.
- Schnepf, Andrea, Daniel Leitner, Magdalena Landl, Guillaume Lobet, Trung Hieu Mai, Shehan Morandage, Cheng Sheng, Mirjam Zörner, Jan Vanderborght, and Harry Vereecken. 2018. "CRootBox: A Structural–functional Modelling Framework for Root Systems." *Ann. Bot.*, February.