Package 'rsofun'

October 9, 2025

Title The P-Model and BiomeE Modelling Framework

Version 5.1.0

Description Implements the Simulating Optimal FUNctioning framework for site-scale simulations of ecosystem processes, including model calibration. It contains 'Fortran 90' modules for the P-model (Stocker et al. (2020) <doi:10.5194/gmd-13-1545-2020>), SPLASH (Davis et al. (2017) <doi:10.5194/gmd-10-689-2017>) and BiomeE (Weng et al. (2015) <doi:10.5194/bg-12-2655-2015>).

URL https://github.com/geco-bern/rsofun

BugReports https://github.com/geco-bern/rsofun/issues

License GPL-3

Depends R (>= 4.1.0)

Imports dplyr, purrr, tidyr, magrittr, GenSA, BayesianTools, lubridate, multidplyr, stats, utils

LazyData true

LazyDataCompression xz

ByteCompile true

NeedsCompilation yes

RoxygenNote 7.3.2

Suggests covr, constructive, cowplot, rcmdcheck, testthat (>= 3.0.0), rmarkdown, ggplot2, knitr, sensitivity, rpmodel, rlang

VignetteBuilder knitr

Encoding UTF-8

Config/testthat/edition 3

Author Benjamin Stocker [aut, cre] (ORCID: <https://orcid.org/0000-0003-2697-9096>), Koen Hufkens [aut] (ORCID: <https://orcid.org/0000-0002-5070-8109>), Josefa Arán Paredes [aut] (ORCID: <https://orcid.org/0009-0006-7176-2311>),

Laura Marqués [ctb] (ORCID: <https://orcid.org/0000-0002-3593-5557>),
Mayeul Marcadella [ctb] (ORCID:
<https://orcid.org/0000-0001-8555-3808>),
Ensheng Weng [ctb] (ORCID: <https://orcid.org/0000-0002-1858-4847>),
Fabian Bernhard [aut] (ORCID: <https://orcid.org/0000-0003-0338-0961>),
Geocomputation and Earth Observation, University of Bern [cph, fnd]

Maintainer Benjamin Stocker <benjamin.stocker@gmail.com>

Repository CRAN

Date/Publication 2025-10-09 14:00:02 UTC

Contents

`biomee_gs_leuning_drivers` *rsofun BiomeE driver data (Leuning photosynthesis model)*

Description

Example driver to run the BiomeE-model at the CH-LAE site using the Leuning photosynthesis specification (and half-hourly time step) It can also be used together with leaf trait data from CH-LAE (`biomee_validation`) to optimize model parameters.

Usage

```
biomee_gs_leuning_drivers
```

Format

A tibble of driver data.

sitename Site name

params_siml Simulation parameters as a data.frame, including the following data:

- **spinup** Flag indicating whether this simulation does spin-up (deprecated).
- **spinupyears** Number of spin-up years. Set to 0 for no spinup.
- **recycle** Number of first N years of forcing data.frame that are recycled for spin-up.
- **firstyeartrend** Year of first transient year (AD) (optional). Is only used to set years in output data frames. Defaults to 0 if not provided.
- **nyeartrend** Number of transient years (optional). Determines the length of simulation output after spin-up. Defaults to number of years contained in the forcing data. (If longer than forcing data, last year of forcing is repeated until the end (spin-down).)
- **steps_per_day** Time resolution of the forcing (day-1).
- **do_U_shaped_mortality** Flag indicating whether U-shaped mortality is used.
- **do_closedN_run** Flag indicating whether doing N closed runs to recover N balance enforcing 0.2 kg N m-2 in the inorganic N pool.
- **method_photosynth** String specifying the method of photosynthesis used in the model, either "pmodel" or "gs_leuning".document()
- **method_mortality** String indicating the type of mortality in the model. One of the following: "dbh" is size-dependent mortality, "const_selfthin" is constant self thinning (in development), "cstarvation" is carbon starvation, and "growthrate" is growth rate dependent mortality.
- **do_daily_diagnostics** Whether to output daily diagnostics ('output_daily_tile'). Default: True.

site_info Site meta info in a data.frame. This data structure can be freely used for documenting the dataset, but must include at least the following data:

- **lon** Longitude of the site location in degrees east.
- **lat** Latitude of the site location in degrees north.

elv Elevation of the site location, in meters above sea level.

forcing Forcing data.frame used as input

ppfd Photosynthetic photon flux density (mol m-2 s-1)

tair Air temperature (deg C)

vpd Vapor pressure deficit (Pa)

rain Precipitation (kgH2O m-2 s-1 == mm s-1)

wind Wind velocity (m s-1)

pair Atmospheric pressure (Pa)

co2 Atmospheric CO_2 concentration in ppm.

params_tile Tile-level model parameters, into a single row data.frame, including the following data:

soiltype Integer indicating the type of soil: Sand = 1, LoamySand = 2, SandyLoam = 3, SiltLoam = 4, FrittedClay = 5, Loam = 6, Clay = 7.

FLDCAP Field capacity (vol/vol). Water remaining in a soil after it has been thoroughly saturated and allowed to drain freely.

WILTPT Wilting point (vol/vol). Water content of a soil at which plants wilt and fail to recover.

K1 Fast soil C decomposition rate (yr^{-1}).

K2 Slow soil C decomposition rate (yr^{-1}).

K_nitrogen Mineral Nitrogen turnover rate (yr^{-1}).

MLmixRatio Ratio of C and N returned to litters from microbes.

etaN N loss rate through runoff (organic and mineral) (yr^{-1}).

LMAmin Minimum LMA, leaf mass per unit area, kg C m^{-2}.

fsc_fine Fraction of fast turnover carbon in fine biomass.

fsc_wood Fraction of fast turnover carbon in wood biomass.

GR_factor Growth respiration factor.

l_fract Fraction of the carbon retained after leaf drop.

retransN Retranslocation coefficient of nitrogen.

f_initialBSW Coefficient for setting up initial sapwood.

f_N_add Re-fill of N for sapwood.

tf_base Calibratable scalar for respiration, used to increase LUE levels.

par_mort Canopy mortality parameter.

par_mort_under Parameter for understory mortality.

params_species A data.frame containing species-specific model parameters, with one species per row, including the following data:

The following columns pertaining to the plant type: lifeform Integer set to 0 for grasses and 1 for trees.

phenotype Integer set to 0 for deciduous and 1 for evergreen.

pt Integer indicating the type of plant according to photosynthesis: 0 for C3; 1 for C4

The following columns pertaining to the root parameters: alpha_FR Fine root turnover rate (yr^{-1}).

rho_FR Material density of fine roots (kg C m^{-3}).

root_r Radius of the fine roots, in m.

root_zeta e-folding parameter of root vertical distribution, in m.

Kw_root Fine root water conductivity (mol m^{-2} s^{-1} MPa^{-1}).

leaf_size Characteristic leaf size.

The following columns pertaining to the photosynthesis parameters: Vmax Max RuBisCo rate, in mol m^{-2} s^{-1}.

Vannual Annual productivity per unit area at full sun (kg C m^{-2} $year^{-2}$).

wet_leaf_dreg Wet leaf photosynthesis down-regulation.

m_cond Factor of stomatal conductance.

alpha_phot Photosynthesis efficiency.

gamma_L Leaf respiration coefficient, in yr^{-1}.

gamma_LN Leaf respiration coefficient per unit N.

gamma_SW Sapwood respiration rate, in kg C m^{-2} yr^{-1}.

gamma_FR Fine root respiration rate, kg C kg C^{-1} yr^{-1}.

tk_crit Critical temperature triggerng offset of phenology, in Kelvin.

tk_crit_on Critical temperature triggerng onset of phenology, in Kelvin.

gdd_crit Critical value of GDD5 for turning ON growth season.

betaON Critical soil moisture for phenology onset.

betaOFF Critical soil moisture for phenology offset.

The following columns pertaining to the allometry parameters: alphaHT Coefficient for allometry (height = alphaHT * DBH_m ** thetaHT), in m $m^{-thetaHT}$.

thetaHT Coefficient for allometry (height = alphaHT * DBH_m ** thetaHT), in m $m^{-thetaHT}$.

alphaCA Coefficient for allometry (projected crown area = pi * (alphaCA * DBH_m) ** thetaCA), in $m^{2/thetaCA-1}$.

thetaCA Coefficient for allometry (projected crown area = pi * (alphaCA * DBH_m) ** thetaCA), unitless. Dybzinski (eq. G1) showed that thetaCA = theatBM - 1.

alphaBM Coefficient for allometry (biomass = alphaBM * DBH ** thetaBM), in kg C $m^{-thetaBM}$.

thetaBM Coefficient for allometry (biomass = alphaBM * DBH ** thetaBM), unitless. Dybzinski (eq. G1) showed that thetaCA = theatBM - 1.

The following columns pertaining to the reproduction parameters: seedlingsize Initial size of seedlings, in kg C per individual.

maturalage Age at which trees can reproduce (years).

v_seed Fraction of G_SF to G_F.

The following columns pertaining to the mortality parameters: mortrate_d_c Canopy tree mortality rate (yr^{-1}).

mortrate_d_u Understory tree mortality rate (yr^{-1}).

The following columns pertaining to the leaf parameters: LMA Leaf mass per unit area (kg C m^{-2}).

leafLS TODO

LNbase Basal leaf N per unit area, in kg N m^{-2}.

CNleafsupport TODO

rho_wood Wood density (kg C m^{-3}).

taperfactor TODO

lAImax Maximum crown LAI (leaf area index).

tauNSC TODO

fNSmax Multiplier for NSNmax as sum of potential bl and br.

phiCSA Ratio of sapwood area to leaf area.

The following columns pertaining to the C/N ratios for plant pools: CNleaf0 TODO

CNsw0 TODO

CNwood0 TODO

CNroot0 TODO

CNseed0 TODO

Nfixrate0 Reference N fixation rate (kg N kg C^{-1} root).

NfixCost0 Carbon cost of N fixation (kg C kg N^{-1}).

internal_gap_frac TODO

The following columns pertaining to the calibratable parameters: kphio Quantum yield efficiency φ_0, in mol mol^{-1}.

phiRL Ratio of fine root to leaf area.

LAI_light Maximum LAI limited by light.

init_cohort A data.frame of initial cohort specifications, including the following data:

init_cohort_species Index of a species described in param_species.

init_cohort_nindivs Initial individual density, in individuals per m^2.

init_cohort_bl Initial biomass of leaf, in kg C per individual.

init_cohort_br Initial biomass of fine root, in kg C per individual.

init_cohort_bsw Initial biomass of sapwood, in kg C per individual.

init_cohort_bHW Initial biomass of heartwood, in kg C per individual.

init_cohort_seedC Initial biomass of seed, in kg C per individual.

init_cohort_nsc Initial non-structural biomass, in kg C per individual.

lu_index Land use type this cohorts belongs to (given as index in init_lu aray). Default: 0 (attach to all LU types except thoses which do not accept vegetation – cf init_lu.vegetated).

init_soil A data.frame of initial soil pools, including the following data:

init_fast_soil_C Initial fast soil carbon, in kg C m^{-2}.

init_slow_soil_C Initial slow soil carbon, in kg C m^{-2}.

init_Nmineral Mineral nitrogen pool, in kg N m^{-2}.

N_input Annual nitrogen input to soil N pool, in kg N m^{-2} yr^{-1}.

init_lu A data.frame of initial land unit (LU) specifications, including the following data:

fraction Initial grid cell fraction occupied by this LU, dimensionless (0 to 1) or m^{-2} LU area per m^{-2} grid cell area. The sum of all fractions is typically equal to 1, but may be less in which case the difference is the fraction of the grid cell occupied by ice/water.

preset Predefined land use type (optional). One of: 'unmanaged', 'urban', 'cropland', 'pasture'. See below for meaning of these presets. Leave empty to not use any preset.

vegetated Whether this LU accepts vegetation. Default for preset 'urban': False, default for other presets: True.

extra_N_input Additional inorg N supply (to account for N fertiliser application), in kg m-2 yr-1. Default for preset 'cropland': 0.01, default other presets: 0.

extra_turnover_rate Additional soil turnover rate (to account for soil management such as tillage), dimensionless. Default for preset 'cropland': 0.2, default for other presets: 0.

oxidized_litter_fraction Fraction of above-ground turnover that is directly oxidized (crop and grass harvest), dimensionless. Default for preset 'cropland': 0.9, default for preset 'pasture': 0.4, default for other presets: 0.

luc_forcing Array of land use change (LUC) used during transient phase. During spinup, the initial land unit fractions are used (i.e. no transition). If there are more transient years than provided LUC data, the last state is maintained until the end of the transient phase (i.e. no transition). The array is a nxn square matrix, where n is the number of LU (i.e. dimension of init_lu). Each entry f(i, j) expresses the grid cell fraction of LU i (row) being transfered to LU j (column). I.e. same units as `init_lu$fraction`. Self transitions are allowed, meaning that a part of the land unit is clear cut, but the area remains in the same land use.

`biomee_gs_leuning_output`

rsofun BiomeE (gs_leuning) output data

Description

Example output dataset from a BiomeE-model run using divers `biomee_gs_leuning_drivers` See `runread_biomee_f` and `run_biomee_f_bysite` for a detailed description of the outputs.

Usage

```
biomee_gs_leuning_output
```

Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 1 rows and 2 columns.

`biomee_p_model_drivers`

rsofun BiomeE driver data (P-model photosynthesis model)

Description

Example driver data to run the BiomeE-model at the CH-LAE site using the P-model photosynthesis specification (and daily time step). It can also be used together with leaf trait data from CH-LAE (`biomee_validation`) to optimize model parameters.

Usage

```
biomee_p_model_drivers
```

Format

See `biomee_gs_leuning_drivers`

`biomee_p_model_luluc_drivers`
rsofun BiomeE driver data (P-model photosynthesis model) with LU-LUC

Description

Example driver data to run the BiomeE-model at the CH-LAE site using the P-model photosynthesis specification (and daily time step). It provides an example of land use change (LUC).

Usage

```
biomee_p_model_luluc_drivers
```

Format

See `biomee_gs_leuning_drivers`

`biomee_p_model_luluc_output`
rsofun BiomeE (P-model) output data

Description

Example output dataset from a BiomeE-model run using divers `biomee_p_model_luluc_drivers` See `runread_biomee_f` and `run_biomee_f_bysite` for a detailed description of the outputs.

Usage

```
biomee_p_model_luluc_output
```

Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 1 rows and 4 columns.

biomee_p_model_output *rsofun BiomeE (P-model) output data*

Description

Example output dataset from a BiomeE-model run using divers `biomee_p_model_drivers` See `runread_biomee_f` and `run_biomee_f_bysite` for a detailed description of the outputs.

Usage

```
biomee_p_model_output
```

Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 1 rows and 2 columns.

biomee_validation *rsofun BiomeE targets validation data*

Description

Small example dataset of target observations (leaf trait data) at the CH-LAE site to optimize model parameters with the function `calib_sofun`

Usage

```
biomee_validation
```

Format

A tibble of validation data:

sitename site name

data validation data

Source

Lukas Hörtnagl, Werner Eugster, Nina Buchmann, Eugenie Paul-Limoges, Sophia Etzold, Matthias Haeni, Peter Pluess, Thomas Baur (2004-2014) FLUXNET2015 CH-Lae Laegern, Dataset. https://doi.org/10.18140/FLX/144

build_luc_matrix *Build LUC matrix*

Description

Build land-use change (LUC) transition matrix from patterns.

Usage

```
build_luc_matrix(patterns, n_lu, n_years, out = vector())
```

Arguments

patterns	A list of patterns. Each pattern must be a sequence of transition values whose size is a multiple of n_luxn_lu. The n_lu first values are the transitions from each LU to the first LU, and so on. If the sequence contains more years than n_years, it will be truncated.
n_lu	Number of land use types (LU).
n_years	Number of years (i.e. length of the 3rd dimension).
out	For internal use only.

Value

An n_luxn_luxn_years transition matrix.

Examples

```
# Example of building a 6 year-long transition matix consisting of 6 times 2x2 matrices

# A one time transfer of 0.5 of the total cell fraction from LU 2 to LU 1
pattern1 <- c(0, 0, 0.5, 0)
# The null pattern (no transition)
null_pattern <- rep(0, 4)
# A repeated self-transition of 0.1 of the total cell fraction from LU 2 to LU 2 every other year
pattern2 <- rep(c(c(0, 0, 0, 0.1), null_pattern), 3)

# Building the transition matrix
build_luc_matrix(list(pattern1, pattern2), 2, 6)
```

`calib_sofun` *Calibrates SOFUN model parameters*

Description

This is the main function that handles the calibration of SOFUN model parameters.

Usage

```
calib_sofun(drivers, obs, settings, optim_out = TRUE, ...)
```

Arguments

`drivers`	A data frame with driver data. See `p_model_drivers` for a description of the data structure. Additional columns can optionally be provided to `drivers` to control e.g. the processing within a personalized cost function.
`obs`	A data frame containing observational data used for model calibration. See `p_model_validation` for a description of the data structure. Additional columns can optionally be provided to `obs` to control e.g. the processing within a personalized cost function.
`settings`	A list containing model calibration settings. See the 'P-model usage' vignette for more information and examples. `method` A string indicating the optimization method, either `'GenSA'` or `'BayesianTools'`. `par` A list of model parameters. For each parameter, an initial value and lower and upper bounds should be provided. The calibratable parameters include model parameters 'kphio', 'kphio_par_a', 'kphio_par_b', 'soilm_thetastar', 'soilm_betao', 'beta_costunitratio', 'rd_to_vcmax', 'tau_acclim', 'kc_jmax' and 'rootzone_whc' , and (if doing Bayesian calibration) error parameters for each target variable, named for example 'err_gpp'. This list must match the input parameters of the calibration metric and the parameters should be given in the order above. `metric` A cost function. See the 'Cost functions for parameter calibration' vignette for examples. `control` A list of arguments passed on to the optimization function. If `method = 'GenSA'`, see GenSA. If `method = 'BayesianTools'` the list should include at least `settings` and `sampler`, see BayesianTools::runMCMC.
`optim_out`	A logical indicating whether the function returns the raw output of the optimization functions (defaults to TRUE).
`...`	Optional arguments, simply passed on to the cost function.

Value

A named list containing the calibrated parameter vector 'par' and the output object from the optimization 'mod'. For more details on this output and how to evaluate it, see runMCMC (also this post) and GenSA.

Examples

```
# Fix model parameters that won't be calibrated
params_fix <- list(
  kphio_par_a        = 0,
  kphio_par_b        = 1.0,
  soilm_thetastar    = 0.6*240,
  soilm_betao        = 0.01,
  beta_unitcostratio = 146,
  rd_to_vcmax        = 0.014,
  tau_acclim         = 30,
  kc_jmax            = 0.41
)

# Define calibration settings
settings <- list(
  method = "BayesianTools",
  par = list(
    kphio = list(lower=0.04, upper=0.09, init=0.05),
    err_gpp = list(lower = 0.01, upper = 4, init = 2)
  ),
  metric = rsofun::cost_likelihood_pmodel,
  control = list(
    sampler = "DEzs",
    settings = list(
      nrChains = 1,
      burnin = 0,
      iterations = 50     # kept artificially low
    )
  )
 )

 # Run the calibration for GPP data
 calib_output <- rsofun::calib_sofun(
   drivers = rsofun::p_model_drivers,
   obs = rsofun::p_model_validation,
   settings = settings,
   # extra arguments for the cost function
   par_fixed = params_fix,
   targets = c("gpp")
 )
```

`cost_likelihood_biomee` *Log-likelihood cost function for BiomeE with different targets*

Description

Cost function for parameter calibration, which computes the log-likelihood for the biomee model fitting several target variables for a given set of parameters.

Usage

```
cost_likelihood_biomee(par, obs, drivers, targets)
```

Arguments

par	A named vector containing parameter values for 'phiRL','LAI_light', 'tf_base', 'par_mort' in that order, and for the error terms corresponding to the target variables, e.g. 'err_GPP' if GPP is a target. Make sure that the order of the error terms in par coincides with the order provided in the targets argument.
obs	A nested data frame of observations, following the structure of biomee_validation, for example.
drivers	A nested data frame of driver data, for example biomee_gs_leuning_drivers.
targets	A character vector indicating the target variables for which the optimization will be done. This should be a subset of c("GPP", "LAI","Density", "Biomass").

Details

The cost function performs a BiomeE model run for the value of par given as argument. The likelihood is calculated assuming that the predicted targets are independent, normally distributed and centered on the observations. The optimization should be run using BayesianTools, so the likelihood is maximized.

Value

The log-likelihood of the simulated targets by the biomee model versus the observed targets.

Examples

```
 # do not run long-running simulations
# Compute the likelihood for a set of
# BiomeE model parameter values
# and the example data
cost_likelihood_biomee(
 par = c(phiRL = 3.5,
         LAI_light = 3.5,
         tf_base = 1,
         par_mort = 1,    # model params
         err_GPP = 0.5),  # err_GPP
 obs = biomee_validation,
 drivers = biomee_p_model_drivers,
 targets = c("GPP")
)
```

`cost_likelihood_pmodel` *Cost function computing a log-likelihood for calibration of P-model parameters*

Description

The cost function performs a P-model run for the input drivers and model parameter values, and computes the outcome's normal log-likelihood centered at the input observed values and with standard deviation given as an input parameter (calibratable).

Usage

```
cost_likelihood_pmodel(
  par,
  obs,
  drivers,
  targets,
  par_fixed = NULL,
  parallel = FALSE,
  ncores = 2
)
```

Arguments

`par`	A named vector of values for the parameters to be calibrated, including a subset of model parameters (described in `runread_pmodel_f`), in order, and error terms for each target variable (for example `'gpp_err'`), in the same order as the targets appear in `targets`.
`obs`	A nested data.frame of observations, with columns `'sitename'` and `'data'` (see `p_model_validation` or `p_model_validation_vcmax25` to check their structure).
`drivers`	A nested data.frame of driver data. See `p_model_drivers` for a description of the data structure.
`targets`	A character vector indicating the target variables for which the optimization will be done and the RMSE computed. This string must be a column name of the `data` data.frame belonging to the validation nested data.frame (for example 'gpp').
`par_fixed`	A named list of model parameter values to keep fixed during the calibration. These should complement the input `par` such that all model parameters are passed on to `runread_pmodel_f`.
`parallel`	A logical specifying whether simulations are to be parallelised (sending data from a certain number of sites to each core). Defaults to `FALSE`.
`ncores`	An integer specifying the number of cores used for parallel computing. Defaults to 2.

Details

To run the P-model, all model parameters must be given. The cost function uses arguments `par` and `par_fixed` such that, in the calibration routine, `par` can be updated by the optimizer and `par_fixed` are kept unchanged throughout calibration.

If the validation data contains a "date" column (fluxes), the simulated target time series is compared to the observed values on those same dates (e.g. for GPP). Otherwise, there should only be one observed value per site (leaf traits), and the outputs (averaged over the growing season, weighted by predicted GPP) will be compared to this single value representative of the site (e.g. Vcmax25). As an exception, when the date of a trait measurement is available, it will be compared to the trait value predicted on that date.

Value

The log-likelihood of the observed target values, assuming that they are independent, normally distributed and centered on the predictions made by the P-model run with standard deviation given as input (via 'par' because the error terms are estimated through the calibration with 'BayesianTools', as shown in the "Parameter calibration and cost functions" vignette).

Examples

```
# Compute the likelihood for a set of
# model parameter values involved in the
# temperature dependence of kphio
# and example data
cost_likelihood_pmodel(
par = c(kphio       = 0.05,
        kphio_par_a = -0.01,
        kphio_par_b = 1,     # model parameters
        err_gpp     = 2),    # err_gpp
 obs = p_model_validation,
 drivers = p_model_drivers,
 targets = c('gpp'),
 par_fixed = list(
  soilm_thetastar    = 0.6 * 240,  # old setup with soil moisture stress
  soilm_betao        = 0.0,
  beta_unitcostratio = 146.0,
  rd_to_vcmax        = 0.014,      # from Atkin et al. 2015 for C3 herbaceous
  tau_acclim         = 30.0,
  kc_jmax            = 0.41
 )
)
```

`cost_rmse_biomee` *RMSE cost function for BiomeE*

Description

Cost function for parameter calibration, which computes the root mean squared error (RMSE) between BiomeE simulations (using the input set of parameters) and observed target variables. Cost function for parameter calibration, which computes the RMSE for the biomee model fitting target variables 'GPP','LAI','Density' and 'Biomass' for a given set of parameters.

Usage

```
cost_rmse_biomee(par, obs, drivers)
```

Arguments

par	A vector containing parameter values for 'phiRL','LAI_light', 'tf_base', 'par_mort' in that order.
obs	A nested data frame of observations, following the structure of biomee_validation, for example.
drivers	A nested data frame of driver data, for example biomee_gs_leuning_drivers.

Value

The root mean squared error (RMSE) between the observed and simulated values of 'GPP','LAI','Density' and 'Biomass' (all variables have the same weight). Relative errors (difference divided by observed values) are used instead of absolute errors. The cost function performs a BiomeE model run for parameter values par and model drivers drivers given as arguments, producing the simulated values used to compute the RMSE.

Examples

```
 # do not run long-running simulations
# Compute RMSE for a set of
# model parameter values
# and example data
cost_rmse_biomee(
 par = c(3.5, 3.5, 1, 1),
 obs = biomee_validation,
 drivers = biomee_p_model_drivers
)
```

---

cost_rmse_pmodel *Cost function computing RMSE for calibration of P-model parameters*

Description

The cost function performs a P-model run for the input drivers and parameter values, and compares the output to observations of various targets by computing the root mean squared error (RMSE).

Usage

```
cost_rmse_pmodel(
  par,
  obs,
  drivers,
  targets,
  par_fixed = NULL,
  target_weights = NULL,
  parallel = FALSE,
  ncores = 2
)
```

Arguments

par	A vector of values for the parameters to be calibrated (a subset of those described in runread_pmodel_f, in order).
obs	A nested data.frame of observations, with columns 'sitename' and 'data' (see p_model_validation or p_model_validation_vcmax25 to check their structure).
drivers	A nested data.frame of driver data. See p_model_drivers for a description of the data structure.
targets	A character vector indicating the target variables for which the optimization will be done and the RMSE computed. This string must be a column name of the data data.frame belonging to the validation nested data.frame (for example 'gpp').
par_fixed	A named list of model parameter values to keep fixed during the calibration. These should complement the input par such that all model parameters are passed on to runread_pmodel_f.
target_weights	A vector of weights to be used in the computation of the RMSE if using several targets. By default (target_weights = NULL) the RMSE is computed separately for each target and then averaged. The provided weights are used to compute a weighted average of RMSE across targets.
parallel	A logical specifying whether simulations are to be parallelised (sending data from a certain number of sites to each core). Defaults to FALSE.
ncores	An integer specifying the number of cores used for parallel computing. Defaults to 2.

Details

To run the P-model, all model parameters must be given. The cost function uses arguments par and par_fixed such that, in the calibration routine, par can be updated by the optimizer and par_fixed are kept unchanged throughout calibration.

If the validation data contains a "date" column (fluxes), the simulated target time series is compared to the observed values on those same dates (e.g. for GPP). Otherwise, there should only be one observed value per site (leaf traits), and the outputs (averaged over the growing season, weighted by predicted GPP) will be compared to this single value representative of the site (e.g. Vcmax25).

As an exception, when the date of a trait measurement is available, it will be compared to the trait value predicted on that date.

Value

The root mean squared error (RMSE) between observed values and P-model predictions. The RMSE is computed for each target separately and then aggregated (mean or weighted average).

Examples

```
# Compute RMSE for a set
# of model parameter values
# and example data
cost_rmse_pmodel(
 par = c(0.05, -0.01, 0.5),  # kphio related parameters
 obs = p_model_validation,
 drivers = p_model_drivers,
 targets = c('gpp'),
 par_fixed = list(
  soilm_thetastar    = 0.6 * 240,  # old setup with soil moisture stress
  soilm_betao        = 0.0,
  beta_unitcostratio = 146.0,
  rd_to_vcmax        = 0.014,      # from Atkin et al. 2015 for C3 herbaceous
  tau_acclim         = 30.0,
  kc_jmax            = 0.41
 )
)
```

---

init_dates_dataframe *Initialises a tibble with dates*

Description

Creates a tibble with rows for each date from 'yrstart' to 'yrend' in 'yyyy-mm-dd' format. Intervals of dates are specified by argument 'freq'. ddf <- init_dates_dataframe(2000, 2003, startmoy=1, startdoy=1, freq="days", endmoy=12, enddom=31, noleap=FALSE)

Usage

```
init_dates_dataframe(
  yrstart,
  yrend,
  startmoy = 1,
  startdoy = 1,
  freq = "days",
  endmoy = 12,
  enddom = 31,
  noleap = FALSE
)
```

Arguments

`yrstart`	An integer defining the start year of dates covered by the dataframe.
`yrend`	An integer defining the end year of dates covered by the dataframe.
`startmoy`	An integer defining the start month-of-year of dates covered by the dataframe. Defaults to 1.
`startdoy`	An integer defining the start day-of-year of dates covered by the dataframe. Defaults to 1.
`freq`	A character string specifying the time steps of dates (in rows). Defaults to `"days"`. Any of `"days"`, `"months"`, `"years"`. If `freq = "months"` the 15^{th} day of the months is used as date, and if `freq = "years"` the 1^{st} of January of each year is returned.
`endmoy`	An integer defining the end month-of-year of dates covered by the dataframe. Defaults to 12.
`enddom`	An integer defining the end day-of-year of dates covered by the dataframe. Defaults to 31.
`noleap`	Whether leap years are ignored, that is, whether the 29^{th} of February is removed. Defaults to `FALSE`.

Value

A tibble with dates.

`p_model_drivers` *rsofun P-model driver data*

Description

Small dataset representing the driver to run the P-model at the FR-Pue site. It can also be used together with daily GPP flux time series data from CH-LAE (`p_model_validation`) to optimize model parameters. To optimize model parameters to leaf traits data use the datasets `p_model_drivers_vcmax25` and `p_model_validation_vcmax25`.

Usage

```
p_model_drivers
```

Format

A tibble of driver data:

sitename A character string containing the site name.

forcing A tibble of a time series of forcing climate data, including the following data:

date Date of the observation in YYYY-MM-DD format.

temp Daytime average air temperature in °C.

vpd Daytime average vapour pressure deficit in Pa.

ppfd Photosynthetic photon flux density (PPFD) in mol m^{-2} s^{-1}. If all values are NA, it indicates that PPFD should be calculated by the SPLASH model and column ccov.

netrad Net radiation in W m^{-2}. WARNING: This is currently ignored as a model forcing.

patm Atmospheric pressure in Pa.

snow Snow in water equivalents mm s^{-1}.

rain Rain as precipitation in liquid form in mm s^{-1}.

tmin Daily minimum air temperature in °C.

tmax Daily maximum air temperature in °C.

fapar Fraction of photosynthetic active radiation (fAPAR), taking values between 0 and 1.

co2 Atmospheric CO_2 concentration in ppm.

ccov Cloud coverage in %. This is only used when either PPFD or net radiation are not prescribed.

params_siml A tibble of simulation parameters, including the following data:

spinup A logical value indicating whether this simulation does spin-up.

spinupyears Number of spin-up years.

recycle Number of first N years of forcing data.frame that are recycled for spin-up.

outdt An integer indicating the output periodicity.

ltre A logical value, TRUE if evergreen tree.

ltne A logical value, TRUE if evergreen tree and N-fixing.

ltrd A logical value, TRUE if deciduous tree.

ltnd A logical value, TRUE if deciduous tree and N-fixing.

lgr3 A logical value, TRUE if grass with C3 photosynthetic pathway.

lgn3 A logical value, TRUE if grass with C3 photosynthetic pathway and N-fixing.

lgr4 A logical value, TRUE if grass with C4 photosynthetic pathway.

site_info A tibble containing site meta information. This data structure can be freely used for documenting the dataset, but must include at least the following data:

lon Longitude of the site location in degrees east.

lat Latitude of the site location in degrees north.

elv Elevation of the site location, in meters above sea level.

whc A numeric value for the rooting zone water holding capacity (in mm)

Source

Pastorello, G., Trotta, C., Canfora, E. et al. The FLUXNET2015 dataset and the ONEFlux processing pipeline for eddy covariance data. Sci Data 7, 225 (2020). https://doi.org/10.1038/s41597-020-0534-3

University of East Anglia Climatic Research Unit; Harris, I.C.; Jones, P.D.; Osborn, T. (2021): CRU TS4.05: Climatic Research Unit (CRU) Time-Series (TS) version 4.05 of high-resolution gridded data of month-by-month variation in climate (Jan. 1901- Dec. 2020). NERC EDS Centre for Environmental Data Analysis, date of citation. https://catalogue.ceda.ac.uk/uuid/c26a65020a5e4b80b20018f148556681

Weedon, G. P., G. Balsamo, N. Bellouin,S. Gomes, M. J. Best, and P. Viterbo(2014), The WFDEI meteorologicalforcing data set: WATCH Forcing Datamethodology applied to ERA-Interimreanalysis data, Water Resour. Res.,50,7505–7514, doi:10.1002/2014WR015638.

Fick, S.E. and R.J. Hijmans, 2017. WorldClim 2: new 1km spatial resolution climate surfaces for global land areas. International Journal of Climatology 37 (12): 4302-4315.

`p_model_drivers_vcmax25`

rsofun P-model driver data (for leaf traits)

Description

Small dataset representing the driver to run the P-model at four separate sites. It can also be used together with leaf traits data from these four sites (`p_model_validation_vcmax25`) to optimize model parameters. To optimize model parameters to GPP flux data use the datasets `p_model_drivers` and `p_model_validation`.

Usage

```
p_model_drivers_vcmax25
```

Format

See `p_model_drivers`

Source

Atkin, O. K., Bloomfield, K. J., Reich, P. B., Tjoelker, M. G., Asner, G. P., Bonal, D., et al. (2015). Global variability in leaf respiration in relation to climate, plant functional types and leaf traits. New Phytol. 206 (2), 614–636. doi:10.1111/nph.13253

University of East Anglia Climatic Research Unit; Harris, I.C.; Jones, P.D.; Osborn, T. (2021): CRU TS4.05: Climatic Research Unit (CRU) Time-Series (TS) version 4.05 of high-resolution gridded data of month-by-month variation in climate (Jan. 1901- Dec. 2020). NERC EDS Centre for Environmental Data Analysis, date of citation. https://catalogue.ceda.ac.uk/uuid/c26a65020a5e4b80b20018f148556681

Weedon, G. P., G. Balsamo, N. Bellouin,S. Gomes, M. J. Best, and P. Viterbo(2014), The WFDEI meteorologicalforcing data set: WATCH Forcing Datamethodology applied to ERA-Interimreanalysis data, Water Resour. Res.,50,7505–7514, doi:10.1002/2014WR015638.

Fick, S.E. and R.J. Hijmans, 2017. WorldClim 2: new 1km spatial resolution climate surfaces for global land areas. International Journal of Climatology 37 (12): 4302-4315.

C.D. Keeling, R.B. Bacastow, A.E. Bainbridge, C.A. Ekdahl, P.R. Guenther, and L.S. Waterman, (1976), Atmospheric carbon dioxide variations at Mauna Loa Observatory, Hawaii, Tellus, vol. 28, 538-551

p_model_output *rsofun P-model output data*

Description

Example output dataset from a p-model run using p_model_drivers See run_pmodel_f_bysite for a detailed description of the outputs.

Usage

```
p_model_output
```

Format

An object of class tbl_df (inherits from tbl, data.frame) with 1 rows and 3 columns.

p_model_output_vcmax25 *rsofun P-model output data (using vcmax25 drivers)*

Description

Example output dataset from a p-model run using p_model_drivers_vcmax25 See run_pmodel_f_bysite for a detailed description of the outputs.

Usage

```
p_model_output_vcmax25
```

Format

An object of class tbl_df (inherits from tbl, data.frame) with 4 rows and 3 columns.

p_model_validation *rsofun P-model GPP validation data*

Description

Small example dataset of target observations (daily GPP flux data) to optimize model parameters with the function `calib_sofun`

Usage

```
p_model_validation
```

Format

A tibble of validation data:

sitename A character string containing the site name (e.g. 'FR-Pue').

data A tibble [2,920 x 3] with time series for the following variables:

date Date vector with format YYYY-MM-DD.

gpp The observed Gross Primary Productivity (GPP) for each time stamp (in gC m^{-2} d^{-1}).

gpp_unc The uncertainty of the GPP (in gC m^{-2} d^{-1}).

Source

Pastorello, G., Trotta, C., Canfora, E. et al. The FLUXNET2015 dataset and the ONEFlux processing pipeline for eddy covariance data. Sci Data 7, 225 (2020). https://doi.org/10.1038/s41597-020-0534-3

Examples

```
require(ggplot2); require(tidyr)
p_model_validation %>% tidyr::unnest(data)
```

p_model_validation_vcmax25 *rsofun P-model Vcmax25 validation data*

Description

Small example dataset of target observations (leaf trait data) to optimize model parameters with the function `calib_sofun`

Usage

```
p_model_validation_vcmax25
```

Format

A tibble of validation data:

sitename A character string containing the site names (e.g. 'Reichetal_Colorado').

data A tibble [1 x 2] with observations for the following variables:

vcmax25 The observed maximum rate of carboxylation (Vcmax), normalised to 25^o C (in mol C m^{-2} d^{-1}), aggregated over different plant species in each site.

vcmax25_unc The uncertainty of the Vcmax25 (in mol C m^{-2} d^{-1}), calculated as the standard deviation among Vcmax25 observations for several species per site or as the total standard deviation across sites for single-plant-species sites.

Source

Atkin, O. K., Bloomfield, K. J., Reich, P. B., Tjoelker, M. G., Asner, G. P., Bonal, D., et al. (2015). Global variability in leaf respiration in relation to climate, plant functional types and leaf traits. New Phytol. 206 (2), 614–636. doi:10.1111/nph.13253

Examples

```
require(ggplot2); require(tidyr)
p_model_validation_vcmax25 %>% tidyr::unnest(data)
```

---

runread_biomee_f *Run BiomeE*

Description

Runs BiomeE model for multiple sites.

Usage

```
runread_biomee_f(drivers, makecheck = TRUE, parallel = FALSE, ncores = 1)
```

Arguments

drivers	A nested data frame with one row for each site and columns named according to the arguments of function run_biomee_f_bysite. Namely sitename, params_siml, site_info, forcing, params_tile,params_species, init_cohort and init_soil.
makecheck	A logical specifying whether checks are performed to verify forcings and model parameters. TRUE by default.
parallel	Deprecated. Use ncores instead.
ncores	An integer specifying the number of cores used for parallel computing (sites processed in parallel). Default: 1 (no parallel execution).

Value

A data frame (tibble) with one row for each site. The columns are the site information `site_info` and one column per land unit (LU) in addition to an aggregated output `aggregated`. By default, the only LU is named `data` and `aggregated` is not present since aggregating one LU is not useful. When multiple LU are configured (using `init_lu`), the columns are named using the LU name provided in `init_lu`. See `run_biomee_f_bysite` for a detailed description of the outputs. Example outputs are provided as `biomee_p_model_output` and `biomee_p_model_luluc_output`.

Examples

```
# Example BiomeE model run
 # do not run long-running simulations
runread_biomee_f(
  drivers = biomee_p_model_drivers
)
## Not run:  # do not run this long-running example at all, only *show* example
runread_biomee_f(
  drivers = biomee_gs_leuning_drivers
)

## End(Not run)
```

`runread_pmodel_f` *Run P-model*

Description

Runs P-model for multiple sites.

Usage

```
runread_pmodel_f(drivers, par, makecheck = TRUE, parallel = FALSE, ncores = 1)
```

Arguments

`drivers`	A nested data frame with one row for each site and columns named according to the arguments of function `runread_pmodel_f`. Namely `sitename`, `params_siml`, `site_info` and `forcing`.
`par`	A named list of free (calibratable) model parameters.

kphio The quantum yield efficiency at optimal temperature φ_0, in mol mol^{-1}. When temperature dependence is used, it corresponds to the multiplicative parameter c (see Details).

kphio_par_a The shape parameter a of the temperature-dependency of quantum yield efficiency (see Details). To disable the temperature dependence, set `kphio_par_a = 0`.

kphio_par_b The optimal temperature parameter b of the temperature dependent quantum yield efficiency (see Details), in ^{o}C.

soilm_thetastar The threshold parameter θ^* in the soil moisture stress function (see Details), given in mm. To turn off the soil moisture stress, set `soilm_thetastar = 0`.

soilm_betao The intercept parameter β_0 in the soil moisture stress function (see Details). This is the parameter calibrated in Stocker et al. 2020 GMD.

beta_unitcostratio The unit cost of carboxylation, corresponding to $\beta = b/a'$ in Eq. 3 of Stocker et al. 2020 GMD.

rd_to_vcmax Ratio of Rdark (dark respiration) to Vcmax25.

tau_acclim Acclimation time scale of photosynthesis, in days.

kc_jmax Parameter for Jmax cost ratio (corresponding to c* in Stocker et al. 2020 GMD).

`makecheck` A logical specifying whether checks are performed to verify forcings and model parameters. `TRUE` by default.

`parallel` A logical specifying whether simulations are to be parallelised (sending data from a certain number of sites to each core). Defaults to `FALSE`.

`ncores` An integer specifying the number of cores used for parallel computing (by default `ncores = 2`).

Details

Depending on the input model parameters, it's possible to run the different P-model setups presented in Stocker et al. 2020 GMD. The P-model version implemented in this package allows more flexibility than the one presented in the paper, with the following functions:

The temperature dependence of the quantum yield efficiency is given by:
$\varphi_0(T) = c(1 + a(T - b)^2)$ if $0 < c(1 + a(T - b)^2) < 1$,
$\varphi_0(T) = 0$ if $c(1 + a(T - b)^2) \leq 0$, and
$\varphi_0(T) = 1$ if $c(1 + a(T - b)^2) \geq 1$.
The ORG setup can be reproduced by setting `kphio_par_a = 0` and calibrating the `kphio` parameter only. The BRC setup (which calibrates $c_L = \frac{a_L b_L}{4}$ in Eq. 18) is more difficult to reproduce, since the temperature-dependency has been reformulated and a custom cost function would be necessary for calibration. The new parameters are related to c_L as follows:
$a = -0.0004919819$
$b = 32.35294$
$c = 0.6910823 c_L$

The soil moisture stress is implemented as
$\beta(\theta) = \frac{\beta_0 - 1}{\theta^{*2}}(\theta - \theta^*)^2 + 1$ if $0 \leq \theta \leq \theta^*$ and
$\beta(\theta) = 1$ if $\theta > \theta^*$.
In Stocker et al. 2020 GMD, the threshold plant-available soil water is set as θ^* `= 0.6 * whc` where `whc` is the site's water holding capacity. Also, the β reduction at low soil moisture ($\beta_0 = \beta(0)$) was parameterized as a linear function of mean aridity (Eq. 20 in Stocker et al. 2020 GMD) but is considered a constant model parameter in this package. Hence, the FULL calibration setup cannot be exactly replicated.

Value

A data frame (tibble) with one row for each site, site information stored in the nested column `site_info` and outputs stored in the nested column `data`. See `run_pmodel_f_bysite` for a detailed description of the outputs. Example outputs are provided as `biomee_p_model_output` and `biomee_gs_leuning_output`.

Examples

```
# Define model parameter values from previous work
params_modl <- list(
  kphio              = 0.04998,    # setup ORG in Stocker et al. 2020 GMD
  kphio_par_a        = 0.0,        # disable temperature-dependence of kphio
  kphio_par_b        = 1.0,
  soilm_thetastar    = 0.6 * 240,  # old setup with soil moisture stress
  soilm_betao        = 0.0,
  beta_unitcostratio = 146.0,
  rd_to_vcmax        = 0.014,      # from Atkin et al. 2015 for C3 herbaceous
  tau_acclim         = 30.0,
  kc_jmax            = 0.41
)

# Run the model for these parameters and the example drivers
output <- rsofun::runread_pmodel_f(
  drivers = rsofun::p_model_drivers,
  par = params_modl)
output_vcmax25 <- rsofun::runread_pmodel_f(
  drivers = rsofun::p_model_drivers_vcmax25,
  par = params_modl)
```

---

run_biomee_f_bysite *Run BiomeE (R wrapper)*

Description

Run BiomeE Fortran model on single site.

Usage

```
run_biomee_f_bysite(
  sitename,
  params_siml,
  site_info,
  forcing,
  params_tile,
  params_species,
  init_cohort,
  init_soil,
  init_lu = NULL,
```

```
  luc_forcing = NULL,
  makecheck = TRUE
)
```

Arguments

sitename	Site name.
params_siml	Simulation parameters.
site_info	Site meta info in a data.frame.
forcing	A data.frame of forcing climate data, used as input.
params_tile	Tile-level model parameters, into a single row data.frame.
params_species	A data.frame containing species-specific model parameters, with one species per row. See examples biomee_gs_leuning_drivers or biomee_p_model_drivers
init_cohort	A data.frame of initial cohort specifications.
init_soil	A data.frame of initial soil pools.
init_lu	A data.frame of initial land unit (LU) specifications.
luc_forcing	An array of land use change (LUC) used during transient phase. For further specifications of above inputs and examples see biomee_gs_leuning_drivers, biomee_p_model_drivers, or biomee_p_model_luluc_drivers.
makecheck	A logical specifying whether checks are performed to verify forcings and model parameters. TRUE by default.

Value

A data.frame with columns containing model output for each land unit (LU). See examples biomee_gs_leuning_output, biomee_p_model_output, or biomee_p_model_luluc_output. If only one land unit (LU) is simulated, the column is named 'data'. If multiple land units (LU) are simulated, the columns are named according to the LU names. If multiple land units (LU) are simulated, an additional column 'aggregated' contains output aggregating all tiles as well as product pools. Model output for each land unit (LU) is provided as a list. Each list has elements: output_daily_tile, output_annual_tile, and output_annual_cohorts. Model output for the aggregated land units (LU) is provided as a list containing output_daily_cell.

output_daily_tile A data.frame with daily outputs at tile level.

year Year of the simulation.

doy Day of the year.

Tk Air temperature (Kelvin).

Prcp Precipitation (mm m^{-2} day^{-1}).

SoilWater Soil water content in root zone (kg m^{-2}).

Transp Transpiration (mm m^{2-} day^{-1}).

Evap Evaporation (mm m^{-2} day^{-1}).

Runoff Water runoff (mm m^{-2} day^{-1}).

ws1 Volumetric soil water content for layer 1.

ws2 Volumetric soil water content for layer 2.

ws3 Volumetric soil water content for layer 3.
LAI Leaf area index (m^2/m^2).
NPP Net primary productivity (kg C m^{-2} day^{-1}).
GPP Gross primary production (kg C m^{-2} day^{-1}).
Rauto Plant autotrophic respiration (kg C m^{-2} day^{-1}).
Rh Heterotrophic respiration (kg C m^{-2} day^{-1}).
NSC Non-structural carbon (kg C m^{-2}).
seedC Biomass of seeds (kg C m^{-2}).
leafC Biomass of leaves (kg C m^{-2}).
rootC Biomass of fine roots (kg C m^{-2}).
sapwoodC Biomass of sapwood (kg C m^{-2}).
heartwoodC Biomass of heartwood (kg C m^{-2}).
NSN Non-structural N pool (kg N m^{-2}).
seedN Nitrogen of seeds (kg N m^{-2}).
leafN Nitrogen of leaves (kg N m^{-2}).
rootN Nitrogen of roots (kg N m^{-2}).
sapwoodN Nitrogen of sapwood (kg N m^{-2}).
heartwoodN Nitrogen of heartwood (kg N m^{-2}).
mcrbC Microbial carbon (kg C m^{-2}).
fastSOM Fast soil carbon pool (kg C m^{-2}).
slowSOM Slow soil carbon pool (kg C m^{-2}).
mcrbN Microbial nitrogen (kg N m^{-2}).
fastSoilN Fast soil nitrogen pool (kg N m^{-2}).
slowSoilN Slow soil nitrogen pool (kg N m^{-2}).
mineralN Mineral nitrogen pool (kg N m^{-2}).
N_uptk Nitrogen uptake (kg N m^{-2} day^{-1}).

`output_annual_tile` A data.frame with annual outputs at tile level.

year Year of the simulation.
CAI Crown area index (m^2/m^2).
LAI Leaf area index (m^2/m^2).
Density Number of trees per area (trees ha^{-1}).
DBH Diameter at tile level (cm).
Density12 Tree density for trees with DBH > 12 cm (individuals ha^{-1}).
DBH12 Diameter at tile level considering trees with DBH > 12 cm(cm).
QMD12 Quadratic mean diameter at tile level considering trees withDBH > 12 cm (cm).
NPP Net primary productivity (kg C m^{-2} yr^{-1}).
GPP Gross primary productivity (kg C m^{-2} yr^{-1}).
Rauto Plant autotrophic respiration (kg C m^{-2} yr^{-1}).
Rh Heterotrophic respiration (kg C m^{-2} yr^{-1}).
Prcp Annual precipitation (mm m^{-2} yr^{-1}).
SoilWater Soil water content in root zone (kg m^{-2}).
Transp Transpiration (mm m^{-2} yr^{-1}).

Evap Evaporation (mm m^{-2} yr^{-1}).
Runoff Water runoff (mm m^{-2} yr^{-1}).
plantC Plant biomass (kg C m^{-2}).
soilC Soil carbon (kg C m^{-2}).
totC Total carbon in plant and soil (kg C m^{-2}).
plantN Plant nitrogen (kg N m^{-2}).
soilN Soil nitrogen (kg N m^{-2}).
totN Total nitrogen in plant and soil (kg N m^{-2}).
NSC Nonstructural carbohydrates (kg C m^{-2}).
seedC Seed biomass (kg C m^{-2}).
leafC Leaf biomass (kg C m^{-2}).
rootC Fine root biomass (kg C m^{-2}).
sapwoodC Sapwood biomass (kg C m^{-2}).
heartwoodC Heartwood biomass (kg C m^{-2}).
NSN Nonstructural nitrogen (kg N m^{-2}).
seedN Seed nitrogen (kg N m^{-2}).
leafN Leaf nitrogen (kg N m^{-2}).
rootN Fine root nitrogen (kg N m^{-2}).
sapwoodN Sapwood nitrogen (kg N m^{-2}).
heartwoodN Heartwood nitrogen (kg N m^{-2}).
mcrbC Microbial carbon (kg C m^{-2}).
fastSOM Fast soil carbon pool (kg C m^{-2}).
slowSOM Slow soil carbon pool (kg C m^{-2}).
mcrbN Microbial nitrogen (kg N m^{-2}).
fastSoilN Fast soil nitrogen pool (kg N m^{-2}).
slowsoilN Slow soil nitrogen pool (kg N m^{-2}).
mineralN Mineral nitrogen pool (kg N m^{-2}).
N_fxed Nitrogen fixation (kg N m^{-2}).
N_uptk Nitrogen uptake (kg N m^{-2}).
N_yrMin Annual available nitrogen (kg N m^{-2}).
N_P2S Annual nitrogen from plants to soil (kg N m^{-2}).
N_loss Annual nitrogen loss (kg N m^{-2}).
totseedC Total seed carbon (kg C m^{-2}).
totseedN Total seed nitrogen (kg N m^{-2}).
Seedling_C Total carbon from all compartments but seeds (kg C m^{-2}).
Seedling_N Total nitrogen from all compartments but seeds(kg N m^{-2}).
MaxAge Age of the oldest tree in the tile (years).
MaxVolume Maximum volume of a tree in the tile (m^3).
MaxDBH Maximum DBH of a tree in the tile (m).
NPPL Growth of a tree, including carbon allocated to leaves(kg C m^{-2} yr^{-1}).
NPPW Growth of a tree, including carbon allocated to sapwood(kg C m^{-2} yr^{-1}).
n_deadtrees Number of trees that died (trees m^{-2} yr^{-1}).

c_deadtrees Carbon biomass of trees that died (kg C m^{-2} yr^{-1}).

m_turnover Continuous biomass turnover (kg C m^{-2} yr^{-1}).

c_turnover_time Carbon turnover rate, calculated as the ratio between plant biomass and NPP (yr^{-1}).

lu_fraction Fraction of BiomeE grid cell that is occupied by this land unit (LU tile) tile (unitless, or m^{-2} LU area per m^{-2} grid cell area).

`output_annual_cohorts` A data.frame of annual outputs at the cohort level.

year Year of the simulation.

cID An integer indicating the cohort identity.

PFT An integer indicating the Plant Functional Type.

layer An integer indicating the crown layer, numbered from top to bottom.

density Number of trees per area (trees ha^{-1}).

flayer Fraction of layer area occupied by this cohort.

DBH Tree diameter (cm).

dDBH Diameter growth of a tree in this cohort (cm yr^{-1}).

height Tree height (m).

age Age of the cohort (years).

BA Basal area a tree in this cohort (m^2 $tree^{-1}$).

dBA Basal area increment of a tree in this cohort (m^2 $tree^{-1}$ yr^{-1}).

Acrown Crown area of a tree in this cohort (m^2 $tree^{-1}$).

Aleaf Total area of leaves (m^2 $tree^{-1}$).

wood Sum of sapwood and heartwood biomass of a tree in this cohort (kg C $tree^{-1}$).

NSC Non-structural carbon of a tree in this cohort (kg C $tree^{-1}$).

seedC Biomass of seeds of a tree in this cohort (kg C $tree^{-1}$).

leafC Biomass of leaves of a tree in this cohort (kg C $tree^{-1}$).

rootC Biomass of fine roots of a tree in this cohort (kg C $tree^{-1}$).

sapwoodC Biomass of sapwood of a tree in this cohort (kg C $tree^{-1}$).

heartwoodC Biomass of heartwood of a tree in this cohort (kg C $tree^{-1}$).

NSN Non-structural nitrogen of a tree in this cohort (kg N $tree^{-1}$).

treeG Total growth of a tree, including carbon allocated to seeds, leaves, fine roots, and sapwood (kg C $tree^{-1}$ yr^{-1}).

fseed Fraction of carbon allocated to seeds to total growth.

fleaf Fraction of carbon allocated to leaves to total growth.

froot Fraction of carbon allocated to fine roots to total growth.

fwood Fraction of carbon allocated to sapwood to total growth.

NPP Net primary productivity of a tree (kg C $tree^{-1}$ yr^{-1}).

GPP Gross primary productivity of a tree (kg C $tree^{-1}$ yr^{-1}).

Rauto Plant autotrophic respiration (kg C $tree^{-1}$ yr^{-1}).

N_uptk Nitrogen uptake (kg N $tree^{-1}$ yr^{-1}).

N_fxed Nitrogen fixation (kg N $tree^{-1}$ yr^{-1}).

deathrate Mortality rate of this cohort, including natural mortality, starvation and any other processes causing a loss of individuals in general (yr^{-1}).

n_deadtrees Plant to soil N flux due to mortality, including natural mortality, starvation and any other processes causing a loss of individuals in general (kg N yr^{-1} m^{-2}).

c_deadtrees Plant to soil C flux due to mortality, including natural mortality, starvation and any other processes causing a loss of individuals in general (kg C yr^{-1} m^{-2}).

If there are multiple land units (LU) there will also be a column named 'aggregated' containing a data.frame in the column 'output_annual_cell' with annual outputs aggregating all tiles present in the simulation cell. Note that quantities per m2 refer to m2 of grid cell area, i.e. the full area of the BiomeE simulation. 'lu_fraction' refers to the sum of all the tiles, which must remain constant and which represents the fraction of the cell area that is not water/ice. In most cases, it would be close to 1. It contains columns:

`output_annual_cell` A data.frame with annual outputs aggregating all tiles present in the simulation cell. Note that quantities per m^2 refer to m^2 of grid cell area, i.e. the full area of the BiomeE simulation. 'lu_fraction' refers to the sum of all the tiles, which must remain constant and which represents the fraction of the cell area that is not water/ice. In most cases, it would be close to 1.

all columns from 'output_yearly_tile' See above for output_yearly_tile, but now expressed per unit area of the BiomeE grid cell.

lu_fraction Fraction of BiomeE grid cell that is occupied by this land unit (LU tile) tile (unitless, or m^2 LU area per m^2 grid cell area).

prod_pool_1_C Carbon in product pool 1 (kg C m^{-2} grid cell).

prod_pool_1_N Nitrogen in product pool 1 (kg N m^{-2} grid cell).

prod_pool_2_C Carbon in product pool 2 (kg C m^{-2} grid cell).

prod_pool_2_N Nitrogen in product pool 2 (kg N m^{-2} grid cell).

Rprod_0_C Carbon loss rate directly from land use change (LUC) (kg C m^{-2} grid cell yr^{-1}).

Rprod_0_N Nitrogen loss rate directly from land use change (LUC) (kg C m^{-2} grid cell yr^{-1}).

Rprod_1_C Carbon loss rate from product pool 1 (kg C m^{-2} grid cell yr^{-1}).

Rprod_1_N Nitrogen loss rate from product pool 1 (kg N m^{-2} grid cell yr^{-1}).

Rprod_2_C Carbon loss rate from product pool 2 (kg C m^{-2} grid cell yr^{-1}).

Rprod_2_N Nitrogen loss rate from product pool 2 (kg N m^{-2} grid cell yr^{-1}).

Examples

```
 # do not run long-running simulations
# Example BiomeE model run

# Use example drivers data
drivers <- biomee_p_model_drivers

# Run BiomeE for the first site
mod_output <- run_biomee_f_bysite(
 sitename = drivers$sitename[1],
 params_siml = drivers$params_siml[[1]],
 site_info = drivers$site_info[[1]],
 forcing = drivers$forcing[[1]],
 params_tile = drivers$params_tile[[1]],
```

```
 params_species = drivers$params_species[[1]],
 init_cohort = drivers$init_cohort[[1]],
 init_soil = drivers$init_soil[[1]]
)
```

`run_pmodel_f_bysite` *Run P-model (time series)*

Description

Run P-model on a single site for a forcing time series.

Usage

```
run_pmodel_f_bysite(
  sitename,
  params_siml,
  site_info,
  forcing,
  params_modl,
  makecheck = TRUE,
  verbose = TRUE
)
```

Arguments

`sitename`	Site name.
`params_siml`	Simulation parameters.
`site_info`	Site meta info in a data.frame.
`forcing`	A data frame of forcing climate data, used as input.
`params_modl`	A named list of free (calibratable) model parameters. See `runread_pmodel_f`
`makecheck`	A logical specifying whether checks are performed to verify forcings and model parameters. `TRUE` by default.
`verbose`	A logical specifying whether to print warnings. Defaults to `TRUE`.
	For further specifications of above inputs and examples see `p_model_drivers` or `p_model_drivers_vcmax25`

Details

Depending on the input model parameters, it's possible to run the different P-model setups presented in Stocker et al. 2020 GMD. The P-model version implemented in this package allows more flexibility than the one presented in the paper, with the following functions:

The temperature dependence of the quantum yield efficiency is given by:
$\varphi_0(T) = c(1 + a(T - b)^2)$ if $0 < c(1 + a(T - b)^2) < 1$,

$\varphi_0(T) = 0$ if $c(1 + a(T - b)^2) \leq 0$, and
$\varphi_0(T) = 1$ if $c(1 + a(T - b)^2) \geq 1$.
The ORG setup can be reproduced by setting `kphio_par_a = 0` and calibrating the `kphio` parameter only. The BRC setup (which calibrates $c_L = \frac{a_L b_L}{4}$ in Eq. 18) is more difficult to reproduce, since the temperature-dependency has been reformulated and a custom cost function would be necessary for calibration. The new parameters are related to c_L as follows:
$a = -0.0004919819$
$b = 32.35294$
$c = 0.6910823 c_L$

The soil moisture stress is implemented as
$\beta(\theta) = \frac{\beta_0 - 1}{\theta^{*2}}(\theta - \theta^*)^2 + 1$ if $0 \leq \theta \leq \theta^*$ and
$\beta(\theta) = 1$ if $\theta > \theta^*$.
In Stocker et al. 2020 GMD, the threshold plant-available soil water is set as θ^* `= 0.6 * whc` where `whc` is the site's water holding capacity. Also, the β reduction at low soil moisture ($\beta_0 = \beta(0)$) was parameterized as a linear function of mean aridity (Eq. 20 in Stocker et al. 2020 GMD) but is considered a constant model parameter in this package. Hence, the FULL calibration setup cannot be exactly replicated.

Value

Model output is provided as a tidy dataframe, with columns:

`date` Date of the observation in YYYY-MM-DD format.

`year_dec` Decimal representation of year and day of the year (for example, 2007.000 corresponds to 2007-01-01 and 2007.003 to 2007-01-02.

`fapar` Fraction of photosynthetic active radiation (fAPAR), taking values between 0 and 1.

`gpp` Gross Primary Productivity (GPP) for each time stamp (in gC m^{-2} d^{-1}).

`aet` Actual evapotranspiration (AET), calculated by SPLASH following Priestly-Taylor (in mm d^{-1}).

`le` Latent heat flux (in J m^{-2} d^{-1}).

`pet` Potential evapotranspiration (PET), calculated by SPLASH following Priestly-Taylor (in mm d^{-1}).

`vcmax` Maximum rate of RuBisCO carboxylation (Vcmax) (in mol C m^{-2} s^{-1}).

`jmax` Maximum rate of electron transport for RuBP regeneration (in mol CO_2 m^{-2} s^{-1}).

`vcmax25` Maximum rate of carboxylation (Vcmax), normalised to 25^{o}C (in mol C m^{-2} s^{-1}).

`jmax25` Maximum rate of electron transport, normalised to 25^{o}C (in mol C m^{-2} s^{-1}).

`gs_accl` Acclimated stomatal conductance (in mol C (mol photons)$^{-1}$ Pa^{-1}. (Multiply by ppfd (mol photons m^{-2} d^{-1}) and fapar to express per unit ground area and time.)

`wscal` Relative soil water content, between 0 (permanent wilting point, PWP) and 1 (field capacity, FC).

`chi` Ratio of leaf-internal to ambient CO_2, ci:ca (unitless).

`iwue` Intrinsic water use efficiency (iWUE) (unitless, multiply with patm (Pa) to get iWUE in Pa).

`rd` Dark respiration (Rd) in gC m^{-2} s^{-1}. (Multiply by 1/12 (mol C / gC) to convert to mol C m^{-2} s^{-1}.)

`tsoil` Soil temperature, in ^{o}C.

`netrad` Net radiation, in W m^{-2}. WARNING: this is currently ignored as a model forcing. Instead, net radiation is internally calculated by SPLASH.

`wcont` Soil water content, in mm.

`snow` Snow water equivalents, in mm.

`cond` Water input by condensation, in mm d^{-1}

`cleaf` C mass of a virtual leaf carbon pool to keep track of isotopic composition, in gC m^{-2}

`cleafd13c` 13C isotopic signature (delta) of `cleaf`, in permil.

Examples

```
# Define model parameter values from previous work
params_modl <- list(
  kphio              = 0.04998,    # setup ORG in Stocker et al. 2020 GMD
  kphio_par_a        = 0.0,        # disable temperature-dependence of kphio
  kphio_par_b        = 1.0,
  soilm_thetastar    = 0.6 * 240,  # old setup with soil moisture stress
  soilm_betao        = 0.0,
  beta_unitcostratio = 146.0,
  rd_to_vcmax        = 0.014,      # from Atkin et al. 2015 for C3 herbaceous
  tau_acclim         = 30.0,
  kc_jmax            = 0.41
)

# Run the Fortran P-model
mod_output <- run_pmodel_f_bysite(
  # unnest drivers example data
  sitename = p_model_drivers$sitename[1],
  params_siml = p_model_drivers$params_siml[[1]],
  site_info = p_model_drivers$site_info[[1]],
  forcing = p_model_drivers$forcing[[1]],
  params_modl = params_modl
 )
```

---

run_pmodel_onestep_f_bysite

Run P-model (single time step)

Description

Run P-model on a single site for a single time step. This does not include the simulation of ecosystem-level quantities, water limitation, nor a simulation of water fluxes. Instead, this corresponds to a leaf-level representation of the acclimation of photosynthesis.

Usage

```
run_pmodel_onestep_f_bysite(lc4, forcing, params_modl, makecheck = TRUE)
```

Arguments

lc4	Locigical specifying whether P-model simulation is for C4 (as opposed to C3). Defaults to FALSE.
forcing	A data frame of forcing climate data, used as input (single row).
params_modl	A named list of free (calibratable) model parameters. See runread_pmodel_f
makecheck	A logical specifying whether checks are performed to verify forcings and model parameters. TRUE by default.
	For further specifications of above inputs and examples see p_model_drivers or p_model_drivers_vcmax25

Details

TBC

Value

Model output is provided as a tidy dataframe, with columns:

vcmax Maximum rate of RuBisCO carboxylation (Vcmax) (in mol C m^{-2} s^{-1}).

jmax Maximum rate of electron transport for RuBP regeneration (in mol CO_2 m^{-2} s^{-1}).

vcmax25 Maximum rate of carboxylation (Vcmax), normalised to 25^{o}C (in mol C m^{-2} s^{-1}).

jmax25 Maximum rate of electron transport, normalised to 25^{o}C (in mol C m^{-2} s^{-1}).

gs_accl Acclimated stomatal conductance (in mol C (mol photons)$^{-1}$ Pa^{-1}. (Multiply by ppfd (mol photons m^{-2} d^{-1}) and fapar to express per unit ground area and time.)

chi Ratio of leaf-internal to ambient CO_2, ci:ca (unitless).

iwue Intrinsic water use efficiency (iWUE) (unitless, multiply with patm (Pa) to get iWUE in Pa).

rd Dark respiration (Rd) in gC m^{-2} s^{-1}. (Multiply by 1/12 (mol C / gC) to convert to mol C m^{-2} s^{-1}.)

bigdelta 13C isotope discrimination of leaf assimilates against atmospheric signature (permil).

Examples

```
# Define model parameter values from previous work
params_modl <- list(
  kphio              = 0.04998,    # setup ORG in Stocker et al. 2020 GMD
  kphio_par_a        = 0.0,        # disable temperature-dependence of kphio
  kphio_par_b        = 1.0,
  beta_unitcostratio = 146.0,
  rd_to_vcmax        = 0.014,      # from Atkin et al. 2015 for C3 herbaceous
  kc_jmax            = 0.41
)

# Run the Fortran P-model
run_pmodel_onestep_f_bysite(
  lc4 = FALSE,
  forcing = data.frame(
    temp  = 20,           # temperature, deg C
```

```
    vpd   = 1000,         # Pa,
    ppfd  = 300/10^6,     # mol/m2/s
    co2   = 400,          # ppm,
    patm  = 101325        # Pa
  ),
  params_modl = list(
    kphio              = 0.04998,    # setup ORG in Stocker et al. 2020 GMD
    kphio_par_a        = 0.0,        # disable temperature-dependence of kphio
    kphio_par_b        = 1.0,
    beta_unitcostratio = 146.0,
    rd_to_vcmax        = 0.014,      # from Atkin et al. 2015 for C3 herbaceous
    kc_jmax            = 0.41
  ),
  makecheck = TRUE
)
```

Index