**User guide of SurEau-Ecos v2.0.0**

Authors : Julien Ruffault ([julien.ruff@gmail.com](mailto:julien.ruff@gmail.com)), François Pimont ([francois.pimont@inrae.fr](mailto:francois.pimont@inrae.fr)), Hervé Cochard ([herve.cochard@inrae.fr](mailto:herve.cochard@inrae.fr)), Jean-Luc Dupuy (jean-luc.dupuy@inrae.fr), Nicolas Martin-StPaul ([nicolas.martin@inrae.fr](mailto:nicolas.martin@inrae.fr))

\*\*Please contact Julien Ruffault and Nicolas Martin-StPaul to report any bugs or for assistance

# Introduction

*SurEau-Ecos* is a mechanistic plant water model designed to simulate water fluxes, fuel moisture content and plant mortality at stand level according to a set of vegetation traits and soil physical properties. This manual presents the essential steps to configure and run *SurEau-Ecos.v2.0.0* in *R*

SurEau-Ecos v2.0.0 can be downloaded at the following link (download the last release):

[**https://github.com/julien-ruffault/SurEau-Ecos**](https://github.com/julien-ruffault/SurEau-Ecos)

*SurEau-Ecos* is delivered as a zip file. Files within the main directory must not be changed.

Before starting, the directory of the model (*mainDir*) should be defined, such as:

mainDir = "/Users/Name/SurEau-Ecos\_v2.0.0"

Then, source all functions in your R environment with the following line of code:

source(paste0(mainDir,'/functions/load.SurEau-Ecos.R'))

*Sureau-Ecos* can be run with the function run.SurEau-Ecos(). This function has **six** different arguments containing the required variables to run the model. All these arguments must be specified and declared in the following order:

1. modeling options

2. simulation parameters

3. climate data

4. stand parameters

5. soil parameters

6. vegetation parameters

The R script ‘/*quick\_start/example\_launcher.R*’ is an example on how to run a basic simulation of *SurEau-Ecos* (with a ‘standard’ configuration of modeling options) that can be used by the users as a template to build their own simulation.

The six input lists of *SurEau-Ecos*’s parameters and how they can be initialized are described in the following chapters. A more specific description on how vegetation and soil is represented in *SurEau-Ecos* and how to determine those parameters will soon be provided in a more specific documentation.

# Modeling options

Modeling options indicate the options for the implementation of *SurEau-Ecos* and allows the user to choose between several mechanisms affecting the behavior of the model.

This list **must** be created by the function create.modeling.options(). The function takes several arguments as input (see Table 1). Note that some of these options might also change the list of climate data and the stand, soil or vegetation input parameters that are required to run the model (e.g., StomatalRegFormulation indicates the type of stomatal regulation used in the model and therefore determine the parameters that should be provided).

Examples

# basic configuration of *SurEau-Ecos*

modeling\_options = create.modeling.options()

# configuration to run SurEau-Ecos with a constant climate

modeling\_options = create.modeling.options(constantClimate=T)

|  |  |  |
| --- | --- | --- |
| **R parameter** | **Description** | **Comment** |
| constantClimate | a logical value indicating whether a constant climate will be used during the simulation (default = F). If set to ‘T’, the first line of the climate input file will be repeated to generate the climate data.frame in create.climate.data() |  |
| timeStepForEvapo | a numerical value (in hours) indicating the time step for the main evapotranspiration loop. Should be one of the following 1,2,4,6,8 (default = 1).  NB: This is the time step at which climate data are sampled (before being interpolated in the small time step loop). Computations becomes inaccurate for timeStepForEvapo>2h, as the range of daily variations of climate conditions and associated stomatal regulation and fluxes are not well discretized | for advanced users only |
| numericalScheme | The numerical scheme used to solve water balances : ’Implicit' (default), 'Semi-Implicit' or ‘Explicit’ | for advanced users only |
| compOptionsForEvapo | "Normal" (Default): adaptive time step (10, 6, 3, 1 min)  "Accurate" : fixed time step (10 s)  "Fast" : adaptive time step (1 hour, 10 min)  "Custom" : specify your small time step (parameter “customSmallTimeStepInSec”, default is 600 s =10 min) | for advanced users only |
| soilEvap | A logical value to activate or deactivate the computation of soil evaporation. NB: this is computed on the large time step and might raise convergence issues on large time steps larger than 2h. | for advanced users only |
| Lcav and Tcav | Water released by cavitation is by default redistributed to adjacent cells (for respectively Leaf and Trunk apo). Default is 1 (for redistribution). 0 corresponds to absence of redistribution. | for advanced users only |
| ETPFormulation | the formulation of ETP to be used, either 'PT' (Priestley-Taylor, Default) or 'P' (Penman). | Penman formulation not implemented yet |
| RnFormulation | The method to be used to derive net radiation from global radiation, either 'Linacre' (default) or 'Linear' | the linear method is not implemented yet |
| stomatalRegFormulation | The type of regulation to be used for stomatal response to leaf symplasmic water potential, either 'Sigmoid' (default) 'PiecewiseLinear' or ‘Turgor’. |  |
| avoidWaterSoilTransfer | a logical value indicating whether the transfer of water between soil layers should be avoided by disconnecting the soil layers that get refilled from the soil-plant system (default =F). | Not implemented yet |
| defoliation | a logical value indicating whether trees should loose leaves when occurs.cavitation occurs of the above part of plant (default =F) . Defoliation starts only when PLC\_TL > 10% . |  |
| resetSWC | a logical value indicating whether soil layers should be refilled at the beginning of each year (default=F) |  |
| printProg | A logical value indicating wheter model progression should be printed on screen (default =F) |  |
| thresholdMortality | A numeric value indicating the PLC value (in %) above which the plant is considered dead and simulation stops for the current year. Default value is 90 |  |
| soilEvapo | A logical value indicating whether soil evaporation should be simulated (T) or set to 0 (F) (default=T) |  |

# Simulation parameters

Simulation parameters is a list that indicates the time period for the simulation and also configures the writing and format of the output simulation file. Simulation parameters must be created by the function create.simulation.parameters()(see table below).

startYearSimulation and endYearSimulation arguments specify the time period for the simulation and must be set according to the period covered by the climate input data (see section 5).

Output data is written in an output csv file whose path must be specified by the argument outputPath. outputResolution specifies which one of the three resolutions is chosen for output writing in file : ‘subdaily’ (i.e., similar timestep of timeStepForEvapo specified in modeling options), ‘daily’ or ‘yearly’ time scale.

The type of output that must be written is specified by the outpuType argument. By default,a “simple'' output type is chosen (‘simple\_subdaily’, ‘simple\_daily’ or ‘simple\_yearly’ type) according to the chosen resolution. Two more output types are already implemented for the sub-daily time scale, ‘diagnostic subdaily’ (which writes all possible outputs) and ‘LFMC-subdaily’ (used for fuel moisture simulation purposes). Users can also specify their own output configuration. In that case, output names should be provided as a csv file with ‘;’ as separator and placed in the “functions/ouput\_types' directory.

Examples

# create simulation parameters to run SurEau-Ecos on the period from 1990 to 1992 with ‘LFMC’ output type at the subdaily time scale

output\_path = paste0(mainDir, '/quick\_start/example\_output\_subdaily.csv')

simulation\_parameters = create.simulation.parameters(startYearSimulation = 1990,endYearSimulation = 1992, mainDir = mainDir, outputType = 'LFMC\_subdaily', outputPath = output\_path)

|  |  |  |
| --- | --- | --- |
| **R parameter** | **Description** | **Comment** |
| mainDir | Main directory of the model |  |
| startYearSimulation | a numeric indicating the starting year for the simulation (must match the dates of the input climate file) |  |
| endYearSimulation | a numeric indicating the last year for the simulation (must match the dates of the input climate file) |  |
| resolutionOutput | the resolution for the output simulation file. Must be 'subdaily' (default), 'daily' or 'yearly' |  |
| outputType | the type of output chosen. if not provided set to ‘simple\_subdaily’, ‘simple\_daily’ or ‘simple\_yearly’ according to resolutionOutput. |  |
| outputPath | the path of the output simulation file |  |
| overWrite | a logical value indicating whether the chosen output path can be overwritten if it already exists (default = F) |  |

# Climate data

Weather input data must include variables at the **daily** scale. Daily data will be disaggregated at the time step specified in modeling options (R parameter: timeStepForEvapo). Note that at this stage, **it is not possible to use subdaily data** as input in *SurEau-Ecos*. This will be added in the next version of the code.

run.SurEau-Ecos.R() is the function to create climate data for SurEau-Ecos. Itstakes three arguments : filePath (path a csv file containing the climate data), modeling\_options (a list containing the modeling options created with create.modeling.options()) and simulation\_parameters a list containing the simulation parameters created with create.simulation.parameters())

Weather data should be arranged in a csv file with days in rows and variables in columns, ‘;’ as field separator character and ‘.’ as decimal character. For each row the Date must be provided in the following format : “dd/mm/yyy”. Only the data on corresponding to startingYear and endYear specified in the list of simulation parameters will be retained. Table 3 indicates the symbols, units, definitions and variable name.

Example

# load climate data from test and select the period from 2005 to 2006

climate\_path = paste0(‘mainDir’, test\_simulation/climate\_data\_test.csv’)

modeling\_options = create.modeling.options()

simulation\_parameters = create.simulation.parameters(starting\_year =2005 ,endYear = 2006)

climate\_data = create.climate.data(filePath = climate\_path, modeling\_options, simulation\_parameters)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Symbol** | **Unit** | **R parameter** | **Description** | **Comment** |
| Date | - | DATE | Date of the day ( "dd/mm/yyyy") | if 'constantClimate' is set to T |
| Tmean | °C | Tair\_mean | Mean daily temperature | - |
| Tmin | °C | Tair\_min | minimum daily temperature | - |
| Tmax | °C | Tair\_max | maximum daily temperature | - |
| Rg | MJ.m-2 | Rg\_sum | daily global radiation | - |
| ppt | mm | PPT\_sum | precipitation | - |
| RHmean | % | RH\_mean | mean daily relative humidity | - |
| RHmin | % | RH\_min | minimum daily relative humidity | - |
| RHmax | % | RH\_max | maximum relative humidity | - |
| u | m.s-1 | WS\_mean | mean daily wind speed | - |

# Stand parameters

Stand parameters include information on the stand.? Stand parameters are created with the function create.stand.params(). Stand parameters should be provided as a csv file with ‘;’ as separator and ‘.’ for decimal or can be provided directly as arguments of the function. The following table indicates the symbols, unit and description of each stand parameter.

Exemple

stand\_parameters=create.stand.parameters(LAImax=4,lat=48.73,lon=6.23)

|  |  |  |  |
| --- | --- | --- | --- |
| **Symbol** | **Unit** | **R parameter** | **Description** |
|  | - | LAImax | Maximum Leaf area index |
| *lat* | *deg* | Lat | Latitude of the stand |
| *lon* | *deg* | lon | Longitude of the stand |

# Soil parameters

Soil is represented as a three-layer bucket whose physical properties allow the estimation of soil water retention and hydraulic properties. Note that in *Sureau-Ecos,* the word soil refers to the depth that plant rooting systems can reach, including cracks within the bedrock. Specifying layers with an elevated rock fragment content may be important in seasonally-arid climates like the Mediterranean, because plants often extend their roots into cracks existing in the parent rock to access water during summer.

The number of soil layers is fixed. Variations of soil and rhizosphere conductance (ksoil), and mean soil water potential in the root zone (Psi\_soil) are calculated with the van Genuchten–Mualem equations.

Soil parameters are created with the function create.soil.params()Soil parameters should be provided as a csv file with ‘;’ as separator and ‘.’ for decimal. The following table indicates the symbols, unit and description of each soil parameter.

Example

soilPath=‘/quick\_start/soil\_example.csv' # path to the example soil dataset

# Create a soil

soil\_parameters=create.soil.parameters(filePath=soilPath, modeling\_options)

|  |  |  |  |
| --- | --- | --- | --- |
| **Symbol** | **Unit** | **R parameter** | **Description** |
| *d1* | m | depth\_1 | Maximum depth of the 1st soil layer |
| *d2* | m | depth\_2 | Maximum depth of the 2nd soil layer |
| *d3* | m | depth\_3 | Maximum depth of the 3rd soil layer |
| *rfc1* | % | RFC\_1 | Rock fragment content of the 1st layer |
| *rfc2* | % | RFC\_2 | Rock fragment content of the 2nd layer |
| *rfc3* | % | RFC\_3 | Rock fragment content of the 3rd layer |
| *𝛂* | MPa-1 | alpha\_vg | Shape parameter |
| *n* | - | n\_vg | Pore size distribution index |
| *I* | - | I\_vg | shape parameter |
| *Ksat* | mmol.m-2.s-1.MPa-1 | Ksat | Soil hydraulic conductivity at saturation |
|  | cm3.cm-3 | saturation\_capacity\_vg | Soil water content at saturation |
|  | cm3.cm-3 | residual\_capacity\_vg | Residual soil water content |
| *gsoil0* | mmol.m-2.s-1 | gSoil0 | Soil conductance at saturation |

# 8. Vegetation parameters

The plant is represented organs that have an apoplasm and a symplasm. SurEau-Ecos has **two plant main organs** : (1) the **canopy leaves**, and (2) a **stem** which includes sapwood of the branches, the trunk and the roots. SurEau-Ecos has four plant compartments: leaf apoplasm and leaf symplasm, stem apoplasm and stem symplasm.

To simplify the parameterization of the hydraulic patchway, mainly one **total plant hydraulic conductance per unit leaf area** kPlantInit is required. It is internally distributed among plant organs assuming that 50% of the resistance is belowground in accordance with the root distribution between the three soil layers, and 50% is located aboveground and mostly in the leaf symplasm

Vegetation parameters are created with the function create.vegetation.parameter()Soil parameters should be provided as a csv file with ‘;’ as separator and ‘.’ for decimal. The following table indicates the symbols, unit and description of each parameter.

|  |  |  |  |
| --- | --- | --- | --- |
| **Symbol** | **Unit** | **R parameter** | **Description** |
|  | - | Foliage | Foliage type (‘Evergreen’ or ‘Deciduous’) |
|  |  | DayStart | Initial date of the forcing period for leaf phenology (for ‘Deciduous’) |
|  |  | Tbase | Minimum temperature to start cumulating temperature for budburst (for ‘Deciduous’) |
| *F\** |  | Fcrit | amount of forcing temperature to reach budburst (for ‘Deciduous’) |
|  |  | nbdayLAI | LAI growth rate per day (for ‘Deciduous’) |
|  |  | CanopyStorageParam | Canopy water storage capacity |
| *k* | *-* | K | Light extinction parameter |
|  |  | EpsilonSymp\_Leaf | modulus of elasticity of the leaf symplasm |
|  |  | PiFullTurgor\_Leaf | Osmotic potential at full turgor of the leaf symplasm |
|  |  | EpsilonSymp\_Trunk | modulus of elasticity of the stem symplasm |
|  |  | PiFullTurgor\_Trunk | osmotic potential at full turgor of the stem symplasm |
|  |  | slope\_VC\_Leaf | Slope of rate of leaf embolism spread at . |
|  |  | P50\_VC\_Leaf | Water potential causing 50% loss of leaf hydraulic conductance |
|  |  | slope\_VC\_Trunk | Slope of rate of stem embolism spread at . |
|  |  | P50\_VC\_Trunk | Water potential causing 50% loss of stem hydraulic conductance |
|  |  | kPlantInit | Maximum conductance from the root surface to the stem apoplasm |
|  |  | k\_TSymInit | Conductance from the stem apoplasm to stem symplasm |
|  | *-* | ApoplasmicFrac\_Leaf | Leaf apoplasmic fraction (from RWC leaf) |
|  | *-* | ApoplasmicFrac\_Trunk | Stem apoplasmic fraction of the wood water volume |
|  | *-* | SymplasmicFrac\_Trunk | Stem symplasmic fraction of the wood water volume |
|  |  | C\_LApoInit | Capacitance of the leaf apoplasm |
|  |  | C\_TApoInit | Capacitance of the stem apoplasm |
|  |  | VolumeLiving\_TRB | Volume of tissue of the stem compartment (includes the root, trunk and branches) |
|  |  | Succulence | Leaf succulence (water content per unit leaf area) |
|  |  | LDMC | Leaf dry mater content (Dry mass over saturated mass) |
|  |  | LMA | Leaf mass per area |
|  | *-* | betaRootProfile | Shape parameter for root distribution |
|  | *-* | fRootToLeaf | Root to leaf area ratio |
|  |  | rootRadius | Root diameter |
|  |  | P50\_gs | Water potential causing 50% stomatal closure |
|  |  | slope\_gs | Rate of decrease in stomatal conductance at |
|  |  | gsNight | Minimum stomatal conductance |
|  |  | gsMax | Maximum stomatal conductance |
| *δ* | *-* | JarvisPAR | Response of to light |
|  |  | Tgs\_optim | Temperature at maximal stomatal conductance |
|  |  | Tgs\_sens | Stomatal sensitivity to temperature |
|  |  | gCrown0 | Reference crown conductance |
|  |  | gmin20 | Cuticular conductance at 20°C |
|  | *-* | Q10\_1\_gmin | Temperature dependance of |
|  | *-* | Q10\_2\_gmin | Temperature dependance of |
|  |  | TPhase\_gmin | Temperature for transition phase of |