

Exercise 3: General solution of stoichiometric equations

ETH Zurich Course 701-0426-00L: Modelling Aquatic Ecosystems (Schuwirth)

March 20, 2024

Goals:

- Understand the concepts of stoichiometric calculations introduced in section 4.3.
- Be able to do stoichiometric calculations based on the general solution of stoichiometric equations (section 4.3.3) using the R package `stoichcalc`.
- Learn to be attentive to the results and to error messages and to correct the code accordingly (execute the code piece by piece and proceed only if the results are as expected and everything was correct).

Task 1: Simple stoichiometry using the `stoichcalc`-package

Study the implementation of the simple stoichiometry of algae growth as formulated in exercise 1 based on the `stoichcalc`-package.

```
# load required packages:  
if ( !require("stoichcalc") ) {install.packages("stoichcalc"); library(stoichcalc) }
```

To use the `stoichcalc`-package, we first need to construct a composition matrix.

```
HP04 <- c(P = 1) # gP/gHPO4-P  
ALG <- c(P = 0.01) # gP/gALG  
  
subst.comp <- list(HP04 = HP04,  
                  ALG = ALG)  
  
alpha.1 <- calc.comp.matrix(subst.comp)  
  
print(alpha.1)
```

Next, we use the function `calc.stoich.coef` to calculate the stoichiometric coefficients of the processes that describe growth and respiration of algae.

```
nu.gro.ALG <-  
  calc.stoich.coef(alpha = alpha.1,  
                  name = "growth.ALG",  
                  subst = c("HP04", "ALG"),  
                  subst.norm = "ALG",  
                  nu.norm = 1)  
  
nu.resp.ALG <-  
  calc.stoich.coef(alpha = alpha.1,  
                  name = "resp.ALG",  
                  subst = c("HP04", "ALG"),  
                  subst.norm = "ALG",  
                  nu.norm = -1)
```

```

nu.1 <- rbind(nu.gro.ALG,
              nu.resp.ALG)

print(nu.1)

```

Task 2: Stoichiometry of a complex process model

Study the implementation of the stoichiometry of a process model of algae and zooplankton growth, respiration and death given below.

First we set the parameter values:

```

# Complex stoichiometry
# ~~~~~

param <- list(a.O.ALG      = 0.50,      # gO/gALG
              a.H.ALG      = 0.07,      # gH/gALG
              a.N.ALG      = 0.06,      # gN/gALG
              a.P.ALG      = 0.005,     # gP/gALG
              a.O.ZOO      = 0.50,      # gO/gZOO
              a.H.ZOO      = 0.07,      # gH/gZOO
              a.N.ZOO      = 0.06,      # gN/gZOO
              a.P.ZOO      = 0.01,      # gP/gZOO
              a.O.POM      = 0.40,      # gO/gPOM
              a.H.POM      = 0.07,      # gH/gPOM
              a.N.POM      = 0.04,      # gN/gPOM
              a.P.POM      = 0.007,     # gP/gPOM
              Y.ZOO        = 0.2,       # gZOO/gALG
              f.e          = 0.4)       # gPOM/gALG

# choose carbon fractions to guarantee that the fractions sum to unity:
param$a.C.ALG = 1-(param$a.O.ALG+param$a.H.ALG+param$a.N.ALG+param$a.P.ALG)
param$a.C.ZOO = 1-(param$a.O.ZOO+param$a.H.ZOO+param$a.N.ZOO+param$a.P.ZOO)
param$a.C.POM = 1-(param$a.O.POM+param$a.H.POM+param$a.N.POM+param$a.P.POM)

```

Next, we construct a composition matrix containing all the involved substances and organisms.

```

NH4  <- c(H      = 4,      # molH/molNH4
          N      = 1,      # molN/molNH4
          charge = 1)      # chargeunits/molNH4
NO3  <- c(O      = 3,      # molO/molNO3
          N      = 1,      # molN/molNO3
          charge = -1)     # chargeunits/molNO3
HPO4 <- c(O      = 4,      # molO/molHPO4
          H      = 1,      # molH/molHPO4
          P      = 1,      # molP/molHPO4
          charge = -2)     # chargeunits/molHPO4
HCO3 <- c(C      = 1,      # molC/molHCO3
          O      = 3,      # molO/molHCO3
          H      = 1,      # molH/molHCO3
          charge = -1)     # chargeunits/molHCO3
O2   <- c(O      = 2)      # molO/molO2
H    <- c(H      = 1)      # molH/molH

```

```

      charge = 1)           # chargeunits/molH
H2O   <- c(O = 1,         # molO/molH2O
          H = 2)         # molH/molH2O
ALG   <- c(C = param$a.C.ALG/12, # molC/gALG
          O = param$a.O.ALG/16, # molO/gALG
          H = param$a.H.ALG,    # molH/gALG
          N = param$a.N.ALG/14, # molN/gALG
          P = param$a.P.ALG/31) # molP/gALG
ZOO   <- c(C = param$a.C.ZOO/12, # molC/gZOO
          O = param$a.O.ZOO/16, # molO/gZOO
          H = param$a.H.ZOO,    # molH/gZOO
          N = param$a.N.ZOO/14, # molN/gZOO
          P = param$a.P.ZOO/31) # molP/gZOO
POM   <- c(C = param$a.C.POM/12, # molC/gPOM
          O = param$a.O.POM/16, # molO/gPOM
          H = param$a.H.POM,    # molH/gPOM
          N = param$a.N.POM/14, # molN/gPOM
          P = param$a.P.POM/31) # molP/gPOM

```

```

subst.comp <- list(NH4 = NH4,
                  NO3  = NO3,
                  HPO4 = HPO4,
                  HCO3 = HCO3,
                  O2   = O2,
                  H    = H,
                  H2O  = H2O,
                  ALG  = ALG,
                  ZOO  = ZOO,
                  POM  = POM)

```

```
alpha.2 <- calc.comp.matrix(subst.comp)
```

```
print(alpha.2)
```

Next, we define the yields for algae and zooplankton such that death does not require nutrients (note: oxygen content of POM was reduced to avoid need of oxygen).

```

param$Y.ALG.death = min(1,param$a.N.ALG/param$a.N.POM,param$a.P.ALG/param$a.P.POM)
param$Y.ZOO.death = min(1,param$a.N.ZOO/param$a.N.POM,param$a.P.ZOO/param$a.P.POM)

```

```

print(param$Y.ALG.death)
print(param$Y.ZOO.death)

```

Finally, we calculate the stoichiometric coefficients for the involved processes: Growth of algae with ammonium or nitrate as nitrogen sources, growth of zooplankton, and respiration and death processes for both zooplankton and algae.

```
# Calculate stoichiometric coefficients of selected processes:
```

```

nu.gro.ALG.NH4 <-
  calc.stoich.coef(alpha = alpha.2,
                  name   = "gro.ALG.NH4",
                  subst  = c("NH4", "HPO4", "HCO3", "O2", "H", "H2O", "ALG"),
                  subst.norm = "ALG",

```

```

        nu.norm      = 1)

nu.gro.ALG.NO3 <-
  calc.stoich.coef(alpha      = alpha.2,
                  name       = "gro.ALG.NO3",
                  subst      = c("NO3", "HP04", "HCO3", "O2", "H", "H2O", "ALG"),
                  subst.norm  = "ALG",
                  nu.norm     = 1)

nu.resp.ALG <-
  calc.stoich.coef(alpha      = alpha.2,
                  name       = "resp.ALG",
                  subst      = c("NH4", "HP04", "HCO3", "O2", "H", "H2O", "ALG"),
                  subst.norm  = "ALG",
                  nu.norm     = -1)

nu.death.ALG <-
  calc.stoich.coef(alpha      = alpha.2,
                  name       = "death.ALG",
                  subst      = c("NH4", "HP04", "HCO3", "O2", "H", "H2O", "ALG", "POM"),
                  subst.norm  = "ALG",
                  nu.norm     = -1,
                  constraints = list(c("ALG" = param$Y.ALG.death,
                                      "POM" = 1)))

nu.gro.ZOO <-
  calc.stoich.coef(alpha      = alpha.2,
                  name       = "groZOO",
                  subst      = c("NH4", "HP04", "HCO3", "O2", "H", "H2O", "ALG", "ZOO", "POM"),
                  subst.norm  = "ZOO",
                  nu.norm     = 1,
                  constraints = list(c("ZOO" = 1,
                                      "ALG" = param$Y.ZOO),
                                    c("POM" = 1,
                                      "ALG" = param$f.e)))

nu.resp.ZOO <-
  calc.stoich.coef(alpha      = alpha.2,
                  name       = "resp.ZOO",
                  subst      = c("NH4", "HP04", "HCO3", "O2", "H", "H2O", "ZOO"),
                  subst.norm  = "ZOO",
                  nu.norm     = -1)

nu.death.ZOO <-
  calc.stoich.coef(alpha      = alpha.2,
                  name       = "death.ZOO",
                  subst      = c("NH4", "HP04", "HCO3", "O2", "H", "H2O", "ZOO", "POM"),
                  subst.norm  = "ZOO",
                  nu.norm     = -1,
                  constraints = list(c("ZOO" = param$Y.ZOO.death,
                                      "POM" = 1)))

```

```

nu.2 <- rbind(nu.gro.ALG.NH4,
             nu.gro.ALG.NO3,
             nu.resp.ALG,
             nu.death.ALG,
             nu.gro.ZOO,
             nu.resp.ZOO,
             nu.death.ZOO)

```

```
print(round(nu.2,3))
```

To conclude, study the following (correct or incorrect) examples of calls to the function ‘calc.stoich.coef’ and the corresponding outputs.

Correct stoichiometry of growth of zooplankton:

```

round(calc.stoich.coef(alpha      = alpha.2,
                      name       = "groZOO",
                      subst      = c("NH4", "HPO4", "HCO3", "O2", "H", "H2O", "ALG", "ZOO", "POM"),
                      subst.norm = "ZOO",
                      nu.norm    = 1,
                      constraints = list(c("ZOO" = 1,
                                           "ALG" = param$Y.ZOO),
                                         c("POM" = 1,
                                           "ALG" = param$f.e))),3)

```

Missing constraint error for the stoichiometry of growth of zooplankton:

```

round(calc.stoich.coef(alpha      = alpha.2,
                      name       = "groZOO",
                      subst      = c("NH4", "HPO4", "HCO3", "O2", "H", "H2O", "ALG", "ZOO", "POM"),
                      subst.norm = "ZOO",
                      nu.norm    = 1,
                      constraints = list(c("ZOO" = 1,
                                           "ALG" = param$Y.ZOO))),3)

```

Too many substances error for the stoichiometry of growth of zooplankton:

```

round(calc.stoich.coef(alpha      = alpha.2,
                      name       = "groZOO",
                      subst      = c("NH4", "NO3", "HPO4", "HCO3", "O2", "H", "H2O", "ALG", "ZOO", "POM"),
                      subst.norm = "ZOO",
                      nu.norm    = 1,
                      constraints = list(c("ZOO" = 1,
                                           "ALG" = param$Y.ZOO),
                                         c("POM" = 1,
                                           "ALG" = param$f.e))),3)

```

Missing substance error for the stoichiometry of growth of zooplankton:

```

round(calc.stoich.coef(alpha      = alpha.2,
                      name       = "groZOO",
                      subst      = c("HPO4", "HCO3", "O2", "H", "H2O", "ALG", "ZOO", "POM"),
                      subst.norm = "ZOO",
                      nu.norm    = 1,
                      constraints = list(c("ZOO" = 1,

```

```
"ALG" = param$Y.Z00),  
c("POM" = 1,  
  "ALG" = param$f.e)),3)
```

Task 3: Homework: Extend the process stoichiometry to sulfur

Extend the process model implemented in task 2 to the consideration of sulfur (S), assuming the same content of 0.3% of S in algae, zooplankton and particulate organic matter (decrease the carbon content by 0.3%) and the uptake and release of sulfur in the form of sulfate SO_4^{2-} (release may also be in the form of hydrogen sulfide, H_2S , which later is oxidized to sulfate; we aggregate these processes).

Theory questions

1. How do you find out, whether you need additional constraints to elemental mass balance and charge? What could be a drawback of the simplified approach to this question?
2. Where to get the required additional constraints from (if needed)?
3. Why do we add H^+ , but not OH^- to the compounds considered for the calculation of stoichiometric coefficients?