Package 'methanogenlab'

April 15, 2022

Type Package

Title Methanogenesis Model

Version 0.1.2

Description Evolves a closed system undergoing methanogenesis.

Imports microbialkitchen, CHNOSZ, mathjaxr

Remotes KopfLab/microbialkitchen

RdMacros mathjaxr

License What license is it under?

Encoding UTF-8

LazyData true

RoxygenNote 7.1.1

R topics documented:

alkalinity *Calculates solution alkalinity*

Description

'alkalinity()' Uses the calculate_closed_system_alkalinity function from Sebastian Kopf's microbialkitchen R package to calculate media alkalinity

Usage

```
alkalinity(
  pH,
  nDIC,
  VolumeSolution,
  VolumeHeadspace,
  temperature,
  K.CO2HCO3,
  K.HCO3CO3
)
```

Arguments

pH	pH of the system.
nDIC	Moles of dissolved inorganic carbon
VolumeSolution	Volume of liquid in the closed system, in liters.
VolumeHeadspace	Volume of gaseous head space in the closed system, in liters.
temperature	Temperature of the system, in Kelvin.
K.CO2HCO3	Equilibrium constant for the dissociation of CO2(aq) to HCO3-(aq).
K.HCO3CO3	Equilibrium constant for the dissociation of HCO3- (aq) to CO3– (aq).

Value

alkalinity of the solution, in millimolar.

aqueous.step *Uses Henry's Law to calculate dissolved gas concentrations*

Description

'aqueous.step()' calculates the aqueous concentration at each step using the current pressure.

Usage

```
aqueous.step(P.step, K)
```

Arguments

P.step	the current pressure in the system.
K	Henry's equilibrium constant of the gas.

Value

Dissolved gas concentration, in molar.

---

biomass.coefficient *Amount of biomass produced for every DIC consumed*

Description

'biomass.coefficient()' calculates the amount of biomass produced for every mol of DIC consumed

Usage

```
biomass.coefficient(biomass.yield, carbon.fraction)
```

Arguments

biomass.yield	mass of dry biomass produced per mol of product.
carbon.fraction	w/w percent C of biomass, expressed as a decimal.

Value

biomass produced, in mol/mol DIC

---

calculate.Kgas *Calculates Henry's constant for a gas*

Description

'calculate.Kgas()' uses the provided Bunsen coefficient and moles per volume of gas to calculate Henry's constant.

Usage

```
calculate.Kgas(bunsen, temperature, pressure)
```

Arguments

bunsen	The Bunsen coefficient for the gas.
temperature	Temperature of the system, in Kelvin.
pressure	The pressure of the system, in atm.

calculate.KH *Calculates equilibrium constants using CHNOSZ*

Description

'calculate.KH()' utilizes the subcrt function from Jeffrey Dick's CHNOSZ R package to calculate equilibrium constants.

Usage

```
calculate.KH(reactants, moles, phases, temperature, pressure)
```

Arguments

reactants	A vector of all components involved in the reaction, both reactants and products.
moles	A vector of the molar coefficients for the reaction, with negative values indicating reactants and positive values indicating products.
phases	A vector of the phases for all components in the reaction, either "aq", "l", or "g".
temperature	Temperature of the system, in Kelvin.
pressure	The pressure of the system, in atm.

Details

For the following chemical equilibrium:

$$CO_2(aq) + H_2O(l) \rightleftharpoons {HCO_3}^{2-}(aq) + H^+(aq)$$

an example can found below

Examples

```
calculate_KH(c("CO2","H2O","HCO3-","H+"),c(-1,-1,1,1),c("aq","l","aq","aq"), 273.15+37,1.7)
```

carbon.yield *Biomass to carbon yield*

Description

'carbon.yield()' converts the known organism-specific biomass yield per mol of product into carbon yield.

Usage

```
carbon.yield(biomass.yield, carbon.fraction)
```

Arguments

`biomass.yield`	mass of dry biomass produced per mol of product.
`carbon.fraction`	w/w percent C of biomass, expressed as a decimal.

Value

carbon yield, in mol C/mol product

`CH4.coefficient` *Amount of CH4 produced for every DIC consumed*

Description

'CH4.coefficient()' calculates the amount of CH4 produced for every mol of DIC consumed.

Usage

```
CH4.coefficient(biomass.yield, carbon.fraction)
```

Arguments

`biomass.yield`	mass of dry biomass produced per mol of product.
`carbon.fraction`	w/w percent C of biomass, expressed as a decimal.

Value

CH4 produced, in mol/mol DIC

`gibbs.step` *Calculates Gibbs free energy*

Description

'gibbs.step()' calculates the Gibbs free energy at each step in the model for a given Q and standard Gibbs free energy.

Usage

```
gibbs.step(standard.gibbs, Q, temperature)
```

Arguments

`standard.gibbs`	The standard Gibbs free energy of formation of the reaction in J/mol
`Q`	Current reaction quotient
`temperature`	Temperature of the system, in Kelvin.

`growth` *Determines the number of new cells generated for a given time step size and*

Description

Determines the number of new cells generated for a given time step size and

Usage

```
growth(
  cells.initial,
  CO2,
  is.CO2.limiting = NA,
  Ks.CO2,
  H2,
  is.H2.limiting = NA,
  Ks.H2,
  umax,
  time.step
)
```

Arguments

`cells.initial`	Number of cells currently in the system.
`CO2`	Current dissolved CO2
`is.CO2.limiting`	Boolean. Determines whether to use the Monod equation for CO2. NA by default (limiting contribution determined by the proportion of current CO2 and Ks.CO2).
`Ks.CO2`	Half-saturation constant for CO2 (microorganism dependent).
`H2`	Current dissolved H2
`is.H2.limiting`	Boolean. Determines whether to use the Monod equation for H2. NA by default (limiting contribution determined by the proportion of current H2 and Ks.H2).
`Ks.H2`	Half-saturation constant for H2 (microorganism dependent).
`umax`	Maximum growth rate for the microorganism, hr^-1
`time.step`	Step size in time, in hours. 0.1 hrs by default.

Value

Number of new cells made

H2.coefficient *Amount of H2 consumed for every DIC consumed*

Description

'H2.coefficient()' calculates the amount of H2 consumed for every mol of DIC consumed.

Usage

```
H2.coefficient(biomass.yield, carbon.fraction)
```

Arguments

`biomass.yield`	mass of dry biomass produced per mol of product. 2.4 g/mol product by default.
`carbon.fraction`	w/w percent C of biomass, expressed as a decimal. 0.44 by default.

Value

H2 consumed, in mol/mol DIC

init *Determines initial conditions from initial inputs*

Description

'init()' sets up the initial environment to be used by the methanogenesis model.

Usage

```
init(
  CH4.initial,
  K.CH4,
  H2.initial,
  K.H2,
  DIC.initial,
  pH.initial,
  K.CO2,
  temperature,
  VolumeSolution,
  VolumeHeadspace,
  inoculum.cell.number,
  biomass.yield,
  carbon.fraction,
  cell.weight,
  K.CO2HCO3,
  K.HCO3CO3
)
```

Arguments

CH4.initial	Concentration of initial dissolved CH4, in molarity.
K.CH4	Henry's constant for CH4. NA by default (calculated by CHNOSZ).
H2.initial	Concentration of initial dissolved H2, in molarity.
K.H2	Henry's constant for H2. NA by default (calculated by CHNOSZ).
DIC.initial	Concentration of initial dissolved inorganic carbon, in molarity.
pH.initial	initial pH.
K.CO2	Henry's constant for CH4. NA by default (calculated by CHNOSZ).
temperature	Temperature of the system, in Kelvin.
VolumeSolution	Volume of liquid in the closed system, in liters.
VolumeHeadspace	Volume of gaseous head space in the closed system, in liters.
inoculum.cell.number	Initial number of cells. 1e6 by default.
biomass.yield	mass of dry biomass produced per mol of product. 2.4 g/mol product by default.
carbon.fraction	w/w percent C of biomass, expressed as a decimal. 0.44 by default.
cell.weight	individual cell mass, in grams. 30e-15 by default.
K.CO2HCO3	Equilibrium constant for the dissociation of CO2(aq) to HCO3-(aq). NA by default (calculated by CHNOSZ).
K.HCO3CO3	Equilibrium constant for the dissociation of HCO3- (aq) to CO3– (aq). NA by default (calculated by CHNOSZ).

Value

A data frame to be used for the methanogenesis function.

`methanogenesis.DIC` *Steps through DIC consumption during a methanogenesis reaction*

Description

'methanogenesis.DIC()' calculates CH4 produced, H2 consumed, CO2 consumed, pH, and Gibbs free energy changes as dissolved inorganic carbon is consumed.

Usage

```
methanogenesis.DIC(
  CH4.initial,
  K.CH4 = NA,
  H2.initial,
  K.H2 = NA,
  DIC.initial,
  pH.initial,
  K.CO2 = NA,
  temperature,
```

```
  VolumeSolution,
  VolumeHeadspace,
  K.CO2HCO3 = NA,
  K.HCO3CO3 = NA,
  delta.DIC = 1e-04,
  inoculum.cell.number = 1e+06,
  biomass.yield = 2.4,
  carbon.fraction = 0.44,
  cell.weight = 3e-14
)
```

Arguments

CH4.initial	Concentration of initial dissolved CH4, in molarity.
K.CH4	Henry's constant for CH4. NA by default (calculated by CHNOSZ).
H2.initial	Concentration of initial dissolved H2, in molarity.
K.H2	Henry's constant for H2. NA by default (calculated by CHNOSZ).
DIC.initial	Concentration of initial dissolved inorganic carbon, in molarity.
pH.initial	initial pH.
K.CO2	Henry's constant for CH4. NA by default (calculated by CHNOSZ).
temperature	Temperature of the system, in Kelvin.
VolumeSolution	Volume of liquid in the closed system, in liters.
VolumeHeadspace	Volume of gaseous head space in the closed system, in liters.
K.CO2HCO3	Equilibrium constant for the dissociation of CO2(aq) to HCO3-(aq). NA by default (calculated by CHNOSZ).
K.HCO3CO3	Equilibrium constant for the dissociation of HCO3- (aq) to CO3– (aq). NA by default (calculated by CHNOSZ).
delta.DIC	step size, in millimolar DIC. 0.1 mM by default.
inoculum.cell.number	Initial number of cells. 1e6 by default.
biomass.yield	mass of dry biomass produced per mol of product. 2.4 g/mol product by default.
carbon.fraction	w/w percent C of biomass, expressed as a decimal. 0.44 by default.
cell.weight	individual cell mass, in grams. 30e-15 by default.

Value

A data frame of the model results

Examples

```
methanogenesis(CH4.initial = 1e-6,H2.initial = 5e-4,DIC.initial = 3.2e-3,pH.initial = 7.5,temperature = 273.15
```

`methanogenesis.time` *Steps through time during a methanogenesis reaction*

Description

'methanogenesis.time()' calculates CH4 produced, H2 consumed, CO2 consumed, pH, and Gibbs free energy changes as dissolved inorganic carbon is consumed over time.

Usage

```
methanogenesis.time(
  CH4.initial,
  K.CH4 = NA,
  H2.initial,
  K.H2 = NA,
  is.H2.limiting = NA,
  Ks.H2,
  DIC.initial,
  pH.initial,
  K.CO2 = NA,
  is.CO2.limiting = NA,
  Ks.CO2,
  umax,
  temperature,
  VolumeSolution,
  VolumeHeadspace,
  K.CO2HCO3 = NA,
  K.HCO3CO3 = NA,
  time.step = 0.1,
  total.time = 5,
  inoculum.cell.number = 1e+06,
  biomass.yield = 2.4,
  carbon.fraction = 0.44,
  cell.weight = 3e-14
)
```

Arguments

`CH4.initial`	Concentration of initial dissolved CH4, in molarity.
`K.CH4`	Henry's constant for CH4. NA by default (calculated by CHNOSZ).
`H2.initial`	Concentration of initial dissolved H2, in molarity.
`K.H2`	Henry's constant for H2. NA by default (calculated by CHNOSZ).
`is.H2.limiting`	Boolean. Determines whether to use the Monod equation for H2. NA by default (limiting contribution determined by the proportion of current H2 and Ks.H2).
`Ks.H2`	Half-saturation constant for H2 (microorganism dependent).
`DIC.initial`	Concentration of initial dissolved inorganic carbon, in molarity.
`pH.initial`	initial pH.
`K.CO2`	Henry's constant for CH4. NA by default (calculated by CHNOSZ).

`is.CO2.limiting`
: Boolean. Determines whether to use the Monod equation for CO2. NA by default (limiting contribution determined by the proportion of current CO2 and Ks.CO2).

`Ks.CO2`
: Half-saturation constant for CO2 (microorganism dependent).

`umax`
: Maximum growth rate for the microorganism, hr^-1

`temperature`
: Temperature of the system, in Kelvin.

`VolumeSolution`
: Volume of liquid in the closed system, in liters.

`VolumeHeadspace`
: Volume of gaseous head space in the closed system, in liters.

`K.CO2HCO3`
: Equilibrium constant for the dissociation of CO2(aq) to HCO3-(aq). NA by default (calculated by CHNOSZ).

`K.HCO3CO3`
: Equilibrium constant for the dissociation of HCO3- (aq) to CO3– (aq). NA by default (calculated by CHNOSZ).

`time.step`
: Step size in time, in hours. 0.1 hrs by default.

`total.time`
: Length of time the model will run, in hours. Note: the model may break before the total.time is reached if H2 or DIC runs out.

`inoculum.cell.number`
: Initial number of cells. 1e6 by default.

`biomass.yield`
: mass of dry biomass produced per mol of product. 2.4 g/mol product by default.

`carbon.fraction`
: w/w percent C of biomass, expressed as a decimal. 0.44 by default.

`cell.weight`
: individual cell mass, in grams. 30e-15 by default.

Value

A data frame of the model results

Examples

```
methanogenesis(CH4.initial = 1e-6,H2.initial = 5e-4,DIC.initial = 3.2e-3,pH.initial = 7.5,temperature = 273.15
```

`PCO2` *Calculates CO2 pressure in the head space of the closed system.*

Description

'PCO2()' Uses the calculate_closed_system_pCO2 function from Sebastian Kopf's microbialkitchen R package to calculate pressure of CO2 gas.

Usage

```
PCO2(
  pH,
  nDIC,
  VolumeSolution,
  VolumeHeadspace,
  temperature,
  K.CO2HCO3,
  K.HCO3CO3
)
```

Arguments

pH	pH of the system.
nDIC	Moles of dissolved inorganic carbon
VolumeSolution	Volume of liquid in the closed system, in liters.
VolumeHeadspace	Volume of gaseous head space in the closed system, in liters.
temperature	Temperature of the system, in Kelvin.
K.CO2HCO3	Equilibrium constant for the dissociation of CO2(aq) to HCO3-(aq).
K.HCO3CO3	Equilibrium constant for the dissociation of HCO3- (aq) to CO3– (aq).

Value

CO2 pressure in the head space, in atm.

---

pH *Calculates pH of the solution*

Description

'pH()' Uses the calculate_closed_system_pH function from Sebastian Kopf's microbialkitchen R package to calculate pH.

Usage

```
pH(
  nDIC,
  VolumeSolution,
  VolumeHeadspace,
  temperature,
  alkalinity,
  K.CO2HCO3,
  K.HCO3CO3
)
```

Arguments

nDIC	Moles of dissolved inorganic carbon
VolumeSolution	Volume of liquid in the closed system, in liters.
VolumeHeadspace	Volume of gaseous head space in the closed system, in liters.
temperature	Temperature of the system, in Kelvin.
alkalinity	initial alkalinity of the system, in millimolar.
K.CO2HCO3	Equilibrium constant for the dissociation of CO2(aq) to HCO3-(aq).
K.HCO3CO3	Equilibrium constant for the dissociation of HCO3- (aq) to CO3– (aq).

Value

pH of the solution.

---

pressure.step *Calculates head space pressure*

Description

'pressure.step()' calculates head space partial pressure from the total moles of a particular gas in the system using Henry's law and the Ideal Gas law.

Usage

```
pressure.step(n.total.step, K, VolumeSolution, VolumeHeadspace, temperature)
```

Arguments

n.total.step	The total number of moles of a gas in the system.
K	Henry's constant of the gas.
VolumeSolution	Volume of liquid in the closed system, in liters.
VolumeHeadspace	Volume of gaseous head space in the closed system, in liters.
temperature	Temperature of the system, in Kelvin.

Value

Partial pressure of the head space.

`standard.gibbs` *Calculates standard Gibbs free energy for a given temperature and pressure*

Description

'standard.gibbs()' determines the standard Gibbs free energy to be used in 'gibbs.step()'

Usage

```
standard.gibbs(reactants, moles, phases, temperature, pressure = 1)
```

Arguments

`reactants`	A vector of all components involved in the reaction, both reactants and products.
`moles`	A vector of the molar coefficients for the reaction, with negative values indicating reactants and positive values indicating products.
`phases`	A vector of the phases for all components in the reaction, either "aq", "l", or "g".
`temperature`	Temperature of the system, in Kelvin.
`pressure`	The pressure of the system, in atm.

Index