

# Package ‘rsofun’

May 9, 2026

**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](https://doi.org/10.5194/gmd-13-1545-2020)>), SPLASH (Davis et al. (2017) <[doi:10.5194/gmd-10-689-2017](https://doi.org/10.5194/gmd-10-689-2017)>) and BiomeE (Weng et al. (2015) <[doi:10.5194/bg-12-2655-2015](https://doi.org/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, multplyr, 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](mailto:benjamin.stocker@gmail.com)>

**Repository** CRAN

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

## Contents

biomee_gs_leuning_drivers . . . . .	3
biomee_gs_leuning_output . . . . .	7
biomee_p_model_drivers . . . . .	7
biomee_p_model_luluc_drivers . . . . .	8
biomee_p_model_luluc_output . . . . .	8
biomee_p_model_output . . . . .	9
biomee_validation . . . . .	9
build_luc_matrix . . . . .	10
calib_sofun . . . . .	11
cost_likelihood_biomee . . . . .	12
cost_likelihood_pmodel . . . . .	14
cost_rmse_biomee . . . . .	15
cost_rmse_pmodel . . . . .	16
init_dates_dataframe . . . . .	18
p_model_drivers . . . . .	19
p_model_drivers_vcmax25 . . . . .	21
p_model_output . . . . .	22
p_model_output_vcmax25 . . . . .	22
p_model_validation . . . . .	23
p_model_validation_vcmax25 . . . . .	23
runread_biomee_f . . . . .	24
runread_pmodel_f . . . . .	25
run_biomee_f_bysite . . . . .	27
run_pmodel_f_bysite . . . . .	33
run_pmodel_onestep_f_bysite . . . . .	35

**Index**

**38**

---

 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<sup>-2</sup> 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 ( $\text{mol m}^{-2} \text{s}^{-1}$ )
- tair** Air temperature (deg C)
- vpd** Vapor pressure deficit (Pa)
- rain** Precipitation ( $\text{kgH}_2\text{O m}^{-2} \text{s}^{-1}$  ==  $\text{mm s}^{-1}$ )
- wind** Wind velocity ( $\text{m s}^{-1}$ )
- pair** Atmospheric pressure (Pa)
- co2** Atmospheric  $\text{CO}_2$  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 ( $\text{yr}^{-1}$ ).
- K2** Slow soil C decomposition rate ( $\text{yr}^{-1}$ ).
- K\_nitrogen** Mineral Nitrogen turnover rate ( $\text{yr}^{-1}$ ).
- MLmixRatio** Ratio of C and N returned to litters from microbes.
- etan** N loss rate through runoff (organic and mineral) ( $\text{yr}^{-1}$ ).
- LMamin** Minimum LMA, leaf mass per unit area,  $\text{kg C m}^{-2}$ .
- 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 ( $\text{yr}^{-1}$ ).
- rho\_FR** Material density of fine roots ( $\text{kg C m}^{-3}$ ).

**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 ( $\text{mol m}^{-2} \text{s}^{-1} \text{MPa}^{-1}$ ).  
**leaf\_size** Characteristic leaf size.

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

**Vannual** Annual productivity per unit area at full sun ( $\text{kg C m}^{-2} \text{year}^{-2}$ ).  
**wet\_leaf\_dreg** Wet leaf photosynthesis down-regulation.  
**m\_cond** Factor of stomatal conductance.  
**alpha\_phot** Photosynthesis efficiency.  
**gamma\_L** Leaf respiration coefficient, in  $\text{yr}^{-1}$ .  
**gamma\_LN** Leaf respiration coefficient per unit N.  
**gamma\_SW** Sapwood respiration rate, in  $\text{kg C m}^{-2} \text{yr}^{-1}$ .  
**gamma\_FR** Fine root respiration rate,  $\text{kg C kg C}^{-1} \text{yr}^{-1}$ .  
**tk\_crit** Critical temperature triggering offset of phenology, in Kelvin.  
**tk\_crit\_on** Critical temperature triggering 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 ( $\text{height} = \text{alphaHT} * \text{DBH}_m ** \text{thetaHT}$ ), in  $\text{m m}^{-\text{thetaHT}}$ .

**thetaHT** Coefficient for allometry ( $\text{height} = \text{alphaHT} * \text{DBH}_m ** \text{thetaHT}$ ), in  $\text{m}^{-\text{thetaHT}}$ .  
**alphaCA** Coefficient for allometry (projected crown area =  $\text{pi} * (\text{alphaCA} * \text{DBH}_m ** \text{thetaCA})$ ), in  $\text{m}^{2/\text{thetaCA}-1}$ .  
**thetaCA** Coefficient for allometry (projected crown area =  $\text{pi} * (\text{alphaCA} * \text{DBH}_m ** \text{thetaCA})$ ), unitless. Dybzinski (eq. G1) showed that  $\text{thetaCA} = \text{thetaBM} - 1$ .  
**alphaBM** Coefficient for allometry ( $\text{biomass} = \text{alphaBM} * \text{DBH} ** \text{thetaBM}$ ), in  $\text{kg C m}^{-\text{thetaBM}}$ .  
**thetaBM** Coefficient for allometry ( $\text{biomass} = \text{alphaBM} * \text{DBH} ** \text{thetaBM}$ ), unitless. Dybzinski (eq. G1) showed that  $\text{thetaCA} = \text{thetaBM} - 1$ .

**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 ( $\text{yr}^{-1}$ ).

**mortrate\_d\_u** Understory tree mortality rate ( $\text{yr}^{-1}$ ).

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

**leafLS** TODO  
**LNbase** Basal leaf N per unit area, in  $\text{kg N m}^{-2}$ .  
**CNleafsupport** TODO  
**rho\_wood** Wood density ( $\text{kg C m}^{-3}$ ).

**taperfactor** TODO

**LAI<sub>max</sub>** Maximum crown LAI (leaf area index).

**tauNSC** TODO

**fNS<sub>max</sub>** Multiplier for NSN<sub>max</sub> 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: CN<sub>leaf0</sub>** TODO

**CN<sub>sw0</sub>** TODO

**CN<sub>wood0</sub>** TODO

**CN<sub>root0</sub>** TODO

**CN<sub>seed0</sub>** TODO

**Nfix<sub>rate0</sub>** Reference N fixation rate (kg N kg C<sup>-1</sup> root).

**Nfix<sub>Cost0</sub>** Carbon cost of N fixation (kg C kg N<sup>-1</sup>).

**internal\_gap\_frac** TODO

**The following columns pertaining to the calibratable parameters: k<sub>phio</sub>** Quantum yield efficiency  $\varphi_0$ , in mol mol<sup>-1</sup>.

**phiRL** Ratio of fine root to leaf area.

**LAI<sub>light</sub>** 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<sup>2</sup>.

**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 array). 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<sup>-2</sup>.

**init\_slow\_soil\_C** Initial slow soil carbon, in kg C m<sup>-2</sup>.

**init\_N<sub>mineral</sub>** Mineral nitrogen pool, in kg N m<sup>-2</sup>.

**N<sub>input</sub>** Annual nitrogen input to soil N pool, in kg N m<sup>-2</sup> yr<sup>-1</sup>.

**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<sup>-2</sup> LU area per m<sup>-2</sup> 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<sub>input</sub>** Additional inorg N supply (to account for N fertiliser application), in kg m<sup>-2</sup> yr<sup>-1</sup>. 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  $n \times n$  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 transferred 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/14>

---

build_luc_matrix	<i>Build LUC matrix</i>
------------------	-------------------------

---

### 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 <code>n_luxn_lu</code> . The <code>n_lu</code> first values are the transitions from each LU to the first LU, and so on. If the sequence contains more years than <code>n_years</code> , 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 matrix 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	<i>Calibrates SOFUN model parameters</i>
-------------	--

---

### 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 <a href="#">p_model_drivers</a> 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 <a href="#">p_model_validation</a> 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 <a href="#">GenSA</a> . If method = 'BayesianTools' the list should include at least settings and sampler, see <a href="#">BayesianTools::runMCMC</a> .
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 <a href="#">runread_pmodel_f</a> ), 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 <a href="#">p_model_validation</a> or <a href="#">p_model_validation_vcmax25</a> to check their structure).
drivers	A nested data.frame of driver data. See <a href="#">p_model_drivers</a> 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 <a href="#">runread_pmodel_f</a> .
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
  )
)
```

**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
)
```

---

code 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 <a href="#">runread_pmodel_f</a> , in order).
obs	A nested data.frame of observations, with columns 'sitename' and 'data' (see <a href="#">p_model_validation</a> or <a href="#">p_model_validation_vcmax25</a> to check their structure).
drivers	A nested data.frame of driver data. See <a href="#">p_model_drivers</a> 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 <a href="#">runread_pmodel_f</a> .
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.
startday	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 <sup>th</sup> day of the months is used as date, and if freq = "years" the 1 <sup>st</sup> 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 <sup>th</sup> of February is removed. Defaults to FALSE.

**Value**

A tibble with dates.

---

p_model_drivers	<i>rsofun P-model driver data</i>
-----------------	-----------------------------------

---

**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  $\text{mol m}^{-2} \text{s}^{-1}$ . If all values are NA, it indicates that PPFD should be calculated by the SPLASH model and column ccov.  
**netrad** Net radiation in  $\text{W m}^{-2}$ . WARNING: This is currently ignored as a model forcing.  
**patm** Atmospheric pressure in Pa.  
**snow** Snow in water equivalents  $\text{mm s}^{-1}$ .  
**rain** Rain as precipitation in liquid form in  $\text{mm s}^{-1}$ .  
**tmin** Daily minimum air temperature in  $^{\circ}\text{C}$ .  
**tmax** Daily maximum air temperature in  $^{\circ}\text{C}$ .  
**fapar** Fraction of photosynthetic active radiation (fAPAR), taking values between 0 and 1.  
**co2** Atmospheric  $\text{CO}_2$  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 meteorological forcing data set: WATCH Forcing Data methodology applied to ERA-Interim reanalysis 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

*rsfun 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 meteorological forcing data set: WATCH Forcing Data methodology applied to ERA-Interim reanalysis 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

---

<code>p_model_output</code>	<i>rsofun P-model output data</i>
-----------------------------	-----------------------------------

---

**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.

---

<code>p_model_output_vcmax25</code>	<i>rsofun P-model output data (using vcmax25 drivers)</i>
-------------------------------------	---

---

**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  $\text{gC m}^{-2} \text{d}^{-1}$ ).

**gpp\_unc** The uncertainty of the GPP (in  $\text{gC m}^{-2} \text{d}^{-1}$ ).

### 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 ( $V_{cmax}$ ), normalised to 25° C (in  $\text{mol C m}^{-2} \text{d}^{-1}$ ), aggregated over different plant species in each site.

**vcmax25\_unc** The uncertainty of the  $V_{cmax25}$  (in  $\text{mol C m}^{-2} \text{d}^{-1}$ ), calculated as the standard deviation among  $V_{cmax25}$  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	<i>Run BiomeE</i>
------------------	-------------------

---

**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 <a href="#">run_biomee_f_bysite</a> . Namely sitename, params_simpl, 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)
```

---

<code>runread_pmodel_f</code>	<i>Run P-model</i>
-------------------------------	--------------------

---

**Description**

Runs P-model for multiple sites.

**Usage**

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

**Arguments**

<code>drivers</code>	A nested data frame with one row for each site and columns named according to the arguments of function <a href="#">runread_pmodel_f</a> . Namely <code>sitename</code> , <code>params_siml</code> , <code>site_info</code> and <code>forcing</code> .
<code>par</code>	A named list of free (calibratable) model parameters. <ul style="list-style-type: none"> <li><b>kphio</b> The quantum yield efficiency at optimal temperature <math>\varphi_0</math>, in <math>\text{mol mol}^{-1}</math>. When temperature dependence is used, it corresponds to the multiplicative parameter <i>c</i> (see Details).</li> <li><b>kphio_par_a</b> The shape parameter <i>a</i> of the temperature-dependency of quantum yield efficiency (see Details). To disable the temperature dependence, set <code>kphio_par_a = 0</code>.</li> </ul>

	<b>kphio_par_b</b> The optimal temperature parameter $b$ of the temperature dependent quantum yield efficiency (see Details), in °C.
	<b>soilm_thetastar</b> The threshold parameter $\theta^*$ in the soil moisture stress function (see Details), given in mm. To turn off the soil moisture stress, set <code>soilm_thetastar = 0</code> .
	<b>soilm_betao</b> The intercept parameter $\beta_0$ in the soil moisture stress function (see Details). This is the parameter calibrated in Stocker et al. 2020 GMD.
	<b>beta_unitcostratio</b> The unit cost of carboxylation, corresponding to $\beta = b/a'$ in Eq. 3 of Stocker et al. 2020 GMD.
	<b>rd_to_vcmax</b> Ratio of Rdark (dark respiration) to Vcmax25.
	<b>tau_acclim</b> Acclimation time scale of photosynthesis, in days.
	<b>kc_jmax</b> 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 <code>ncores = 2</code> ).

## 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:

$$\varphi_0(T) = c(1 + a(T - b)^2) \text{ if } 0 < c(1 + a(T - b)^2) < 1,$$

$$\varphi_0(T) = 0 \text{ if } c(1 + a(T - b)^2) \leq 0, \text{ and}$$

$$\varphi_0(T) = 1 \text{ if } c(1 + a(T - b)^2) \geq 1.$$

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:

$$a = -0.0004919819$$

$$b = 32.35294$$

$$c = 0.6910823c_L$$

The soil moisture stress is implemented as

$$\beta(\theta) = \frac{\beta_0 - 1}{\theta^{*2}}(\theta - \theta^*)^2 + 1 \text{ if } 0 \leq \theta \leq \theta^* \text{ and}$$

$$\beta(\theta) = 1 \text{ if } \theta > \theta^*.$$

In Stocker et al. 2020 GMD, the threshold plant-available soil water is set as  $\theta^* = 0.6 * whc$  where `whc` is the site's water holding capacity. Also, the  $\beta$  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_simpl,
  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 <a href="#">biomee_gs_leuning_drivers</a> or <a href="#">biomee_p_model_drivers</a>
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 <a href="#">biomee_gs_leuning_drivers</a> , <a href="#">biomee_p_model_drivers</a> , or <a href="#">biomee_p_model_luluc_drivers</a> .
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.
- day** Day of the year.
- Tk** Air temperature (Kelvin).
- Prcp** Precipitation ( $\text{mm m}^{-2} \text{day}^{-1}$ ).
- SoilWater** Soil water content in root zone ( $\text{kg m}^{-2}$ ).
- Transp** Transpiration ( $\text{mm m}^{-2} \text{day}^{-1}$ ).
- Evap** Evaporation ( $\text{mm m}^{-2} \text{day}^{-1}$ ).
- Runoff** Water runoff ( $\text{mm m}^{-2} \text{day}^{-1}$ ).
- 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 ( $\text{m}^2/\text{m}^2$ ).  
**NPP** Net primary productivity ( $\text{kg C m}^{-2} \text{day}^{-1}$ ).  
**GPP** Gross primary production ( $\text{kg C m}^{-2} \text{day}^{-1}$ ).  
**Rauto** Plant autotrophic respiration ( $\text{kg C m}^{-2} \text{day}^{-1}$ ).  
**Rh** Heterotrophic respiration ( $\text{kg C m}^{-2} \text{day}^{-1}$ ).  
**NSC** Non-structural carbon ( $\text{kg C m}^{-2}$ ).  
**seedC** Biomass of seeds ( $\text{kg C m}^{-2}$ ).  
**leafC** Biomass of leaves ( $\text{kg C m}^{-2}$ ).  
**rootC** Biomass of fine roots ( $\text{kg C m}^{-2}$ ).  
**sapwoodC** Biomass of sapwood ( $\text{kg C m}^{-2}$ ).  
**heartwoodC** Biomass of heartwood ( $\text{kg C m}^{-2}$ ).  
**NSN** Non-structural N pool ( $\text{kg N m}^{-2}$ ).  
**seedN** Nitrogen of seeds ( $\text{kg N m}^{-2}$ ).  
**leafN** Nitrogen of leaves ( $\text{kg N m}^{-2}$ ).  
**rootN** Nitrogen of roots ( $\text{kg N m}^{-2}$ ).  
**sapwoodN** Nitrogen of sapwood ( $\text{kg N m}^{-2}$ ).  
**heartwoodN** Nitrogen of heartwood ( $\text{kg N m}^{-2}$ ).  
**mcrbC** Microbial carbon ( $\text{kg C m}^{-2}$ ).  
**fastSOM** Fast soil carbon pool ( $\text{kg C m}^{-2}$ ).  
**slowSOM** Slow soil carbon pool ( $\text{kg C m}^{-2}$ ).  
**mcrbN** Microbial nitrogen ( $\text{kg N m}^{-2}$ ).  
**fastSoilN** Fast soil nitrogen pool ( $\text{kg N m}^{-2}$ ).  
**slowSoilN** Slow soil nitrogen pool ( $\text{kg N m}^{-2}$ ).  
**mineralN** Mineral nitrogen pool ( $\text{kg N m}^{-2}$ ).  
**N\_uptk** Nitrogen uptake ( $\text{kg N m}^{-2} \text{day}^{-1}$ ).

output\_annual\_tile A data.frame with annual outputs at tile level.

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

**Evap** Evaporation ( $\text{mm m}^{-2} \text{yr}^{-1}$ ).  
**Runoff** Water runoff ( $\text{mm m}^{-2} \text{yr}^{-1}$ ).  
**plantC** Plant biomass ( $\text{kg C m}^{-2}$ ).  
**soilC** Soil carbon ( $\text{kg C m}^{-2}$ ).  
**totC** Total carbon in plant and soil ( $\text{kg C m}^{-2}$ ).  
**plantN** Plant nitrogen ( $\text{kg N m}^{-2}$ ).  
**soilN** Soil nitrogen ( $\text{kg N m}^{-2}$ ).  
**totN** Total nitrogen in plant and soil ( $\text{kg N m}^{-2}$ ).  
**NSC** Nonstructural carbohydrates ( $\text{kg C m}^{-2}$ ).  
**seedC** Seed biomass ( $\text{kg C m}^{-2}$ ).  
**leafC** Leaf biomass ( $\text{kg C m}^{-2}$ ).  
**rootC** Fine root biomass ( $\text{kg C m}^{-2}$ ).  
**sapwoodC** Sapwood biomass ( $\text{kg C m}^{-2}$ ).  
**heartwoodC** Heartwood biomass ( $\text{kg C m}^{-2}$ ).  
**NSN** Nonstructural nitrogen ( $\text{kg N m}^{-2}$ ).  
**seedN** Seed nitrogen ( $\text{kg N m}^{-2}$ ).  
**leafN** Leaf nitrogen ( $\text{kg N m}^{-2}$ ).  
**rootN** Fine root nitrogen ( $\text{kg N m}^{-2}$ ).  
**sapwoodN** Sapwood nitrogen ( $\text{kg N m}^{-2}$ ).  
**heartwoodN** Heartwood nitrogen ( $\text{kg N m}^{-2}$ ).  
**mcrbC** Microbial carbon ( $\text{kg C m}^{-2}$ ).  
**fastSOM** Fast soil carbon pool ( $\text{kg C m}^{-2}$ ).  
**slowSOM** Slow soil carbon pool ( $\text{kg C m}^{-2}$ ).  
**mcrbN** Microbial nitrogen ( $\text{kg N m}^{-2}$ ).  
**fastSoilN** Fast soil nitrogen pool ( $\text{kg N m}^{-2}$ ).  
**slowsoilN** Slow soil nitrogen pool ( $\text{kg N m}^{-2}$ ).  
**mineralN** Mineral nitrogen pool ( $\text{kg N m}^{-2}$ ).  
**N\_fxed** Nitrogen fixation ( $\text{kg N m}^{-2}$ ).  
**N\_uptk** Nitrogen uptake ( $\text{kg N m}^{-2}$ ).  
**N\_yrMin** Annual available nitrogen ( $\text{kg N m}^{-2}$ ).  
**N\_P2S** Annual nitrogen from plants to soil ( $\text{kg N m}^{-2}$ ).  
**N\_loss** Annual nitrogen loss ( $\text{kg N m}^{-2}$ ).  
**totseedC** Total seed carbon ( $\text{kg C m}^{-2}$ ).  
**totseedN** Total seed nitrogen ( $\text{kg N m}^{-2}$ ).  
**Seedling\_C** Total carbon from all compartments but seeds ( $\text{kg C m}^{-2}$ ).  
**Seedling\_N** Total nitrogen from all compartments but seeds ( $\text{kg N m}^{-2}$ ).  
**MaxAge** Age of the oldest tree in the tile (years).  
**MaxVolume** Maximum volume of a tree in the tile ( $\text{m}^3$ ).  
**MaxDBH** Maximum DBH of a tree in the tile (m).  
**NPPL** Growth of a tree, including carbon allocated to leaves ( $\text{kg C m}^{-2} \text{yr}^{-1}$ ).  
**NPPW** Growth of a tree, including carbon allocated to sapwood ( $\text{kg C m}^{-2} \text{yr}^{-1}$ ).  
**n\_deadtrees** Number of trees that died ( $\text{trees m}^{-2} \text{yr}^{-1}$ ).

**c\_deadtrees** Carbon biomass of trees that died ( $\text{kg C m}^{-2} \text{ yr}^{-1}$ ).

**m\_turnover** Continuous biomass turnover ( $\text{kg C m}^{-2} \text{ yr}^{-1}$ ).

**c\_turnover\_time** Carbon turnover rate, calculated as the ratio between plant biomass and NPP ( $\text{yr}^{-1}$ ).

**lu\_fraction** Fraction of BiomeE grid cell that is occupied by this land unit (LU tile) tile (unitless, or  $\text{m}^{-2}$  LU area per  $\text{m}^{-2}$  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 ( $\text{trees ha}^{-1}$ ).

**flayer** Fraction of layer area occupied by this cohort.

**DBH** Tree diameter (cm).

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

**height** Tree height (m).

**age** Age of the cohort (years).

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

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

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

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

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

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

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

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

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

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

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

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

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

**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 ( $\text{kg C tree}^{-1} \text{ yr}^{-1}$ ).

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

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

**N\_uptk** Nitrogen uptake ( $\text{kg N tree}^{-1} \text{ yr}^{-1}$ ).

**N\_fixed** Nitrogen fixation ( $\text{kg N tree}^{-1} \text{ yr}^{-1}$ ).

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

**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 ( $\text{kg N yr}^{-1} \text{ m}^{-2}$ ).

**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 ( $\text{kg C yr}^{-1} \text{ m}^{-2}$ ).

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  $\text{m}^2$  refer to  $\text{m}^2$  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. It contains columns:

**output\_annual\_cell** A data.frame with annual outputs aggregating all tiles present in the simulation cell. Note that quantities per  $\text{m}^2$  refer to  $\text{m}^2$  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  $\text{m}^2$  LU area per  $\text{m}^2$  grid cell area).

**prod\_pool\_1\_C** Carbon in product pool 1 ( $\text{kg C m}^{-2}$  grid cell).

**prod\_pool\_1\_N** Nitrogen in product pool 1 ( $\text{kg N m}^{-2}$  grid cell).

**prod\_pool\_2\_C** Carbon in product pool 2 ( $\text{kg C m}^{-2}$  grid cell).

**prod\_pool\_2\_N** Nitrogen in product pool 2 ( $\text{kg N m}^{-2}$  grid cell).

**Rprod\_0\_C** Carbon loss rate directly from land use change (LUC) ( $\text{kg C m}^{-2}$  grid cell  $\text{yr}^{-1}$ ).

**Rprod\_0\_N** Nitrogen loss rate directly from land use change (LUC) ( $\text{kg C m}^{-2}$  grid cell  $\text{yr}^{-1}$ ).

**Rprod\_1\_C** Carbon loss rate from product pool 1 ( $\text{kg C m}^{-2}$  grid cell  $\text{yr}^{-1}$ ).

**Rprod\_1\_N** Nitrogen loss rate from product pool 1 ( $\text{kg N m}^{-2}$  grid cell  $\text{yr}^{-1}$ ).

**Rprod\_2\_C** Carbon loss rate from product pool 2 ( $\text{kg C m}^{-2}$  grid cell  $\text{yr}^{-1}$ ).

**Rprod\_2\_N** Nitrogen loss rate from product pool 2 ( $\text{kg N m}^{-2}$  grid cell  $\text{yr}^{-1}$ ).

## 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 <a href="#">runread_pmodel_f</a>
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:

$$\varphi_0(T) = c(1 + a(T - b)^2) \text{ if } 0 < c(1 + a(T - b)^2) < 1,$$

$\varphi_0(T) = 0$  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:

$$a = -0.0004919819$$

$$b = 32.35294$$

$$c = 0.6910823c_L$$

The soil moisture stress is implemented as

$$\beta(\theta) = \frac{\beta_0 - 1}{\theta^{*2}}(\theta - \theta^*)^2 + 1 \text{ if } 0 \leq \theta \leq \theta^* \text{ and}$$

$$\beta(\theta) = 1 \text{ if } \theta > \theta^*.$$

In Stocker et al. 2020 GMD, the threshold plant-available soil water is set as  $\theta^* = 0.6 * whc$  where  $whc$  is the site's water holding capacity. Also, the  $\beta$  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  $\text{gC m}^{-2} \text{d}^{-1}$ ).

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

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

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

`vcmax` Maximum rate of RuBisCO carboxylation ( $V_{cmax}$ ) (in  $\text{mol C m}^{-2} \text{s}^{-1}$ ).

`jmax` Maximum rate of electron transport for RuBP regeneration (in  $\text{mol CO}_2 \text{m}^{-2} \text{s}^{-1}$ ).

`vcmax25` Maximum rate of carboxylation ( $V_{cmax}$ ), normalised to  $25^\circ\text{C}$  (in  $\text{mol C m}^{-2} \text{s}^{-1}$ ).

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

`gs_accl` Acclimated stomatal conductance (in  $\text{mol C (mol photons)}^{-1} \text{Pa}^{-1}$ . (Multiply by `ppfd` ( $\text{mol photons m}^{-2} \text{d}^{-1}$ ) 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  $\text{CO}_2$ ,  $c_i:c_a$  (unitless).

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

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

tsoil Soil temperature, in °C.  
 netrad Net radiation, in  $\text{W m}^{-2}$ . 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  $\text{mm d}^{-1}$   
 cleaf C mass of a virtual leaf carbon pool to keep track of isotopic composition, in  $\text{gC m}^{-2}$   
 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	Logical 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 <a href="#">runread_pmodel_f</a>
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 <a href="#">p_model_drivers</a> or <a href="#">p_model_drivers_vcmax25</a>

**Details**

TBC

**Value**

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

vcmax	Maximum rate of RuBisCO carboxylation (Vcmax) (in mol C m <sup>-2</sup> s <sup>-1</sup> ).
jmax	Maximum rate of electron transport for RuBP regeneration (in mol CO <sub>2</sub> m <sup>-2</sup> s <sup>-1</sup> ).
vcmax25	Maximum rate of carboxylation (Vcmax), normalised to 25°C (in mol C m <sup>-2</sup> s <sup>-1</sup> ).
jmax25	Maximum rate of electron transport, normalised to 25°C (in mol C m <sup>-2</sup> s <sup>-1</sup> ).
gs_accl	Acclimated stomatal conductance (in mol C (mol photons) <sup>-1</sup> Pa <sup>-1</sup> . (Multiply by ppfd (mol photons m <sup>-2</sup> d <sup>-1</sup> ) and fapar to express per unit ground area and time.)
chi	Ratio of leaf-internal to ambient CO <sub>2</sub> , 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 <sup>-2</sup> s <sup>-1</sup> . (Multiply by 1/12 (mol C / gC) to convert to mol C m <sup>-2</sup> s <sup>-1</sup> .)
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_mod1 = 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

## \* datasets

- biomee\_gs\_leuning\_drivers, [3](#)
- biomee\_gs\_leuning\_output, [7](#)
- biomee\_p\_model\_drivers, [7](#)
- biomee\_p\_model\_luluc\_drivers, [8](#)
- biomee\_p\_model\_luluc\_output, [8](#)
- biomee\_p\_model\_output, [9](#)
- biomee\_validation, [9](#)
- p\_model\_drivers, [19](#)
- p\_model\_drivers\_vcmax25, [21](#)
- p\_model\_output, [22](#)
- p\_model\_output\_vcmax25, [22](#)
- p\_model\_validation, [23](#)
- p\_model\_validation\_vcmax25, [23](#)

BayesianTools::runMCMC, [11](#)

- biomee\_gs\_leuning\_drivers, [3](#), [7](#), [8](#), [28](#)
- biomee\_gs\_leuning\_output, [7](#), [27](#), [28](#)
- biomee\_p\_model\_drivers, [7](#), [9](#), [28](#)
- biomee\_p\_model\_luluc\_drivers, [8](#), [8](#), [28](#)
- biomee\_p\_model\_luluc\_output, [8](#), [25](#), [28](#)
- biomee\_p\_model\_output, [9](#), [25](#), [27](#), [28](#)
- biomee\_validation, [3](#), [7](#), [9](#)
- build\_luc\_matrix, [10](#)

calib\_sofun, [9](#), [11](#), [23](#)

cost\_likelihood\_biomee, [12](#)

cost\_likelihood\_pmodel, [14](#)

cost\_rmse\_biomee, [15](#)

cost\_rmse\_pmodel, [16](#)

GenSA, [11](#)

init\_dates\_dataframe, [18](#)

p\_model\_drivers, [11](#), [14](#), [17](#), [19](#), [21](#), [22](#), [33](#),  
[36](#)

p\_model\_drivers\_vcmax25, [19](#), [21](#), [22](#), [33](#),  
[36](#)

p\_model\_output, [22](#)

p\_model\_output\_vcmax25, [22](#)

p\_model\_validation, [11](#), [14](#), [17](#), [19](#), [21](#), [23](#)

p\_model\_validation\_vcmax25, [14](#), [17](#), [19](#),  
[21](#), [23](#)

run\_biomee\_f\_bysite, [7–9](#), [24](#), [25](#), [27](#)

run\_pmodel\_f\_bysite, [22](#), [27](#), [33](#)

run\_pmodel\_onestep\_f\_bysite, [35](#)

runMCMC, [11](#)

runread\_biomee\_f, [7–9](#), [24](#)

runread\_pmodel\_f, [14](#), [17](#), [25](#), [25](#), [33](#), [36](#)