**General**

**Compiling**

The model programs are written in C++ and can be compiled by typing “make” on the command line in the main directory (the directory containing the include/ src/ bin/ obj/ directory and Makefile).

**Running the model**

The executable (binary) of a model can be run by typing “./bin/digitalroot xxx” on the command line in this same main directory. Here xxx stands for the directory in which output will be placed which needs to be created a priori by the user. Please make sure to create a novel directory with a different name if you do not want your prior output to be overwritten.

**Single cell**

**Non-sweep settings**

To perform individual single cell simulations, go to directory singlecell.

To achieve single cell settings in the file include/Plant\_layout.hh Y2 and Y4 are set to to zero and Y3 at 1.

If the single cell model is run in non-sweep settings, output is generated by the code in src/Plotting.cc, generating a file with the values of the different model variables for every so manyth timepoint. This output can be easily visualized with programs like Xmgrace, gnuplot, excel etc.

In the file include/Param\_config.hh precise model settings can be varied through making use of 4 so-called define statements (starting with #define).

#define SINGLECELL1 corresponding to medium strength HDZIPIII ANT repression, this setting was used to generate Fig S8A.

#define STRONGANTREPRESSION corresponding to maximum strength HDZIPIII ANT repression, these model settings are also used in a subset of multicellular simulations

#define STRONGSEQUESTRATION corresponding to simulations in which PXY mediated TDIF binding actually reduces the amount of free TDIF, and TDIF-PXY complex undergoes degradation

In the paper these settings are not actually used for single cell simulations, but only for a subset of the multicellular simulations. They are provided here to enable the user to investigate the consequences of these different settings.

To vary the amount of auxin experienced by the cell change the value of auxmax\_min in include/Param\_config.hh. To vary the amount of total TDIF experienced by the cell change the value of globalCLEprod\_min in include/Param\_config.hh. By varying these two the FigS8A results can be created.

To create a series of simulations, in which all parameters are kept the same, but this parameter set is subjected to the matrix of 66 auxin \* TDIF values also used in the parameter sweep, in include/Param\_config.hh uncomment the #define MULTIRUN and put the values of auxmax\_min and globalGLEprod\_min both to o.

**Sweep settings**

To perform a single cell parameter sweep go to directory singlecellsweep.

To achieve single cell settings in the file include/Plant\_layout.hh Y2 and Y4 are set to to zero and Y3 at 1.

Instead of having single valued parameters, in include/Param\_config.hh we now defined upper and lower boundaries as well as increment sizes for the parameters used in the sweep. As was the case for the multirun settings in the singlecell directory, also for auxin and TDIF production upper and lower bounds and increment sizes are set to expose each different parameter combination to a matrix of auxin and TDIF values.

In sweep settings we use #define SIMPLEPLOT in include/Param\_config.hh file to suppress the plotting done in no sweep settings as this would generate an excessive amount of output that would substantially enhance run time. Instead separate routines are written that generate PNG files depicting the averaged final outcome of the parameter sweep. These routines can be found in the src/Main.cc. PNG files created represent output for the auxin level times TDIF level matrix for which the parameter sweep is done, with auxin levels running from high to low from left to right, and TDIF levels running from low to high from bottom to top, similar to the parameter sweep figures in the manuscript. PNG files are created to depict the fraction of runs converging for a particular auxin-TDIF combination to xylem (blue), phloem (green) or cambium (grey fate) (SWEEPresults.png) as well as to depict the average expression of ANT, PLT, HDZIP3, non-bound PXY and TDIF/CLE-PXY complex (titles of files are self-explanatory). Note that these functions are not recreating exactly the figures of the paper (somewhat different coloring scheme used in these simplified functions), but should enable the user to compare in detail outputs to results in the paper.

Note that in the include/sweep/Param\_config\_sweep.hh file provided we have doubled the stepsize applied in the varying of parameters between their minimum and maximum values, as a consequence instead of having a 10 dimensional parameter space encompassing 5^7\*7^3 different parameter settings to run, this results in a 10 dimensional parameter space encompassing 3^7\*4^3 different parameter settings to run resulting in an approximately 100 fold reduction in run time of the parameter sweep. If desired users can reverse back to the orginal parameter step sizes, their values can be found uncommented directly behind the currently applied values.

Note that as part of the parameter sweep simulation a hierarchical directory structure is generated, which is not filled with files but instead serves as a means of checking the dimensionality of the parameter sweep and its progress.

**Multicellular**

To perform multicellular simulations, go to the directory multicellular.

To control the number of cells in your simulation go to include/Plant\_layout.hh. Here put Y2 and Y4 to 1, and put Y3 to the number of cells you want to simulate minus 2. The model is currently set to simulate a total of 4 cells.

To control the precise modeling settings go to include/Param\_config.hh. Here you find 3 so-called define statements (starting with #define) that can be used to switch between different model settings used in the different figures of the article:

#define INITIALMULTICELL corresponding to model settings of figure Fig S9A,B;

#define STRONGANTREPRESSION corresponding to model settings of Fig 4A,E ; S9C

#define STRONGSEQUESTRATION corresponding to model settings of Fig 4B, F; S9D; S11

Note that only one of these define statements should be switched on, the other ones should be outcommented (have a // in front of them).

To control the level of PXY expression (varied in figures 4A, E), change the value of maxPXY in include/Param\_config.hh

To control the level of TDIF added to all cambium cells (done in figure 4B,F) change the value of globalCLEprod\_min in include/Param\_config.hh

To vary diffusion rate of TDIF or ANT/PLT (varied in Fig S ….) change the value of CLE41diff or pltsdiff, respectively in include/Param\_config.hh

In the multicellular simulations output is generated in terms of png files depicting the levels of auxin, HDZIPIII, TDIF free, PXY free, TDIF-PXY complex, ANT and PLT every so many timesteps. In addition to separate png files per model variable also a combined PNG file is created.

Additionally flat text files are generated per model cell, storing variable dynamics as a function of time for that cell. This output can be easily visualized with programs like Xmgrace, gnuplot, excel etc.

**Alternative multicellular code**

An alternatively, but similar working, multicellular code is provided in the directory multicellular\_homogeneousTDIFdiffusion

In this code, instead of only implementing diffusion of TDIF in the cell walls separating the cells in 2D, we implemented homogeneous diffusion across the simulated 2D tissue, so in both the cell walls “overlaying” the simulated 2D cells and those separating them in 2D.

Note that if the diffusion coefficient is the same as in the original code, diffusion is too slow and no effective gradient reaching PXY receptors is formed. This is caused by the fact that in the original code effectively there is infinitely rapid diffusion in cell walls “overlaying” the cells and finite diffusion in the cell walls separating the cells in the 2D plane. It can be shown through a homogenesation approach that, given the used sizes of cells and walls, therefore effective diffusion is approximately 23 times higher than suggested by the value of the diffusion coefficient. Therefore to achieve similar results a 23-fold increase in diffusion coefficient is necessary in this alternative code.