

FORHYCS Technical Documentation

Model version: 1.0.0
Date: September 2019
Author: Matthias Speich, WSL

LICENSE	4
1. INTRODUCTION	5
2. MODEL OVERVIEW	5
2.1. Model structure and flow	5
2.2. Format of input and output grids	6
2.3. Required inputs	7
2.3.1. Parameter values	7
2.3.2. Static gridded data	7
2.3.3. Meteorological data	7
2.4. Configurations	8
2.5. Auxiliary files	8
3. MODIFICATIONS FROM SOURCE MODELS	8
3.1. Local water balance	8
3.2. Canopy variables	9
3.2.1. Maximum Annual Leaf Area Index (LAI)	9
3.2.2. Fractional Canopy Cover (FCC)	9
3.2.3. Phenology	9
3.2.4. Interception storage	9
3.3. Minimum stomatal resistance	9
3.4. Drought index	9
3.5. Dynamic maximum tree height	10
3.6. Leaf Area Index reduction	10
3.7. Snow-induced seedling mortality	10
3.8. Rooting depth estimation	10
3.9. Dynamic land cover	10
3.10. Additional output routines	11
3.11. Glacier retreat module	11
4. PRE-PROCESSING / NEW INPUTS	12
4.1. Soil water holding capacity	12
4.2. Ecoregions for forest output	12
4.3. Glacier files	12
4.4. Control parameters	12

5. RUNNING FORHYCS ON AN HPC CLUSTER	14
5.1. Directory structure	14
5.2. Modules	15
5.3. Launch scripts	16
5.3.1. imet files	16
5.3.2. run files	16
5.4. Spin-up and future simulations	17
5.4.1. Spin-up	17
5.4.2. Future simulations	17
REFERENCES	18

License

FORHYCS v1.0.0 is free software, distributed under the terms of the General Public License (GPL) v3. Please see <https://www.gnu.org/licenses/>. Copyright (C) 2014-2019 Matthias Speich, Massimiliano Zappa, Marc Scherstjanoi, Heike Lischke.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

1. Introduction

This document describes the technical aspects of the spatially distributed ecohydrological model FORHYCS. FORHYCS is based on the coupling of two pre-existing models, the hydrological model PREVAH and the forest landscape model TreeMig. Both these models have already been documented, so that this document focuses on the changes made to the original source code, as well as on the new interface modules. Furthermore, this document shows how to launch a FORHYCS simulation, including pre-processing steps (excluding pre- and post-processing tools of the source models). A TreeMig user guide can be obtained from the model page on WSL's website (<https://www.wsl.ch/en/projects/treemig.html>), while the technical description of PREVAH is given by Viviroli et al. (2007).

The source code of FORHYCS is entirely written in Fortran (using the Fortran 77 format for most of the PREVAH code and the Fortran 90 format for the TreeMig code as well as the new model components), whereas the new pre- and post-processing tools are mostly written in R. It is assumed that the reader has access to the source code and is familiar with these two languages. FORHYCS was written for parallel execution using the MPI (Message Passing Interface) standard. The model code was written for compilation and execution with Intel Fortran (version 11 or higher) and Intel MPI, and has not been tested with other compilers and MPI implementations.

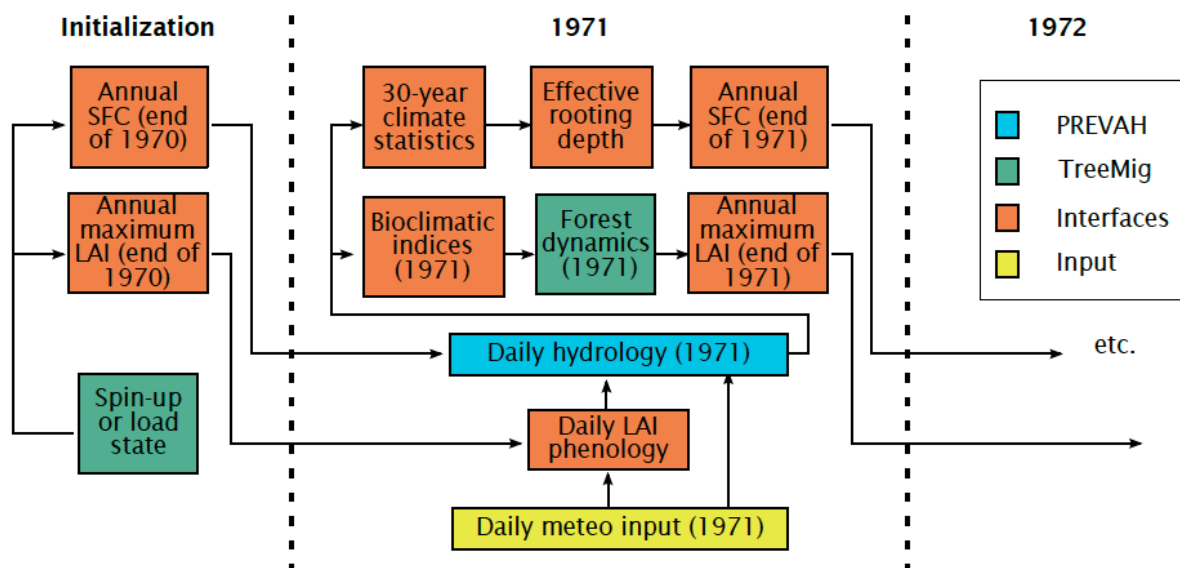
The following applies for the description of the code in this document: for better legibility, variable names are written in *italic*, subroutine names in ***bold italic***, and file names in ***bold italic***, with the file extension (e.g. ***example.f90***). If not indicated otherwise, the code for a subroutine is stored in a file of the same name.

Many of the input and output files contain the name of the catchment under study. A convention is to use a short catchment name, consisting of three letters and a number, the latter expressing the cell size length in meters. For example, for a study in the catchment of a river named "Gougra" with a cell size length of 200 meters, one might use the name "Gou200". In the following descriptions, the placeholder "CatXXX" is used.

2. Model overview

2.1. Model structure and flow

The model logic of FORHYCS is described in detail in the paper on the Val d'Anniviers case study (Speich et al., GMDD). In this section, a summary of the model's structure and functioning is given. The parts of the code that have been added during the development of FORHYCS are described in more detail in Section 3.



The model flow of FORHYCS is schematically shown above. In this example, the model run starts in the year 1971. To initialize the forest state, it is necessary to either run a spin-up, or load the saved state from a previous run (see Section 5.4.). The main interfaces between the forest and hydrological processes are leaf area index (LAI) and rooting zone water storage capacity (SFC). Both are updated annually, based on the simulated forest dynamics (for LAI) and long-term climatic variables (for SFC). LAI is furthermore transformed to daily values, to include leaf phenology. Daily meteorological data are used to drive the hydrological simulations. From the daily water balance calculations, an annual drought index is derived. Together with temperature-based indices derived from daily temperature, this drought index is passed to the forest model, where it influences the growth and mortality of the different tree species. In the next year, the updated canopy characteristics are again passed to the hydrological modules.

2.2. Format of input and output grids

As FORHYCS is a spatially distributed model, many of its inputs and outputs are in the form of grids. The format for these grids is the same binary format that was defined for PREVAH. Its structure is somewhat similar to that of Esri ASCII grids: it consists of a header containing metadata, followed by the actual data. Unlike for ASCII grids, the header consists of 12 instead of 6 elements. The first six elements are similar as for ASCII grids:

- Number of rows
- Number of columns
- Bottom left x coordinate
- Bottom left y coordinate
- Cell side length
- No Data value

The following six elements represent summary statistics on the data: sum, mean, minimum, maximum, standard deviation and coefficient of variation. In some of the new output routines, the calculation of these statistics is omitted, as they are not used by the new post-processing scripts.

The two R scripts *prevah_to_raster.r* and *raster_to_prevah.r* are an interface to read and write files in this format, respectively. In these scripts, grids are converted to/from raster objects, using the R **raster** package.

2.3. Required inputs

This section lists all of the inputs required to run FORHYCS. It differentiates between parameter values, static gridded data, and meteorological data. For the location of these files in the directory structure, please see Section 5.

2.3.1. Parameter values

The values of the tuneable parameters for PREVAH are given in two files, *CatXXX.inp* and *CatXXX.run*. See Sections 4.4 and 5.1 for more information on these files.

Parameter values for TreeMig are given in two separate files: one for spatially constant parameters, and one for species-specific parameters. The first file is named *Tst1100_2.3*, and contains the name of the second file (for now, invariably *TreePars2.3.txt*).

2.3.2. Static gridded data

FORHYCS requires a number of gridded input files, containing information e.g. on physiography, land cover soil properties. In addition, some of the model parameters of PREVAH are spatially distributed, and also need to be read in as a grid. For Switzerland, a consistent gridded dataset for 14 parameters was generated based on the regionalization procedure of Viviroli et al. (2009). This dataset may be obtained from the authors.

Aside from the regionalized parameter values, the following grids are required:

File name	Meaning	Units
CatXXX.dhm	Elevation	m asl
CatXXX.pus	Land cover	Code
CatXXX.exp	Aspect	°
CatXXX.slp	Slope	°
CatXXX.whc	Soil water holding capacity for 1 m depth	mm
CatXXX.pfc	Soil water holding capacity (old version)	mm
CatXXX.btk	Soil depth	m
CatXXX.kwt	Soil hydraulic conductivity (k-value)	

In addition, FORHYCS requires two grids dividing the study area into subcatchments and ecoregions, respectively. These grids are needed to aggregate daily hydrological outputs and annual forest outputs, respectively. In both grids, each zone is assigned a number, and outputs are aggregated (averaged) for all cells that share the same number. Ecoregions may be defined, for example, by dividing cells by aspect (north vs. south-facing) and elevation band. The subcatchments are given in the file *CatXXX.pwr*, and the ecoregions in the file *CatXXX.ecr*.

Note: all gridded data is numeric. In some cases, the values are integer numbers, e.g. the codes for different land cover types or the catchments and ecoregions. Nevertheless, all grids read by FORHYCS should contain real numbers only. Where necessary, the conversion to integer is done internally.

2.3.3 Meteorological data

PREVAH and FORHYCS require daily gridded input of precipitation [mm], air temperature [°C], relative humidity [%], global radiation [$W m^{-2}$], wind speed [$m s^{-1}$] and sunshine duration [h]. A separate gridded file (in the format described in Section 2.2) must be provided for each variable and day.

If FORHYCS is run in uncoupled mode (see Section 2.4 below), a separate file with annual bioclimatic indices for each cell must be provided. This is a single file in the form of a table, containing the following columns:

- X: x-coordinate of cell
- Y: y-coordinate of cell
- Year
- muDD: degree-day sum
- muWiT: minimum winter temperature
- muDrStr: (abiotic) drought stress index
- smday: last day of snow cover

The drought stress index provided in this file differs from the index used in coupled FORHYCS, and instead follows the definition of Bugmann and Cramer (1998).

2.4. Configurations

The intended purpose of FORHYCS is to run the model in coupled mode, to include all connections between hydrological processes and forest dynamics. Nevertheless, one might want to deactivate some of the connections, e.g. to test model behavior. It is possible to run FORHYCS in coupled mode, with specific processes not represented. Also, FORHYCS can be run in uncoupled mode (the hydrological and forest parts of the model are run simultaneously, but without any information transfer) or in one-way coupled mode (transfer of information from the forest model to the hydrological model only). Note that it is theoretically possible to set up FORHYCS for a reverse one-way coupling (transfer of information from the hydrological model to the forest model only). However, this configuration has not been tested.

The control parameters to define the configuration are defined in the first launch script (see Section 2.5 below) and written to the parameter file *CatXXX.inp*. See Section 4.4. for a list of the control parameters.

2.5. Auxiliary files

Two launch scripts, written in bash, are required to run FORHYCS. In a first script (*imet.sh*), the study region, control parameters and meteorological data sources are defined. It is possible to specify several study regions and several values for each control parameter. The script iterates over each of the specified options and launches a second script (*run.sh*) for each selected configuration. This second script sets up the directory structure for model output, then launches the FORHYCS executable.

3. Modifications from source models

3.1. Local water balance

The local water balance module of PREVAH calculates a bulk “soil evapotranspiration” term, implicitly combining transpiration and soil evaporation. In FORHYCS, (canopy) transpiration and understory/soil evaporation are calculated separately. The PREVAH subroutine *epemontd* was replaced by the new subroutine *evap*. These subroutines are called from *prevahcore*. In PREVAH, evaporation terms are calculated daily, and then disaggregated with the subroutine *tvert*. In FORHYCS, they are calculated hourly. The subroutine *evap* uses the new water balance module for forested cells, and the standard module of PREVAH for non-forested cells.

The balance of the soil water reservoir is calculated in the subroutine *abfgPatch* (*abfg6eth* in original PREVAH). The main difference between the new and original

subroutines is the two evaporative fluxes instead of one. This only affects the formulation of the rooting zone water storage – the runoff generation processes are unchanged between PREVAH and FORHYCS.

3.2. Canopy variables

3.2.1. Maximum Annual Leaf Area Index (LAI)

Annual Leaf Area Index (LAI) is calculated from the simulated number of trees in each species and height class, using the allometric equations of (Bugmann, 1994), which are already used in TreeMig to calculate light distribution. LAI is calculated in the subroutine *vegPars*. The allometric equations give the double- or all-sided LAI, i.e. the total area of leaves per unit ground area. FORHYCS uses a one-sided definition of LAI (so that the values are divided by 2 in *vegPars*), as well as a clumping factor of 0.8 for conifers (applied in *LeafPhenology*).

3.2.2. Fractional Canopy Cover (FCC)

In addition to LAI, the fractional canopy cover (FCC) is calculated in *vegPars*, using the formulation and parameterization of Zurbriggen et al. (2014).

3.2.3. Phenology

The annual LAI represents the conditions at full foliage cover. To account for seasonal variations in leaf area for deciduous trees, FORHYCS includes a daily phenology module. A species-specific variable, indicating the phenological status, is updated daily in each cell. This variable ranges from 0 (no leaves / dormant period) to 1 (full foliage). In spring, this index is a function of accumulated degree days, as well as the number of chill days (calculated daily in *ChillDaysAndGDD5*, called from *prevahcore*). The calculation of the phenological index and daily LAI and FCC is done in *LeafPhenology*, called from *prevahcore*. In autumn, the phenological status is reduced as a function of temperature and day length.

3.2.4. Interception storage

The interception storage is calculated as a function of Leaf Area Index and an empirical parameter, depending on forest type. This calculation is done in *prevahcore* using daily LAI values.

3.3. Minimum stomatal resistance

Minimum stomatal resistance of the forest canopy (RSMIN) is implemented as a function of the simulated species-size distribution. Tree species are divided in three classes of low, intermediate and high stomatal resistance. Furthermore, a linear increase of stomatal conductance (the inverse of resistance) with tree height is assumed. The range of values that RSMIN may take is between 125 and 555 s m⁻¹ (corresponding to a short tree with low resistance, and a tall tree with high resistance, respectively). The values for each species-size class are calculated by the subroutine *parstom* (called from *prevahopt* during model initialization). At the end of each TreeMig run, RSMIN is calculated for each cell in *vegPars*.

3.4. Drought index

The annual drought index used in FORHYCS is based on the ratio of actual and potential transpiration sums. Both these variables are calculated with an hourly time step in *evap*, and aggregated to daily values at the end of *prevahcore*. The 2D array *drstr* is updated at the end of each day in *prevahcore*. At the end of the year, its values are passed to the array *drstrOld* in *prevahopt*. On the next TreeMig call (March 1st), the

value in *dstrOld* is used in ***ReadNewBioclimIfNecessary***. There, the array *dstr30* is updated, which stores the drought index values of the last 30 years. By storing these past values, it is possible to use an average of several years, instead of an annual drought index. Currently, an averaging period of 3 years is hard-coded in ***ReadNewBioclimIfNecessary***.

3.5. Dynamic maximum tree height

In TreeMig, the species-specific parameter *HMax* denotes the maximum height that trees of a certain species can attain. In FORCLIM, a model that shares many process formulations with TreeMig, (Rasche et al., 2012) made this parameter variable in space and time, as a function of bioclimatic variables (drought and stress). For example, limited water availability may place a cap on tree height due to its effects on tree internal hydraulics. This formulation was also implemented in FORHYCS. The new subroutine ***adjHmax***, called from ***TimeLoop***, performs this calculation. The new value of *HMax* in each species and cell, *curhcmx*, is then used in ***CalcCurrentDepFuncsInCell***.

3.6. Leaf Area Index reduction

Leaf area may be reduced under drier conditions. For example, the model 3-PG (Landsberg and Waring, 1997) contains a drought- and temperature-dependent reduction of leaf area. Their formulation has been implemented in FORHYCS. The corresponding code was implemented in ***CalcCurrentDepFuncsInCell***. First results indicate that this LAI reduction lead to implausibly high fluctuations in stand biomass, so that it was disabled in the first FORHYCS case study (Val d'Anniviers).

3.7. Snow-induced seedling mortality

An additional mortality term for *Pinus* and *Larix* seedlings due to snow fungi was implemented in TreeMig by (Zurbriggen et al., 2014). This mortality term depends on the last day of snow cover. In TreeMig, this was estimated as a function of May and June temperature. Here, the snow-off date is obtained from the simulated snowpack in PREVAH (subroutine ***prevahcore***). The additional mortality term is calculated in ***CalcCurrentDepFuncsInCell***.

3.8. Rooting depth estimation

The plant-available soil moisture reservoir SFC is defined as the product of soil water holding capacity (WHC) and rooting depth. WHC is kept static (see Section on Pre-Processing/ New inputs), whereas rooting depth is updated annually. This is done in the subroutine ***RZstorage***, called from ***prevahopt*** at the end of each simulation year. The climate statistics needed by the rooting depth module are calculated from daily values in ***prevahcore***.

3.9. Dynamic land cover

Each cell belongs to a land cover type. In stand-alone PREVAH, most surface properties of a cell (e.g. LAI, RSMIN, rooting depth) are prescribed for each land cover class. (Schattan et al., 2013) summarized these classes into three types: currently forested, potentially forested, and not forested. Potentially forested areas are mostly (sub)alpine meadows that might be abandoned in the future. Non-forested areas include all classes where forest cannot grow due to natural (open water, bare rock) or anthropogenic factors (built-up, agriculture).

3.10. Additional output routines

FORHYCS produces output files for variables that were not produced by the original models. These outputs can be divided into three categories: grids over the whole simulation area, tables of daily hydrological variables, and tables of annual forest characteristics.

Grids of annual maximal LAI and vegetation height are written out after each TreeMig run, e.g. on each March 1st. For now, this code is located in the subroutine *prevahcore*. Daily hydrological variables (precipitation, soil evaporation, forest transpiration and evaporation from non-forested cells) are aggregated at the level of sub-catchments, and annual forest variables (biomass, LAI and stem number by species, as well as rooting zone storage capacity) are aggregated at the level of ecoregions. The grids assigning each cell to one catchment and ecoregion, defined in Section 2.3.2, are read by the subroutines *catchGridInit* and *ecoGridInit*, in the files ***catchGrid.f90*** and ***ecoGrid.f90***. The subroutines used to write the outputs, *MPIcatchWrite* (daily streamflow), *MPIcatchPrec* (daily precipitation), *MPIcatchEvap* (daily evaporation components) and *ecoGridWrite* (annual forest characteristics) are contained in the same files. These subroutines are called from *prevaho*.

3.11. Glacier retreat module

In future simulations with climate change scenarios, important changes to the volume and extent of glaciers should be expected. FORHYCS offers two options to consider glacier dynamics. The first option is to update the extent of glaciated cells based on new land-cover grids. These grids can be generated externally (e.g. using a glacier dynamics model) and the update takes place at pre-defined simulation times (e.g. every five years). As long as a cell is glaciated, the model assumes an unlimited quantity of ice available for melting. This procedure is also implemented in PREVAH, and has been used e.g. for the CCHydro simulations.

A second option is a dynamic glacier retreat module, based on the deltaH algorithm of Huss et al. (2010). This method explicitly considers ice thickness as a cell-level state variable. Thickness is updated yearly, based on mass balance of all the cells belonging to a glacier, as well as the position of each cell within a glacier. This method requires additional input files: a grid of bedrock elevation, and a grid identifying the glacier each cell belongs to (described in greater detail in the section on “Pre-processing / New inputs”).

The glacier retreat routine is implemented in the file ***TransGlac.f90***. Currently, all subroutines are written for parallel execution with MPI, and single-core execution was not tested. At the beginning of a simulation, the module ***glacInit*** (called from ***prevahinit***) reads the glacier input files and determines the number of glaciers, their initial extent, as well as the initial ice thickness in each cell. Information for each glacier is stored in an object of type *glacier*, defined in the module ***GlacGlob***.

During the simulation, the updating of glaciers is done each year on October 1st in the subroutine ***glacDeltaH*** (called from ***prevaho***). The water balance of each cell is calculated as the difference between the minimum amount of snow cover since last October (positive) and ice melt since last October (negative). Regarding snow, the assumption is that snow that remains for over a year becomes part of the glacier. The per-cell mass balance is then aggregated to the level of each glacier. Following the deltaH algorithm, the change in ice thickness resulting from the mass balance is distributed between the cells of a glacier depending on their relative position (elevation).

4. Pre-processing / New inputs

4.1. Soil water holding capacity

The grids of soil water holding capacity (WHC) have the extension `.whc` and are stored in the directory `run/prevah2d/grids/CatXXX/`. For Switzerland, the dataset used was provided by Jan Remund (Meteotest) and Sabine Augustin (BAFU), and is described in (Remund and Augustin, 2015). WHC is given in mm water that can be stored in the soil, assuming a depth of up to 1 m (where the soil is assumed to be shallower, the value is already reduced in the original dataset). As there are no reliable datasets of soil depth for Switzerland, the following workaround is used in FORHYCS: the WHC values are assumed to correspond to the volume of water that can be stored in 1 m of soil.

The original dataset has a cell width of 250 m. The R script file `soil_ch2018.r` contains a function to clip and resample this dataset to match the resolution and extent of a FORHYCS simulation region, and to write a `.whc` file in the binary PREVAH format.

4.2. Ecoregions for forest output

It may be desirable to evaluate the forest outputs (biomass, LAI, rooting depth) at the level of “regions” of cells with similar characteristics. These “ecoregions” can be specified in the `.ecr` files, to be placed in the directory `run/prevah2d/grids/CatXXX/`. Cells of the same ecoregion have the same code (an integer number). The R script `ecor_ch2018.r` sets the ecoregion codes based on elevation (using 200 m elevation bands) and aspect (north vs. south-facing cells). It is, however, possible to set ecoregions based on other criteria (e.g. slope, soil, etc.).

4.3. Glacier files

If FORHYCS is to be run with the `deltaH` glacier retreat module activated, additional input grids are necessary to represent ice thickness [m]. A separate grid must be provided for each glacier to which the `deltaH` module will be applied. In addition, a text file must be provided, with the complete file path of each glacier grid file, separated by line returns. The grid files must have the name structure ***GlacYYY.bin*** (where YYY is a three-digit code, padded with leading zeros if necessary), and must be placed in the directory `run/prevah2d/grids/CatXXX/`. The text file must be placed into the directory `run/prevah2d/inp/orig/`, and has the name structure ***CatXXX.glac***.

4.4. Control parameters

The new control parameters are specified in the `imet` launch script and passed to the `run` script. They are written to the `.inp` file, and in FORHYCS, they are read in ***readpin2d.inc*** (included in ***prevahinit.for***). The table below gives an overview of the new FORHYCS control parameters.

Parameter	Signification	Values
PRtoTM (1 st digit of <code>tpswitchlist</code>)	PREVAH to TreeMig coupling?	0: no PREVAH to TreeMig coupling, i.e. use BioClim drought index 1: enable PREVAH to TreeMig coupling, i.e. use FORHYCS drought index
TMtoPR (2 nd digit of <code>tpswitchlist</code>)	TreeMig to PREVAH coupling?	0: no TreeMig to PREVAH coupling, i.e. use prescribed vegetation properties 1: enable TreeMig to PREVAH

		coupling
prevahBC (3 rd digit of tpswitchlist)	Spin-up with TreeMig or FORHYCS	0: TreeMig spin-up 1: FORHYCS spin-up
LCdyn	Allow land cover change (i.e. whether forest may grow in potentially forested areas)	0: no land cover change allowed 1: land cover change allowed after spin-up only 2: land cover change allowed during and after spin-up
glacSwitch	Enable transient glacier module	0: no transient glacier module (glacier area is updated by reading in new land cover grids) 1: enable glacier module
rootslag	Number of years over which climate statistics are calculated for the rooting depth module	Any integer ≥ 1 (FORHYCS test runs in Val d'Anniviers used a value of 30).
dT	For idealized climate change: temperature modifier [K]	Any number (FORHYCS test runs used the values 0, 2, 4, 6 K)
dP	For idealized climate change: precipitation modifier [%]	Any number (FORHYCS test runs used the values -10%, 0%, 10%)
doHmax	Enable drought and temperature-based maximum height reduction	0: disable 1: enable
doLared	Enable drought and temperature-based leaf area reduction	0: disable 1: enable
doSFC	Enable dynamic rooting depth module	0: disable (rooting depth of 1 m is assumed) 1: enable
doSmort	Enable snow-induced mortality of Pinus/Larix seedlings	0: disable 1: enable
cdtol	Species-specific drought tolerance parameter set (for FORHYCS drought index only)	1: (Lischke and Zierl, 2002) 2: FORHYCS
doground	Do "Groundhog" spin-up (use meteo forcing for the same year, repeated many times)	0: disable (standard case) 1: enable
ghogyear	For groundhog spin-up: year for which meteorological forcing is used	Any year number (integer) for which meteorological forcing is available
grsmin	Enable effect of atmospheric CO ₂ on	0: disable 1 (or any integer > 0): enable

	stomatal resistance	
readFH	Read state files from a previous run	0: do not read state 1: read state
writeFH	Write state files (years for which this is done are currently hard-coded in prevahopt)	0: do not write state 1: write state

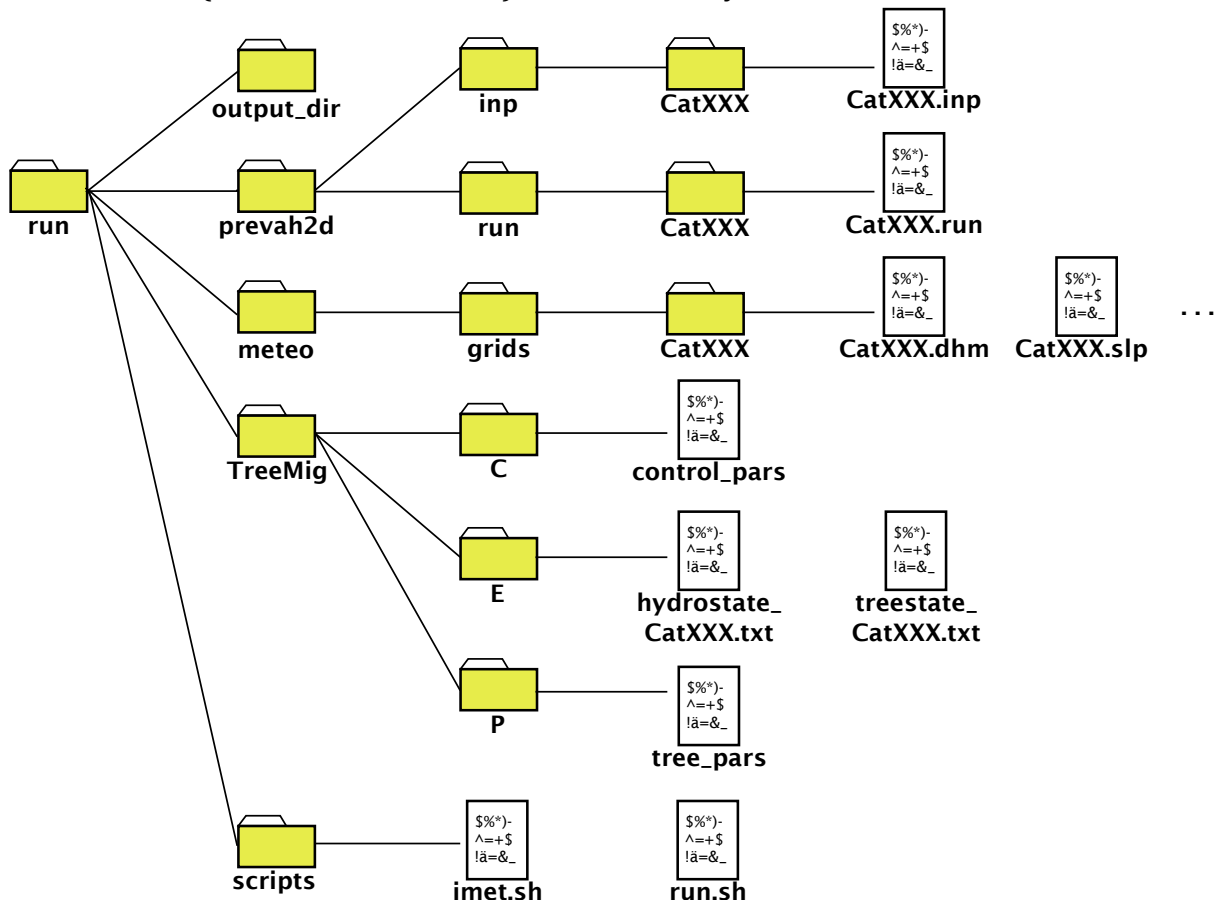
5. Running FORHYCS on an HPC cluster

5.1. Directory structure

The directory structure for FORHYCS is shown below. All files required for running FORHYCS need to be placed in a directory named *run*. Typically, its path will look like */home/yourname/yourproject/run/*. Hereafter, for simplicity, the paths to subdirectories of *run* are simply given as */run/subdir/*. The *run* directory and its subdirectories are organized as follows:

- Launch scripts (*/run/scripts/*): This directory contains the shell/bash scripts needed to run FORHYCS (the content of these files is explained below). In addition, the subdirectory */run/scripts/runlogs/* contains the log files of each run, which may contain valuable debugging information.
- Binary files (*/run/bin/*): This directory contains the executable files created when compiling the source code. There is usually no reason for the user to access these files directly.
- PREVAH inputs (*/run/prevah2d/*): Contains three subdirectories with the inputs for PREVAH:
 - grids (*/run/prevah2d/grids/CatXXX/*): Contains raster grids of physiographic inputs and regionalized parameter values. All grids are in PREVAH's own (binary) file format, and are generated with PREVAH's pre-processing tools. For each study area, a new directory must be created. In the file path given here, *CatXXX* serves as a placeholder for the name of the study area, which usually consists of three letters (abbreviation of the catchment name) and three digits (cell side length in meters, e.g. 200).
 - .inp files (*/run/prevah2d/inp/*): These files contain control parameters. The template file is located in */run/prevah2d/inp/orig/common_settings_2014*, and contains placeholders for all parameters. For each model run, a new .inp file is generated by the launch scripts, and placed in a new directory with the name of the current run.
 - .run files (*/run/prevah2d/run/*): Similarly to the .inp files, these files contain control parameters, as well as tuneable parameter values. A template file must be created for each study area (*/run/prevah2d/run/orig/CatXXX.run.orig*).
- TreeMig inputs (*/run/TreeMig/*): Contains all TreeMig inputs. The files *listOfCtrlPars.txt* and *logfile.txt* contain some control parameters. The TreeMig directory contains the following subdirectories:
 - C: Contains the file with control parameters. The name of the file to be read is specified in */run/TreeMig/listOfCtrlPars.txt*.

- E: Contains environmental information such as bioclimatic indices per year and cell (for uncoupled TreeMig) and stockability file. In FORHYCS, this is also the directory in which state files from a previous run (*hydrostate.txt* and *treestate.txt*) should be placed.
- P: Contains file with species-specific parameter values.
- Meteorological data (*/run/meteo/*): Directory where daily, gridded meteorological data are stored. On a shared cluster with several users of PREVAH or FORHYCS, it is sensible to have only one instance of this directory and link to it.
- Output directories: For each model run, an output directory is created in */run/*. Main outputs are:
 - Catchment-level averages of daily precipitation (CatXXX.prc) and streamflow (CatXXX.cat).
 - Ecoregion-level averages of annual biomass (*bio_ecor_ZZZZ.txt*, where **ZZZZ** corresponds to the ecoregion code), LAI (*lai_ecor_ZZZZ.txt*) and stem number (*stn_ecor_ZZZZ.txt*), as well as rooting zone storage capacity (*sd_ecor.txt*).
 - Grids of monthly and annual values (can be (de-)activated in the *common_settings_2014* file). These grids are stored in the directories *mm* and *yyyy*.
 - State files for hydrological (*hydrostate_CatXXX.txt*) and forest variables (*treestate_CatXXX.txt*), in the directory *states*.



5.2. Modules

To compile and run FORHYCS, it is necessary to load the modules containing libraries for Intel Fortran as well as Intel MPI.

5.3. Launch scripts

For each experiment, two shell/bash scripts are necessary to launch the model. The file whose name starts with “imet” is executed via command-line interface, and calls the second script, starting with “run”, one or more times.

5.3.1. imet files

The “imet” file allows the user to specify a number of control parameters, among which the new control parameters described in this document. The values for each control parameter are given as an array of length ≥ 1 . In addition to the new control parameters, the study areas can also be specified here.

Nested loops are performed over all arrays of control parameter values. For each combination of parameter values, the script submits a job by invoking the “run” script. For example, the cluster Hyperion at WSL uses the software SLURM to manage the scheduling of jobs on the cluster. The SLURM command to submit a job is SBATCH. The following parameters are used:

- --ntasks: number of nodes on which the model shall be run. Selecting 20 nodes has proven satisfactory so far (1 master and 19 slaves).
- --cpus-per-task: must be set to 1.
- --time: maximum duration of model run (hh:mm:ss). The job is terminated once this time is reached, so that it is important to allow a long enough duration. Specifying a time is optional, and omitting this parameter lets the model run without a time limit. The advantage of specifying a time is that the job gets a higher priority (especially for shorter durations).
- --mem-per-cpu: with this optional parameter, the amount of memory for each process can be specified. In some cases, using the default value might cause out-of-memory errors.
- -J: job name, which will be given to the output directory. The job name is also passed as an argument to the “run” script (there may be a better way to retrieve the job name as a SLURM parameter...).

After each job submission, the script pauses (*sleep*) for 300 seconds. This is to avoid model crashes during initialization of the meteo input routine (not sure why this is happening). If several jobs are submitted while the cluster is busy, they might be placed in the queue and the execution will be delayed until there are enough resources available. In this case, the 300 seconds pause does not apply, which may cause one or several runs to crash. Therefore, caution is advised when submitting several jobs on a busy cluster.

5.3.2. run files

The “run” files allow the user to specify the following:

- Start and end date of the simulation.
- Name of source for meteorological data (e.g. “ch2018”).
- Path to the executable file (variable *PREVAH*).
- Path to the input files for TreeMig (bioclimatic indices per cell and year (for uncoupled TreeMig), stockability file, species-specific parameter file).
- Path to the state files.

5.4. Spin-up and future simulations

5.4.1. Spin-up

A spin-up run serves to generate plausible starting conditions, by simulating a forest succession over many years, starting from “bare soil” (no trees). This is done by bootstrapping years from the period covered by meteorological data. Currently, a series of years randomly selected from the period 1981-2000 is hard-coded into FORHYCS (array *spinupyearsFH*, stored in ***forhycsglobals*** and initialized in ***prevahinit***). This way, it is possible to perform a simulation over 515 years by setting the start year to 1500.

If the control parameter *writeFH* is set to 1 (see above), FORHYCS writes the states of hydrological and forest variables to two files. Note that the structure of these state files differs from that used by stand-alone PREVAH and TreeMig. The simulation years for which this is done are currently hard-coded in ***prevahopt***.

5.4.2. Future simulations

To launch a future simulation using state files from a spin-up simulation, the following steps are necessary:

- make sure that the control parameter *readFH* is set to 1 in the “imet” file.
- copy the files ***hydrostate_1971.txt*** and ***treestate_1971.txt*** from the spin-up run into the directory */run/TreeMig/E/* (you may also choose another year, e.g. to start with younger forests). It may be sensible to rename these files, e.g. ***hydrostate_CatXXX.txt*** and ***treestate_CatXXX.txt***.
- set the starting year to 1971 and the end year to 2099 in the “run” file (for CH2018 simulations).
- edit the “run” file so that the state files get copied into the directory of the new run

References

- Bugmann, H., 1994. On the Ecology of Mountainous Forests in a Changing Climate: A Simulation Study. ETH Zurich, Zurich.
- Bugmann, H., Cramer, W., 1998. Improving the behaviour of forest gap models along drought gradients. *Forest Ecology and Management* 103, 247–263.
- Huss, M., Jouvett, G., Farinotti, D., Bauder, A., 2010. Future high-mountain hydrology: a new parameterization of glacier retreat. *Hydrology and Earth System Sciences* 14, 815–829. <https://doi.org/10.5194/hess-14-815-2010>
- Landsberg, J.J., Waring, R.H., 1997. A generalised model of forest productivity using simplified concepts of radiation-use efficiency, carbon balance and partitioning. *Forest Ecology and Management* 95, 209–228. [https://doi.org/10.1016/S0378-1127\(97\)00026-1](https://doi.org/10.1016/S0378-1127(97)00026-1)
- Lischke, Zierl, B., 2002. Feedback between structured vegetation and soil water in a changing climate: A simulation study, in: Beniston, M. (Ed.), *Climatic Change: Implications for the Hydrological Cycle and for Water Management*. Kluwer Academic Publishers, pp. 349–377.
- Rasche, L., Fahse, L., Zingg, A., Bugmann, H., 2012. Enhancing gap model accuracy by modeling dynamic height growth and dynamic maximum tree height. *Ecological Modelling* 232, 133–143. <https://doi.org/10.1016/j.ecolmodel.2012.03.004>
- Remund, J., Augustin, S., 2015. Zustand und Entwicklung der Trockenheit in Schweizer Wäldern. *Schweizerische Zeitschrift für Forstwesen* 166, 352–360. <https://doi.org/10.3188/szf.2015.0352>
- Schattan, P., Zappa, M., Lischke, H., Bernhard, L., Thürig, E., Diekkrüger, B., 2013. An approach for transient consideration of forest change in hydrological impact studies, in: IAHS (Ed.), *Climate and Land Surface Changes in Hydrology*. IAHS–IAPSO–IASPEI, Gothenburg, Sweden, pp. 311–319.
- Viviroli, Gurtz, Zappa, M., 2007. The Hydrological Modelling System PREVAH. Part II: Physical Model Description. Institute of Geography, University of Bern.
- Viviroli, D., Mittelbach, H., Gurtz, J., Weingartner, R., 2009. Continuous simulation for flood estimation in ungauged mesoscale catchments of Switzerland – Part II: Parameter regionalisation and flood estimation results. *Journal of Hydrology* 377, 208–225. <https://doi.org/10.1016/j.jhydrol.2009.08.022>
- Zurbriggen, N., Nabel, J.E.M.S., Teich, M., Bebi, P., Lischke, H., 2014. Explicit avalanche-forest feedback simulations improve the performance of a coupled avalanche-forest model. *Ecol. Complexity* 17, 56–66. <https://doi.org/10.1016/j.ecocom.2013.09.002>