Package ‘RPhosFate’

March 22, 2025

Type Package

Title Soil and Chemical Substance Emission and Transport Model

Version 2.0.1

Description An enhanced version of the semi-empirical, spatially distributed emission and transport model PhosFate implemented in 'R' and 'C++'. It is based on the D-infinity, but also supports the D8 flow method. The currently available substances are suspended solids (SS) and particulate phosphorus (PP). A major feature is the allocation of substance loads entering surface waters to their sources of origin, which is a basic requirement for the identification of critical source areas and in consequence a cost-effective implementation of mitigation measures. References: Hepp et al. (2022) <doi:10.1016/j.jenvman.2022.114514>; Hepp and Zessner (2019) <doi:10.3390/w11102161>; Kovacs (2013) <http://hdl.handle.net/20.500.12708/9468>.

License AGPL (>= 3)

URL https://gisler.github.io/RPhosFate/

BugReports https://github.com/gisler/RPhosFate/issues

Language en-GB

Encoding UTF-8

ByteCompile true

Depends R (>= 3.5.0)

Imports checkmate, graphics, methods, Rcpp, stats, terra, tools, utils, yaml

Suggests knitr, rmarkdown, tinytest, whitebox (>= 2.0.0)

LinkingTo Rcpp, RcppArmadillo

RoxygenNote 7.3.2

VignetteBuilder knitr

Collate 'GOFmetrics.R' 'aaa.R' 'RPhosFate.R' 'RcppExports.R' 'assertives.R' 'gettersSetters.R' 'input.R' 'interface.R' 'utils.R' 'zzz.R'

NeedsCompilation yes

Author Gerold Hepp [aut, cre]

Maintainer Gerold Hepp <gisler@hepp.cc>

Repository CRAN

Date/Publication 2025-03-22 10:40:04 UTC

Contents

`autoCalibrate,RPhosFate-method`

One dimensional automatic model calibration

Description

Automatically calibrates the model with the help of a combination of golden section search and successive parabolic interpolation.

Usage

```
## S4 method for signature 'RPhosFate'
autoCalibrate(
  x,
  substance,
  col,
  interval,
  metric,
  tol = min(interval) * 0.1,
  parameter = NULL
)
```

Arguments

`x` An S4 RPhosFate river catchment object.

`substance` A character string specifying the substance to calculate.

`col` A character string specifying the calibration data column with the respective substance river loads.

`interval` A numeric vector specifying the end-points of the interval to be searched.

`metric` A character string specifying the metric to optimise. See calibrationQuality for available metrics.

`tol` A numeric scalar specifying the desired accuracy of the parameter used for optimisation (not the metric).

`parameter` By default, SS are calibrated utilising the overland deposition rate and all other substances are calibrated utilising their respective enrichment ratio. This argument can be used to specify a dedicated parameter utilised for calibration via a character string: "ns_dep_ovl" for overland or "ns_dep_cha" for channel deposition rate.

Value

An S4 RPhosFate river catchment object and side effects in the form of raster files.

See Also

snapGauges, optimize

Examples

```
# temporary demonstration project copy
cv_dir <- demoProject()
# load temporary demonstration project
x <- RPhosFate(
  cv_dir = cv_dir,
  ls_ini = TRUE
)
# presupposed method calls
x <- firstRun(x, "SS")
```

```
x <- snapGauges(x)

x <- autoCalibrate(
  x,
  "SS",
  col = "SS_load",
  interval = c(1e-3, 2e-3),
  metric = "KGE"
)
```

autoCalibrate2,RPhosFate-method

Two dimensional automatic model calibration

Description

Automatically calibrates the model with the help of a general-purpose optimisation function. In contrast to autoCalibrate, this method always utilises the overland and channel deposition rate at the same time and never the respective enrichment ratio for calibration. Beware of local optima and parameters approximately within the convergence tolerance of interval end-points.

Usage

```
## S4 method for signature 'RPhosFate'
autoCalibrate2(
  x,
  substance,
  col,
  metric,
  method = "Nelder-Mead",
  lower = 0,
  upper = 0.1,
  control = list(fnscale = if (metric %in% c("NSE", "mNSE", "KGE")) -1 else 1)
)
```

Arguments

`x` An S4 RPhosFate river catchment object.

`substance` A character string specifying the substance to calculate.

`col` A character string specifying the calibration data column with the respective substance river loads.

`metric` A character string specifying the metric to optimise. See calibrationQuality for available metrics.

`method` A character string specifying the utilised optimisation method. See optim for further information (use autoCalibrate instead of method "Brent").

`lower` A numeric scalar or vector specifying the lower end-point(s) of the interval(s) to be searched.

`upper` A numeric scalar or vector specifying the upper end-point(s) of the interval(s) to be searched.

`control` A `list` of control parameters passed on to `optim`. See `optim` for further information.

Value

An S4 `RPhosFate` river catchment object and side effects in the form of raster files.

See Also

`snapGauges`

Examples

```
# temporary demonstration project copy
cv_dir <- demoProject()
# load temporary demonstration project
x <- RPhosFate(
  cv_dir = cv_dir,
  ls_ini = TRUE
)
# presupposed method calls
x <- firstRun(x, "SS")
x <- snapGauges(x)

x <- autoCalibrate2(
  x,
  "SS",
  col = "SS_load",
  metric = "KGE",
  method = "L-BFGS-B",
  lower = c(1e-3, 0),
  upper = c(2e-3, 2e-3),
  control = list(fnscale = -1, parscale = c(1e-3, 1e-3), factr = 1e12)
)
```

`calibrationQuality,RPhosFate-method`

Calibration quality

Description

Assesses the model's calibration quality with the help of the pairwise complete modelled as well as observed loads and the following metrics:

- *NSE:* Nash-Sutcliffe Efficiency
- *mNSE:* Modified Nash-Sutcliffe Efficiency (`j = 1`)
- *KGE:* Modified Kling-Gupta Efficiency
- *RMSE:* Root Mean Square Error
- *PBIAS:* Percent Bias
- *RSR:* Ratio of the RMSE to the standard deviation of the observations
- *RCV:* Ratio of the coefficients of variation
- *GMRAE:* Geometric Mean Relative Absolute Error
- *MdRAE:* Median Relative Absolute Error

In addition, a scatter plot with the observed river loads on the x- and the modelled river loads on the y-axis is displayed and provides a visual impression of the model performance. Other elements of this plot are an identity line (solid) and plus/minus 30% deviation lines (dashed).

Usage

```
## S4 method for signature 'RPhosFate'
calibrationQuality(x, substance, col)
```

Arguments

`x`	An S4 `RPhosFate` river catchment object.
`substance`	A character string specifying the substance to calculate.
`col`	A character string specifying the calibration data column with the respective substance river loads.

Value

A named numeric vector containing the assessed metrics along with the in-channel retention ratio (one minus sum of *xxt* at catchment outlet(s) divided by sum of *xxt_inp*).

References

Nash, J.E., Sutcliffe, J.V., 1970. River flow forecasting through conceptual models part I – a discussion of principles. Journal of Hydrology 10, 282–290. https://doi.org/10.1016/0022-1694(70)90255-6

Legates, D.R., McCabe Jr., G.J., 1999. Evaluating the use of "goodness-of-fit" measures in hydrologic and hydroclimatic model validation. Water Resources Research 35, 233–241. https://doi.org/10.1029/1998WR900018

Kling, H., Fuchs, M., Paulin, M., 2012. Runoff conditions in the upper Danube basin under an ensemble of climate change scenarios. Journal of Hydrology 424–425, 264–277. https://doi.org/10.1016/j.jhydrol.2012.01.011

Moriasi, D.N., Arnold, J.G., Van Liew, M.W., Bingner, R.L., Harmel, R.D., Veith, T.L., 2007. Model evaluation guidelines for systematic quantification of accuracy in watershed simulations. Transactions of the ASABE 50, 885–900.

See Also

snapGauges, autoCalibrate, autoCalibrate2

Examples

```
# temporary demonstration project copy
cv_dir <- demoProject()
# load temporary demonstration project
x <- RPhosFate(
  cv_dir = cv_dir,
  ls_ini = TRUE
)
# presupposed method calls
x <- firstRun(x, "SS")
x <- snapGauges(x)

calibrationQuality(x, "SS", "SS_load")
```

demoProject *Demonstration project*

Description

Copies a demonstration project to an existing or a temporary directory.

The demonstration project data are a derivative of the

- *Geoland.at* (digital elevation model),
- *AMA* (field data; utilised dataset with the ID 35e36014-ec69-439b-8629-389f52ffaa92 was removed from Offene Daten Österreich),
- *BMLRT* (channel data) and
- *GIP.at* (road data)

data sets, used and licensed under *(CC BY 4.0)* by Gerold Hepp.

While the data represent a real catchment *(HOAL)*, some of them are fictitious, but plausible. These are, among others, R- and C-factors, soil and related data, existence of subsurface drainage at road embankments as well as substance river loads.

Usage

```
demoProject(cs_dir = tempdir(TRUE))
```

Arguments

cs_dir	An optional character string specifying an existing directory.

Value

A character string containing the demonstration project root directory.

See Also

`RPhosFate`, `catchment`

Examples

```
demoProject()
```

`DEMrelatedInput` *DEM related input*

Description

Clips, pre-processes and calculates or determines all input data related to the digital elevation model (DEM) in the broader sense: *acc_inf, cha, dem, dir_inf, rds, slp_inf,* and *wsh.*

Requires the *WhiteboxTools* binary (`whitebox::install_whitebox`) to be installed on your computer.

Usage

```
DEMrelatedInput(
  cv_dir,
  cs_dem,
  cs_cha,
  sp_msk,
  sp_olp,
  sp_sds,
  cs_rds = NULL,
  ns_cha = NULL,
  ns_brn = 50,
  is_adj = 1L,
  is_ths = 1L,
  ls_mD8 = FALSE,
  ls_tmp = FALSE
)
```

Arguments

`cv_dir` A character vector specifying the desired project root directory (first position).

`cs_dem` A character string specifying a path to a potentially large raster digital elevation model.

`cs_cha` A character string specifying a path to a potentially large raster providing channels.

`sp_msk` A `terra::SpatVector` providing a somewhat oversized catchment polygon mask used to clip the potentially large input rasters for further processing.

`sp_olp` A `terra::SpatVector` providing the desired catchment outlet point(s).

`sp_sds` A `terra::SpatVector` providing channel source points.

`cs_rds` An optional character string specifying a path to a potentially large raster providing roads.

`ns_cha` An optional numeric scalar specifying the minimum D8 flow accumulation in number of upslope grid cells determining a channel.

`ns_brn` A numeric scalar specifying the stream burning step size in m.

`is_adj` A numeric scalar specifying how many cells adjacent to channels shall be burnt.

`is_ths` An integer scalar specifying the number of threads to use for processing, where applicable.

`ls_mD8` A logical scalar specifying if D8 flow directions shall be mimicked, i.e. the D-infinity flow directions are rounded to the nearest multiple of 45 degrees. Please note that this treatment is always applied to channel cells independently of this argument.

`ls_tmp` A logical scalar specifying if the temporary files created during computation shall be kept.

Details

This function applies the following (pre-processing) steps to ensure hydrologic consistency of the generated input data:

- Stream burning and orientation of cells adjacent to channel cells approximately into the direction of channel cells (no effect with `ns_brn = 0`).
- Depression breaching.
- Tracing of downslope flowpaths from the provided channel sources.

When roads are provided, they are considered as flow obstacles breaking the continuity of the calculated flow accumulations.

`ns_cha` can be used to enhance the channel network obtained by the tracing of downslope flowpaths from the provided channel sources.

dem represents the breached DEM with reversed stream burning if applicable. The basis for the calculation of the D-infinity slopes provided by *slp_inf,* however, is the original DEM.

Value

A two column numeric `matrix` specifying one or more catchment outlet coordinates and side effects in the form of raster files.

References

Lindsay, J.B., 2016. Efficient hybrid breaching-filling sink removal methods for flow path enforcement in digital elevation models. Hydrological Processes 30, 846–857. https://doi.org/10.1002/hyp.10648

Tarboton, D.G., 1997. A new method for the determination of flow directions and upslope areas in grid digital elevation models. Water Resour. Res. 33, 309–319. https://doi.org/10.1029/96WR03137

See Also

RPhosFate, catchment

Examples

```
## Not run:
# obtain temporary project root directory
cv_dir <- normalizePath(
  tempfile("cmt"),
  winslash = .Platform$file.sep,
  mustWork = FALSE
)
# obtain directory holding "large" rasters and other required data sets
cs_dir_lrg <- system.file("tinytest", "largeData", package = "RPhosFate")

nm_olc <- DEMrelatedInput(
  cv_dir = cv_dir,
  cs_dem = file.path(cs_dir_lrg, "dem_lrg.tif"),
  cs_cha = file.path(cs_dir_lrg, "cha_lrg.tif"),
  sp_msk = terra::vect(file.path(cs_dir_lrg, "msk.shp")),
  sp_olp = terra::vect(file.path(cs_dir_lrg, "olp.shp")),
  sp_sds = terra::vect(file.path(cs_dir_lrg, "sds.shp")),
  cs_rds = file.path(cs_dir_lrg, "rds_lrg.tif"),
  ls_tmp = TRUE
)
## End(Not run)
```

emission,RPhosFate-method

Emission

Description

Calculates and writes substance emissions to disk.

Usage

```
## S4 method for signature 'RPhosFate'
emission(x, substance = "PP")
```

Arguments

x	An S4 RPhosFate river catchment object.
substance	A character string specifying the substance to calculate.

Value

An S4 RPhosFate river catchment object and side effects in the form of raster files.

See Also

firstRun, subsequentRun

Examples

```
# temporary demonstration project copy
cv_dir <- demoProject()
# load temporary demonstration project
x <- RPhosFate(
  cv_dir = cv_dir,
  ls_ini = TRUE
)
# presupposed method calls
x <- erosionPrerequisites(x)
x <- erosion(x)

x <- emission(x, "PP")
```

erosion,RPhosFate-method

Erosion

Description

Calculates and writes (R)USLE erosion to disk.

Usage

```
## S4 method for signature 'RPhosFate'
erosion(x)
```

Arguments

x	An S4 RPhosFate river catchment object.

Value

An S4 RPhosFate river catchment object and side effects in the form of raster files.

References

Renard, K.G., Foster, G.R., Weesies, G.A., McCool, D.K., Yoder, D.C., 1997. Predicting soil erosion by water: a guide to conservation planning with the Revised Universal Soil Loss Equation (RUSLE), Agriculture Handbook. U.S. Government Printing Office, Washington, DC.

Wischmeier, W.H., Smith, D.D., 1978. Predicting rainfall erosion losses. A guide to conservation planning, Agriculture Handbook. U.S. Government Printing Office, Washington, DC.

See Also

firstRun, subsequentRun

Examples

```
# temporary demonstration project copy
cv_dir <- demoProject()
# load temporary demonstration project
x <- RPhosFate(
  cv_dir = cv_dir,
  ls_ini = TRUE
)
# presupposed method call
x <- erosionPrerequisites(x)

x <- erosion(x)
```

erosionPrerequisites,RPhosFate-method

Erosion prerequisites

Description

Calculates and writes capped slopes, L- and RUSLE S-factors (equations for summer conditions and slopes $\geq$ 15 ft) to disk.

Usage

```
## S4 method for signature 'RPhosFate'
erosionPrerequisites(x)
```

Arguments

x	An S4 RPhosFate river catchment object.

Value

An S4 RPhosFate river catchment object and side effects in the form of raster files.

References

Desmet, P.J.J., Govers, G., 1996. A GIS procedure for automatically calculating the USLE LS factor on topographically complex landscape units. Journal of Soil and Water Conservation 51, 427–433.

Renard, K.G., Foster, G.R., Weesies, G.A., McCool, D.K., Yoder, D.C., 1997. Predicting soil erosion by water: a guide to conservation planning with the Revised Universal Soil Loss Equation (RUSLE), Agriculture Handbook. U.S. Government Printing Office, Washington, DC.

See Also

firstRun, subsequentRun

Examples

```
# temporary demonstration project copy
cv_dir <- demoProject()
# load temporary demonstration project
x <- RPhosFate(
  cv_dir = cv_dir,
  ls_ini = TRUE
)

x <- erosionPrerequisites(x)
```

---

firstRun,RPhosFate-method

First run

Description

Calls erosionPrerequisites, erosion, emission, transportPrerequisites and transport in the mentioned order. While transport is called for the specified substance only, emission is called for all substances whose top soil concentrations have been provided.

Usage

```
## S4 method for signature 'RPhosFate'
firstRun(x, substance = "PP")
```

Arguments

x	An S4 RPhosFate river catchment object.
substance	A character string specifying the substance to calculate.

Value

An S4 RPhosFate river catchment object and side effects in the form of raster files.

See Also

subsequentRun

Examples

```
# temporary demonstration project copy
cv_dir <- demoProject()
# load temporary demonstration project
x <- RPhosFate(
  cv_dir = cv_dir,
  ls_ini = TRUE
)

x <- firstRun(x, "SS")
```

getLayer,RPhosFate-method

Get layer

Description

Obtains a project raster layer for further analysis.

Usage

```
## S4 method for signature 'RPhosFate'
getLayer(x, i, j = NULL)

## S4 method for signature 'RPhosFate,ANY,ANY'
x[i, j]
```

Arguments

x	An S4 RPhosFate river catchment object.
i	A character string specifying a layer name. Substance related layers whose names start with *xx* are treated differently. They have to be queried by their name (not filename), for example, "xxc" in combination with "PP" in argument j queries the particulate phosphorus concentrations in top soils. See subdirectory sections for further information.
j	A character string specifying a substance if applicable.

Value

A terra::SpatRaster object.

Input subdirectory

This directory holds all possible user input raster data (flow obstacles like roads must be considered during the generation of the flow accumulation layer and must also be cut out from it in order to be properly respected):

- *acc_inf:* D-infinity flow accumulations in number of upslope grid cells required for everything.
- *CFa:* (R)USLE C-factors required for `erosion`.
- *cha:* Channel cells required for everything (`1`: channel cell, `NA`: no channel cell).
- *clc:* Clay contents of top soils in % required for substance `emission`s.
- *dem:* Digital elevation model in m a.s.l. (optional).
- *dir_inf:* D-infinity flow directions in azimuth degrees measured from north (0 to 360 clockwise) required for `transportPrerequisites` and substance `transport`.
- *fid:* Field IDs (optional).
- *KFa:* (R)USLE K-factors required for `erosion`.
- *lue:* Land use classes (optional).
- *man:* Manning's roughness coefficients required for substance `transport`.
- *xxc:* Substance contents of top soils in mg/kg required for substance `emission`s, for example, *ppc* for PP top soil contents.
- *rds:* Road cells required for `transportPrerequisites` (`0`: road cell without subsurface drainage, `1`: road cell with subsurface drainage, `NA`: no road cell).
- *RFa:* (R)USLE R-factors required for `erosion`.
- *slp_inf:* D-infinity slopes in % required for everything.
- *wsh:* Watershed (optional).

Intermediate subdirectory

This directory holds intermediate calculations:

- *inl:* Cells representing inlets at roads (storm drains).
- *LFa:* L-factors.
- *rip:* Cells representing the riparian zones within channel cells.
- *SFa:* RUSLE S-factors.
- *slp_cap:* Capped slopes in %.

Result subdirectory

This directory holds the model results:

- *ero:* Erosion in t/cell/yr.
- *xxe:* Substance emissions in kg/cell/yr, for example, *ppe* for PP emissions.
- *xxr:* Substance retentions in t/cell/yr (SS) or kg/cell/yr, for example, *ppr* for PP retentions.
- *xxt:* Substance transports in t/cell/yr (SS) or kg/cell/yr, for example, *ppt* for PP transports.

- *xxt_cld:* Substance cell loads in t/cell/yr (SS) or kg/cell/yr, for example, *ppt_cld* for PP cell loads.
- *xxt_ctf:* Substance cell transfers in t/cell/yr (SS) or kg/cell/yr, for example, *ppt_ctf* for PP transfers.
- *xxt_inp:* Substance inputs into surface waters in t/cell/yr (SS) or kg/cell/yr, for example, *ppt_inp* for PP inputs into surface waters.
- *xxt_out:* Substance outlet loads of subsurface drainages in t/cell/yr (SS) or kg/cell/yr, for example, *ppt_out* for PP outlet loads.

Examples

```
# temporary demonstration project copy
cv_dir <- demoProject()
# load temporary demonstration project
x <- RPhosFate(
  cv_dir = cv_dir,
  ls_ini = TRUE
)
# presupposed method call
x <- firstRun(x, "SS")

getLayer(x, "dir_inf")
getLayer(x, "xxt", "SS")
getLayer(x, "xxe", "PP")
```

getParameter,RPhosFate-method

Get parameter(s)

Description

Obtains a single model parameter or all model parameters at once.

Usage

```
## S4 method for signature 'RPhosFate'
getParameter(x, parameter = NULL)
```

Arguments

x	An S4 RPhosFate river catchment object.
parameter	A character string specifying a parameter name or NULL for a list of all parameters. See model parameter arguments section for further information.

Value

Depends on the queried parameter or a list in case of all parameters. See model parameter arguments section for further information.

Model parameter arguments

- ns_slp_min: A numeric scalar specifying the minimum bounding slope in % (defaults to 1.0).
- ns_slp_max: A numeric scalar specifying the maximum bounding slope in % (defaults to 999.0).
- ns_rhy_a: A numeric scalar specifying a network constant depending on the discharge frequency needed for the calculation of the hydraulic radius, which in turn is a prerequisite for substance transport (defaults to 0.09 representing a discharge frequency of approximately six years).
- ns_rhy_b: A numeric scalar specifying a geometry scaling exponent depending on the discharge frequency needed for the calculation of the hydraulic radius, which in turn is a prerequisite for substance transport (defaults to 0.50 representing a discharge frequency of approximately six years).
- ns_cha_rto: A numeric scalar specifying the ratio of the channel to the cell width determining the widths of the riparian zones required for substance transport (defaults to 0.5).
- ns_man_rip: A numeric scalar specifying Manning's roughness coefficient of the riparian zones within channel cells required for substance transport (defaults to 0.32).
- ns_man_cha: A numeric scalar specifying Manning's roughness coefficient of the channel within channel cells required for substance transport (defaults to 0.04).
- ns_dep_ovl: A numeric scalar specifying the overland deposition rate per second required for substance transport (calibration parameter; no default).
- ns_dep_cha: A numeric scalar specifying the channel deposition rate per second required for substance transport (calibration parameter; no default).
- nv_tfc_inl: A named numeric vector specifying the inlet transfer coefficients required for substance transport, for example, c(SS = 0.6, PP = 0.6) (no default).
- nv_enr_rto A named numeric vector specifying the substance enrichment ratios required for substance except SS transport, for example, c(PP = 2.0) (calibration parameter; no default).
- nm_olc: A two column numeric matrix specifying one or more catchment outlet coordinates required for the in-channel retention ratio of calibrationQuality (no default).
- df_cdt: A data.frame with calibration data, which must have at least the following three columns and one or more columns with substance river loads in t/yr (no default):
  - *ID:* ID(s) of the gauge(s)
  - *x:* x-coordinate(s) of the gauge(s)
  - *y:* y-coordinate(s) of the gauge(s)

See Also

setParameter

Examples

```
# temporary demonstration project copy
cv_dir <- demoProject()
# load temporary demonstration project
```

```
x <- RPhosFate(
  cv_dir = cv_dir,
  ls_ini = TRUE
)

getParameter(x)
getParameter(x, "ns_dep_ovl")
```

img2tif *Convert* ERDAS IMAGINE *to* GeoTIFF *raster files*

Description

Converts all *ERDAS IMAGINE* raster files in a directory and its subdirectories into *GeoTIFF* raster files.

Usage

```
img2tif(cs_dir, cs_crs = NULL)
```

Arguments

cs_dir	A character string specifying an existing directory.
cs_crs	An optional character string used to set the coordinate reference system of all output raster files. See terra::crs for further information.

Value

A character vector containing the paths to the processed *ERDAS IMAGINE* raster files.

RPhosFate *Initialise project*

Description

Initialises a project from scratch or loads the state of an existing one utilising *GeoTIFF* (*.tif) raster files from, by convention, the following three project root subdirectories:

- *Input*
- *Intermediate*
- *Result*

See subdirectory sections for further information.

catchment is an alias for RPhosFate.

Usage

```
RPhosFate(...)

catchment(...)
```

Arguments

`...`	Arguments used to initialise the project. See argument sections for further information.

Value

An S4 `RPhosFate` river catchment object.

Input subdirectory

This directory holds all possible user input raster data (flow obstacles like roads must be considered during the generation of the flow accumulation layer and must also be cut out from it in order to be properly respected):

- *acc_inf:* D-infinity flow accumulations in number of upslope grid cells required for everything.
- *CFa:* (R)USLE C-factors required for `erosion`.
- *cha:* Channel cells required for everything (`1`: channel cell, `NA`: no channel cell).
- *clc:* Clay contents of top soils in % required for substance `emission`s.
- *dem:* Digital elevation model in m a.s.l. (optional).
- *dir_inf:* D-infinity flow directions in azimuth degrees measured from north (0 to 360 clockwise) required for `transportPrerequisites` and substance `transport`.
- *fid:* Field IDs (optional).
- *KFa:* (R)USLE K-factors required for `erosion`.
- *lue:* Land use classes (optional).
- *man:* Manning's roughness coefficients required for substance `transport`.
- *xxc:* Substance contents of top soils in mg/kg required for substance `emission`s, for example, *ppc* for PP top soil contents.
- *rds:* Road cells required for `transportPrerequisites` (`0`: road cell without subsurface drainage, `1`: road cell with subsurface drainage, `NA`: no road cell).
- *RFa:* (R)USLE R-factors required for `erosion`.
- *slp_inf:* D-infinity slopes in % required for everything.
- *wsh:* Watershed (optional).

Intermediate subdirectory

This directory holds intermediate calculations:

- *inl:* Cells representing inlets at roads (storm drains).
- *LFa:* L-factors.
- *rip:* Cells representing the riparian zones within channel cells.
- *SFa:* RUSLE S-factors.
- *slp_cap:* Capped slopes in %.

Result subdirectory

This directory holds the model results:

- *ero:* Erosion in t/cell/yr.
- *xxe:* Substance emissions in kg/cell/yr, for example, *ppe* for PP emissions.
- *xxr:* Substance retentions in t/cell/yr (SS) or kg/cell/yr, for example, *ppr* for PP retentions.
- *xxt:* Substance transports in t/cell/yr (SS) or kg/cell/yr, for example, *ppt* for PP transports.
- *xxt_cld:* Substance cell loads in t/cell/yr (SS) or kg/cell/yr, for example, *ppt_cld* for PP cell loads.
- *xxt_ctf:* Substance cell transfers in t/cell/yr (SS) or kg/cell/yr, for example, *ppt_ctf* for PP transfers.
- *xxt_inp:* Substance inputs into surface waters in t/cell/yr (SS) or kg/cell/yr, for example, *ppt_inp* for PP inputs into surface waters.
- *xxt_out:* Substance outlet loads of subsurface drainages in t/cell/yr (SS) or kg/cell/yr, for example, *ppt_out* for PP outlet loads.

Data management and processing arguments

- `cv_dir`: A character vector specifying the project root (first position) and optionally the Monte Carlo input data directory (second position).
- `ls_ini`: A logical scalar specifying if the state of an existing project shall be loaded from disk (defaults to `FALSE`). Parameters or substance parameter values specified via the `...` argument take precedence over loaded ones.
- `is_ths`: An integer scalar holding the number of threads to use for processing (defaults to 1).
- `is_MCi`: An integer scalar specifying the current Monte Carlo iteration if applicable (defaults to `integer()`, which means Monte Carlo simulation mode is disabled).
- `cv_MCl`: A character vector specifying the names of the layers, which shall be written to disk with the associated Monte Carlo iteration in their filenames upon calling the appropriate methods (defaults to `"xxt"`; no effect in case Monte Carlo simulation mode is disabled).

Model parameter arguments

- `ns_slp_min`: A numeric scalar specifying the minimum bounding slope in % (defaults to `1.0`).
- `ns_slp_max`: A numeric scalar specifying the maximum bounding slope in % (defaults to `999.0`).
- `ns_rhy_a`: A numeric scalar specifying a network constant depending on the discharge frequency needed for the calculation of the hydraulic radius, which in turn is a prerequisite for substance transport (defaults to `0.09` representing a discharge frequency of approximately six years).
- `ns_rhy_b`: A numeric scalar specifying a geometry scaling exponent depending on the discharge frequency needed for the calculation of the hydraulic radius, which in turn is a prerequisite for substance transport (defaults to `0.50` representing a discharge frequency of approximately six years).
- `ns_cha_rto`: A numeric scalar specifying the ratio of the channel to the cell width determining the widths of the riparian zones required for substance `transport` (defaults to `0.5`).
- `ns_man_rip`: A numeric scalar specifying Manning's roughness coefficient of the riparian zones within channel cells required for substance `transport` (defaults to `0.32`).
- `ns_man_cha`: A numeric scalar specifying Manning's roughness coefficient of the channel within channel cells required for substance `transport` (defaults to `0.04`).
- `ns_dep_ovl`: A numeric scalar specifying the overland deposition rate per second required for substance `transport` (calibration parameter; no default).
- `ns_dep_cha`: A numeric scalar specifying the channel deposition rate per second required for substance `transport` (calibration parameter; no default).
- `nv_tfc_inl`: A named numeric vector specifying the inlet transfer coefficients required for substance `transport`, for example, `c(SS = 0.6, PP = 0.6)` (no default).
- `nv_enr_rto` A named numeric vector specifying the substance enrichment ratios required for substance except SS `transport`, for example, `c(PP = 2.0)` (calibration parameter; no default).
- `nm_olc`: A two column numeric `matrix` specifying one or more catchment outlet coordinates required for the in-channel retention ratio of `calibrationQuality` (no default).
- `df_cdt`: A `data.frame` with calibration data, which must have at least the following three columns and one or more columns with substance river loads in t/yr (no default):
  - *ID:* ID(s) of the gauge(s)
  - *x:* x-coordinate(s) of the gauge(s)
  - *y:* y-coordinate(s) of the gauge(s)

Monte Carlo simulation mode

This mode can make use of repeated random samples, i.e. raster data, of distributions of about all input data. The filenames of the Monte Carlo input raster data must contain the specified iteration, for example, *CFa12.tif* for the twelfth iteration of the C-factors input data, and can reside in a separate directory. In case no Monte Carlo raster file is found for a certain layer in the designated directory, the respective project root subdirectory is searched for one and finally the "normal" project input raster data is utilised.

See Also

saveState, demoProject

Examples

```
# temporary demonstration project copy
cv_dir <- demoProject()

# initialise project from scratch
x <- RPhosFate(
  cv_dir = cv_dir,
  ns_dep_ovl = 25e-4,
  ns_dep_cha = 0.0,
  nv_tfc_inl = c(SS = 0.6, PP = 0.6),
  nv_enr_rto = c(PP = 2.0),
  nm_olc = matrix(c(4704255, 2795195), ncol = 2L),
  df_cdt = read.table(
    file.path(cv_dir, "cdt.txt"),
    header = TRUE,
    stringsAsFactors = FALSE
  )
)

# load state of existing project in Monte Carlo simulation mode
x <- RPhosFate(
  cv_dir = c(
    cv_dir,
    system.file("tinytest", "testProject", package = "RPhosFate")
  ),
  ls_ini = TRUE,
  is_MCi = 1L,
  cv_MCl = c("xxt", "xxt_cld")
)
```

RPhosFate-class *RPhosFate class*

Description

An S4 object representing a river catchment.

Slots

- cv_dir A character vector holding the project root (first position) and optionally the Monte Carlo input data directory (second position).
- ls_ini A logical scalar specifying if the state of an existing project was loaded from disk.
- is_ths An integer scalar holding the number of threads to use for processing, where applicable.

is_MCi An integer scalar holding the current Monte Carlo iteration if applicable.

cv_MCl A character vector holding the names of the layers, which shall be written to disk with the associated Monte Carlo iteration in their filenames upon calling the appropriate methods.

parameters An S4 object holding the model parameters.

topo An S4 object holding the raster layers related to topography in the broader sense.

erosion An S4 object holding the raster layers related to erosion.

transport An S4 object holding raster layers required for modelling transport.

substances An S4 object holding the substance raster layer containers.

helpers An S4 object holding helper data.

See Also

RPhosFate, catchment

saveState,RPhosFate-method

Save state

Description

Saves parameters *(parameters.yaml)* to disk.

Usage

```
## S4 method for signature 'RPhosFate'
saveState(x)
```

Arguments

x	An S4 RPhosFate river catchment object.

Value

NULL invisibly and side effects in the form of files.

See Also

RPhosFate, catchment

Examples

```
# temporary demonstration project copy
cv_dir <- demoProject()
# load temporary demonstration project
x <- RPhosFate(
  cv_dir = cv_dir,
  ls_ini = TRUE
)

saveState(x)
```

setParameter,RPhosFate-method

Set parameter(s)

Description

Sets one or more model parameters or substance parameter values.

Usage

```
## S4 method for signature 'RPhosFate'
setParameter(x, ...)
```

Arguments

x	An S4 RPhosFate river catchment object.
...	Names and values of the parameters to set. See model parameter arguments section for further information.

Value

An S4 RPhosFate river catchment object.

Model parameter arguments

- ns_slp_min: A numeric scalar specifying the minimum bounding slope in % (defaults to 1.0).
- ns_slp_max: A numeric scalar specifying the maximum bounding slope in % (defaults to 999.0).
- ns_rhy_a: A numeric scalar specifying a network constant depending on the discharge frequency needed for the calculation of the hydraulic radius, which in turn is a prerequisite for substance transport (defaults to 0.09 representing a discharge frequency of approximately six years).

- ns_rhy_b: A numeric scalar specifying a geometry scaling exponent depending on the discharge frequency needed for the calculation of the hydraulic radius, which in turn is a prerequisite for substance transport (defaults to 0.50 representing a discharge frequency of approximately six years).
- ns_cha_rto: A numeric scalar specifying the ratio of the channel to the cell width determining the widths of the riparian zones required for substance transport (defaults to 0.5).
- ns_man_rip: A numeric scalar specifying Manning's roughness coefficient of the riparian zones within channel cells required for substance transport (defaults to 0.32).
- ns_man_cha: A numeric scalar specifying Manning's roughness coefficient of the channel within channel cells required for substance transport (defaults to 0.04).
- ns_dep_ovl: A numeric scalar specifying the overland deposition rate per second required for substance transport (calibration parameter; no default).
- ns_dep_cha: A numeric scalar specifying the channel deposition rate per second required for substance transport (calibration parameter; no default).
- nv_tfc_inl: A named numeric vector specifying the inlet transfer coefficients required for substance transport, for example, c(SS = 0.6, PP = 0.6) (no default).
- nv_enr_rto A named numeric vector specifying the substance enrichment ratios required for substance except SS transport, for example, c(PP = 2.0) (calibration parameter; no default).
- nm_olc: A two column numeric matrix specifying one or more catchment outlet coordinates required for the in-channel retention ratio of calibrationQuality (no default).
- df_cdt: A data.frame with calibration data, which must have at least the following three columns and one or more columns with substance river loads in t/yr (no default):
  - *ID:* ID(s) of the gauge(s)
  - *x:* x-coordinate(s) of the gauge(s)
  - *y:* y-coordinate(s) of the gauge(s)

See Also

getParameter

Examples

```
# temporary demonstration project copy
cv_dir <- demoProject()
# load temporary demonstration project
x <- RPhosFate(
  cv_dir = cv_dir,
  ls_ini = TRUE
)

x <- setParameter(x, ns_dep_ovl = 15e-4)
x <- setParameter(
  x,
  nv_tfc_inl = c(SS = 0.6, PP = 0.6),
  nv_enr_rto = c(PP = 1.4)
)
```

snapGauges,RPhosFate-method

Snap gauge(s)

Description

Snaps the coordinates of the provided calibration gauges to the respective midpoint of the nearest channel cell.

Usage

```
## S4 method for signature 'RPhosFate'
snapGauges(x)
```

Arguments

x	An S4 RPhosFate river catchment object.

Value

An S4 RPhosFate river catchment object.

See Also

calibrationQuality, autoCalibrate, autoCalibrate2

Examples

```
# temporary demonstration project copy
cv_dir <- demoProject()
# load temporary demonstration project
x <- RPhosFate(
  cv_dir = cv_dir,
  ls_ini = TRUE
)

x <- snapGauges(x)
```

subsequentRun,RPhosFate-method

Subsequent run

Description

Calls transport for the specified substance and optionally erosionPrerequisites, erosion, emission and transportPrerequisites beforehand.

Usage

```
## S4 method for signature 'RPhosFate'
subsequentRun(
  x,
  substance = "PP",
  erosionPrerequisites = FALSE,
  erosion = FALSE,
  emission = FALSE,
  transportPrerequisites = FALSE
)
```

Arguments

x
: An S4 RPhosFate river catchment object.

substance
: A character string specifying the substance to calculate.

erosionPrerequisites
: A logical scalar specifying if erosionPrerequisites is called.

erosion
: A logical scalar specifying if erosion is called.

emission
: A logical scalar specifying if emission is called. It is never called with substance = "SS" though.

transportPrerequisites
: A logical scalar specifying if transportPrerequisites is called.

Value

An S4 RPhosFate river catchment object and side effects in the form of raster files.

See Also

firstRun

Examples

```
# temporary demonstration project copy
cv_dir <- demoProject()
# load temporary demonstration project
x <- RPhosFate(
  cv_dir = cv_dir,
  ls_ini = TRUE
)
# presupposed method call
x <- firstRun(x, "SS")

x <- subsequentRun(x, "PP")
```

---

transport,RPhosFate-method

Transport

Description

Calculates and writes substance retentions, transports and cell loads as well as transfers to disk.

Usage

```
## S4 method for signature 'RPhosFate'
transport(x, substance = "PP")
```

Arguments

x	An S4 RPhosFate river catchment object.
substance	A character string specifying the substance to calculate.

Value

An S4 RPhosFate river catchment object and side effects in the form of raster files.

References

Engman, E.T., 1986. Roughness coefficients for routing surface runoff. Journal of Irrigation and Drainage Engineering 112, 39–53.

Molnár, P., Ramírez, J.A., 1998. Energy dissipation theories and optimal channel characteristics of river networks. Water Resources Research 34, 1809–1818. https://doi.org/10.1029/98WR00983

See Also

firstRun, subsequentRun

Examples

```
# temporary demonstration project copy
cv_dir <- demoProject()
# load temporary demonstration project
x <- RPhosFate(
  cv_dir = cv_dir,
  ls_ini = TRUE
)
# presupposed method calls
x <- erosionPrerequisites(x)
x <- erosion(x)
x <- emission(x, "PP")
x <- transportPrerequisites(x)

x <- transport(x, "PP")
```

---

transportPrerequisites,RPhosFate-method

Transport prerequisites

Description

Determines cells representing inlets as well as riparian zones before writing them to disk.

Usage

```
## S4 method for signature 'RPhosFate'
transportPrerequisites(x)
```

Arguments

x	An S4 RPhosFate river catchment object.

Value

An S4 RPhosFate river catchment object and side effects in the form of raster files.

See Also

firstRun, subsequentRun

Examples

```
# temporary demonstration project copy
cv_dir <- demoProject()
# load temporary demonstration project
x <- RPhosFate(
  cv_dir = cv_dir,
  ls_ini = TRUE
)

x <- transportPrerequisites(x)
```

Index