Package ‘RAC’

October 12, 2022

Version 1.5

Date 2018-04-24

Title R Package for Aqua Culture

Maintainer Baldan D. `<damiano.baldan91@gmail.com>`

Depends R (>= 3.1.0), matrixStats, rstudioapi, plotrix

Suggests MASS, knitr, rmarkdown

Imports maps, maptools, ncdf4, rgdal, rgeos, sp, stats, utils, raster

Description Solves the individual bioenergetic balance for different aquaculture sea fish (Sea Bream and Sea Bass; Brigolin et al., 2014 <doi:10.3354/aei00093>) and shellfish (Mussel and Clam; Brigolin et al., 2009 <doi:10.1016/j.ecss.2009.01.029>; Solidoro et al., 2000 <doi:10.3354/meps199137>). Allows for spatialized model runs and population simulations.

RoxygenNote 5.0.1

VignetteBuilder knitr

NeedsCompilation no

Author Baldan D. [aut, cre],
Palazzo D. [ctb],
Porporato E.M.D [ctb],
Brigolin D. [ctb]

Repository CRAN

Date/Publication 2018-05-02 07:38:09 UTC

R topics documented:

`Bass_ind_dataloader` *Function that loads forcings data for Seabass individual bioenergetic model and performs the interpolation*

Description

Function that loads forcings data for Seabass individual bioenergetic model and performs the interpolation

Usage

```
Bass_ind_dataloader(userpath)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees] and feeding rate [g/individual x d]

`Bass_ind_equations` *Seabass bioenergetic individual model differential equations*

Description

Seabass bioenergetic individual model differential equations

Usage

```
Bass_ind_equations(Param, Temp, G, Food, weight)
```

Arguments

`Param`	vector containing all metabolic parameters
`Temp`	water temperature forcing at time t
`G`	food entering the cage at time t
`Food`	food characterization (Proteins, Lipids, Carbohydrates)
`weight`	individual weight at time t

Value

model output at time t

`Bass_ind_main` *Seabass bioenergetic individual model*

Description

Solves the bioenergetic balance for Seabass

Usage

```
Bass_ind_main(userpath, forcings)
```

Arguments

`userpath`	the path where forcing are located
`forcings`	a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees] and feeding rate [g/individual x d]

Value

A list containing model outputs: weight, excreted quantities and quantities to waste, actual and potential ingestion, temperature limitation functions and metabolic rates

---

`Bass_ind_post` *Seabass bioenergetic individual model postprocessor*

Description

Seabass bioenergetic individual model postprocessor

Usage

```
Bass_ind_post(userpath, output, times, Dates, CS)
```

Arguments

`userpath`	the path where the working folder is located
`output`	output list containing the output of the RK solver
`times`	the vector containing informations on integration extremes
`Dates`	the vector containing the date
`CS`	the commercial size of Seabass

Value

a list containing the fish weight, proteines, lipids and carbohydrates wasted or produced with excretions, potential and actual ingestion rates, temperature limitation functions and metabolic rates

---

`Bass_ind_pre` *Seabass bioenergetic individual model preprocessor*

Description

Seabass bioenergetic individual model preprocessor

Usage

```
Bass_ind_pre(userpath, forcings)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located
`forcings`	a list containing model forcings

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees] and feeding rate [g/individual x d]

`Bass_ind_RKsolver` *Solves the Seabass bioenergetic balance with a 4th order Runge Kutta method*

Description

Solves the Seabass bioenergetic balance with a 4th order Runge Kutta method

Usage

```
Bass_ind_RKsolver(Param, Temperature, G, Food, IC, times)
```

Arguments

`Param`	vector containing all metabolic parameters
`Temperature`	water temperature forcing time series
`G`	food entering the cage time series
`Food`	food characterization (Proteins, Lipids, Carbohydrates)
`IC`	initial conditions
`times`	vector containing integration extremes and integration timestep

Value

a list containing the fish weight, proteines, lipids and carbohydrates wasted or produced with excretions, potential and actual ingestion rates, temperature limitation functions and metabolic rates

---

`Bass_ind_skeleton` *Creates the folders structure for Seabass individual bioenergetic model*

Description

Creates the folders structure for Seabass individual bioenergetic model

Usage

```
Bass_ind_skeleton(userpath)
```

Arguments

`userpath`	the path where forcing are located

`Bass_pop_dataloader` *Function that loads forcings data for Seabass population model and performs the interpolation*

Description

Function that loads forcings data for Seabass population model and performs the interpolation

Usage

```
Bass_pop_dataloader(userpath)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees] and feeding rate [g/individual x d]

`Bass_pop_equations` *Seabass bioenergetic population model differential equations*

Description

Seabass bioenergetic population model differential equations

Usage

```
Bass_pop_equations(Param, N, Temp, G, Food, weight)
```

Arguments

`Param`	vector containing all metabolic parameters
`N`	the number of individuals at time t
`Temp`	water temperature forcing at time t
`G`	food entering the cage at time t
`Food`	food characterization (Proteins, Lipids, Carbohydrates)
`weight`	individual weight at time t

Value

model output at time t

`Bass_pop_loop` *Function that runs the Monte Carlo simulation for the Seabass population model*

Description

Function that runs the Monte Carlo simulation for the Seabass population model

Usage

```
Bass_pop_loop(Param, Tint, Gint, Food, IC, times, N, userpath)
```

Arguments

`Param`	a vector containing model parameters
`Tint`	the interpolated water temperature time series
`Gint`	the interpolated feeding rate time series
`Food`	the food characterization
`IC`	initial condition
`times`	integration extremes and integration timestep
`N`	time series with number of individuals
`userpath`	the path where the working folder is located

Value

a list with RK solver outputs

`Bass_pop_main` *Seabass bioenergetic population model*

Description

Seabass bioenergetic population model

Usage

```
Bass_pop_main(userpath, forcings)
```

Arguments

`userpath`	the path where the working folder is located
`forcings`	a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees] and feeding rate [g/individual x d]

Value

A list containing model outputs: weight, excreted quantities and quantities to waste, actual and potential ingestion, temperature limitation functions and metabolic rates

---

`Bass_pop_post` *Postprocess the Seabass population bioenergetic model results*

Description

Postprocess the Seabass population bioenergetic model results

Usage

```
Bass_pop_post(userpath, output, times, Dates, N, CS)
```

Arguments

`userpath`	the path where the working folder is located
`output`	list containing the output of the RK solver
`times`	the vector containing informations on integration extremes
`Dates`	the vector containing the date
`N`	the number of individuals time series
`CS`	the commercial size of Seabass

Value

output: a list containing the fish weight, proteines, lipids and carbohydrates wasted or produced with excretions, potential and actual ingestion rates, temperature limitation functions and metabolic rates

---

`Bass_pop_pre` *Seabass bioenergetic population model preprocessor*

Description

Seabass bioenergetic population model preprocessor

Usage

```
Bass_pop_pre(userpath, forcings)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located
`forcings`	a list containing model forcings

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees] and feeding rate [g/individual x d]

`Bass_pop_RKsolver` *Solves the Seabass population bioenergetic balance with a 4th order Runge Kutta method*

Description

Solves the Seabass population bioenergetic balance with a 4th order Runge Kutta method

Usage

```
Bass_pop_RKsolver(Param, Temperature, G, Food, IC, times, N)
```

Arguments

`Param`	vector containing all metabolic parameters
`Temperature`	water temperature forcing time series
`G`	food entering the cage at time series
`Food`	food characterization (Proteins, Lipids, Carbohydrates)
`IC`	initial condition on weight
`times`	integration times
`N`	number of individuals time series

Value

a list containing the fish weight, proteines, lipids and carbohydrates wasted or produced with excretions, potential and actual ingestion rates, temperature limitation functions and metabolic rates

---

`Bass_pop_skeleton` *Creates the folders structure for Seabass population model*

Description

Creates the folders structure for Seabass population model

Usage

```
Bass_pop_skeleton(userpath)
```

Arguments

`userpath`	the path where forcing are located

`Bass_spatial_dataloader` *Function that loads forcings data for Bass spatialized model and performs the interpolation*

Description

Function that loads forcings data for Bass spatialized model and performs the interpolation

Usage

```
Bass_spatial_dataloader(userpath)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees] and feeding rate [g/individual x d]

`Bass_spatial_loop` *Bass bioenergetic spatialized model - spatialization loop*

Description

Solves the bioenergetic balance for Bass

Usage

```
Bass_spatial_loop(userpath, forcings)
```

Arguments

`userpath`	the path where the working folder is located
`forcings`	a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees] and feeding rate [g/individual x d]

Value

a list containing the outputs that main script saves to .nc; .csv and .asc files

---

`Bass_spatial_main` *Bass bioenergetic spatialized model - spatialization loop*

Description

Solves the bioenergetic balance for Bass

Usage

```
Bass_spatial_main(userpath, forcings)
```

Arguments

`userpath`	the path where the working folder is located
`forcings`	list containing the time series in the odd positions and realted forcings in the even positions. Forcings imputted are: Water temperature [Celsius degrees] and feeding rate [g/individual x d]

Value

saves .nc; .csv and .asc outputs in the 'Outputs' folder

`Bass_spatial_post` *Postprocess the Mussel spatialized model results*

Description

Postprocess the Mussel spatialized model results

Usage

```
Bass_spatial_post(userpath, output, times, Dates, CS)
```

Arguments

`userpath`	the path where the working folder is located
`output`	output list containing the output of the RK solver
`times`	the vector containing informations on integration extremes
`Dates`	the vector containing the date
`CS`	the commercial size of Bass

Value

a list containing the fish weight, proteines, lipids and carbohydrates wasted or produced with excretions, potential and actual ingestion rates, temperature limitation functions and metabolic rates

---

`Bass_spatial_pre` *Bass bioenergetic spatialized model preprocessor*

Description

Bass bioenergetic spatialized model preprocessor

Usage

```
Bass_spatial_pre(userpath, forcings)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located
`forcings`	a list containing forcings used by the model

Value

a list containing the data used by the main script

`Bass_spatial_pre_int` *Bass bioenergetic spatialized model preprocessor - used inside spatialization loop*

Description

Bass bioenergetic spatialized model preprocessor - used inside spatialization loop

Usage

```
Bass_spatial_pre_int(userpath, forcings)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located
`forcings`	a list containing forcings used by the model

Value

a list containing data used by the main script

`Bass_spatial_RKsolver` *Solves the Seabass bioenergetic balance with a 4th order Runge Kutta method - used in spatialized model*

Description

Solves the Seabass bioenergetic balance with a 4th order Runge Kutta method - used in spatialized model

Usage

```
Bass_spatial_RKsolver(Param, Temperature, G, Food, IC, times)
```

Arguments

`Param`	vector containing all metabolic parameters
`Temperature`	water temperature forcing time series
`G`	food entering the cage time series
`Food`	food characterization (Proteins, Lipids, Carbohydrates)
`IC`	initial conditions
`times`	vector containing integration extremes and integration timestep

Value

a list containing the fish weight, proteines, lipids and carbohydrates wasted or produced with excretions, potential and actual ingestion rates, temperature limitation functions and metabolic rates

---

`Bass_spatial_skeleton` *Creates the folders structure for Bass spatialized model*

Description

Creates the folders structure for Bass spatialized model

Usage

```
Bass_spatial_skeleton(userpath)
```

Arguments

`userpath`	the path where forcing are located

`Bream_ind_dataloader` *Function that loads forcings data for Seabream individual bioenergetic model and performs the interpolation*

Description

Function that loads forcings data for Seabream individual bioenergetic model and performs the interpolation

Usage

```
Bream_ind_dataloader(userpath)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees] and feeding rate [g/individual x d]

`Bream_ind_equations` *Seabream bioenergetic individual model differential equations*

Description

Seabream bioenergetic individual model differential equations

Usage

```
Bream_ind_equations(Param, Temp, G, Food, weight)
```

Arguments

`Param`	vector containing all metabolic parameters
`Temp`	water temperature forcing at time t
`G`	food entering the cage at time t
`Food`	food characterization (Proteins, Lipids, Carbohydrates)
`weight`	individual weight at time t

Value

model output at time t

`Bream_ind_main` *Seabream bioenergetic individual model*

Description

Seabream bioenergetic individual model

Usage

```
Bream_ind_main(userpath, forcings)
```

Arguments

`userpath`	the path where the working folder is located
`forcings`	a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees] and feeding rate [g/individual x d]

Value

A list containing model outputs: weight, excreted quantities and quantities to waste, actual and potential ingestion, temperature limitation functions and metabolic rates

---

`Bream_ind_post` *Postprocess the Seabream indivual bioenergetic model results*

Description

Postprocess the Seabream indivual bioenergetic model results

Usage

```
Bream_ind_post(userpath, output, times, Dates, CS)
```

Arguments

`userpath`	the path where the working folder is located
`output`	output list containing the output of the RK solver
`times`	the vector containing informations on integration extremes
`Dates`	the vector containing the date
`CS`	the commercial size of Seabream

Value

a list containing the fish weight, proteines, lipids and carbohydrates wasted or produced with excretions, potential and actual ingestion rates, temperature limitation functions and metabolic rates

---

`Bream_ind_pre` *Seabream bioenergetic individual model preprocessor*

Description

Preprocesses the data for the bioenergetic balance for Sea Bream

Usage

```
Bream_ind_pre(userpath, forcings)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located
`forcings`	a list containing model forcings

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees] and feeding rate [g/individual x d]

`Bream_ind_RKsolver` *Solves the Seabream individual bioenergetic balance with a 4th order Runge Kutta method*

Description

Solves the Seabream individual bioenergetic balance with a 4th order Runge Kutta method

Usage

```
Bream_ind_RKsolver(Param, Temperature, G, Food, IC, times)
```

Arguments

`Param`	vector containing all metabolic parameters
`Temperature`	water temperature forcing time series
`G`	food entering the cage time series
`Food`	food characterization (Proteins, Lipids, Carbohydrates)
`IC`	vector containing initial conditions on weight
`times`	vector containing integration extremes and timestep

Value

a list containing the fish weight, proteines, lipids and carbohydrates wasted or produced with excretions, potential and actual ingestion rates, temperature limitation functions and metabolic rates

---

`Bream_ind_skeleton` *Creates the folders structure for Seabream individual bioenergetic model*

Description

Creates the folders structure for Seabream individual bioenergetic model

Usage

```
Bream_ind_skeleton(userpath)
```

Arguments

`userpath`	the path where forcing are located

`Bream_pop_dataloader` *Function that loads forcings data for Seabream population model and performs the interpolation*

Description

Function that loads forcings data for Seabream population model and performs the interpolation

Usage

```
Bream_pop_dataloader(userpath)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees] and feeding rate [g/individual x d]

`Bream_pop_equations` *Seabream bioenergetic population model differential equations*

Description

Seabream bioenergetic population model differential equations

Usage

```
Bream_pop_equations(Param, N, Temp, G, Food, weight)
```

Arguments

`Param`	vector containing all metabolic parameters
`N`	the number of individuals at time t
`Temp`	water temperature forcing at time t
`G`	food entering the cage at time t
`Food`	food characterization (Proteins, Lipids, Carbohydrates)
`weight`	individual weight at time t

Value

model output at time t

`Bream_pop_loop` *Function that runs the Monte Carlo simulation for the Seabream population model*

Description

Function that runs the Monte Carlo simulation for the Seabream population model

Usage

```
Bream_pop_loop(Param, Tint, Gint, Food, IC, times, N, userpath)
```

Arguments

`Param`	a vector containing model parameters
`Tint`	the interpolated water temperature time series
`Gint`	the interpolated feeding rate time series
`Food`	the food characterization
`IC`	initial condition
`times`	integration extremes and integration timestep
`N`	time series with number of individuals
`userpath`	the path where the working folder is located

Value

a list with RK solver outputs

`Bream_pop_main` *Seabream bioenergetic population model*

Description

Seabream bioenergetic population model

Usage

```
Bream_pop_main(userpath, forcings)
```

Arguments

`userpath`	the path where the working folder is located
`forcings`	a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees] and feeding rate [g/individual x d]

Value

A list containing model outputs: weight, excreted quantities and quantities to waste, actual and potential ingestion, temperature limitation functions and metabolic rates

---

`Bream_pop_post` *Postprocess the Bream population bioenergetic model results*

Description

Postprocess the Bream population bioenergetic model results

Usage

```
Bream_pop_post(userpath, output, times, Dates, N, CS)
```

Arguments

`userpath`	the path where the working folder is located
`output`	output list containing the output of the RK solver
`times`	the vector containing informations on integration extremes
`Dates`	the vector containing the date
`N`	the number of individuals
`CS`	the commercial size of Seabream

Value

a list containing the fish weight, proteines, lipids and carbohydrates wasted or produced with excretions, potential and actual ingestion rates, temperature limitation functions and metabolic rates

---

`Bream_pop_pre` *Seabream bioenergetic population model preprocessor*

Description

Preprocesses the data for the bioenergetic balance for Sea Bream

Usage

```
Bream_pop_pre(userpath, forcings)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located
`forcings`	a list containing model forcings

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees] and feeding rate [g/individual x d]

`Bream_pop_RKsolver` *Solves the Seabream population bioenergetic balance with a 4th order Runge Kutta method*

Description

Solves the Seabream population bioenergetic balance with a 4th order Runge Kutta method

Usage

```
Bream_pop_RKsolver(Param, Temperature, G, Food, IC, times, N)
```

Arguments

`Param`	vector containing all metabolic parameters
`Temperature`	water temperature forcing time series
`G`	food entering the cage time series
`Food`	food characterization (Proteins, Lipids, Carbohydrates)
`IC`	initial condition on weight
`times`	integration times
`N`	number of individuals time series

Value

a list containing the fish weight, proteines, lipids and carbohydrates wasted or produced with excretions, potential and actual ingestion rates, temperature limitation functions and metabolic rates

---

`Bream_pop_skeleton` *Creates the folders structure for Seabream population model*

Description

Creates the folders structure for Seabream population model

Usage

```
Bream_pop_skeleton(userpath)
```

Arguments

`userpath`	the path where forcing are located

`Bream_spatial_dataloader` *Function that loads forcings data for Bream spatialized model and performs the interpolation*

Description

Function that loads forcings data for Bream spatialized model and performs the interpolation

Usage

```
Bream_spatial_dataloader(userpath)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees] and feeding rate [g/individual x d]

`Bream_spatial_loop` *Bream bioenergetic spatialized model - spatialization loop*

Description

Solves the bioenergetic balance for Bream

Usage

```
Bream_spatial_loop(userpath, forcings)
```

Arguments

`userpath`	the path where the working folder is located
`forcings`	a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees] and feeding rate [g/individual x d]

Value

A list containing model outputs: weight, excreted quantities and quantities to waste, actual and potential ingestion, temperature limitation functions and metabolic rates

---

`Bream_spatial_main` *Bream bioenergetic spatialized model - spatialization loop*

Description

Solves the bioenergetic balance for Bream

Usage

```
Bream_spatial_main(userpath, forcings)
```

Arguments

`userpath`	the path where the working folder is located
`forcings`	list containing the time series in the odd positions and realted forcings in the even positions. Forcings imputted are: Water temperature [Celsius degrees] and feeding rate [g/individual x d]

Value

saves .nc; .csv and .asc outputs in the 'Outputs' folder

`Bream_spatial_post` *Postprocess the Bream spatialized model results*

Description

Postprocess the Bream spatialized model results

Usage

```
Bream_spatial_post(userpath, output, times, Dates, CS)
```

Arguments

`userpath`	the path where the working folder is located
`output`	output list containing the output of the RK solver
`times`	the vector containing informations on integration extremes
`Dates`	the vector containing the date
`CS`	the commercial size of Bream

Value

a list containing the model outputs saved by the main script to .nc; .csv and .asc files

---

`Bream_spatial_pre` *Bream bioenergetic spatialized model preprocessor*

Description

Bream bioenergetic spatialized model preprocessor

Usage

```
Bream_spatial_pre(userpath, forcings)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located
`forcings`	a list containing forcings used by the model

Value

a list containing the data used in the main script

`Bream_spatial_pre_int` *Bream bioenergetic spatialized model preprocessor - used inside spatialization loop*

Description

Bream bioenergetic spatialized model preprocessor - used inside spatialization loop

Usage

```
Bream_spatial_pre_int(userpath, forcings)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located
`forcings`	a list containing forcings used by the model

Value

a list containing the data used by the main script

`Bream_spatial_RKsolver` *Solves the Seabream bioenergetic balance with a 4th order Runge Kutta method - used in spatialized model*

Description

Solves the Seabream bioenergetic balance with a 4th order Runge Kutta method - used in spatialized model

Usage

```
Bream_spatial_RKsolver(Param, Temperature, G, Food, IC, times)
```

Arguments

`Param`	vector containing all metabolic parameters
`Temperature`	water temperature forcing time series
`G`	food entering the cage time series
`Food`	food characterization (Proteins, Lipids, Carbohydrates)
`IC`	initial conditions
`times`	vector containing integration extremes and integration timestep

Value

a list containing the fish weight, proteines, lipids and carbohydrates wasted or produced with excretions, potential and actual ingestion rates, temperature limitation functions and metabolic rates

---

`Bream_spatial_skeleton` *Creates the folders structure for Bream spatialized model*

Description

Creates the folders structure for Bream spatialized model

Usage

```
Bream_spatial_skeleton(userpath)
```

Arguments

`userpath`	the path where forcing are located

`ClamF_ind_dataloader` *Function that loads forcings data for Clam individual bioenergetic model (alternative version) and performs the interpolation*

Description

Function that loads forcings data for Clam individual bioenergetic model (alternative version) and performs the interpolation

Usage

```
ClamF_ind_dataloader(userpath)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3]

`ClamF_ind_equations` *Clam bioenergetic individual model differential equations (alternative version)*

Description

Clam bioenergetic individual model differential equations (alternative version)

Usage

```
ClamF_ind_equations(Param, Tint, Chlint, Ww)
```

Arguments

`Param`	a vector containing model parameters
`Tint`	the interpolated water temperature at time t
`Chlint`	the interpolated chlorophyll at time t
`Ww`	clam wet weight at time t

Value

a list containing the clam weights, temperature limitation functions and metabolic rates at time t

`ClamF_ind_main` *Clam bioenergetic individual model (alternative version)*

Description

Clam bioenergetic individual model (alternative version)

Usage

```
ClamF_ind_main(userpath, forcings)
```

Arguments

`userpath`	the path where the working folder is located
`forcings`	a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3]

Value

A list containing model outputs: weights, temperature limitation functions and metabolic rates

`ClamF_ind_post` *Postprocess the Clam indivual bioenergetic model (alternative version) results*

Description

Postprocess the Clam indivual bioenergetic model (alternative version) results

Usage

```
ClamF_ind_post(userpath, output, times, Dates, CS)
```

Arguments

`userpath`	the path where the working folder is located
`output`	output list containing the output of the RK solver
`times`	the vector containing informations on integration extremes
`Dates`	the vector containing the date
`CS`	the commercial size of Clam

Value

a list containing the clam weights, temperature limitation functions and metabolic rates

---

`ClamF_ind_pre` *Clam bioenergetic individual model preprocessor (alternativer version)*

Description

Clam bioenergetic individual model preprocessor (alternativer version)

Usage

```
ClamF_ind_pre(userpath, forcings)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located
`forcings`	a list containing model forcings

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3]

`ClamF_ind_RKsolver` *Solves the Clam bioenergetic balance (alternative version) with a 4th order Runge Kutta method*

Description

Solves the Clam bioenergetic balance (alternative version) with a 4th order Runge Kutta method

Usage

```
ClamF_ind_RKsolver(Param, times, IC, Tint, Chlint)
```

Arguments

`Param`	a vector containing model parameters
`times`	integration extremes and integration timestep
`IC`	initial condition
`Tint`	the interpolated water temperature time series
`Chlint`	the interpolated chlorophyll a time series

Value

a list containing the clam weights, temperature limitation functions and metabolic rates

---

`ClamF_ind_skeleton` *Creates the folders structure for Clam individual bioenergetic model (alternative version)*

Description

Creates the folders structure for Clam individual bioenergetic model (alternative version)

Usage

```
ClamF_ind_skeleton(userpath)
```

Arguments

`userpath`	the path where forcing are located

`ClamF_pop_dataloader` *Function that loads forcings data for Clam population model (alternative version) and performs the interpolation*

Description

Function that loads forcings data for Clam population model (alternative version) and performs the interpolation

Usage

```
ClamF_pop_dataloader(userpath)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3]

`ClamF_pop_equations` *Clam bioenergetic population model differential equations (alternative version)*

Description

Clam bioenergetic population model differential equations (alternative version)

Usage

```
ClamF_pop_equations(Param, Tint, Chlint, Ww)
```

Arguments

`Param`	a vector containing model parameters
`Tint`	the interpolated water temperature at time t
`Chlint`	the interpolated chlorophyll at time t
`Ww`	clam wet weight at time t

Value

a list containing the clam weights, temperature limitation functions and metabolic rates at time t

---

`ClamF_pop_loop` *Function that runs the Monte Carlo simulation for the Clam population model (alternative version)*

Description

Function that runs the Monte Carlo simulation for the Clam population model (alternative version)

Usage

```
ClamF_pop_loop(Param, times, IC, Tint, Chlint, N, userpath)
```

Arguments

`Param`	a vector containing model parameters
`times`	integration extremes and integration timestep
`IC`	initial condition
`Tint`	the interpolated water temperature time series
`Chlint`	the interpolated chlorophyll a time series
`N`	time series with number of individuals
`userpath`	the path where the working folder is located

Value

a list with RK solver outputs

`ClamF_pop_main` *Clam bioenergetic population model (alternative version)*

Description

Clam bioenergetic population model (alternative version)

Usage

```
ClamF_pop_main(userpath, forcings)
```

Arguments

`userpath`	the path where the working folder is located
`forcings`	a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3]

Value

A list containing model outputs: weights, temperature limitation functions and metabolic rates

---

`ClamF_pop_post` *Postprocess the Clam population bioenergetic model results (alternative model)*

Description

Postprocess the Clam population bioenergetic model results (alternative model)

Usage

```
ClamF_pop_post(userpath, output, times, Dates, N, CS)
```

Arguments

`userpath`	the path where the working folder is located
`output`	output list containing the output of the RK solver
`times`	the vector containing informations on integration extremes
`Dates`	the vector containing the date
`N`	the number of individuals
`CS`	the commercial size of Clam

Value

a list containing the clam weights, temperature limitation functions and metabolic rates

---

`ClamF_pop_pre` *Clam bioenergetic population model (alternative version) preprocessor*

Description

Clam bioenergetic population model (alternative version) preprocessor

Usage

```
ClamF_pop_pre(userpath, forcings)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located
`forcings`	a list containing model forcings

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3]

`ClamF_pop_RKsolver` *Solves the Clam bioenergetic balance (alternative version) with a 4th order Runge Kutta method*

Description

Solves the Clam bioenergetic balance (alternative version) with a 4th order Runge Kutta method

Usage

```
ClamF_pop_RKsolver(Param, times, IC, Tint, Chlint)
```

Arguments

`Param`	a vector containing model parameters
`times`	integration extremes and integration timestep
`IC`	initial condition on weight
`Tint`	the interpolated water temperature time series
`Chlint`	the interpolated chlorophyll a time series

Value

a list containing the clam weights, temperature limitation functions and metabolic rates

---

`ClamF_pop_skeleton` *Creates the folders structure for Clam population model (alternative version)*

Description

Creates the folders structure for Clam population model (alternative version)

Usage

```
ClamF_pop_skeleton(userpath)
```

Arguments

`userpath`	the path where forcing are located

`Clam_ind_dataloader` *Function that loads forcings data for Clam individual bioenergetic model and performs the interpolation*

Description

Function that loads forcings data for Clam individual bioenergetic model and performs the interpolation

Usage

```
Clam_ind_dataloader(userpath)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3], particulated organic carbon (POC) concentration [mgC/l], particulated organic matter (POM) concentration [mgC/l], total suspended solids (TSS) concentration [mg/l]

`Clam_ind_equations` *Clam bioenergetic individual model differential equations*

Description

Clam bioenergetic individual model differential equations

Usage

```
Clam_ind_equations(Param, Tint, Phy, DT, POCint, POMint, TSSint, Wd)
```

Arguments

`Param`	a vector containing model parameters
`Tint`	the interpolated water temperature at time t
`Phy`	the interpolated phytoplankton at time t
`DT`	the interpolated detritus at time t
`POCint`	the interpolated POC at time t
`POMint`	the interpolated POM at time t
`TSSint`	the interpolated TSS at time t
`Wd`	the weight of the clam at time t

Value

a list containing the clam weights, temperature limitation functions and metabolic rates at time t

---

`Clam_ind_main` *Clam bioenergetic individual model*

Description

Clam bioenergetic individual model

Usage

```
Clam_ind_main(userpath, forcings)
```

Arguments

`userpath`	the path where the working folder is located
`forcings`	a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3], particulated organic carbon (POC) concentration [mgC/l], particulated organic matter (POM) concentration [mgC/l], total suspended solids (TSS) concentration [mg/l]

Value

A list containing model outputs: weights, temperature limitation functions and metabolic rates

---

`Clam_ind_post` *Postprocess the Clam individual bioenergetic model results*

Description

Postprocess the Clam individual bioenergetic model results

Usage

```
Clam_ind_post(userpath, output, times, Dates, CS)
```

Arguments

`userpath`	the path where the working folder is located
`output`	output list containing the output of the RK solver
`times`	the vector containing informations on integration extremes
`Dates`	the vector containing the date
`CS`	the commercial size of Clam

Value

a list containing the clam weights, temperature limitation functions and metabolic rates

---

`Clam_ind_pre` *Clam bioenergetic individual model preprocessor*

Description

Clam bioenergetic individual model preprocessor

Usage

```
Clam_ind_pre(userpath, forcings)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located
`forcings`	a list containing model forcings

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3], particulated organic carbon (POC) concentration [mgC/l], particulated organic matter (POM) concentration [mgC/l], total suspended solids (TSS) concentration [mg/l]

`clam_ind_RKsolver` *Solves the Clam individual bioenergetic balance with a 4th order Runge Kutta method*

Description

Solves the Clam individual bioenergetic balance with a 4th order Runge Kutta method

Usage

```
clam_ind_RKsolver(Param, times, IC, Tint, Phyint, DTint, POCint, POMint, TSSint)
```

Arguments

`Param`	a vector containing model parameters
`times`	integration extremes and integration timestep
`IC`	initial condition
`Tint`	the interpolated water temperature time series
`Phyint`	the interpolated phytoplankton time series
`DTint`	the interpolated detritus time series
`POCint`	the interpolated POC time series
`POMint`	the interpolated POM time series
`TSSint`	the interpolated TSS time series

Value

a list containing the clam weights, temperature limitation functions and metabolic rates

---

`Clam_ind_skeleton` *Creates the folders structure for Clam individual bioenergetic model*

Description

Creates the folders structure for Clam individual bioenergetic model

Usage

```
Clam_ind_skeleton(userpath)
```

Arguments

`userpath`	the path where forcing are located

`Clam_pop_dataloader` *Function that loads forcings data for Clam population model and performs the interpolation*

Description

Function that loads forcings data for Clam population model and performs the interpolation

Usage

```
Clam_pop_dataloader(userpath)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3], particulated organic carbon (POC) concentration [mgC/l], particulated organic matter (POM) concentration [mgC/l], total suspended solids (TSS) concentration [mg/l]

`Clam_pop_equations` *Clam bioenergetic population model differential equations*

Description

Clam bioenergetic population model differential equations

Usage

```
Clam_pop_equations(Param, Tint, Phy, DT, POCint, POMint, TSSint, Wd)
```

Arguments

`Param`	a vector containing model parameters
`Tint`	the interpolated water temperature at time t
`Phy`	the interpolated phytoplankton at time t
`DT`	the interpolated detritus at time t
`POCint`	the interpolated POC at time t
`POMint`	the interpolated POM at time t
`TSSint`	the interpolated TSS at time t
`Wd`	the weight of the clam at time t

Value

a list containing the clam weights, temperature limitation functions and metabolic rates at time t

---

`Clam_pop_loop` *Function that runs the Monte Carlo simulation for the Clam population model*

Description

Function that runs the Monte Carlo simulation for the Clam population model

Usage

```
Clam_pop_loop(Param, times, IC, Tint, Phyint, DTint, POCint, POMint, TSSint, N,
  userpath)
```

Arguments

`Param`	a vector containing model parameters
`times`	integration extremes and integration timestep
`IC`	initial condition
`Tint`	the interpolated water temperature time series
`Phyint`	the interpolated phytoplankton time series
`DTint`	the interpolated detritus time series
`POCint`	the interpolated POC time series
`POMint`	the interpolated POM time series
`TSSint`	the interpolated TSS time series
`N`	time series with number of individuals
`userpath`	the path where the working folder is located

Value

a list with RK solver outputs

`Clam_pop_main` *Clam bioenergetic population model*

Description

Clam bioenergetic population model

Usage

```
Clam_pop_main(userpath, forcings)
```

Arguments

`userpath`	the path where the working folder is located
`forcings`	a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3], particulated organic carbon (POC) concentration [mgC/l], particulated organic matter (POM) concentration [mgC/l], total suspended solids (TSS) concentration [mg/l]

Value

A list containing model outputs: weights, temperature limitation functions and metabolic rates

---

`Clam_pop_post` *Postprocess the Clam population bioenergetic model results*

Description

Postprocess the Clam population bioenergetic model results

Usage

```
Clam_pop_post(userpath, output, times, Dates, N, CS)
```

Arguments

`userpath`	the path where the working folder is located
`output`	output list containing the output of the RK solver
`times`	the vector containing informations on integration extremes
`Dates`	the vector containing the date
`N`	the number of individuals
`CS`	the commercial size of Clam

Value

a list containing the clam weights, temperature limitation functions and metabolic rates

---

`Clam_pop_pre` *Clam bioenergetic population model preprocessor*

Description

Clam bioenergetic population model preprocessor

Usage

```
Clam_pop_pre(userpath, forcings)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located
`forcings`	a list containing model forcings

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3], particulated organic carbon (POC) concentration [mgC/l], particulated organic matter (POM) concentration [mgC/l], total suspended solids (TSS) concentration [mg/l]

`Clam_pop_RKsolver` *Solves the Clam bioenergetic balance for population with a 4th order Runge Kutta method*

Description

Solves the Clam bioenergetic balance for population with a 4th order Runge Kutta method

Usage

```
Clam_pop_RKsolver(Param, times, IC, Tint, Phyint, DTint, POCint, POMint, TSSint)
```

Arguments

`Param`	a vector containing model parameters
`times`	integration extremes and integration timestep
`IC`	initial condition
`Tint`	the interpolated water temperature time series
`Phyint`	the interpolated phytoplankton time series
`DTint`	the interpolated detritus time series
`POCint`	the interpolated POC time series
`POMint`	the interpolated POM time series
`TSSint`	the interpolated TSS time series

Value

a list containing the clam weights, temperature limitation functions and metabolic rates

---

`Clam_pop_skeleton` *Creates the folders structure for Clam population model*

Description

Creates the folders structure for Clam population model

Usage

```
Clam_pop_skeleton(userpath)
```

Arguments

`userpath`	the path where forcing are located

`Mussel_ind_dataloader` *Function that loads forcings data for Mussel individual bioenergetic model and performs the interpolation*

Description

Function that loads forcings data for Mussel individual bioenergetic model and performs the interpolation

Usage

```
Mussel_ind_dataloader(userpath)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3], particulated organic carbon (POC) concentration [mgC/l] and its characterization in terms of C/P and N/P molar ratios, particulated organic matter (POM) concentration [mgC/l], total suspended solids (TSS) concentration [mg/l]

`Mussel_ind_equations` *Mussel bioenergetic individual model differential equations*

Description

Mussel bioenergetic individual model differential equations

Usage

```
Mussel_ind_equations(Param, Tint, Phyint, DTint, POCint, Ccont, Ncont, Pcont,
  POMint, TSSint, Wb, R, t, trip)
```

Arguments

`Param`	a vector containing model parameters
`Tint`	the interpolated water temperature at time t
`Phyint`	the interpolated phytoplankton at time t
`DTint`	the interpolated detritus at time t
`POCint`	the interpolated POC at time t

`Ccont`	the C/C content of the POC at time t
`Ncont`	the N/C content of POC at time t
`Pcont`	the P/C content of POC at time t
`POMint`	the interpolated POM at time t
`TSSint`	the interpolated TSS at time t
`Wb`	the somatic tissue dry weight at time t
`R`	the gondadic tissue dry weight at time t
`t`	the time
`trip`	vector containing the flags with resting periods

Value

the outputs at time t

`Mussel_ind_main` *Mussel bioenergetic individual model*

Description

Solves the bioenergetic balance for Mussel

Usage

```
Mussel_ind_main(userpath, forcings)
```

Arguments

`userpath`	the path where the working folder is located
`forcings`	a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3], particulated organic carbon (POC) concentration [mgC/l] and its characterization in terms of C/P and N/P molar ratios, particulated organic matter (POM) concentration [mgC/l], total suspended solids (TSS) concentration [mg/l]

Value

A list containing model outputs: weight, length mussel CNP, pseudofaeces CNP production, faeces CNP production, temperature limitation functions, metabolic rates and oxygen consumption

---

`Mussel_ind_post` *Postprocess the Mussel indivual bioenergetic model results*

Description

Postprocess the Mussel indivual bioenergetic model results

Usage

```
Mussel_ind_post(userpath, output, times, Dates, CS)
```

Arguments

`userpath`	the path where the working folder is located
`output`	output list containing the output of the RK solver
`times`	the vector containing informations on integration extremes
`Dates`	the vector containing the date
`CS`	the commercial size of Mussel

Value

a list containing the weights of the mussel, the excreted CNP, the mussel CNP, temperature limitation functions, metabolic rates, oxygen consumption

---

`Mussel_ind_pre` *Mussel bioenergetic individual model preprocessor*

Description

Mussel bioenergetic individual model preprocessor

Usage

```
Mussel_ind_pre(userpath, forcings)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located
`forcings`	a list containing model forcings

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3], particulated organic carbon (POC) concentration [mgC/l] and its characterization in terms of C/P and N/P molar ratios, particulated organic matter (POM) concentration [mgC/l], total suspended solids (TSS) concentration [mg/l]

`Mussel_ind_RKsolver` *Solves the Mussel individual bioenergetic balance with a 4th order Runge Kutta method*

Description

Solves the Mussel individual bioenergetic balance with a 4th order Runge Kutta method

Usage

```
Mussel_ind_RKsolver(Param, times, IC, Tint, Phyint, DTint, POCint, Ccont, Ncont,
  Pcont, POMint, TSSint)
```

Arguments

`Param`	a vector containing model parameters
`times`	integration extremes and integration timestep
`IC`	initial condition
`Tint`	the interpolated water temperature time series
`Phyint`	the interpolated phytoplankton time series
`DTint`	the interpolated detritus time series
`POCint`	the interpolated POC time series
`Ccont`	the C/C content of the POC
`Ncont`	the N/C content of POC
`Pcont`	the P/C content of POC
`POMint`	the interpolated POM time series
`TSSint`	the interpolated TSS time series

Value

a list containing the weights of the mussel, the excreted CNP, the mussel CNP, temperature limitation functions, metabolic rates, oxygen consumption

---

`Mussel_ind_skeleton` *Creates the folders structure for Mussel individual bioenergetic model*

Description

Creates the folders structure for Mussel individual bioenergetic model

Usage

```
Mussel_ind_skeleton(userpath)
```

Arguments

`userpath`	the path where forcing are located

`Mussel_pop_dataloader` *Function that loads forcings data for Mussel population model and performs the interpolation*

Description

Function that loads forcings data for Mussel population model and performs the interpolation

Usage

```
Mussel_pop_dataloader(userpath)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3], particulated organic carbon (POC) concentration [mgC/l] and its characterization in terms of C/P and N/P molar ratios, particulated organic matter (POM) concentration [mgC/l], total suspended solids (TSS) concentration [mg/l]

`Mussel_pop_equations` *Mussel bioenergetic population model differential equations*

Description

Mussel bioenergetic population model differential equations

Usage

```
Mussel_pop_equations(Param, N, Tint, Phyint, DTint, POCint, Ccont, Ncont, Pcont,
  POMint, TSSint, Wb, R, t, trip)
```

Arguments

`Param`	a vector containing model parameters
`N`	the number of individuals at time t
`Tint`	the interpolated water temperature at time t
`Phyint`	the interpolated phytoplankton at time t
`DTint`	the interpolated detritus at time t
`POCint`	the interpolated POC at time t
`Ccont`	the C/C content of the POC at time t
`Ncont`	the N/C content of POC at time t
`Pcont`	the P/C content of POC at time t
`POMint`	the interpolated POM at time t
`TSSint`	the interpolated TSS at time t
`Wb`	the somatic tissue dry weight at time t
`R`	the gondadic tissue dry weight at time t
`t`	the time
`trip`	vector containing the flags with resting periods

Value

the outputs at time t

`Mussel_pop_loop` *Function that runs the Monte Carlo simulation for the Mussel population model*

Description

Function that runs the Monte Carlo simulation for the Mussel population model

Usage

```
Mussel_pop_loop(Param, times, IC, Tint, Phyint, DTint, POCint, Ccont, Ncont,
  Pcont, POMint, TSSint, N, userpath)
```

Arguments

`Param`	a vector containing model parameters
`times`	integration extremes and integration timestep
`IC`	initial condition
`Tint`	the interpolated water temperature time series
`Phyint`	the interpolated phytoplankton time series
`DTint`	the interpolated detritus time series
`POCint`	the interpolated POC time series
`Ccont`	the C/C content of the POC
`Ncont`	the N/C content of POC
`Pcont`	the P/C content of POC
`POMint`	the interpolated POM time series
`TSSint`	the interpolated TSS time series
`N`	time series with number of individuals
`userpath`	the path where the working folder is located

Value

a list with RK solver outputs

`Mussel_pop_main` *Mussel bioenergetic population model*

Description

Solves the bioenergetic balance for Mussel and simulates a population

Usage

```
Mussel_pop_main(userpath, forcings)
```

Arguments

`userpath`	the path where the working folder is located
`forcings`	a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3], particulated organic carbon (POC) concentration [mgC/l] and its characterization in terms of C/P and N/P molar ratios, particulated organic matter (POM) concentration [mgC/l], total suspended solids (TSS) concentration [mg/l]

Value

A list containing model outputs: weight, length mussel CNP, pseudofecies CNP production, temperature limitation functions, metabolic rates and oxygen consumption

---

`Mussel_pop_post` *Postprocess the Mussel population bioenergetic model results*

Description

Postprocess the Mussel population bioenergetic model results

Usage

```
Mussel_pop_post(userpath, output, times, Dates, N, CS)
```

Arguments

`userpath`	the path where the working folder is located
`output`	output list containing the output of the RK solver
`times`	the vector containing informations on integration extremes
`Dates`	the vector containing the date
`N`	the number of individuals
`CS`	the commercial size of Seabass

Value

a list containing the weights of the mussel, the excreted CNP, the mussel CNP, temperature limitation functions, metabolic rates, oxygen consumption

---

`Mussel_pop_pre` *Mussel bioenergetic population model preprocessor*

Description

Mussel bioenergetic population model preprocessor

Usage

```
Mussel_pop_pre(userpath, forcings)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located
`forcings`	a list containing forcings used by the model

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3], particulated organic carbon (POC) concentration [mgC/l] and its characterization in terms of C/P and N/P molar ratios, particulated organic matter (POM) concentration [mgC/l], total suspended solids (TSS) concentration [mg/l]

`Mussel_pop_RKsolver` *Solves the Mussel population bioenergetic balance with a 4th order Runge Kutta method*

Description

Solves the Mussel population bioenergetic balance with a 4th order Runge Kutta method

Usage

```
Mussel_pop_RKsolver(Param, times, IC, Tint, Phyint, DTint, POCint, Ccont, Ncont,
  Pcont, POMint, TSSint, N)
```

Arguments

`Param`	a vector containing model parameters
`times`	integration extremes and integration timestep
`IC`	initial condition
`Tint`	the interpolated water temperature time series
`Phyint`	the interpolated phytoplankton time series
`DTint`	the interpolated detritus time series
`POCint`	the interpolated POC time series
`Ccont`	the C/C content of the POC
`Ncont`	the N/C content of POC
`Pcont`	the P/C content of POC
`POMint`	the interpolated POM time series
`TSSint`	the interpolated TSS time series
`N`	the number of indivduals time series

Value

a list containing the weights of the mussel, the excreted CNP, the mussel CNP, temperature limitation functions, metabolic rates, oxygen consumption

---

`Mussel_pop_skeleton` *Creates the folders structure for Mussel population model*

Description

Creates the folders structure for Mussel population model

Usage

```
Mussel_pop_skeleton(userpath)
```

Arguments

`userpath`	the path where forcing are located

`Mussel_spatial_dataloader` *Function that loads forcings data for Mussel spatialized model and performs the interpolation*

Description

Function that loads forcings data for Mussel spatialized model and performs the interpolation

Usage

```
Mussel_spatial_dataloader(userpath)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3], particulated organic carbon (POC) concentration [mgC/l] and its characterization in terms of C/P and N/P molar ratios, particulated organic matter (POM) concentration [mgC/l], total suspended solids (TSS) concentration [mg/l]

`Mussel_spatial_loop` *Mussel bioenergetic spatialized model - spatialization loop*

Description

Solves the bioenergetic balance for Mussel

Usage

```
Mussel_spatial_loop(userpath, forcings)
```

Arguments

`userpath`	the path where the working folder is located
`forcings`	a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3], particulated organic carbon (POC) concentration [mgC/l] and its characterization in terms of C/P and N/P molar ratios, particulated organic matter (POM) concentration [mgC/l], total suspended solids (TSS) concentration [mg/l]

Value

A list containing model outputs that main script saves to .nc; .csv and .asc files

---

`Mussel_spatial_main` *Mussel bioenergetic spatialized model - spatialization loop*

Description

Solves the bioenergetic balance for Mussel

Usage

```
Mussel_spatial_main(userpath, forcings)
```

Arguments

`userpath`	the path where the working folder is located
`forcings`	a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3], particulated organic carbon (POC) concentration [mgC/l] and its characterization in terms of C/P and N/P molar ratios, particulated organic matter (POM) concentration [mgC/l], total suspended solids (TSS) concentration [mg/l]

Value

saves .nc; .csv and .asc outputs in the 'Outputs' folder

`Mussel_spatial_post` *Postprocess the Mussel spatialized model results*

Description

Postprocess the Mussel spatialized model results

Usage

```
Mussel_spatial_post(userpath, output, times, Dates, CS)
```

Arguments

`userpath`	the path where the working folder is located
`output`	output list containing the output of the RK solver
`times`	the vector containing informations on integration extremes
`Dates`	the vector containing the date
`CS`	the commercial size of Mussel

Value

a list containing the weights of the mussel, the excreted CNP, the mussel CNP, temperature limitation functions, metabolic rates, oxygen consumption

---

`Mussel_spatial_pre` *Mussel bioenergetic spatialized model preprocessor*

Description

Mussel bioenergetic spatialized model preprocessor

Usage

```
Mussel_spatial_pre(userpath, forcings)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located
`forcings`	a list containing forcings used by the model

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3], particulated organic carbon (POC) concentration [mgC/l] and its characterization in terms of C/P and N/P molar ratios, particulated organic matter (POM) concentration [mgC/l], total suspended solids (TSS) concentration [mg/l]

`Mussel_spatial_pre_int` *Mussel bioenergetic spatialized model preprocessor - used inside spatialization loop*

Description

Mussel bioenergetic spatialized model preprocessor - used inside spatialization loop

Usage

```
Mussel_spatial_pre_int(userpath, forcings)
```

Arguments

`userpath`	the path where folder containing model inputs and outputs is located
`forcings`	a list containing forcings used by the model

Value

a list containing the time series in the odd positions and realted forcings in the even positions. Forcings returned are: Water temperature [Celsius degrees], Chlorophyll a concentration [mgChl-a/m^3], particulated organic carbon (POC) concentration [mgC/l] and its characterization in terms of C/P and N/P molar ratios, particulated organic matter (POM) concentration [mgC/l], total suspended solids (TSS) concentration [mg/l]

`Mussel_spatial_RKsolver`

Solves the Mussel individual bioenergetic balance with a 4th order Runge Kutta method for spatialized model

---

Description

Solves the Mussel individual bioenergetic balance with a 4th order Runge Kutta method for spatialized model

Usage

```
Mussel_spatial_RKsolver(Param, times, IC, Tint, Phyint, DTint, POCint, Ccont,
  Ncont, Pcont, POMint, TSSint)
```

Arguments

`Param`	a vector containing model parameters
`times`	integration extremes and integration timestep
`IC`	initial condition
`Tint`	the interpolated water temperature time series
`Phyint`	the interpolated phytoplankton time series
`DTint`	the interpolated detritus time series
`POCint`	the interpolated POC time series
`Ccont`	the C/C content of the POC
`Ncont`	the N/C content of POC
`Pcont`	the P/C content of POC
`POMint`	the interpolated POM time series
`TSSint`	the interpolated TSS time series

Value

a list containing the weights of the mussel, the excreted CNP, the mussel CNP, temperature limitation functions, metabolic rates, oxygen consumption

---

Mussel_spatial_skeleton *Creates the folders structure for Mussel spatialized model*

Description

Creates the folders structure for Mussel spatialized model

Usage

```
Mussel_spatial_skeleton(userpath)
```

Arguments

`userpath`	the path where forcing are located

Pop_fun *Function that solves the population dynamics equations including discontinuities*

Description

Function that solves the population dynamics equations including discontinuities

Usage

```
Pop_fun(Nseed, mort, manag, times)
```

Arguments

`Nseed`	number of seeded individuals
`mort`	mortality rate
`manag`	list of management actions (seeded/harvested individuals)
`times`	vector containing informations on integration times

Value

a time series with the number of individuals

Index