Collaboratium

Guillaume Lobet[1], nathan.taitano25[1], mks91551[1], lh65982[1], Alexander Bucksch[2], ankita.roy[1], annalisa cartabia[1], Dilhan Ozturk[1], Guillaume De Meue[1], henri.moens[1], hermann.prodjinoto[1], and simon.chavee[1]

[1]Affiliation not available
[2]University of Georgia

April 14, 2020

Abstract

Plants are complex organisms living in a changing environment and, because of that, they have to develop many strategies to perceive physical, chemical or biological stimuli to characterize their momentary environment. Our study aims to understand how plants can interact with those stimuli to communicate with other plants present in their environment. We hypothesized that this communication appears through the root system and we focused our work to apprehend the possible vectors.

First, *in-vitro* experimentation was developed by a group of American students in order to identify four components responsible for the communication between plants. They conclude: " *this pilot study has indicated that neighbor plants communicate underground when submitted to different concentrations of nutrients*".

Then, we developed models to predict the vector of the signal with two hypotheses: by an active transport through mycorrhizae or by a passive transport through diffusion in the medium. Based on the response time noticed in the American's experiment, our results showed that the observed response happened too quickly to be explained by the communication ways we tested.

Introduction

It is known that plants react to many stimuli that are present in the growing environment, such as physical (i.e. light, temperature, humidity), chemical (i.e. nutrient concentration, volatile compound, chemical signal in the ground, ...) or biological (i.e. pathogens, symbiont).

How plant can interact with those stimuli? Via allelopathy, it is accepted that they are able to communicate between them. But, what are the vector of this communication? The soil, the air, the water or some living organisms. In this study case, we tried to explore this question following two steps:

The first one is based on an *in-vitro* experimentation developed by a group of American students. Their purpose was to identify four components responsible of the communication between plants: a clear motivation or reason for the communication, the presence of a signal and/or channel, a receiver, and a clear response. To identify the interaction between plants, they build an *in-vitro* system physically separated into three sections, each of them had a different nutrition condition. They hypothesized that this discrimination in term of nutritive context could be the reason to induce communication. In order to test this hypothesis, one plant has been placed in each compartment. Therefore, they supposed that the signal would be transmitted by chemical molecules excreted by the roots, called exudate, and that the channel will be passive, through the medium or active via the mycorrhizae, that is the reason why they used pre-mycorrhized plants. To allow the perception of the signal by a receiver, the roots of each plant were placed in a shared compartment without any physical contact in the middle of the system (Figure 1). Finally, they wanted to observe a clear response, such as a root direction growth that would indicate the perception of a more favorable nutritive context.

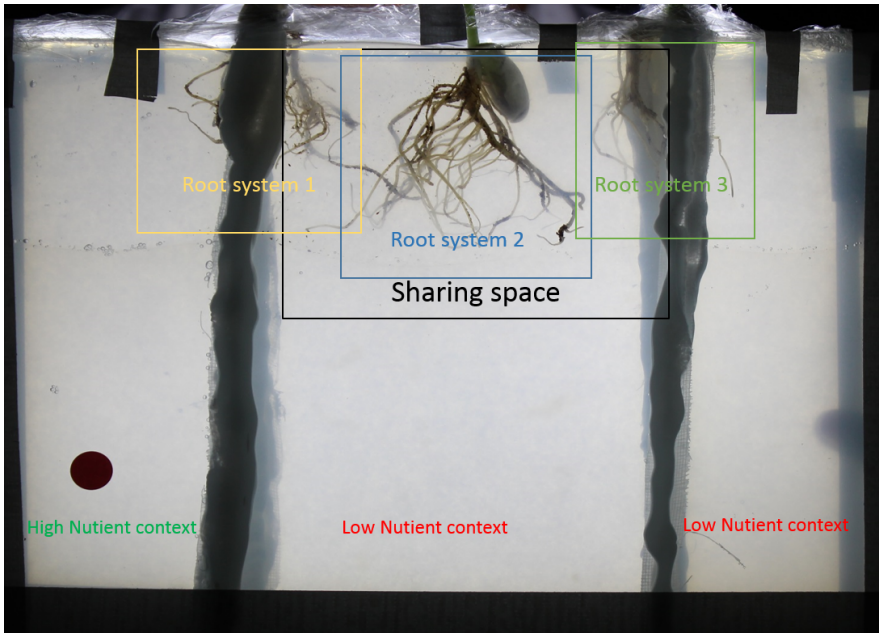

Figure 1: Experimental in-vitro system

To validate one or the other hypothesis, we developed predictive models fed by the data provided by the experiment and by data founding in literature.

Predictions based on a model

The aim of the modeling is to predict the channel of the signal based on the response time observed in the experiment.

To achieve this objective, we employed the CRootBox model to create a realistic root and mycorrhizae architecture that we used to check if the signal could be transmitted using the mycorrhizae as a vector of information transport and to characterize the production sites of signal molecules. Then, we created a model of passive diffusion through a homogeneous matrix simulating the soil in order to estimate the time necessary

to transmit a chemical signal from a plant A to a plant B. By comparing experimental results with those obtained by modeling, we want to determine the type of signal that would have been used to communicate between plants.

Material and Methods

Quantification of root architecture

Description experimental setup

Thanks to the American *in-vitro* trial, it was possible to create a photo suite of the roots of each plant in their culture medium for each day until the end of the experiment. We analyzed these images in ImageJ, software used to process/analyze them. In particular, we used SmartRoot, a semi-automatic image analysis software, considered an ImageJ plugin, allowing to quantify root growth and other parameters of plant architecture.

Quantification of root traits (SmartRoot)

With SmartRoot software, one image is processed at a defined time. We delimited the primary root on the image and SmartRoot makes a trace over the entire length of this root. It is possible to correct the traces that were incorrectly carried out. We have also demarcated the secondary roots and SmartRoot continues the route for this secondary root. This route can then be attached to the primary root line.

The entire route can then be imported into another image. We imported this set in the photo made the next day and continue the plots on the imported set. By performing this step on each photo for the duration of the experiment, it is possible to get a root list with their root type, length and insertion distance on the primary root if it is a secondary root.

All these data were processed on R and Excel to obtain the growth rate of the primary and secondary roots as well as the distance between the insertion points of the secondary roots.

The length between the last branching, the length between the first branching and the beginning of the basal root and the maximum length of the roots were arbitrarily chosen to resemble the plants used in the American experimentation.

Modeling root and mycorrhizae architecture

Description of CRootBox

CRootBox is a software that aims to simulate different types of root architecture. To do this, CRootBox creates a primary root segment with a specific type assigned. This type gives at the segment different characteristics: growth rate, length, basal length, tropism, diameter, distance between secondary roots, etc. Therefore, this root can produce secondary root segments to which it can also be assigned a different type and so on. It is also possible to specify the number of primary roots of the plant (in this model, we will assume there are 3 primary roots).

CRootBox will then create a sequence of segments over a defined simulation time creating a file where each segment is referenced with different information: its starting point in (x_i, y_i, z_i), its arrival point, age, type, etc.

Description of the root classes in the model in R

In this model, we have defined 4 classes of roots that CRootBox will use to create the architecture of roots and mycorrhizae (Annex I).

The first type is attributed to the primary root: its growth rate characteristics and its distance between secondary roots were calculated using SmartRoot.

The second type is attributed to the secondary roots where the growth rate was determined using Smart-Root. We considered that the secondary roots do not make branching, which corresponds to the reality until the growth time of the plants tested is not too long. The other parameters needed for the first type and the second one have been arbitrarily chosen to be similar to the American experimentation.

The third type is attributed to the mycorrhizae. Indeed, from CRootBox, it is not possible to create two separate objects developing at the same time in a defined space. It is therefore impossible to simulate the growth of root and mycorrhizal at the same time in CRootBox , if they are considered as two different entities. To circumvent this problem, we considered the mycorrhizae as roots that can grow on any root type. In order to be plausible, we attributed to the third type a plagiotropism allowing mycorrhizae to develop in a mainly horizontal plane. The growth rate and branching distance were determined at the Louvain-la-Neuve mycology laboratory in Petri dish. We recognize that the rate of growth of mycorrhizae can vary greatly depending on the environment. We have therefore produced a script to carry out a simulation suite for a fixed growth rate that will increase for each simulation (Annex II). This script could be used with other scripts performed in this work to determine a time panel so that two different plant mycorrhizae can touch and send a signal.

The fourth type is attributed to the ramifications of mycorrhizae that have exactly the same characteristics as type 3 except that we consider that they do not branch to avoid overloading the amount of information that CRootBox needs to process in order not to significantly slow down the simulation.

Experimental design in R language

(Post processing of CRootBox, script is available in Annex III)

Using the simulation results of the roots architecture implemented with mycorrhizae from CRootBox program, we were able to estimate with R the contact time between the two radical systems with a fixed distance. We assume that the growth rate and the density of mycorrhizae are enough high so that the contact happens always between mycorrhizae and not between real roots of the plants.

For doing that, we positioned the two root systems in the XZ-plane leading to the representation of the experimental design. After loading the files, we applied a translation along the X axis to one of the two root systems. This translation corresponds to the distance between the two collars of the plants. Notice that we applied another translation, this time for both root systems, we avoid that segment could have a negative x coordinate. The minimal x coordinate is thus 0 and the maximal z coordinate is also 0. This will

greatly simplify the next step consisting of the representation of the 2D space by matrices. By plotting with a different color for the different root types, we could check our assumption that contacts happen between mycorrhizae (mycorrhizae correspond to type 20 and type 50).

We divided the experimental space in a grid composed of cells characterized by a specific size depending on the distance between the collars of the two plants divided by 10. The idea is to increase the precision when the distance between the plants is shorter. For each root system, we developed a matrix containing for each cell the number of root's segments placed inside. After, we overlapped the two matrices in order to see if there is contact. If the same cell contains segments in both matrices, we assume there is a mycorrhizal association. It is probable that two segments in the same cell are not really in contact but, when closed off each other, mycorrhizae can connect thanks to chemotropism, justifying this assumption.

Compare the two matrices thus allows to find the cells where there is contact. From this information, we looked in those cells to find the youngest segment from each plant. The age of the older of the two corresponds to the minimal time needed for what we consider a contact. The minimum of the results obtained for the different cells is our estimation of the time needed for a mycorrhizal association between the two plants.

Modeling signal transfer in the environment

Describe the diffusion model

We kept the space discretization into cells like seen before. 3D array for each of both plants have been made, let's call them A and B. First two dimensions correspond to the XZ space and the third dimension correspond at the time. This is discretized so that we have a one-time layer per day. We considered one plant emitting a signal and the other plant able to receive the signal. For the emitting plant, the array contains the number of emitting segments in each cell. We consider that all the root segments exudate if they are at most 3 days old. For the receiving plant, the cells contain a 1 if there is at least one root segment, and a 0 if there is not. A layer contains more 1 if it corresponds to an advanced time value. The code to model A and B is in Annex IV.

To verify that a chemical-type signal can be sent from a root of plant A to the root of plant B a two-dimensional numerical diffusion and reaction model has been created. The numerical model has been solved according to the explicit finite difference method on the software RStudio©. The model includes 3-dimensional arrays, first two are A, B and the third one called C represent the concentration of signaling molecules. C characterizes the concentration in molecules signals in every point of the space and with each step of time. The signal is transmitted when the term product of the matrices B and C is non-zero, which is equivalent to saying that the signal is transmitted when a receiving root or a receiving mycorrhizae is located in (t_i, x_i, y_i) and that the molecule concentration in (t_i, x_i, y_i) is greater than 1 / 100,000 units of concentration/cm^2. Since the type of exudate is unknown, it has been arbitrarily considered at first approach that a root emits a unit of molecular concentration per day per square centimeter and that the quantity required for detecting the signal must be at least 1/100 000 times the concentration emitted. If the emission rate and the concentration required for detecting the compound are known, the model can take it into account for more realistic modeling. The code of the model is reported in annex V.

Results

Roots architecture

After simulation of CRootBox for a given parameter set (annex II) in which we exclude the data set of mycorrhizae and their ramifications, we get the following root architecture for 30 days simulation.

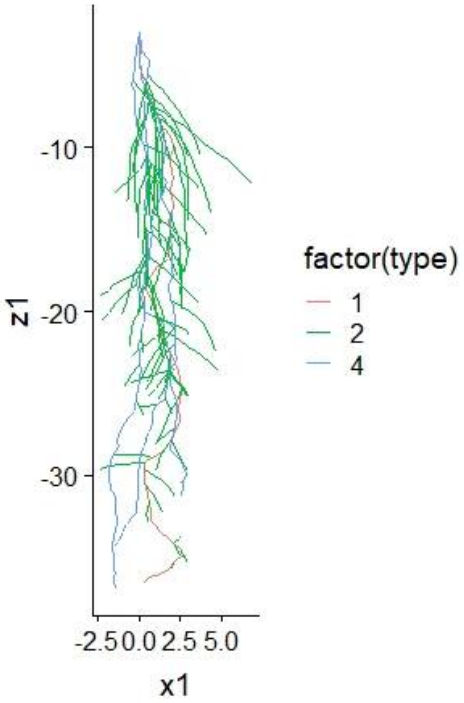

Figure 2: Roots architecture

Type 1 represents the primary root. Type 2 the secondary root. Type 4 is automatically calculated by CRootBox as define basal roots, in our case we did not define this root type. Then, CRootBox uses the primary root parameters (type 1) to define type 4.

Roots with mycorrhizae architecture

If we use the parameter included the mycorrhizae and their ramifications, we get the following architecture after 30 days simulation.

Type 1, 2 and 4 are defined in the same way as in Figure 2. Type 20 corresponds to mycorrhizae. If the simulation lasted longer for this parameter set or if we increased the growth rate of mycorrhizae, we could observe a type 50 that correspond to the mycorrhizae ramifications.

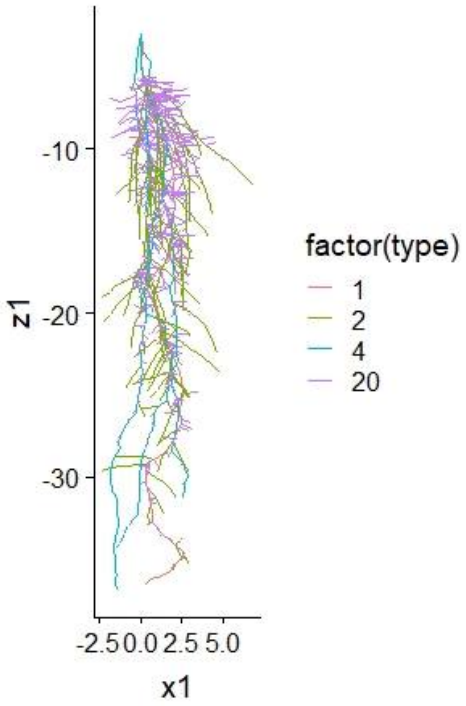

Figure 3: Roots with mycorrhizae architecture

Mycorrhizae development

In this section, we present some results of the estimation of the time needed for the establishment of a mycorrhizae relation between two plants.

In all the cases, the distance between the two collars is set to 10.8 cm, corresponding to the distance set in the *in-vitro* experiment of the American team. We used several root architectures from CRootBox with a different mycorrhizae growth rate.

The first image below represents a mycorrhizae growth rate of 0.08 cm/day, which seems a coherent value for the culture media used in the trial. After the 30 days of the simulation, we did not verify a contact between the two plants. If a response occurs before, we could deduce that it is not due to mycorrhizae effect.

Moreover, we tried with higher values of mycorrhizae growth rate susceptible to appear in a real soil condition.

When it is set to 0.2 cm/day, we found 19 points of contact appearing on average after 26.7 days. The first of them appears after 18.4 days.

When it is set to 0.4 cm/day, we found 34 points of contact appearing on average after 28.2 days. The first of them appears after 21.8 days.

Diffusion of exudates

We calculated the response time based on three diffusion values ranging from 0.0018 cm^2 / day to 0.18 cm^2 / day to cover a wide range of possible values for the diffusion in an unknown homogeneous medium for

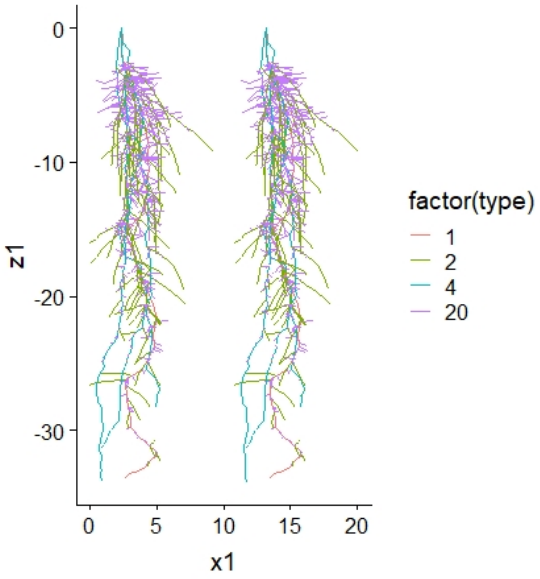

Figure 4: Mycorrhizae growth rate = 0.08 cm/day

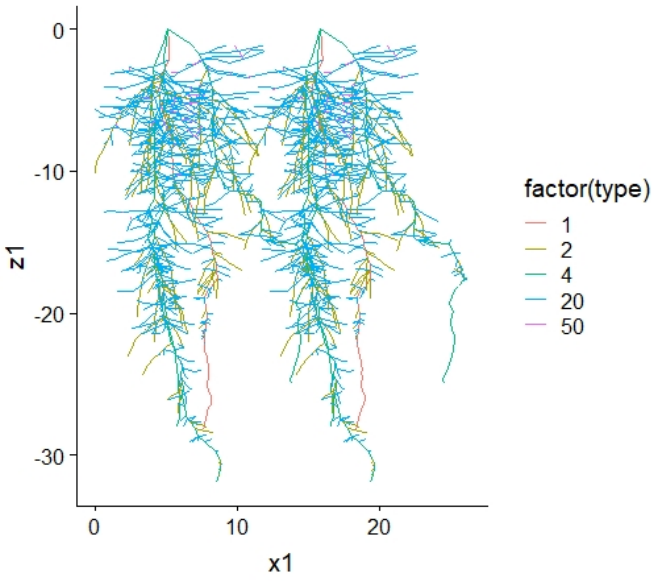

Figure 5: Mycorrhizae growth rate = 0.2 cm/day

unknown substances. Values less than 0.0018 have not been calculated because below this value the model calculated that the roots touch each other before the chemical signal has been transmitted. In fact, the root contact seems to be established in the modeling after 28 days. With a diffusion coefficient of 0.18, which is the maximum estimated value, the signal is transmitted after 11 days. Therefore, we can exclude the possibility that the change of direction of the roots of the experiment is caused by the transmission of a chemical signal because it could not reach the root segment before the change of direction which happened before one week.

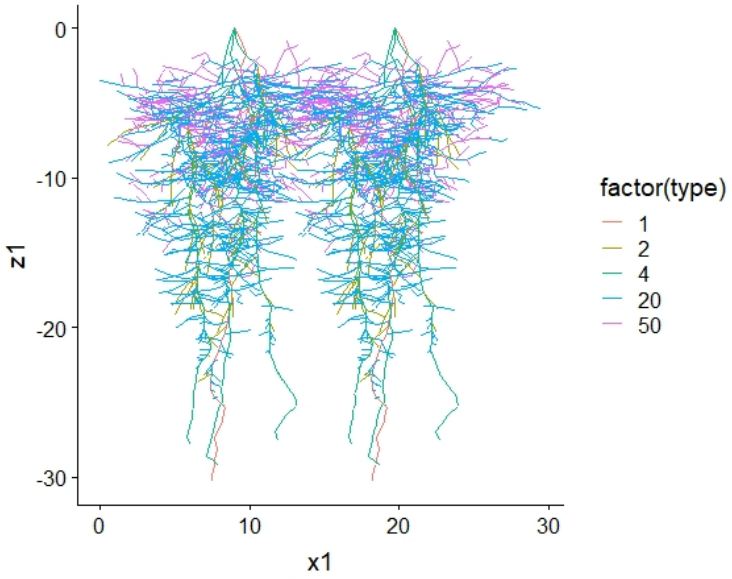

Figure 6: Mycorrhizae growth rate = 0.4 cm/day

Discussion

Limits of the experimental systems

The experience suffers from certain methodological flaws. Firstly, the lack of repetition of the experiment makes impossible to observe the variability of the root structures or to compare the reaction times of the plants in order to obtain information on the type of signal transmitted if it takes place. A repetition would also have made it possible to estimate if the rapid reaction observed in this experiment was systematic or if it was a particular phenomenon. The second lacunar aspect of the experiment lies in the assumptions of the communication device. In order for the signal to allow the plant to orient its growth, it is necessary for the emitting plant to have an evolutionary interest in emitting a signal and for this signal to contain information on the relative direction of the nutrient quality gradient and, finally, that signal can be interpreted by the receptor plant. It is possible that all plants continuously emit similar signals in amounts proportional to the quality of their environment and thus create a signal gradient characterizing the soil quality gradients but this is unlikely from an evolutionary point of view because it means that the plant spends energy to increase competition in his environment. Except if we hypothesized that the signal comes from the plant in a disadvantageous environment, then it might be an evolutive advantage for a population to avoid a hostile environment.

Crosstalk between the model and the experiment

In the model, we considered that the plants were enough separated so that there is no direct contact between their roots in the period of time taken into account. Each contact is thus due to mycorrhizae. From the data we got, it seems that the growth rate of mycorrhizae in the media is so low that there may be no contact

in this period of time. By reducing the distance between the two plants, it is possible that contacts happen between roots, without mycorrhizae. Therefore we must remain critical about the nature of the contact. On the one hand, we can check on the generated graph and wonder if the parameters chosen seem to allow a contact between real roots. On the other hand, we have to compare it with the experiment. Possible contacts between roots can easily be seen with the naked eye. See mycorrhizae requires a microscope.

On the isolated simulations we did, we obtained a surprising result. The contact appears quicker with a mycorrhizae growth rate of 0.2 cm/day than with 0.4 cm/day while the distance has not changed. This can be partly explained since the structure from CRootBox is made with a part of stochasticity. This highlights the importance of repeating simulations and experiments and using statistical tools to interpret the results.

Improvement of the diffusion model

The model designed seems to be robust on the scale of phenomena studied. Plausible results are observed for all scales of values analyzed. The most obvious improvement is the determination of the parameters. Indeed the diffusion coefficients, the rate of production of signal molecules and the concentration necessary for the detection have been chosen arbitrarily and must be adapted to the medium, the signal and the types of roots studied. Only one type of plant has been analyzed by modeling, even if the results are a priori similar to any type of plants, the model should be run with different root phenotype in order to validate this assumption. More differentiation in the model of the collecting roots could be used to characterize the signal transmission time for each type of root and thus improve the specificity of the results that remain very general in this model.

We consider that the emitting plant exudes the signal directly after the germination of the seeds. This is a huge assumption and it is important to remind that an active communication process is composed, in addition, at least of an identification of the signal to transfer and the creation of the signal, or at least a reaction to the environment in the case of passive communication. To this are added the transfer of the signal and possibly a response. Those phenomena take time and extend the communication time. We did not need to work on those phenomena because we find that transfer time is long enough to exclude the communication hypothesis.

Conclusions

We tried to develop models allowing to test different circumstances. We can use them by varying different parameters such as the growth rate of roots and mycorrhizae, the distance between both plants, the diffusion coefficient, etc. It is very important to be critical regarding the results obtained. The modeling process sometimes provides very different results from the experimentation ones.

In the experiment provided by the American team, the response seems to happen very quickly. None of the communication ways we tested can explain such a quick response.

Bibliography

Draye, X., Lobet, G., Pages, L. (2011). A Novel Image Analysis Toolbox Enabling Quantitative Analysis of Root System Architecture. 29-33. doi: https://doi.org/10.1104/pp.111.179895

Hieu Mai, T., Landl, M., Leitner, D., Lobet, G., Morandage, S., Schnepf, A., Sheng, C., Vanderborght, J., Vereecken, H., Zoerner, M. (2017). CRootBox : A Structural-Functional Modelling Framework For Root Systems. 5-14. doi: https://doi.org/10.1101/139980

Huber, L., Roy, A., Sutton, M., Sutton, K., Taitano, N. (2018). Collaboratorium Report.

Petzoldt, T., Soetaert, K., Woodrow, R. (2010). Solving Differential Equations in R. The R Journal Vol. 2/2. Retrieved from https://journal.r-project.org/archive/2010/RJ-2010-013/RJ-2010-013.pdf (Last consultation: January 25th 2019)

Annex

Annex I : Root Architecture parameter set (Divided there rparam and pparam)

param.rparam

```
# Parameter set for type
type    1

name    primaryroots

lb  3   0.03

la  6.14    3.07

ln  0.154   0.4964

lmax    200 2

r   1.1654  0.116

a   1   0.2

color   0.5 0.5 0.5

tropism 1   2   0.832

dx  1

successors  2   2   20

successorP  2   0.2 0.8
```

```
theta   0   0

rlt 3000    300

gf  1

# Parameter set for type
type    2

name    secondaryroots

lb  0.8 0.008

la  2   0.2

ln  0.56    0.056

lmax    20  5

r   0.38    0.038

a   0.5 0.1

color   0   1   0

tropism 1   2   0.18

dx  1

successors  1   20

successorP  1   1

theta   0.882   0.0882

rlt 3000    300

gf  1

# Parameter set for type
type    20

name    secondaryroots

lb  3   0.03

la  1   0.01

ln  0.8889  0.001

lmax    500 0
```

```
r   0.08    0.008

a   0.1 0.02

color   0   1   0

tropism 0   1   0.5

dx  1

successors  1   50

successorP  1   1

theta   1.01    0

rlt 3000    300

gf  1

# Parameter set for type
type    50

name    secondaryroots

lb  500 5

la  500 5

ln  0   0

lmax    500 5

r   0.08    0.008

a   0.1 0.02

color   0   1   0

tropism 0   1   0.5

dx  1

successors  0   0

successorP  1   1

theta   1.01    0

rlt 3000    300
```

```
gf  1
```

param.pparam

```
plantingdepth   3
firstB  0
delayB  0
maxB    3
nC  0
firstSB 14
delaySB 0
delayRC 7
nz  0
simtime 20
dt  20
```

Annex II : Conducts a simulation suite where the growth rate of mycorrhizae varies for each simulation

Script to change the value of mycorrhizal growth rate and record the result (need of io_function.R) with CRootBox

```
library(tidyverse)
library(data.table)
library(stringi)
library(plyr)
library(Hmisc)
options(scipen=999) # Disable scientific notation
#-------------------------------------------------------------------------------------------
#-------------------------- GENERAL OPTIONS ------------------------------------------------
#-------------------------------------------------------------------------------------------
# Main directory, where everthing is stored
dir.base <- "CRootBox-master/"
# Where is ArchiSimple folder
setwd(dir.base)
# Load custom functions
source("io_function.R")
r_range <- seq(from=0.05, to=5, by=0.01)
#-------------------------------------------------------------------------------------------
#-------------------------- READ THE INITIAL PARAMETER FILES -------------------------------
#-------------------------------------------------------------------------------------------
# This is done to always start from the some parameters when we induce variations
dataset_init <- read_rparam('original/param.rparam')
# plant_init <- read_pparam('original/param.pparam')
#-------------------------------------------------------------------------------------------
#-------------------------- WRITE NEW PARAMETER FILES --------------------------------------
#-------------------------------------------------------------------------------------------
k <- 0
for(r in r_range){
```

```
  k <- k+1
  unid <- stri_rand_strings(1, 10)  # Get a unique ID for the simulation

  # plant <- plant_init
  dataset <- dataset_init

  print("------------------------")
  print(paste0("Simulation ",k," started"))

  # Vary "r" for type 20 and 50
  dataset$val1[dataset$type==20 & dataset$param == "r"]  <- r
  dataset$val1[dataset$type==50 & dataset$param == "r"]  <- r

  # Write the data in the parameter file that will be sued nby CRootBox
  write_rparam(dataset, "www/param.rparam")
  # write_pparam(plant, "www/param.pparam")


  #--------------------------- RUN CODE AND STORE DATA ---------------------------------------------

  system("a.exe")


  file.rename(from = "30_rootsystem.txt",
              to = paste0("resultats/",r,"-30-rootsystem.txt"))

}
```

io_function.R

```
 FUNCTION TO READ THE PPARAM FILES FROM CROOTBOX AND STORE IT INTO A DATAFRAME
read_rparam <- function(path){

  fileName <- path
  param <- read_file(fileName)


  param <- strsplit(param, "#")
  dataset_init <- NULL
  for(k in c(2:length(param[[1]]))){
    spl <- strsplit(param[[1]][k], "\n")
    type <- ""
    name <- ""
    for(i in c(1:length(spl[[1]]))){
      temp <- spl[[1]][i]
      pos <- regexpr("//", temp)
      if(pos != -1) temp <- substr(temp, 0, pos-1)
      if(nchar(temp) > 0){
        temp <- strsplit(temp, "\t")
        temp2 <- data.frame("type" = character(0), "name" = character(0),
```

```
                              "param" = character(0), "val1" = numeric(0),
                              #Addition of val4
                              "val2" = numeric(0), "val3" = numeric(0), "val4" = numeric(0), stringsAsFac

        if(temp[[1]][1] == "type"){ type <- temp[[1]][2]
        } else if(temp[[1]][1] == "name"){ name <- temp[[1]][2]
        } else if(grepl("Param", temp[[1]][1])){
        } else if(temp[[1]][1] == "tropism") {
          temp2[[1,3]] <- "n_tropism"
          temp2$val1 <- temp[[1]][3]
          temp2$type <- type
          temp2$name <- name
          dataset_init <- rbind(dataset_init, temp2)
          temp2$param <- "sigma_tropism"
          temp2$val1 <- temp[[1]][4]
          temp2$type <- type
          temp2$name <- name
          dataset_init <- rbind(dataset_init, temp2)
          temp2$param <- "tropism"
          temp2$val1 <- temp[[1]][2]
          temp2$type <- type
          temp2$name <- name
          dataset_init <- rbind(dataset_init, temp2)
        } else {
          for(j in c(1:5)){
            temp2[[1,j+2]] <- temp[[1]][j]
            temp2$type <- type
            temp2$name <- name
          }
          dataset_init <- rbind(dataset_init, temp2)
        }
      }
    }
  }

  return(dataset_init)
}


# FUNCTION TO READ THE PPARAM FILES FROM CROOTBOX AND STORE IT INTO A DATAFRAME
read_pparam <- function(path){
  ## READ THE PARAMETER FILE AND STORE THE DATA IN A DATAFRAME
  data <- read_file(path)
  # READ THE PARAMETER FILE AND STORE THE DATA IN A DATAFRAME
  plant_init <- NULL
  spl <- strsplit(data, "\n")
  for(i in c(1:length(spl[[1]]))){
    temp <- spl[[1]][i]
    if(nchar(temp) > 0){
      temp <- strsplit(temp, "\t")
      temp2 <- data.frame( "param" = character(0), "val1" = numeric(0), stringsAsFactors = F)
```

```
      for(j in c(1:2)){
        temp2[[1,j]] <- temp[[1]][j]
      }
      plant_init <- rbind(plant_init, temp2)
    }
  }

  colnames(plant_init) <- c("param", "val1")
  return(plant_init)
}




# FUNCTION TO WRITE THE RPARAM FILES FROM CROOTBOX AND STORE IT INTO A DATAFRAME

write_rparam <- function(dataset, files){

  types <- unique(dataset$type)
  text <- NULL
  for(t in types){
    if(is.null(text)){text <- "# Parameter set for type"
    }else{
      text <- paste(text, "# Parameter set for type", sep="\n")
    }

    temp <- dataset[dataset$type == t,]

    str <- paste("type", temp$type[1], sep="\t")
    text <- paste(text, str, sep="\n")

    str <- paste("name", temp$name[1], sep="\t")
    text <- paste(text, str, sep="\n")

    for(i in c(1:nrow(temp))){
      if(temp[i, 3] == "n_tropism"){
        str <- paste("tropism", temp[i+2, 4], temp[i, 4], temp[i+1, 4], sep="\t")
        text <- paste(text, str, sep="\n")
      }else if(temp[i, 3] == "sigma_tropism" | temp[i, 3] == "tropism"){
      }else if(temp[i, 3] == "dx"){
        str <- paste(temp[i, 3], temp[i, 4], sep="\t")
        text <- paste(text, str, sep="\n")
      }else{
        str <- paste(temp[i, 3], temp[i, 4], temp[i, 5], temp[i, 6], temp[i, 7], sep="\t")
        text <- paste(text, str, sep="\n")
      }
    }

  }
  text <- gsub("\tNA", "", text)
  for(f in files){
```

```
    cat(text, file=f)
  }

}

# FUNCTION TO WRITE THE PPARAM FILES FROM CROOTBOX AND STORE IT INTO A DATAFRAME

write_pparam <- function(plant, files){
  text <- NULL
  for(i in c(1:nrow(plant))){
    str <- paste(plant[i, 1], plant[i, 2], sep="\t")
    text <- paste(text, str, sep="\n")
  }

  text <- gsub("\tNA", "", text)

  for(f in files){
    cat(text, file=f)
  }
}
```

Annex III : Contacts between the two root systems

```
# LOAD LIBRARIES (might need to install them first)

library(cowplot)
library(archiDART)
library(viridis)
library(dplyr)
library(readr)

# LOAD FILES
path <- "C:/Users/demo/OneDrive - UCL/LBRAI2219 Modelisation de systemes biologiques/Get_time_before_my
ls <- list.files(path) # Get all the file names in the selected folder
ls <- ls[grepl(".txt", ls)] # Select on the .txt files

# Loop over the files and load them
roots <- NULL  # this dataframe will contain the simulation results
for(l in ls){
  temp <- read_delim(file = paste0(path, l), delim = " ") %>% # Load the file
    mutate(file = l) %>% # Add the file name as a newcolumn, to disciminate the different files in the
    mutate(length = sqrt((x2-x1)^2+(y2-y1)^2+(z2-z1)^2)) # Add the length of each segement as a newcolu
  roots <- rbind(roots, temp)  # Add the file to the main datafile
}


# POSITION THE TWO ROOT SYSTEMS IN THE XZ SPACE

dist <- 10.8 # Set the distance between the two plants

xMin <- min (min(roots$x1), min(roots$x2)) # Find the minimal initial x value
```

```
zMax <- max (max(roots$z1), max(roots$z2)) # Find the maximal initial z value

files <- unique(roots$file) # Get the names of the two files
roots <- roots %>%
  mutate(z1 = z1-zMax)%>%
  mutate(z2 = z2-zMax)%>%
  mutate(x1 = ifelse(file == files[2], x1-xMin+dist, x1-xMin),
         x2 = ifelse(file == files[2], x2-xMin+dist, x2-xMin)) # Apply the translation to root systems

# Plot the two root systems
roots %>%
  ggplot()+
  geom_segment(aes(x1, z1, xend = x2, yend=z2, colour = factor(type))) +  # We use the 'segment' repres
  coord_fixed()


# DISCRETIZE THE SPACE INTO CELLS

region <- dist/10 # Size of a cell

system_width <- max (max(roots$x1), max(roots$x2)) # Width of the system
system_height = abs(min (min(roots$z1), min(roots$z2))) # Height of the system

# Create 2 matrices of discretized space (one for each plant)
MatrixSeg1 <- matrix(0, ceiling(system_height/region), ceiling(system_width/region))
MatrixSeg2 <- matrix(0, ceiling(system_height/region), ceiling(system_width/region))


root2D <- roots %>%
  mutate(z = floor(z1/region),
         x = ceiling(x1/region)) # Round the z and x values, to limit the number of layers

# Separate data per plant
root2D_1 <- root2D %>%
  filter(file == files[1])
root2D_2 <- root2D %>%
  filter(file == files[2])

# Fill the matrices of discretized space with the number of segments in each cell
for (i in 1:dim(root2D_1)[1]){
  MatrixSeg1[abs(root2D_1$z[i]), root2D_1$x[i]] = MatrixSeg1[abs(root2D_1$z[i]), root2D_1$x[i]] + 1
}

for (i in 1:dim(root2D_2)[1]){
  MatrixSeg2[abs(root2D_2$z[i]), root2D_2$x[i]] = MatrixSeg2[abs(root2D_2$z[i]), root2D_2$x[i]] + 1
}


contact <- which(MatrixSeg1 & MatrixSeg2, arr.ind = TRUE) # Get the indices of cells with segments of p

if (dim(contact)[1] == 0){
```

```
  print('There is no mychorhise communication between the two plants')
} else {
  # For plant 1, keep only the segments in a cell where there is a contact
  int1 <- NULL
  for (j in 1:dim(contact)[1]){
    for (n in 1:dim(root2D_1)[1]){
      if (abs(root2D_1$z[n]) == contact[j,1] & root2D_1$x[n] == contact[j,2]){
        int1 <- rbind(int1, root2D_1[n,])
      }
    }
  }

  # For plant 1, keep only the youngest segment in each cell where there is a contact
  for (i in dim(int1)[1]:2){
    if (int1$x[i] == int1$x[i-1] & int1$z[i] == int1$z[i-1] & int1$time[i] >= int1$time[i-1]) {
      int1 <- int1[-i,]
    } else if (int1$x[i] == int1$x[i-1] & int1$z[i] == int1$z[i-1] & int1$time[i] < int1$time[i-1]){
      int1 <- int1[-(i-1),]
    }
  }

  # For plant 2, keep only the segments in a cell where there is a contact
  int2 <- NULL
  for (j in 1:dim(contact)[1]){
    for (n in 1:dim(root2D_2)[1]){
      if (abs(root2D_2$z[n]) == contact[j,1] & root2D_2$x[n] == contact[j,2]){
        int2 <- rbind(int2, root2D_2[n,])
      }
    }
  }

  # For plant 2, keep only the youngest segment in each cell where there is a contact
  for (i in dim(int2)[1]:2){
    if (int2$x[i] == int2$x[i-1] & int2$z[i] == int2$z[i-1] & int2$time[i] >= int2$time[i-1]) {
      int2 <- int2[-i,]
    } else if (int2$x[i] == int2$x[i-1] & int2$z[i] == int2$z[i-1] & int2$time[i] < int2$time[i-1]){
      int2 <- int2[-(i-1),]
    }
  }

  # Add the time needed for contact between mycorhizes of plant 1 and 2 as new column of "contact"
  time <- NULL
  for (t in 1:dim(contact)[1]){
    time[t] <- max(int1$time[t],int2$time[t])
  }
  (contact <- cbind(contact, time))

  # Get the minimal time needed before a mychorhize communication between plant 1 and 2
  (min_time <- min(contact[,3]))
}
```

Annexe IV : Modelling of the matrcices of excudates and root receptors

```
Exudate1 <- array(0, dim = c(ceiling(system_height/region), ceiling(system_width/region),
                             max(ceiling(root2D_1$time))))

for (t in 1:max(ceiling(root2D_1$time))){
  for (i in 1:dim(root2D_1)[1]){
    if ((ceiling(root2D_1$time)[i] == t | ceiling(root2D_1$time)[i] == t-1 | ceiling(root2D_1$time)[i] =
      Exudate1[abs(root2D_1$z[i]), root2D_1$x[i],t] = Exudate1[abs(root2D_1$z[i]), root2D_1$x[i],t] + 1
    }
  }
}

isRoot2 <- array(0, dim = c(ceiling(system_height/region), ceiling(system_width/region),
                            max(ceiling(root2D_2$time))))

for (t in 1:max(ceiling(root2D_2$time))){
  for (i in 1:dim(root2D_2)[1]){
    if (ceiling(root2D_1$time)[i] <= t & root2D_1$type[i] != 20 & root2D_1$type[i] != 50) {
      isRoot2[abs(root2D_2$z[i]), root2D_2$x[i],t] = 1
    }
  }
}
```

Annex V : Diffusion Model

```
#diffusion Model

  Ly = ceiling (hauteur_totale/region)            #  domain length along y  (Cm)
  Lx = ceiling(largeur_totale/region)             #   domain length along x  (cm)
  K  = 0.018                                      # diffusion coefficient((cm?/day))
  q  = 0.0                                        # linear reaction rate (unit of mol?cule / day)
  Ncolonne = dim(isRoot2)[2]   #Nx                # number of elements along x [=> Nx+1 nodes)
  Nligne = dim(isRoot2)[1]     #Ny                # number of elements along y (=> Ny+1 nodes)
  Dx = Lx/Ncolonne                                # grid size along xded
  Dy = Ly/Nligne                                  # grid size along y
  T  = max(ceiling(root2D_2$time))                # integration time    (day)
  dt = 1
  Nt = T*ceiling(T/(T*dt))


  # % Initialization
  # % --------------
  t=seq (from=0, to = Nt, by=dt)
  xi=seq(from=1 ,to = Lx, by = 1)                                  # coordinates of grid nodes
  yi=seq(from=1 ,to = Ly, by = 1)
  #x<-matrix(xi,nrow=length(yi),ncol=length(xi),byrow=TRUE)
  #y<-matrix(xi,nrow=length(yi),ncol=length(xi),byrow=FALSE)



  Cold =  matrix(0,nrow=Nligne,ncol=Ncolonne,byrow=TRUE)
```

```
Cold=Cold*0
Cnew  = Cold;



# % Integration of the equation
# % --------------------------
for (k in 1 :\selectlanguage{ngerman} Nt){
  for (i in 1 :  Nligne){
    for (j in 1 : Ncolonne){
      Cij = Cold[i,j]

      if (j>1){     Cl = Cold[i,j-1] } else{ Cl = Cij}    # end left
      if (j<Ncolonne){  Cr = Cold[i,j+1] } else{ Cr = Cij}# end right
      if (i>1){     Cd = Cold[i-1,j] } else{ Cd = Cij}    # end % down
      if (i<Nligne){  Cu = Cold[i+1,j] } else{ Cu = Cij}  # end % up
      # diffusion
      difx_ij = K*dt*(Cr-2*Cij+Cl)/(Dx*Dx);
      dify_ij = K*dt*(Cu-2*Cij+Cd)/(Dy*Dy);
      # new solution
      Cnew[i,j] = Cij  + difx_ij + dify_ij

    }
  }

  Cold = Cnew;
  Cold=Cold+Exudate1[,,k]



  if (! all((round(isRoot2[,,k]*Cold,5)==0))) {   # we keep 5 significative number, so the isgnal have
  print('limit reached')
    break}                              #produit terme Ã  terme de diffusion et position de racine perme
  print(round(Cnew*100,3))
  print(c('jour=',k*dt,'nti=',k))  }
```