# Package 'r3PG'

October 14, 2022

**Type** Package

**Title** Simulating Forest Growth using the 3-PG Model

**Description** Provides a flexible and easy-to-use interface for the Physiological Processes Predicting Growth (3-PG) model written in Fortran. The r3PG serves as a flexible and easy-to-use interface for the 3-PGpjs (monospecific, evenaged and evergreen forests) described in Landsberg & Waring (1997) <doi:10.1016/S0378-1127(97)00026-1> and the 3-PGmix (deciduous, uneven-aged or mixed-species forests) described in Forrester & Tang (2016) <doi:10.1016/j.ecolmodel.2015.07.010>.

**Date** 2022-05-19

**Version** 0.1.4

**License** GPL-3

**Depends** R (>= 3.5.0)

**Imports**

**Suggests** knitr (>= 1.15.1), rmarkdown (>= 1.3), R.rsp (>= 0.40.0), testthat (>= 1.0.2), roxygen2, BayesianTools, sensitivity, dplyr, ggplot2

**VignetteBuilder** R.rsp

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.2

**URL** https://github.com/trotsiuk/r3PG

**BugReports** https://github.com/trotsiuk/r3PG/issues

**NeedsCompilation** yes

**Author** Volodymyr Trotsiuk [aut, cre] (<https://orcid.org/0000-0002-8363-656X>), Florian Hartig [aut] (<https://orcid.org/0000-0002-6255-9059>), David Forrester [aut], Ramiro Silveyra Gonzalez [aut] (<https://orcid.org/0000-0002-3430-8383>)

**Maintainer** Volodymyr Trotsiuk <volodymyr.trotsiuk@wsl.ch>

**Repository** CRAN

**Date/Publication** 2022-05-19 11:50:02 UTC

# R topics documented:



---

`d_climate` *Climate input*

---

## Description

Table containing the information about monthly values for climatic data.

## Usage

```
d_climate
```

## Format

A `data frame` with 156 rows and 7 variables:

**year** calendar year

**month** month

**tmp_min** monthly mean daily minimum temperature (C)

**tmp_max** monthly mean daily maximum temperature (C)

**tmp_ave** monthly mean daily average temperature (C). (optional)

**prcp** monthly rainfall (mm month-1)

**srad** monthly mean daily solar radiation (MJ m-2 d-1)

**frost_days** frost days per month (d month-1)

**co2** monthly mean atmospheric co2 (ppm), required if calculate_d13c=1 (optional)

**d13catm** Monthly mean isotopic composition of air (%■), required if calculate_d13c=1 (optional)

---

`d_parameters` *Parameters input*

---

## Description

Table containing the information about parameters.

## Usage

```
d_parameters
```

## Format

A `data frame` with 65 rows and x variables:

**parameter** name of the parameter, must be consistent in naming with `i_parameters`

**Fagus sylvatica** parameter values for species 1

**Pinus sylvestris** parameter values for species 2

---

`d_site` *Site input*

---

## Description

Table containing the information about site conditions.

## Usage

```
d_site
```

## Format

A `data frame` with 1 rows and 8 variables:

**latitude** site latitude in the WGS84 coordinate system

**altitude** site altitude, m a.s.l.

**soil_class** soil class, according to table 2 user manual of 3PGpjs. 1 - Sandy; 2 - Sandy loam; 3 - Clay loam; 4 - Clay; 0 - No effect of available soil water on production

**asw_i** initial available soil water (mm)

**asw_max** minimum available soil water (mm)

**asw_min** maximum available soil water (mm)

**from** year and month indicating the start of simulation. Provided in form of year-month. E.g. "2000-01"

**to** year and month indicating the end of simulation. Provided in form of year-month. E.g. "2009-12", will include December 2009 as last simulation month

---

`d_sizeDist` *sizeDist input*

---

## Description

Table containing the information about size distribution.

## Usage

```
d_sizeDist
```

## Format

A data frame with 47 rows and x variables:

**parameter** name of the parameter, must be consistent in naming with `i_sizeDist`

**Fagus sylvatica** parameter values for species 1

**Pinus sylvestris** parameter values for species 2

---

`d_species` *Species input*

---

## Description

Table containing the information about species level data. Each row corresponds to one species/cohort.

## Usage

```
d_species
```

## Format

A data frame with number of rows corresponding to each species/cohort and 8 variables:

**species** species or cohort id/name. It must be consistent with species names in d_thinning, d_parameters and d_sizeDist tables.

**planted** year and month indicating when the species was planted. Provided in form of year-month. E.g. "2000-01"

**fertility** soil fertility for a given species. Range from 0 to 1

**stems_n** number of trees per ha

**biom_stem** stem biomass for a given species (Mg/ha)

**biom_root** root biomass for a given species (Mg/ha)

**biom_foliage** initial foliage biomass (Mg/ha). If this is a leafless period, provide the spring foliage biomass.

---

# d_thinning *Thinning input*

---

## Description

Table containing the information about thinnings

## Usage

```
d_thinning
```

## Format

A data frame with 3 rows and 6 variables:

**species** species or cohort id/name. It must be consistent with species names in d_species, d_parameters and d_sizeDist tables.

**age** age when thinning is performed

**stems_n** number of trees remaining after thinning

**stem** type of thinning (above/below). Default is 1

**root** type of thinning (above/below). Default is 1

**foliage** type of thinning (above/below). Default is 1

get_parameters *Get parameter sets*

## Description

Gets parameter sets from published studies with 3PG

## Usage

```
get_parameters(mode = "overview", sp_names = NULL)
```

## Arguments

| | |
|---|---|
| mode | must be one of the following: full, overview, source, comments, parameters, sizeDist |
| sp_names | names of the species. The 'sp_names' must be either a scientific name (Picea abies). If passing parameters or sizeDist to mode, it also possible to use a scientific name with a integer, which is the value of parset_id, e.g. Picea abies 37. If unsure, consider using first the option mode = overview with sp_names = NULL. See details below for more information. |

## Details

This function access the parameter database stored in the package, which is named i_parameters_lit, and also accessible without this function. If mode = overview, a simplified table with all existing parameter sets will be returned. A extended version including information about to the source and remarks can be obtained with mode = comments. A complete description of the source is returned with mode = source. The full version of this table (species,source, comments, parameters) will be returned if mode = full. Alternatively, with mode = parameters and mode = sizeDist, it is possible to obtain parameters in the format required by run_3PG.

Passing species names with sp_names will selecting from the table the desired species. Please note that one species might have more than one available parameter sets, unless species names contain the parset_id value. Please also note that mode = source is not compatible with sp_names.

The parameter sets were obtained from published studies. For basic information about this dataset check in i_parameters_lit. Relevant information about the parameter sets and the corresponding studies is provided in the table. For further information, please consider checking the original publications.

## Value

a data frame with parameter sets for all available species, or only the requested species, if sp_names is not null.

## See Also

run_3PG, i_parameters_lit

## Examples

```
# see an overview of the existing parameter sets
get_parameters(mode = 'overview')
get_parameters(mode = 'overview', sp_names = c('Eucalyptus globulus', 'Pinus sylvestris' ) )

# see existing parameter sets and comments
get_parameters(mode = 'comments')
get_parameters(mode = 'comments', sp_names = c('Eucalyptus globulus', 'Pinus sylvestris' ) )

# see parameters and source information
get_parameters(mode = 'source')

# obtain parameter sets in for some species in the format required by run_3PG
get_parameters(mode = 'parameters', sp_names = c('Fagus sylvatica', 'Picea abies'))
get_parameters(mode = 'parameters', sp_names = 'Fagus sylvatica 34' )
get_parameters(mode = 'sizeDist', sp_names = c('Fagus sylvatica 9', 'Picea abies 37'))

# see parameter sets with full information (species, source, parameters)
get_parameters(mode = 'full')
get_parameters(mode = 'full', sp_names = c('Fagus sylvatica', 'Pinus radiata' ) )
```

---

# i_output *Information about model outputs*

---

## Description

A dataset containing the list of output variables and their description.

## Usage

```
i_output
```

## Format

A data frame with 150 rows and 7 variables:

**group_id** serial number of the group

**variable_id** serial number of the variable

**variable_group** group name to which variable belongs

**variable_name** variable name as named in output

**description** description of the variable

**unit** unit of the variable

**variable_vba** corresponding name of the variable as output from Excel version of 3-PGmix

## i_parameters *Information about parameters*

### Description

A dataset containing the parameters order and description.

### Usage

```
i_parameters
```

### Format

A data frame with 82 rows and 3 variables:

**parameter** parameter name

**description** description of the parameter

**unit** unit

**default** default value for E.globulus from original 3-PG

## i_parameters_lit *Information about literature parameters*

### Description

A dataset containing parameter sets from published studies with 3PG

### Usage

```
i_parameters_lit
```

### Format

A data frame with 110 rows and 124 variables:

**parset_id** id of the parameter set

**species** species scientific name

**age** whether the parameter set was was used in even or uneven stands.

**type** whether the parameter set was used in monocultures or mixed stands.

**year** year of publication

**region** geographical region in which the parameters were tested. NAs values are allowed.

**country** country or countries in which the parameters were tested.

**notes** any relevant remark about how the parameters were processed. NAs are allowed.

**source** short reference to publication

**source_comments** any relevant comment about the parameters present in the publication. NAs are allowed.

**source_full** full reference to publication

**link** a link to the publication, e.g. doi

**pFS2** Foliage:stem partitioning ratio - D = 2 cm

**pFS20** Foliage:stem partitioning ratio - D = 20 cm

**aWS** Constant in the stem mass v. diam. relationship

**nWS** Power in the stem mass v. diam. relationship

**pRx** Maximum fraction of NPP to roots

**pRn** Minimum fraction of NPP to roots

**gammaF1** Maximum litterfall rate

**gammaF0** Litterfall rate at t = 0

**tgammaF** Age at which litterfall rate has median value

**gammaR** Average monthly root turnover rate

**leafgrow** If deciduous, leaves are produced at end of this month

**leaffall** If deciduous, leaves all fall at start of this month

**Tmin** Minimum temperature for growth

**Topt** Optimum temperature for growth

**Tmax** Maximum temperature for growth

**kF** Days production lost per frost day

**SWconst** Moisture ratio deficit for fq = 0.5

**SWpower** Power of moisture ratio deficit

**fCalpha700** Assimilation enhancement factor at 700 ppm

**fCg700** Canopy conductance enhancement factor at 700 ppm

**m0** Value of m when FR = 0

**fN0** Value of fNutr when FR = 0

**fNn** Power of (1-FR) in fNutr

**MaxAge** Maximum stand age used in age modifier

**nAge** Power of relative age in function for fAge

**rAge** Relative age to give fAge = 0.5

**gammaN1** Mortality rate for large t

**gammaN0** Seedling mortality rate (t = 0)

**tgammaN** Age at which mortality rate has median value

**ngammaN** Shape of mortality response

**wSx1000** Max. stem mass per tree - 1000 trees/hectare

**thinPower** Power in self-thinning rule

**mF** Fraction mean single-tree foliage biomass lost per dead tree

**mR** Fraction mean single-tree root biomass lost per dead tree

**mS** Fraction mean single-tree stem biomass lost per dead tree

**SLA0** Specific leaf area at age 0

**SLA1** Specific leaf area for mature leaves

**tSLA** Age at which specific leaf area = (SLA0+SLA1)/2

**k** Extinction coefficient for absorption of PAR by canopy

**fullCanAge** Age at canopy closure

**MaxIntcptn** Maximum proportion of rainfall evaporated from canopy

**LAImaxIntcptn** LAI for maximum rainfall interception

**cVPD** LAI for 50% reduction of VPD in canopy

**alphaCx** Canopy quantum efficiency

**Y** Ratio NPP/GPP

**MinCond** Minimum canopy conductance

**MaxCond** Maximum canopy conductance

**LAIgcx** LAI for maximum canopy conductance

**CoeffCond** Defines stomatal response to VPD

**BLcond** Canopy boundary layer conductance

**RGcGw** The ratio of diffusivities of CO2 and water vapour in air

**D13CTissueDif** d13C difference of modelled tissue and new photosynthate

**aFracDiffu** Fractionation against 13C in diffusion

**bFracRubi** Enzymatic fractionation by Rubisco

**fracBB0** Branch and bark fraction at age 0

**fracBB1** Branch and bark fraction for mature stands

**tBB** Age at which fracBB = (fracBB0+fracBB1)/2

**rhoMin** Minimum basic density - for young trees

**rhoMax** Maximum basic density - for older trees

**tRho** Age at which rho = (rhoMin+rhoMax)/2

**aH** Constant in the stem height relationship

**nHB** Power of DBH in the stem height relationship

**nHC** Power of competition in the stem height relationship

**aV** Constant in the stem volume relationship

**nVB** Power of DBH in the stem volume relationship

**nVH** Power of height in the stem volume relationship

**nVBH** Power of DBH^2 x height in the stem volume relationship

**crownshape** Crown shape (1=cone, 2=ellipsoid, 3=half-ellipsoid, 4=rectangular)

**aK** Constant in the crown diameter relationship

**nKB** Power of DBH in the crown diameter relationship

**nKH** Power of height in the crown diameter relationship

**nKC** Power of competition in the crown diameter relationship

**nKrh** Power of relative height in the crown diameter relationship

**aHL** Constant in the LCL relationship

**nHLB** Power of DBH in the LCL relationship

**nHLL** Power of LAI in the LCL relationship

**nHLC** Power of competition in the LCL relationship

**nHLrh** Power of relative height in the LCL relationship

**Dscale0** Constant in the relationship for Weibull scale parameter of D distribution

**DscaleB** Slope of DBH in relationship for Weibull scale parameter of D distribution

**Dscalerh** Slope of relative height in relationship for Weibull scale parameter of D distribution

**Dscalet** Slope of age in relationship for Weibull scale parameter of D distribution

**DscaleC** Slope of competition in relationship for Weibull scale parameter of D distribution

**Dshape0** Constant in the relationship for Weibull shape parameter of D distribution

**DshapeB** Slope of DBH in relationship for Weibull shape parameter of D distribution

**Dshaperh** Slope of relative height in relationship for Weibull shape parameter of D distribution

**Dshapet** Slope of age in relationship for Weibull shape parameter of D distribution

**DshapeC** Slope of competition in relationship for Weibull shape parameter of D distribution

**Dlocation0** Constant in the relationship for Weibull location parameter of D distribution

**DlocationB** Slope of DBH in relationship for Weibull location parameter of D distribution

**Dlocationrh** Slope of relative height in relationship for Weibull location parameter of D distribution

**Dlocationt** Slope of age in relationship for Weibull location parameter of D distribution

**DlocationC** Slope of competition in relationship for Weibull location parameter of D distribution

**wsscale0** Constant in the relationship for Weibull scale parameter of ws distribution

**wsscaleB** Slope of DBH in relationship for Weibull scale parameter of ws distribution

**wsscalerh** Slope of relative height in relationship for Weibull scale parameter of ws distribution

**wsscalet** Slope of age in relationship for Weibull scale parameter of ws distribution

**wsscaleC** Slope of competition in relationship for Weibull scale parameter of ws distribution

**wsshape0** Constant in the relationship for Weibull shape parameter of ws distribution

**wsshapeB** Slope of DBH in relationship for Weibull shape parameter of ws distribution

**wsshaperh** Slope of relative height in relationship for Weibull shape parameter of ws distribution

**wsshapet** Slope of age in relationship for Weibull shape parameter of ws distribution

**wsshapeC** Slope of competition in relationship for Weibull shape parameter of ws distribution

**wslocation0** Constant in the relationship for Weibull location parameter of ws distribution

**wslocationB** Slope of DBH in relationship for Weibull location parameter of ws distribution

**wslocationrh** Slope of relative height in relationship for Weibull location parameter of ws distribution

**wslocationt** Slope of age in relationship for Weibull location parameter of ws distribution

**wslocationC** Slope of competition in relationship for Weibull location parameter of ws distribution

**Qa** Intercept of net v. solar radiation relationship

**Qb** Slope of net v. solar radiation relationship

**gDM_mol** Molecular weight of dry matter

**molPAR_MJ** Conversion of solar radiation to PAR

## Details

Each row refers to an unique parameter set. The function `get_parameters` eases the use to this dataset.

---

`i_sizeDist` *Information about size distribution parameters*

---

## Description

A dataset containing the parameters order and description.

## Usage

```
i_sizeDist
```

## Format

A data frame with 30 rows and 3 variables:

**parameter** parameter name

**description** description of the parameter

**unit** unit

**default** default value equal to 0

`prepare_climate` *Subsets or replicate a climate data*

## Description

Prepares the climate table, by either replicating the average climate for the required number of years, or by subsetting from a longer time-series of climate data.

## Usage

```
prepare_climate(climate, from = "2000-04", to = "2010-11")
```

## Arguments

| | |
|---|---|
| `climate` | table containing the information about monthly values for climatic data. If the climate table have exactly 12 rows it will be replicated for the number of years and months specified by `from` - `to`. Otherwise, it will be subsetted to the selected time period. If this is required, `year` and `month` columns must be included in the climate table. The minimum required columns are listed below, but additionally you can include: tmp_ave, c02, d13catm. Please refer to `d_climate` for example.<br>• year: year of observation (only required for subsetting) (numeric).<br>• month: months of observation (only required for subsetting) (numeric).<br>• tmp_min: monthly mean daily minimum temperature (C).<br>• tmp_max: monthly mean daily maximum temperature (C).<br>• tmp_ave: monthly mean daily average temperature (C) (optional).<br>• prcp: monthly rainfall (mm month-1).<br>• srad: monthly mean daily solar radiation (MJ m-2 d-1).<br>• frost_days: frost days per month (d month-1).<br>• co2: monthly mean atmospheric co2 (ppm), required if calculate_d13c=1 (optional).<br>• d13catm: monthly mean isotopic composition of air (‰), required if calculate_d13c=1 (optional). |
| `from` | year and month indicating the start of simulation. Provided in form of year-month. E.g. "2000-01". |
| `to` | year and month indicating the end of simulation. Provided in form of year-month. E.g. "2009-12", will include December 2009 as last simulation month. |

## Details

This function prepares the climate table for `run_3PG`.

In case a user provides only average climate, this is replicated for the desired simulation period.

In case a larger climate file is provided, the simulation period is selected from this.

## Value

a data.frame with number of rows corresponding to number of simulated month and 10 columns

## See Also

run_3PG, prepare_input, prepare_parameters, prepare_sizeDist, prepare_thinning

## Examples

```
# subsetting climate data
prepare_climate( climate = d_climate, from = '2003-04', to = '2010-11')

# replicating climate data
climate = matrix(rnorm(60), ncol = 5)
colnames(climate) = c("tmp_min", "tmp_max", "prcp", "srad", "frost_days")

prepare_climate( climate = climate, from = '2000-04', to = '2010-11')
```

---

# prepare_input *Check and prepare input for running 3-PG model*

---

## Description

Checks and prepares all input tables to be used in run_3PG. For detailed descriptions see Forrester (2020).

## Usage

```
prepare_input(site, species, climate, thinning = NULL, parameters = NULL,
  size_dist = NULL, settings = NULL)
```

## Arguments

`site` table containing the information about site conditions.

- latitude: site latitude in the WGS84 coordinate system.
- altitude: site altitude, m a.s.l.
- soil_class: 1 - Sandy; 2 - Sandy loam; 3 - Clay loam; 4 - Clay; 0 - No effect of asw on production.
- asw_i: initial available soil water (mm).
- asw_min: minimum available soil water (mm).
- asw_max: maximum available soil water (mm).
- from: year and month indicating the start of simulation. Provided in form of year-month. E.g. "2000-01".
- to: year and month indicating the end of simulation. Provided in form of year-month. E.g. "2009-12", will include December 2009 as last simulation month

`species` table containing the information about species level data. Each row corresponds to one species/cohort.

- species: species or cohort id/name. It must be consistent with species names in `thinning`, `parameters` and `sizeDist` tables.
- planted: year and month indicating when species was planted. Provided in form of year-month. E.g. "2000-01".
- fertility: soil fertility for a given species. Range from 0 to 1.
- stems_n: number of trees per ha.
- biom_stem: stem biomass for a given species (Mg/ha).
- biom_root: root biomass for a given species (Mg/ha).
- biom_foliage: initial foliage biomass (Mg/ha). If this is a leafless period, provide the spring foliage biomass.

`climate` table containing the information about monthly values for climatic data. If the climate table has exactly 12 rows it will be replicated for the number of years and months specified by `from` - `to`. Otherwise, it will be subsetted to the selected time period. More details about preparing climate data are at `prepare_climate`.

- year: year of observation (only required for subsetting) (optional).
- month: months of observation (only required for subsetting) (optional).
- tmp_min: monthly mean daily minimum temperature (C).
- tmp_max: monthly mean daily maximum temperature (C).
- tmp_ave: monthly mean daily average temperature (C) (optional).
- prcp: monthly rainfall (mm month-1).
- srad: monthly mean daily solar radiation (MJ m-2 d-1).
- frost_days: frost days per month (d month-1).
- vpd_day: water pressure deficit (mbar) (optional).
- co2: monthly mean atmospheric co2 (ppm), required if calculate_d13c=1 (optional)
- d13catm: monthly mean isotopic composition of air (%■), required if calculate_d13c=1 (optional)

`thinning` table containing the information about thinnings. If there is no thinning, it must be `NULL`.

- species: species or cohort id/name. It must be consistent with species names in `species`, `parameters` and `sizeDist` tables.
- age: age when thinning is performed.
- stems_n: number of trees remaining after thinning
- foliage: type of thinning (above/below). Default is 1.
- root: type of thinning (above/below). Default is 1.
- stem: type of thinning (above/below). Default is 1.

`parameters` table containing the information about parameters to be modified. Values that are not provided are replaced by defaults.

- parameter: name of the parameter, must be consistent in naming with `i_parameters`
- species: each column must correspond to species/cohort id/name, as defined in `species` table

`size_dist` table containing the information about size distribution to be modified. Values that are not provided are replaced by defaults.

- parameter: name of the parameter, must be consistent in naming with `i_sizeDist`
- species: each column must correspond to species/cohort id/name, as defined in `species` table

`settings` a list with settings for the model. Values that are not provided are replaced by defaults.

- light_model: '1' - 3-PGpjs (default); '2' - 3-PGmix
- transp_model: '1' - 3-PGpjs (default); '2' - 3-PGmix
- phys_model: '1' - 3-PGpjs (default); '2' - 3-PGmix
- height_model: '1' - linear (default); '2' - non-linear
- correct_bias: '0' - no (default); '1' - yes
- calculate_d13c: '0' - no (default); '1' - yes

## Details

This function checks and prepares the input data for the `run_3PG`. The output is a list with 7 tables. Each of them corresponds to the one from input.

## Value

a list with seven tables. Each table corresponds to one of the input tables.

## References

Forrester, D. I., 2020. 3-PG User Manual. Swiss Federal Institute for Forest, Snow and Landscape Research WSL, Birmensdorf, Switzerland. 70 p. Available at the following web site: http://sites.google.com/site/davidforresterssite/home/projects/3PGmix/3pgmixdownload

Sands, P. J., 2010. 3PGpjs user manual. Available at the following web site: https://3pg.sites.olt.ubc.ca/files/2014/04/3PGpjs_UserManual.pdf

## See Also

`run_3PG`, `prepare_parameters`, `prepare_sizeDist`, `prepare_thinning`, `prepare_climate`, `prepare_site`

## Examples

```
prepare_input( site = d_site, species = d_species, climate = d_climate, d_thinning)
```

prepare_parameters *Prepare parameters table*

## Description

Prepares the parameters table, by either replicating the defaults or replicating defaults for each of the species.

## Usage

```
prepare_parameters(parameters = NULL, sp_names = c("Fagus sylvatica",
  "Pinus sylvestris"))
```

## Arguments

| | |
|---|---|
| parameters | table containing the information about parameters to be modified. Values that are not provided are replaced by defaults.<br>• parameter: name of the parameter, must be consistent in naming with i_parameters<br>• species: each column must correspond to species/cohort id/name, as defined in species table |
| sp_names | names of the species / cohorts used for the simulations.The 'sp_names' must be identical to those from species table. |

## Details

This function prepares the parameter table for run_3PG

## Value

a data.frame with 47 rows and columns corresponding to each species.

## See Also

run_3PG, prepare_input, prepare_sizeDist, prepare_thinning, prepare_climate

## Examples

```
# replace some
prepare_parameters( parameters = d_parameters[1:4,],
  sp_names = c('Fagus sylvatica', 'Pinus sylvestris' ))

# Make default
prepare_parameters( parameters = NULL, sp_names = c('Quercus', 'Abies'))
```

`prepare_site` *Check the site data for consistency*

## Description

Prepares the site table, by checking whether the input information is consistent.

## Usage

```
prepare_site(site)
```

## Arguments

| | |
|---|---|
| `site` | table containing the information about site data. It shall contain exactly one row.<br>• latitude: site latitude in the WGS84 coordinate system.<br>• altitude: site altitude, m a.s.l.<br>• soil_class: soil class, according to table 2 user manual of 3PGpjs. 1 - Sandy; 2 - Sandy loam; 3 - Clay loam; 4 - Clay; 0 - No effect of available soil water on production.<br>• asw_i: initial available soil water (mm).<br>• asw_max: minimum available soil water (mm).<br>• asw_min: maximum available soil water (mm).<br>• from: year and month indicating the start of simulation. Provided in form of year-month. E.g. "2000-01".<br>• to: year and month indicating the end of simulation. Provided in form of year-month. E.g. "2009-12", will include December 2009 as last simulation month. |

## Details

This function check the site table for `run_3PG`.

## Value

a data.frame with one row

## See Also

`run_3PG`, `prepare_input`, `prepare_parameters`, `prepare_sizeDist`, `prepare_thinning`

## Examples

```
# check site data
prepare_site( site = d_site)
```

prepare_sizeDist *Prepare parameters table*

## Description

Prepares the parameters table, by either replicating the defaults or replicating defaults for each of the species.

## Usage

```
prepare_sizeDist(size_dist = NULL, sp_names = c("Fagus sylvatica",
  "Pinus sylvestris"))
```

## Arguments

| | |
|---|---|
| size_dist | table containing the information about size distribution to be modified. Values that are not provided are replaced by defaults.<br>• parameter: name of the parameter, must be consistent in naming with i_sizeDist.<br>• species: each column must correspond to species/cohort id/name, as defined in species table. |
| sp_names | names of the species / cohorts used for the simulations.The 'sp_names' must be identical to those from species table. |

## Details

This function prepares the parameter table for run_3PG.

## Value

a data.frame with 47 rows and columns corresponding to each species.

## See Also

run_3PG, prepare_input, prepare_parameters, prepare_thinning, prepare_climate

## Examples

```
# replace some
prepare_sizeDist( size_dist = d_sizeDist[1:4,],
  sp_names = c('Fagus sylvatica', 'Pinus sylvestris' ))

# Make default
prepare_sizeDist( size_dist = NULL, sp_names = c('Quercus', 'Abies'))
```

`prepare_species` *Check the species data for consistency*

## Description

Prepares the species table, by checking whether the input information is consistent.

## Usage

```
prepare_species(species)
```

## Arguments

| | |
|---|---|
| `species` | table containing the information about species level data. Each row corresponds to one species/cohort.<br>• species: species or cohort id/name. It must be consistent with species names in `thinning`, `parameters` and `sizeDist` tables.<br>• planted: year and month indicating when species was planted. Provided in form of year-month. E.g. "2000-01".<br>• fertility: soil fertility for a given species. Range from 0 to 1.<br>• stems_n: number of trees per ha.<br>• biom_stem: stem biomass for a given species (Mg/ha).<br>• biom_root: root biomass for a given species (Mg/ha).<br>• biom_foliage: initial foliage biomass (Mg/ha). If this is a leafless period, provide the spring foliage biomass. |

## Details

This function check the species table for `run_3PG`.

## Value

a data.frame with one row

## See Also

`run_3PG`, `prepare_input`, `prepare_parameters`, `prepare_sizeDist`, `prepare_thinning`, `prepare_site`

## Examples

```
# check species data
prepare_species( species = d_species)
```

prepare_thinning *Check and prepare management information.*

## Description

Prepares the management table and checks for consistency.

## Usage

```
prepare_thinning(thinning = NULL, sp_names = c("Fagus sylvatica",
  "Pinus sylvestris"))
```

## Arguments

| | |
|---|---|
| thinning | table containing the information about thinnings. If there is no thinning, it must be NULL. The following columns are required:<br>• species: species or cohort id/name.<br>• age: age at which thinning is done.<br>• stems_n: number of trees remaining after thinning<br>• stem: type of thinning (above/below). Default is 1.<br>• foliage: type of thinning (above/below). Default is 1.<br>• root: type of thinning (above/below). Default is 1. |
| sp_names | names of the species / cohorts used for the simulations. This is required whether 'thinning=NULL' or if not all species are indicated in the 'thinning' table. The 'sp_names' must be identical to those from species table. |

## Details

This function prepares the thinning table for run_3PG.

In case there is no thinning it will return empty 3-d array.

In case there will be thinning it will return 3-d array, where one dimension correspond to each species.

## Value

a 3-dimentional array, where third dimention correspond to each species.

## See Also

run_3PG, prepare_input, prepare_parameters, prepare_sizeDist, prepare_climate

## Examples

```
prepare_thinning( thinning = NULL, sp_names = c('Quercus', 'Abies'))

prepare_thinning( thinning = d_thinning, sp_names = c('Fagus sylvatica', 'Pinus sylvestris'))
```

r3PG *Simulating Forest Growth using the 3-PG Process-Based Vegetation Model*

## Description

The r3PG package provides a flexible and easy-to-use interface for Fortran implementations of the 3-PGpjs (monospecific, evenaged and evergreen forests) or 3-PGmix (deciduous, uneven-aged or mixed-species forests) forest growth models. The user can flexibly switch between various options and submodules, to use the original 3-PGpjs model version for monospecific, even-aged and evergreen forests and the 3-PGmix model, which can also simulate multi-cohort stands (e.g. mixtures, uneven-aged) that contain deciduous species. The core function to run the model is run_3PG. For more background, please consult the vignette via vignette(package = "r3PG")

## Value

None

## References

Forrester, D. I., 2020. 3-PG User Manual. Swiss Federal Institute for Forest, Snow and Landscape Research WSL, Birmensdorf, Switzerland. 70 p. Available at the following web site: http://sites.google.com/site/davidforresterssite/home/projects/3PGmix/3pgmixdownload

Forrester, D. I., & Tang, X. (2016). Analysing the spatial and temporal dynamics of species interactions in mixed-species forests and the effects of stand density using the 3-PG model. Ecological Modelling, 319, 233–254. doi:10.1016/j.ecolmodel.2015.07.010

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(3), 209–228. doi:10.1016/S03781127(97)000261

Sands, P. J., 2010. 3PGpjs user manual. Available at the following web site: https://3pg.sites.olt.ubc.ca/files/2014/04/3PGpjs_UserManual.pdf

## See Also

run_3PG

## Examples

```
out <- run_3PG(
  site = d_site,
  species = d_species,
  climate = d_climate,
  thinning = d_thinning,
  parameters = d_parameters,
  size_dist = d_sizeDist,
  settings = list(light_model = 2, transp_model = 2, phys_model = 2,
                  correct_bias = 1, calculate_d13c = 0),
```

```
  check_input = TRUE, df_out = TRUE) # note that default is TRUE

str(out) # List output format
```

---

run_3PG *Runs a 3-PG model simulation*

---

## Description

Runs the 3-PGpjs (monospecific, evenaged and evergreen forests) or 3-PGmix (deciduous, uneven-aged or mixed-species forests) model. For more details on parameters and structure of input visit prepare_input.

## Usage

```
run_3PG(site, species, climate, thinning = NULL, parameters = NULL,
  size_dist = NULL, settings = NULL, check_input = TRUE, df_out = TRUE)
```

## Arguments

| | |
|---|---|
| site | table as described in prepare_input containing the information about site conditions. |
| species | table as described in prepare_input containing the information about species level data. Each row corresponds to one species/cohort. |
| climate | table as described in prepare_input containing the information about monthly values for climatic data. See also prepare_climate |
| thinning | table as described in prepare_input containing the information about thinnings. See also prepare_thinning |
| parameters | table as described in prepare_input containing the information about parameters to be modified. See also prepare_parameters |
| size_dist | table as described in prepare_input containing the information about size distributions. See also prepare_sizeDist |
| settings | a list as described in prepare_input with settings for the model. |
| check_input | logical if the input shall be checked for consistency. It will call prepare_input function. |
| df_out | logical if the output shall be long data.frame (TRUE) the 4-dimensional array (FALSE). |

## Details

'r3PG' provides an implementation of the Physiological Processes Predicting Growth 3-PG model, which simulates forest growth and productivity. The 'r3PG' serves as a flexible and easy-to-use interface for the '3-PGpjs' (monospecific, evenaged and evergreen forests) and the '3-PGmix' (deciduous, uneven-aged or mixed-species forests) model written in 'Fortran'. The package, allows

for fast and easy interaction with the model, and 'Fortran' re-implementation facilitates computationally intensive sensitivity analysis and calibration. The user can flexibly switch between various options and submodules, to use the original '3-PGpjs' model version for monospecific, even-aged and evergreen forests and the '3-PGmix' model, which can also simulate multi-cohort stands (e.g. mixtures, uneven-aged) that contain deciduous species.

This implementation of 3-PG includes several major variants / modifications of the model in particular the ability to switch between 3-PGpjs (the more classic model version for monospecific stands) vs. 3-PGmix (a version for mixed stands), as well as options for bias corrections and $\delta^1 3C$ calculations (see parameters).

## Value

either a 4-dimentional array or a data.frame, depending on the parameter `df_out`. More details on the output is `i_output`

## Note

The `run_3PG` also checks the quality of input data. When names, or structures are not consistent with requirements it will return an error. Turn this off to optimize for speed.

## References

Forrester, D. I., 2020. 3-PG User Manual. Swiss Federal Institute for Forest, Snow and Landscape Research WSL, Birmensdorf, Switzerland. 70 p. Available at the following web site: http://sites.google.com/site/davidforresterssite/home/projects/3PGmix/3pgmixdownload

Forrester, D. I., & Tang, X. (2016). Analysing the spatial and temporal dynamics of species interactions in mixed-species forests and the effects of stand density using the 3-PG model. Ecological Modelling, 319, 233–254. doi:10.1016/j.ecolmodel.2015.07.010

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(3), 209–228. doi:10.1016/S03781127(97)000261

Sands, P. J., 2010. 3PGpjs user manual. Available at the following web site: https://3pg.sites.olt.ubc.ca/files/2014/04/3PGpjs_UserManual.pdf

## See Also

`prepare_input`, `prepare_parameters`, `prepare_sizeDist`, `prepare_thinning`, `prepare_climate`

## Examples

```
out <- run_3PG(
  site = d_site,
  species = d_species,
  climate = d_climate,
  thinning = d_thinning,
  parameters = d_parameters,
  size_dist = d_sizeDist,
  settings = list(light_model = 2, transp_model = 2, phys_model = 2,
                  correct_bias = 1, calculate_d13c = 0),
```

```
  check_input = TRUE, df_out = TRUE) # note that default is TRUE

str(out) # List output format
```

# Index