Exercise 2: Lake Phyto- and Zooplankton Model

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

March 6, 2024

Goals:

- Understand the process table notation introduced in section 4.1 of the manuscript.
- Understand the formulation of biogeochemical and ecological process rates introduced in section 4.2.
- Increase your familiarity with the R package `ecosim`.
- Understand the behaviour of the solutions of the lake phyto- and zooplankton model described in section 11.2 under constant and seasonally varying environmental conditions.

Task 1: Model with constant driving forces

Study the implementation of the model described in section 11.2 for constant driving forces. Run the model implementation below and interpret the behavior of the solutions.

```
# Model with constant driving forces
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

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

```
## Loading required package: deSolve
```

```
## Warning: package 'deSolve' was built under R version 4.3.2
```

```
if ( !require("ecosim") ) {install.packages("ecosim"); library("ecosim") }
```

```
## Loading required package: ecosim
```

```
## Warning: package 'ecosim' was built under R version 4.3.2
```

```
## Loading required package: stoichcalc
```

```
# definition of model parameters:
param    <- list(k.gro.ALG   = 0.5,      # 1/d
                 k.gro.ZOO   = 0.4,      # m3/gDM/d
                 k.death.ALG = 0.1,      # 1/d
                 k.death.ZOO = 0.05,     # 1/d
                 K.HPO4      = 0.002,    # gP/m3
                 Y.ZOO       = 0.2,      # gDM/gDM
                 alpha.P.ALG = 0.003,    # gP/gDM
```

```
                A           = 5e+006,   # m2
                h.epi       = 5,        # m
                Q.in        = 5,        # m3/s
                C.HPO4.in   = 0.04,     # gP/m3
                C.HPO4.ini  = 0.04,     # gP/m3
                C.ALG.ini   = 0.1,      # gDM/m3
                C.ZOO.ini   = 0.1)      # gDM/m3
```

a) Next, we define the processes of growth and death of algae and zooplankton as objects of the class `process`}of the packageecosim. Each process is defined by its name, rate, and stoichiometry. The rate is defined as an expression that can use parameters (defined for the object of classsystembelow), concentrations defined in objects of classreactor} that are part of the object of class `system`}‘. To define the stoichiometry, a named list of expressions must be provided that identifies the substance or organism concentrations as the names and contains the stoichiometric coefficients as expressions.

```
# definition of transformation processes

# growth of algae:

gro.ALG   <- new(Class  = "process",
                 name   = "Growth of algae",
                 rate   = expression(k.gro.ALG
                                     *C.HPO4/(K.HPO4+C.HPO4)
                                     *C.ALG),
                 stoich = list(C.ALG  = expression(1),              # gDM/gDM
                               C.HPO4 = expression(-alpha.P.ALG)))  # gP/gDM

# death of algae: TO BE COMPLETED

death.ALG <- new(Class = "process",
                 name   = "Death of algae",
                 rate   = expression(k.death.ALG*C.ALG),
                 stoich = list(C.ALG  = expression(-1)))            # gDM/gDM

# growth of zooplankton:

gro.ZOO   <- new(Class  = "process",
                 name   = "Growth of zooplankton",
                 rate   = expression(k.gro.ZOO
                                     *C.ALG
                                     *C.ZOO),
                 stoich = list(C.ZOO  = expression(1),              # gDM/gDM
                               C.ALG  = expression(-1/Y.ZOO)))      # gP/gDM

# death of zooplankton: TO BE COMPLETED

death.ZOO <- new(Class  = "process",
                 name   = "Death of zooplankton",
                 rate   = expression(k.death.ZOO*C.ZOO),
                 stoich = list(C.ZOO  = expression(-1)))            # gDM/gDM
```

b) Next, we define the mixed box describing the epliminion of the lake as an object of the class `reactor`

of the package `ecosim`.

```
# definition of reactor to describe the epilimnion of the lake: TO BE COMPLETED

epilimnion <-
  new(Class            = "reactor",
      name             = "Epilimnion",
      volume.ini       = expression(A*h.epi),
      conc.pervol.ini  = list(C.HPO4 = expression(C.HPO4.ini),     # gP/m3
                              C.ALG  = expression(C.ALG.ini),      # gDM/m3
                              C.ZOO  = expression(C.ZOO.ini)),     # gDM/m3
      inflow           = expression(Q.in*86400),                   # m3/d
      inflow.conc      = list(C.HPO4 = expression(C.HPO4.in),
                              C.ALG  = 0,
                              C.ZOO  = 0),
      outflow          = expression(Q.in*86400),
      processes        = list(gro.ALG,death.ALG,gro.ZOO,death.ZOO))
```

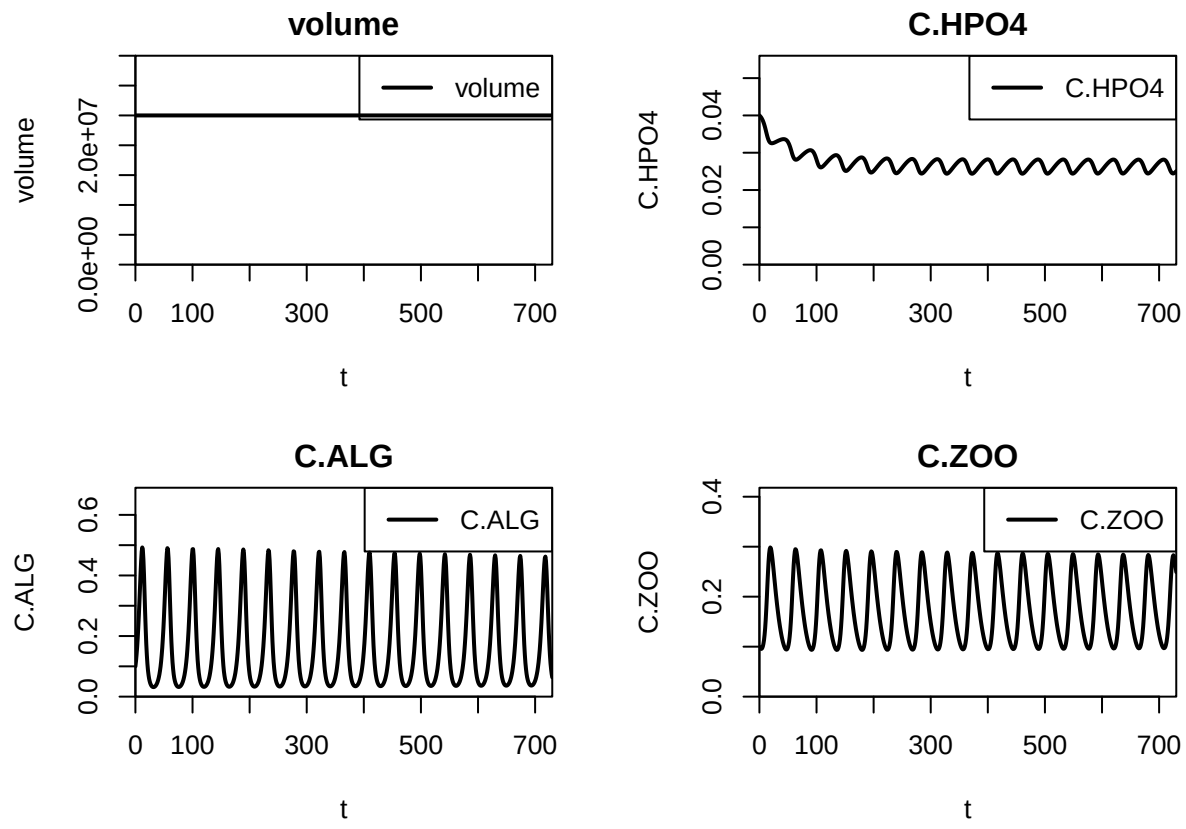
Finally, we combine the reactor, the parameters, and the desired output times in an object of class `system` of the package `ecoval`.

```
# definition of the system consisting of a single reactor:

# observe the time frame of the simulation
system.11.2.a <- new(Class    = "system",
                     name     = "Lake",
                     reactors = list(epilimnion),
                     param    = param,
                     t.out    = seq(0,2*365,by=1))
```

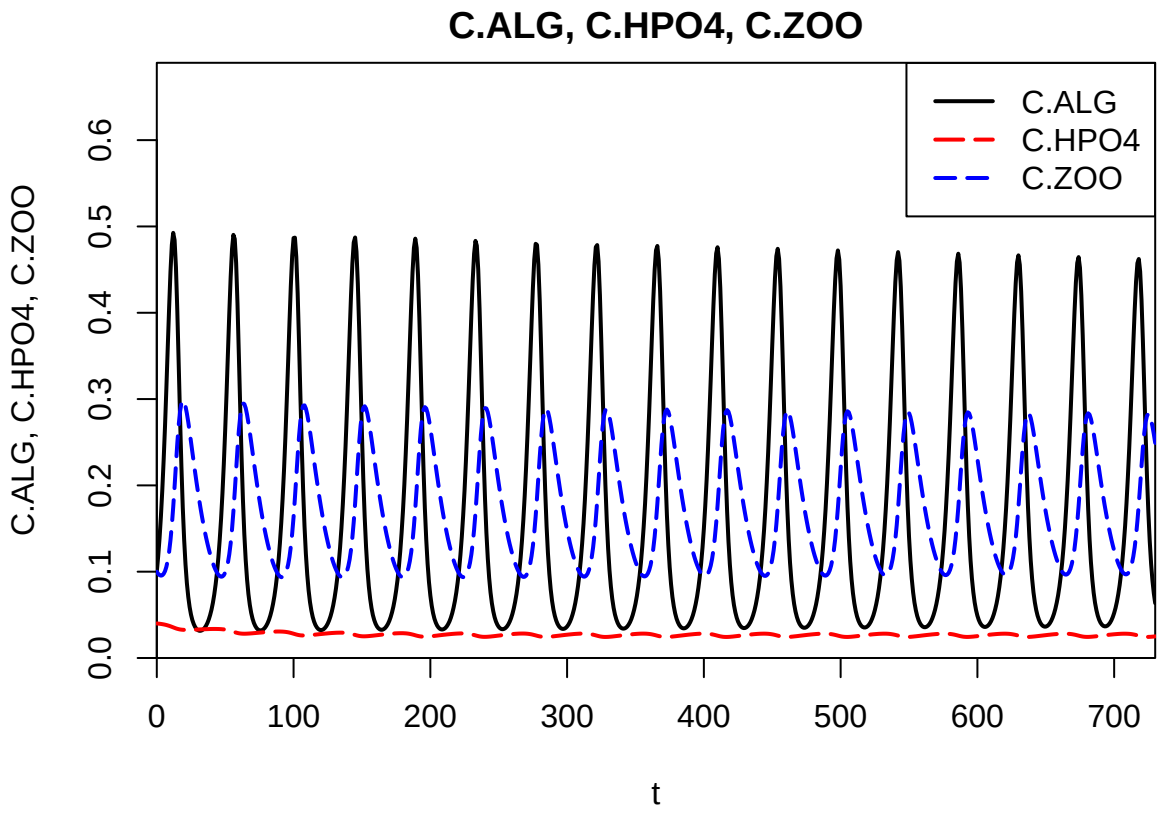
Note that this object contains all definitions of the configuration of reactors (in this case just a single one), the processes active in each reactor, the model parameters, and the output time points. Any simulations carried out will refer to the definitions in this object, and not to the external variables that we used to set up the elements of the system.

c) Perform a simulation and store the results by using the function `calcres` of the package `ecosim`:

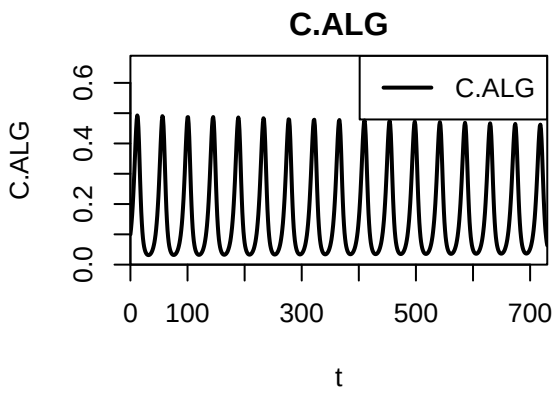
```
# perform simulation:

res.11.2.a <- calcres(system.11.2.a)

# plot results with default options:

plotres(res.11.2.a)
```

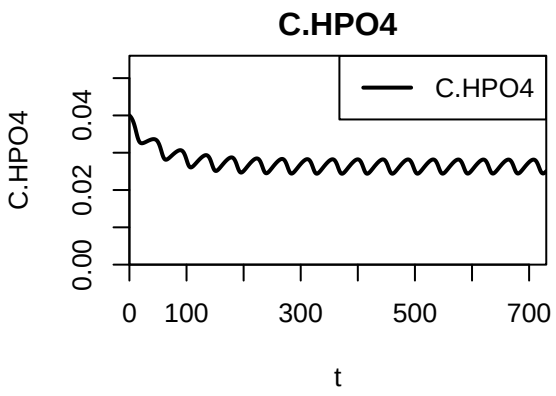


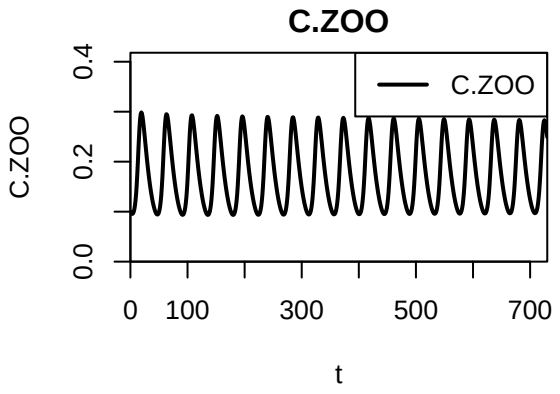

```
# variables in a vector 'c()' are plotted in the same graph

plotres(res=res.11.2.a,colnames=c("C.ALG", "C.HPO4","C.ZOO"))
```

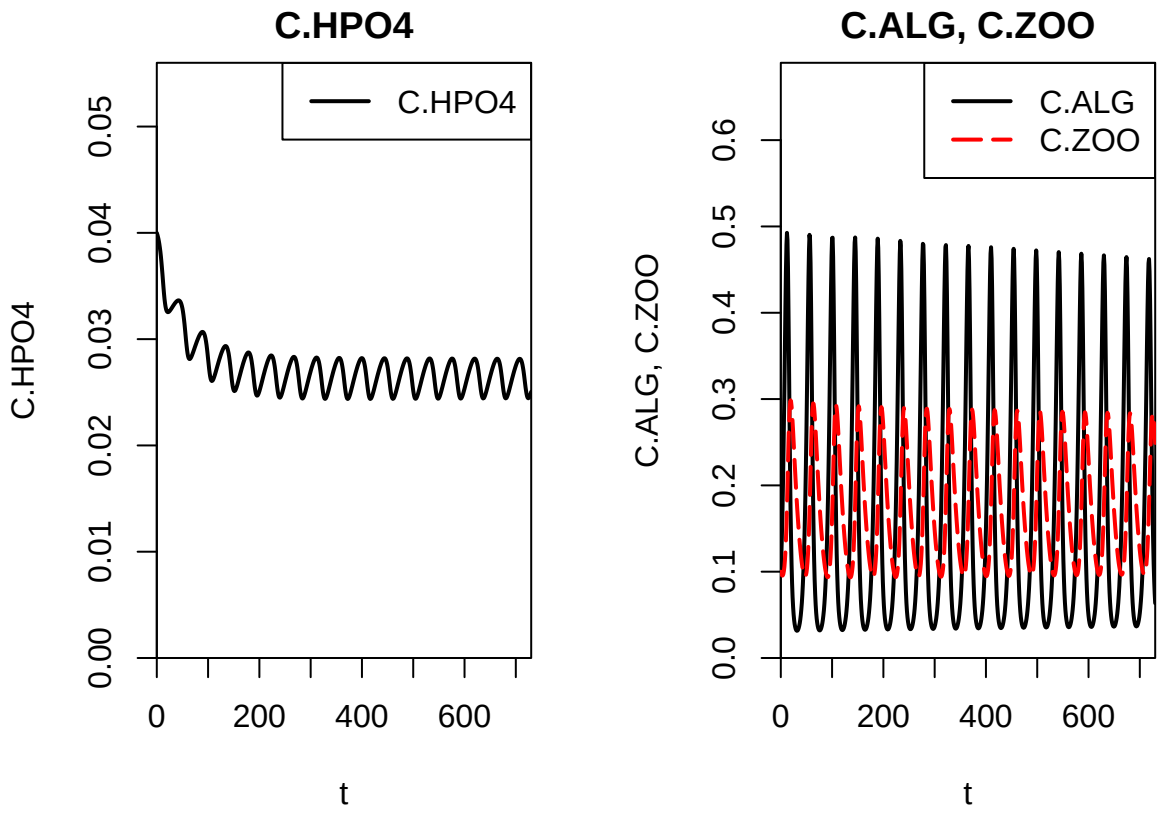



```
# variables in a list 'list()' are plotted in different graphs

plotres(res=res.11.2.a,colnames=list("C.ALG", "C.HPO4", "C.ZOO"))

# combination of the two

plotres(res=res.11.2.a,colnames=list("C.HPO4",c("C.ALG", "C.ZOO")))
```


C.ALG
C.ALG
0.0
0.2
0.4
0.6
0
100
300
500
700
t


C.HPO4
C.HPO4
0.00
0.02
0.04
0
100
300
500
700
t


C.ZOO
C.ZOO
0.0
0.2
0.4
0
100
300
500
700
t




```
# plot and save as pdf

plotres(res      = res.11.2.a,
        colnames = list("C.HPO4",c("C.ALG", "C.ZOO")),
        file     = "exercise_2_results_a.pdf",
        width    = 10,
        height   = 5)
```

```
## pdf
##   2
```

Task 2: Model with seasonally varying driving forces

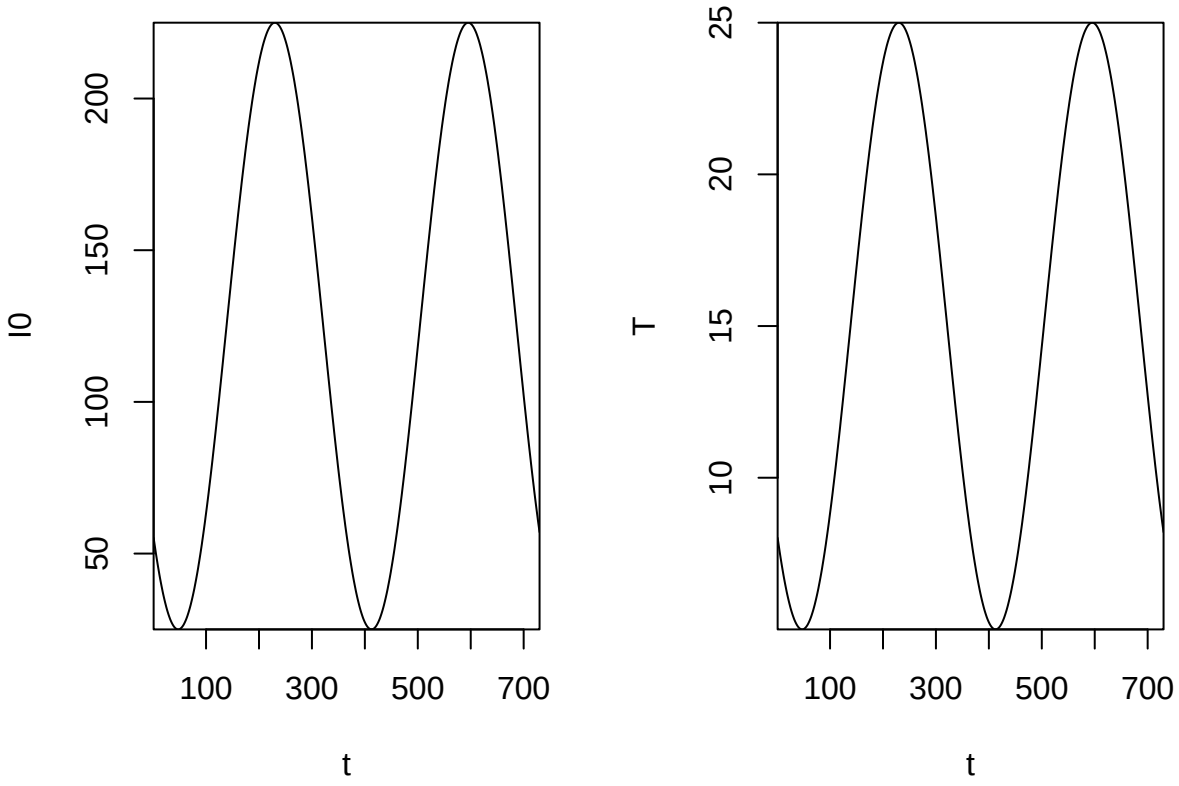
Study the implementation of the model extension to seasonally varying driving forces. First, run the model implementation below.

We adapt system definitions:

```
# Model with seasonally varying driving forces
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

# copy the previous system definition:
system.11.2.b <- system.11.2.a

# extend model parameters:
```

```
param  <- c(param,
            list(beta.ALG    = 0.046,    # 1/degC
                 beta.ZOO    = 0.08,     # 1/degC
                 T0          = 20,       # degC
                 K.I         = 30,       # W/m2
                 lambda.1    = 0.10,     # 1/m
                 lambda.2    = 0.10,     # m2/gDM
                 t.max       = 230,      # d
                 I0.min      = 25,       # W/m2
                 I0.max      = 225,      # W/m2
                 T.min       = 5,        # degC
                 T.max       = 25))      # degC

# extend growth of algae and zooplankton by considering environmental factors:
gro.ALG.ext <-
   new(Class  = "process",
       name   = "Growth of algae extended",
       rate   = expression(k.gro.ALG
                           *exp(beta.ALG*(T-T0))
                           *C.HPO4/(K.HPO4+C.HPO4)
                           *log((K.I+I0)
                                /(K.I+I0*exp(-(lambda.1+lambda.2*C.ALG)*h.epi)))
                            /((lambda.1+lambda.2*C.ALG)*h.epi)
                           *C.ALG),
       stoich = list(C.ALG  = 1,                              # gDM/gDM
                     C.HPO4 = expression(-alpha.P.ALG)))      # gP/gDM

gro.ZOO.ext <-
             new(Class  = "process",
                 name   = "Growth of zooplankton",
                 rate   = expression(k.gro.ZOO
                                     *exp(beta.ZOO*(T-T0))
                                     *C.ALG
                                     *C.ZOO),
                 stoich = list(C.ZOO  = expression(1),              # gDM/gDM
                               C.ALG  = expression(-1/Y.ZOO)))      # gP/gDM

# re-define processes in the reactor "epilimnion":
epilimnion@processes <- list(gro.ALG.ext,death.ALG,gro.ZOO.ext,death.ZOO)

# make environmental conditions (light and temperature) time dependent:
epilimnion@cond <- list(I0 = expression(0.5*(I0.min+I0.max)+
                                        0.5*(I0.max-I0.min)*
                                        cos(2*pi/365.25*(t-t.max))),    # W/m2
                        T  = expression(0.5*(T.min+T.max)+
                                        0.5*(T.max-T.min)*
                                        cos(2*pi/365.25*(t-t.max))))  # degC

# plot the environmental conditions
t <-  seq(1,2*365) # for two years
I0 <- numeric(0)
T <- numeric(0)
for(i in 1:length(t))
```

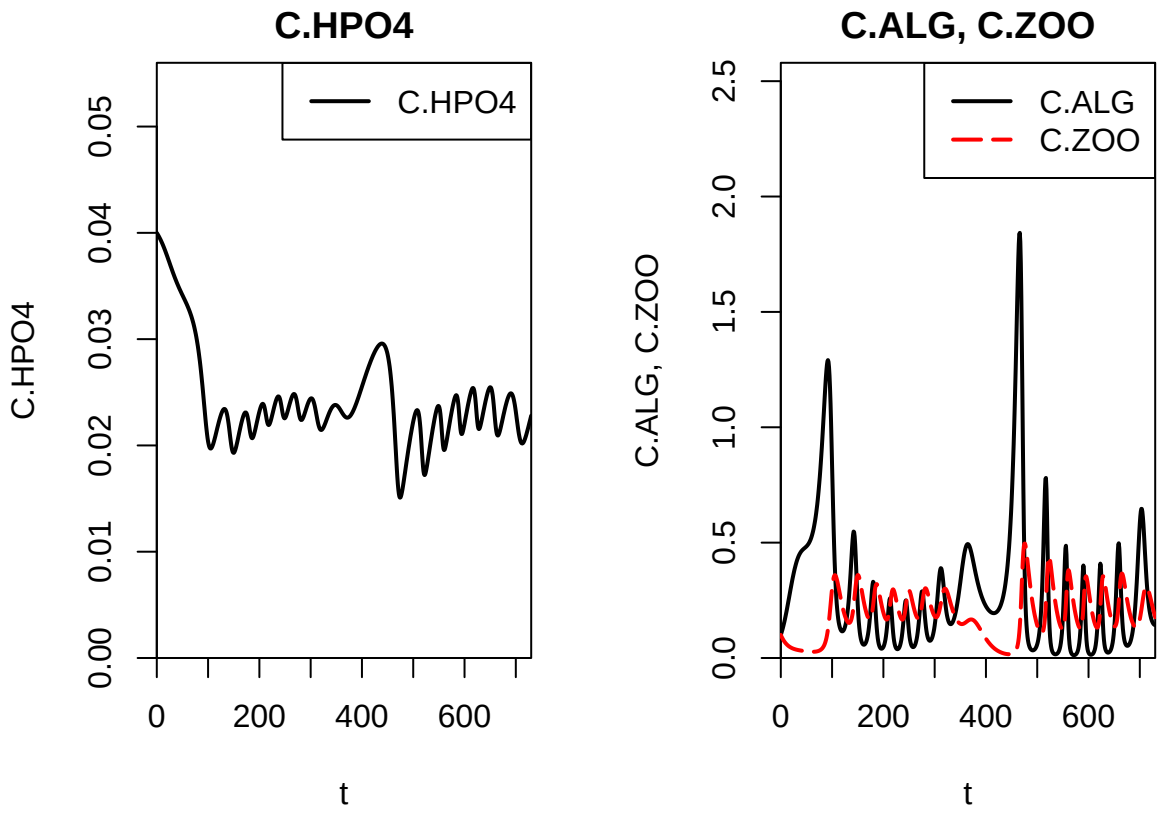
```
{
  I0[i] <- eval(epilimnion@cond$I0, envir=c(param, t=t[i]))
  T[i]  <- eval(epilimnion@cond$T, envir=c(param, t=t[i]))
}
par(mfrow=c(1,2),xaxs="i",yaxs="i",mar=c(4.5,4.5,2,1.5)+0.1)
plot(t, I0, type="l")
plot(t, T, type="l")
```



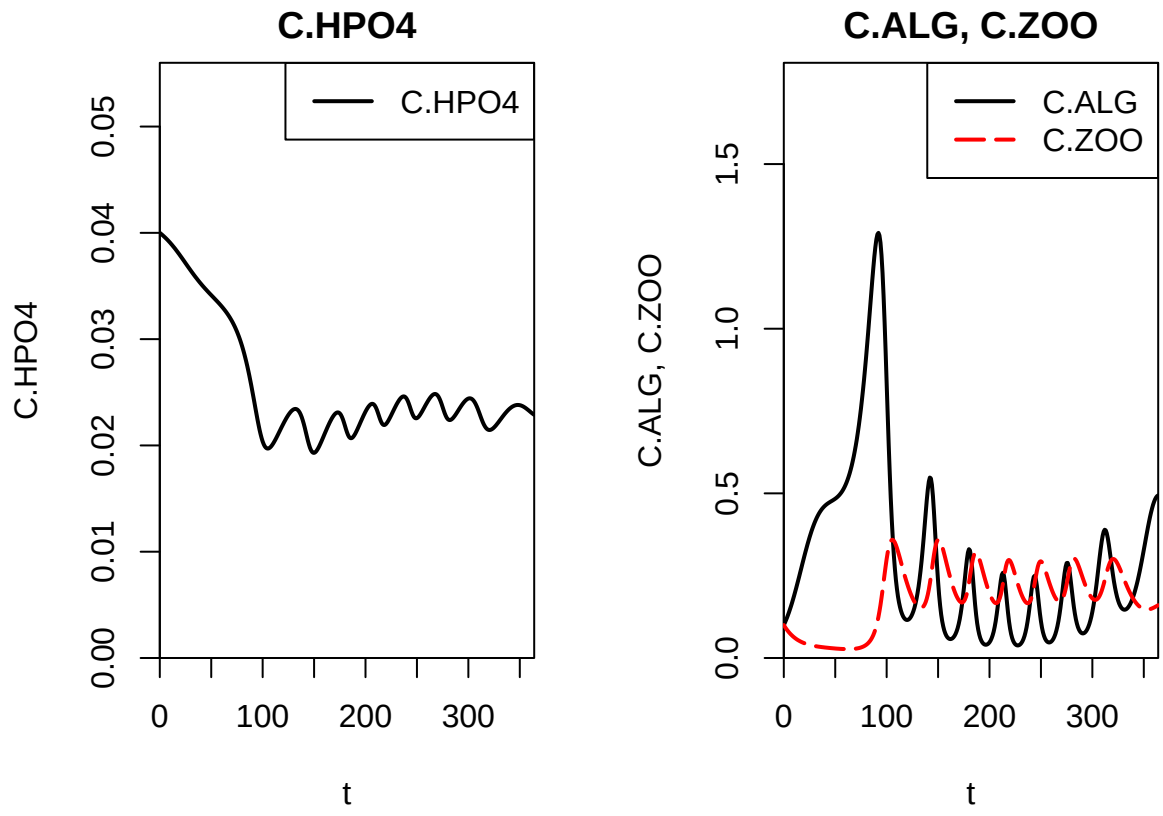

```
# re-define the reactor "epilimnion" in the system definition:
system.11.2.b@reactors <- list(epilimnion)

# increase algal growth rate to compensate for new limitations:
param$k.gro.ALG <- 0.8

# replace parameters in the system definition:
system.11.2.b@param <- param
```

Second, redo the simulations and plot and interpret the behavior of the solutions.

```
# redo simulations and plot results:
res.11.2.b <- calcres(system.11.2.b)

plotres(res=res.11.2.b, colnames=list("C.HPO4",c("C.ALG", "C.ZOO")))
```




```
plotres(res=res.11.2.b[1:365,], colnames=list("C.HPO4",c("C.ALG", "C.ZOO")))
```



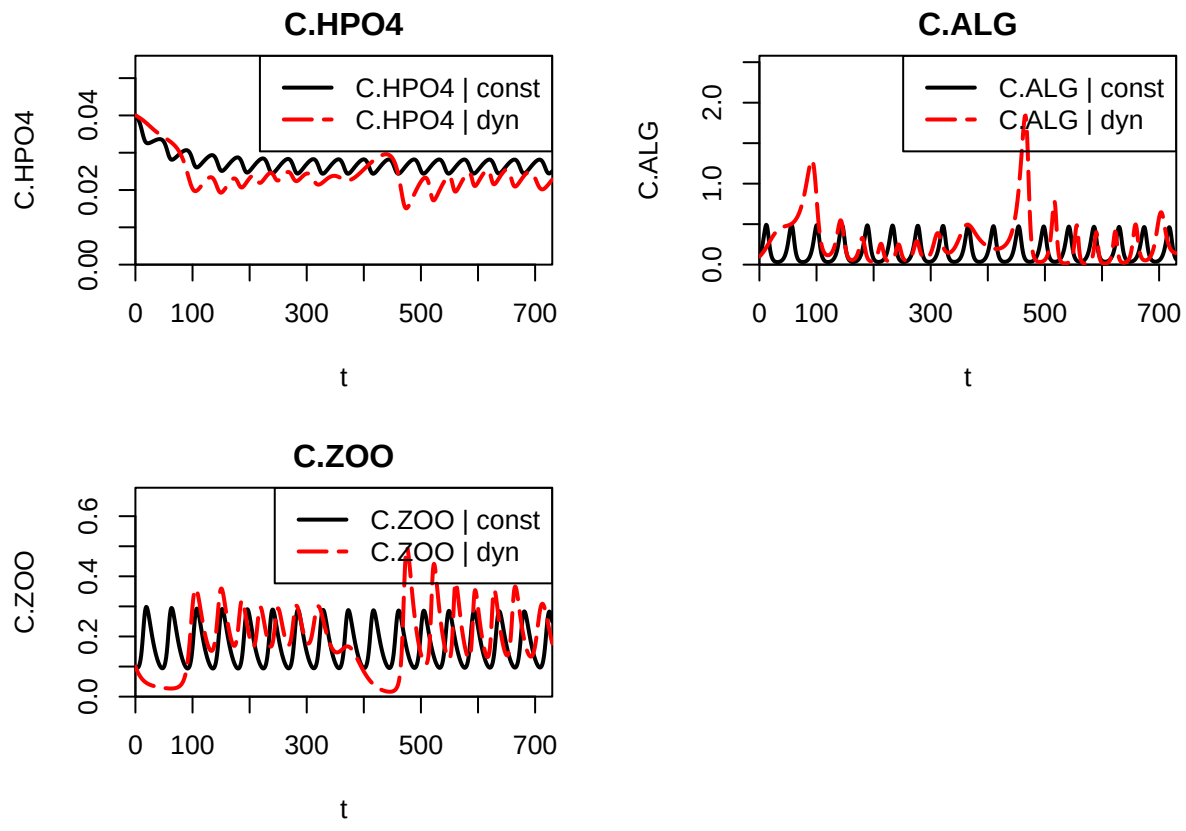

```
plotres(res      = res.11.2.b,    # plot to pdf file
        colnames = list("C.HPO4",c("C.ALG","C.ZOO")),
        file     = "exercise_2_results_b.pdf",
        width    = 10,
        height   = 5)
```

```
## pdf
##   2
```

We compare the two simulations.

```
# comparison of the two simulations:

plotres(res      = list(const=res.11.2.a,dyn=res.11.2.b),
        colnames = list("C.HPO4","C.ALG","C.ZOO"))

plotres(res      = list(const=res.11.2.a,dyn=res.11.2.b),
        colnames = list("C.HPO4","C.ALG","C.ZOO"),
        file     = "exercise_2_results_b.pdf",
        width    = 10,
        height   = 8)
```

```
## pdf
##   2
```



Task 3: Sensitivity analyis for constant driving forces

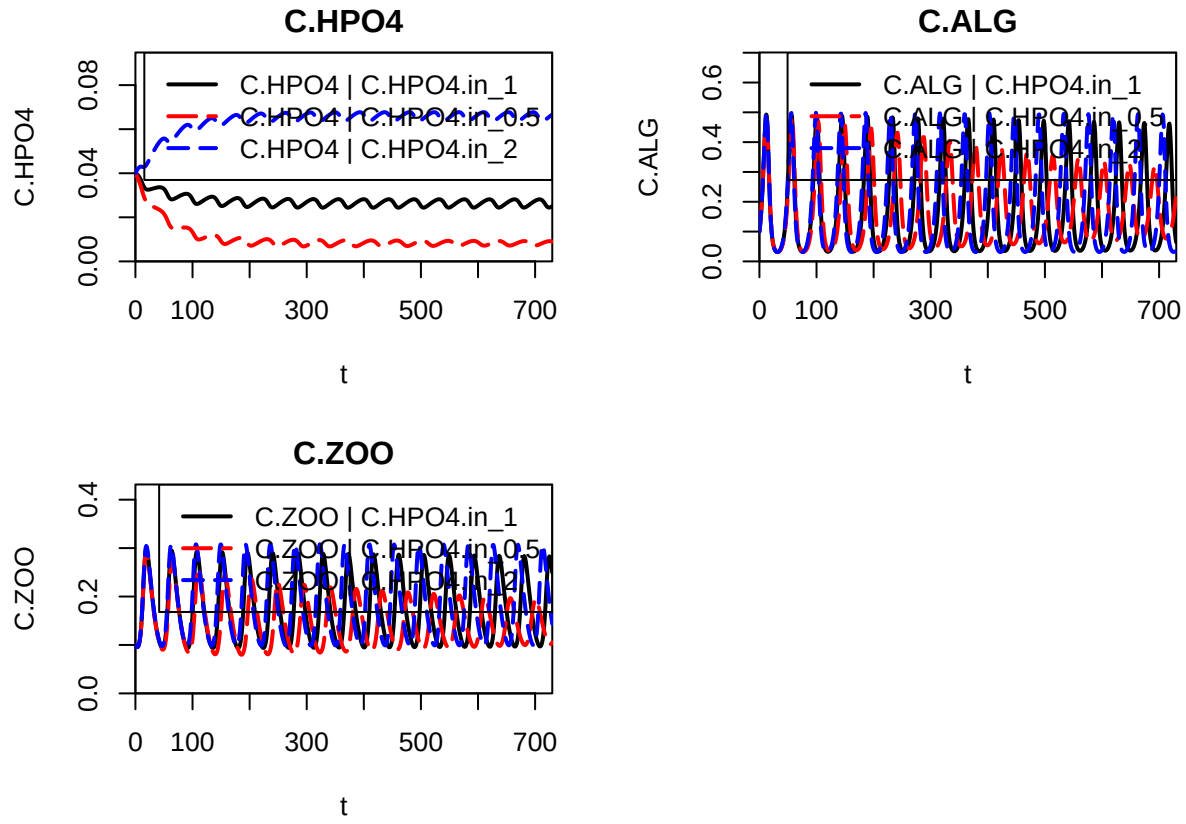
Fill in the missing terms and perform a sensitivity analysis of the model results to the parameters

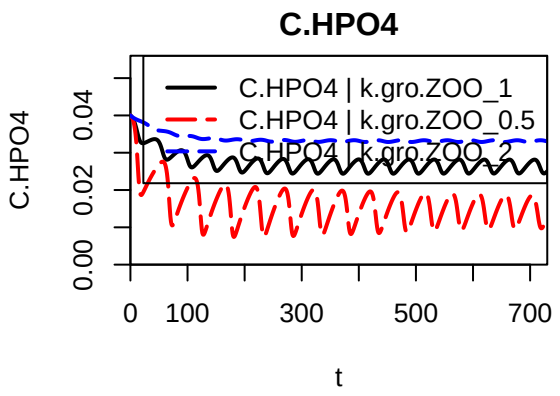
$$C_{in,HPO_4^{2-}},\ Q_{in},\ k_{gro,ALG},\ k_{death,ALG},\ k_{gro,ZOO},\ k_{death,ZOO},\ \text{and}\ Y_{ZOO}$$

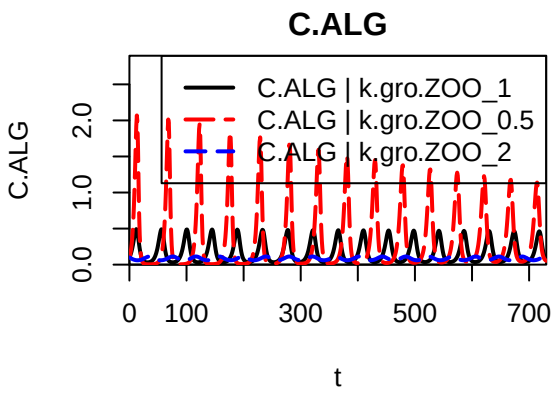
by modifying their values by factors of 2 and 1/2 for the model under constant driving forces using the function `calcsens`. Interpret the results. *Hint:* look at the solution of the Task 5 of Exercise 1.

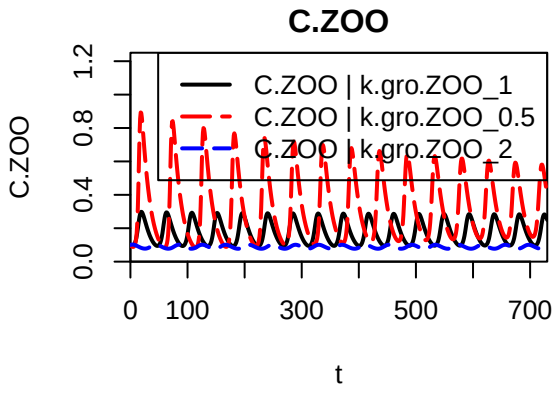
```
# Sensitivity analysis with constant driving forces
# ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

# calculate results of sensitivity analysis: TO BE COMPLETED

res.11.2.a.sens <- calcsens(system.11.2.a,
                            param.sens=c("C.HPO4.in",
                                         "k.gro.ZOO",
                                         "k.death.ZOO",
                                         "k.gro.ALG",
                                         "k.death.ALG",
                                         "K.HPO4",
                                         "Y.ZOO",
                                         "alpha.P.ALG",
                                         "Q.in"))
```

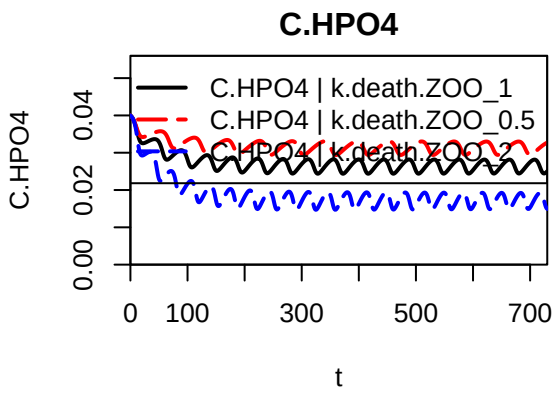
```
# plot results:
plotres(res      = res.11.2.a.sens,
        colnames = list("C.HPO4","C.ALG", "C.ZOO"))
```

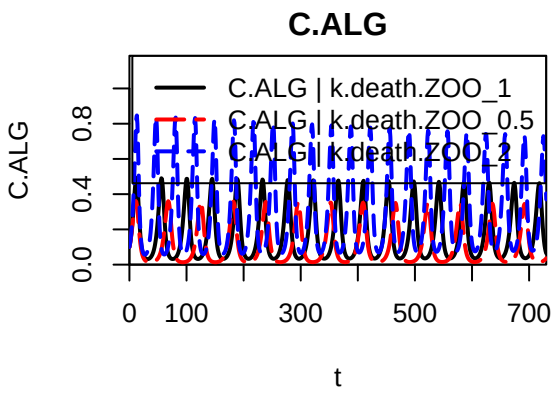


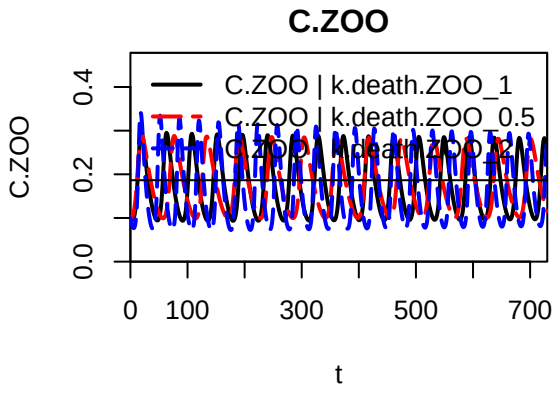

C.HPO4
C.HPO4 | k.gro.ZOO_1
C.HPO4 | k.gro.ZOO_0.5
C.HPO4 | k.gro.ZOO_2
0.00
0.02
0.04
0
100
300
500
700
t

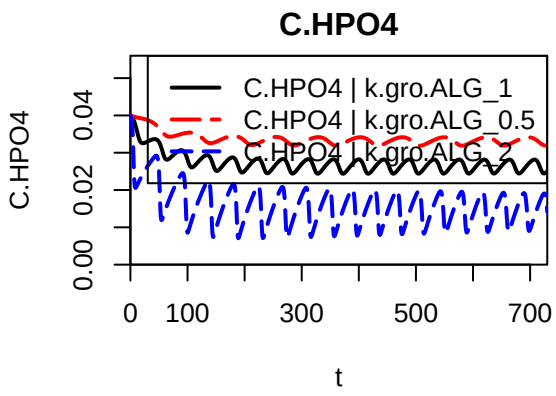

C.ALG
C.ALG | k.gro.ZOO_1
C.ALG | k.gro.ZOO_0.5
C.ALG | k.gro.ZOO_2
0.0
1.0
2.0
0
100
300
500
700
t

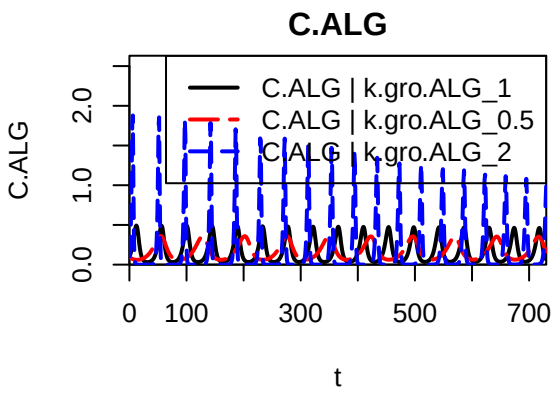

C.ZOO
C.ZOO | k.gro.ZOO_1
C.ZOO | k.gro.ZOO_0.5
C.ZOO | k.gro.ZOO_2
0.0
0.4
0.8
1.2
0
100
300
500
700
t

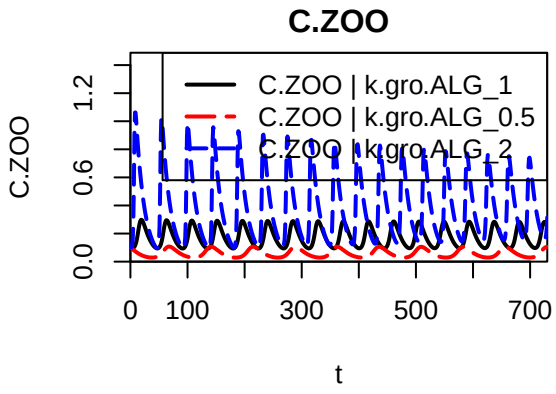

C.HPO4
C.HPO4 | k.death.ZOO_1
C.HPO4 | k.death.ZOO_0.5
C.HPO4 | k.death.ZOO_2
0.00
0.02
0.04
0
100
300
500
700
t

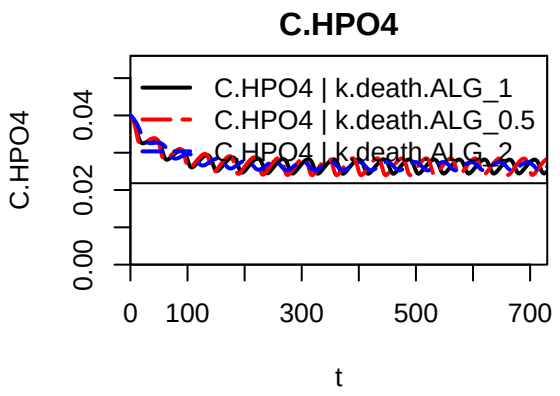

C.ALG
C.ALG | k.death.ZOO_1
C.ALG | k.death.ZOO_0.5
C.ALG | k.death.ZOO_2
0.0
0.4
0.8
0
100
300
500
700
t

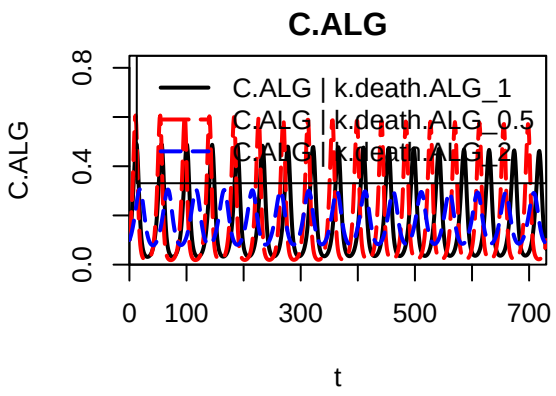

C.ZOO
C.ZOO | k.death.ZOO_1
C.ZOO | k.death.ZOO_0.5
C.ZOO | k.death.ZOO_2
0.0
0.2
0.4
0
100
300
500
700
t

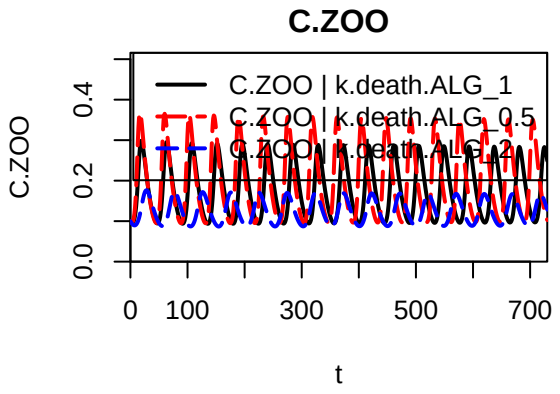

C.HPO4
C.HPO4 | k.gro.ALG_1
C.HPO4 | k.gro.ALG_0.5
C.HPO4 | k.gro.ALG_2
0.00
0.02
0.04
0
100
300
500
700
t

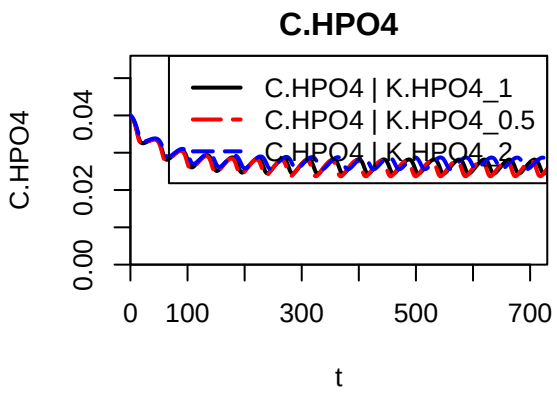

C.ALG
C.ALG | k.gro.ALG_1
C.ALG | k.gro.ALG_0.5
C.ALG | k.gro.ALG_2
0.0
1.0
2.0
0
100
300
500
700
t

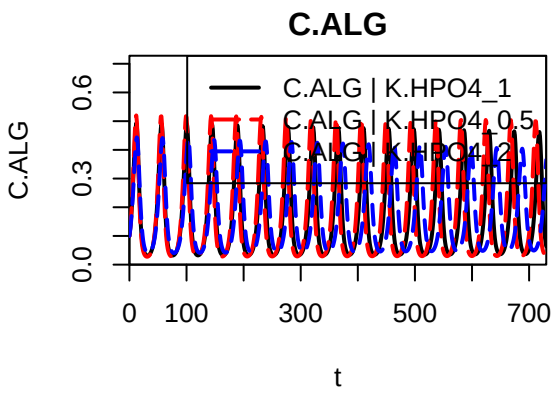

C.ZOO
C.ZOO | k.gro.ALG_1
C.ZOO | k.gro.ALG_0.5
C.ZOO | k.gro.ALG_2
0.0
0.6
1.2
0
100
300
500
700
t

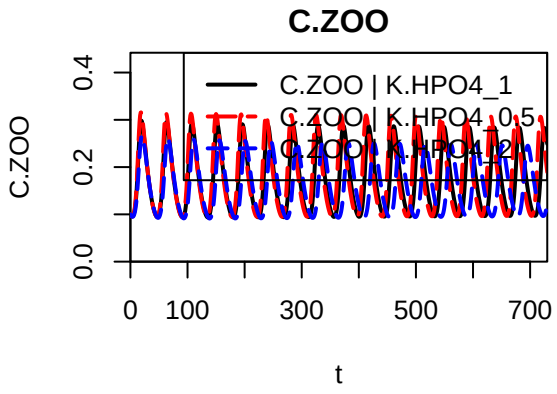

C.HPO4
C.HPO4 | k.death.ALG_1
C.HPO4 | k.death.ALG_0.5
C.HPO4 | k.death.ALG_2
0.00
0.02
0.04
0
100
300
500
700
t

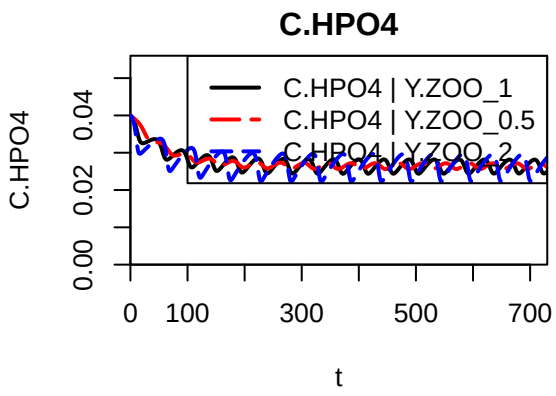

C.ALG
C.ALG | k.death.ALG_1
C.ALG | k.death.ALG_0.5
C.ALG | k.death.ALG_2
0.0
0.4
0.8
0
100
300
500
700
t

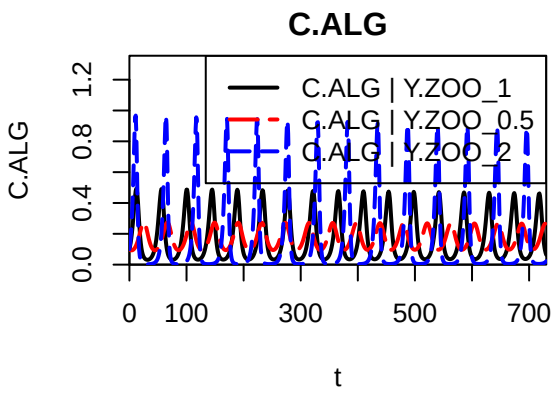

C.ZOO
C.ZOO | k.death.ALG_1
C.ZOO | k.death.ALG_0.5
C.ZOO | k.death.ALG_2
0.0
0.2
0.4
0
100
300
500
700
t

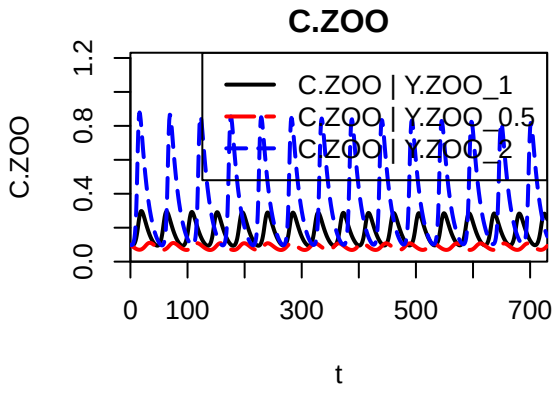

C.HPO4
C.HPO4 | K.HPO4_1
C.HPO4 | K.HPO4_0.5
C.HPO4 | K.HPO4_2
0.00
0.02
0.04
0
100
300
500
700
t

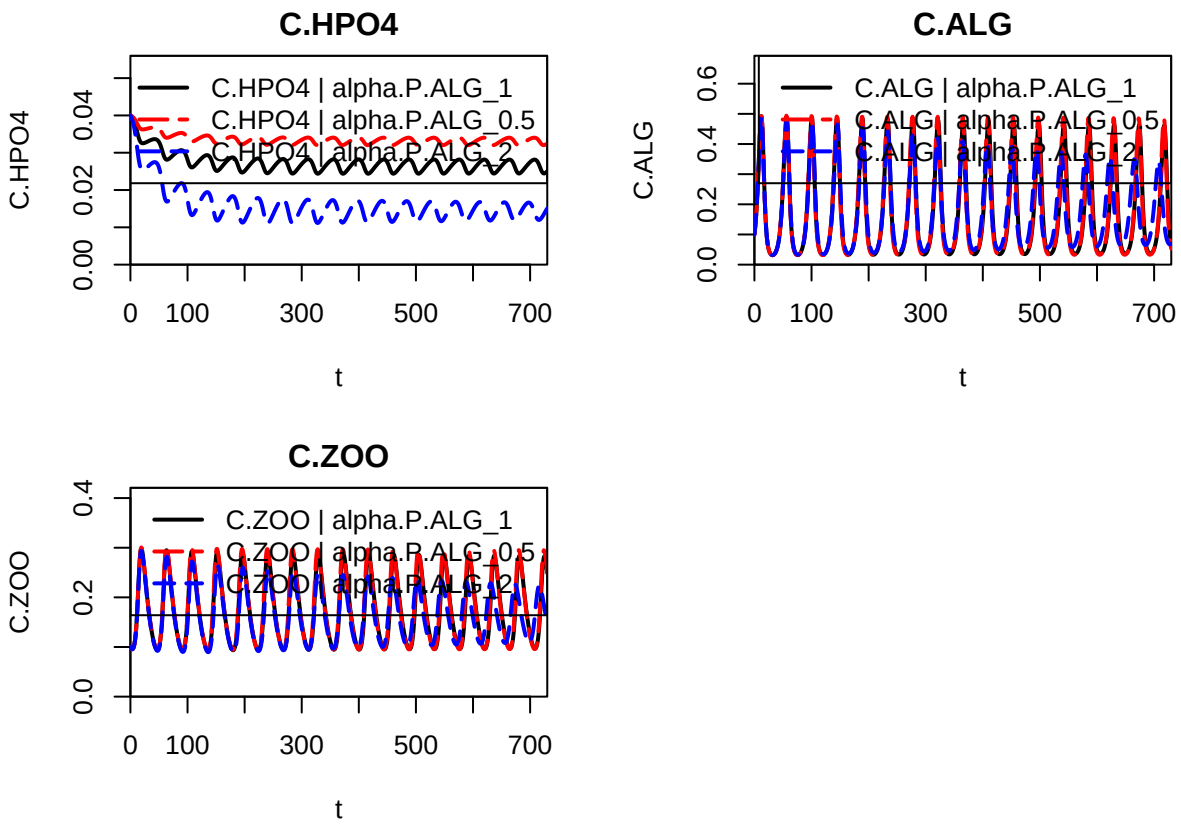

C.ALG
C.ALG | K.HPO4_1
C.ALG | K.HPO4_0.5
C.ALG | K.HPO4_2
0.0
0.3
0.6
0
100
300
500
700
t


C.ZOO
C.ZOO | K.HPO4_1
C.ZOO | K.HPO4_0.5
C.ZOO | K.HPO4_2
0.0
0.2
0.4
0
100
300
500
700
t


C.HPO4
C.HPO4 | Y.ZOO_1
C.HPO4 | Y.ZOO_0.5
C.HPO4 | Y.ZOO_2
0.00
0.02
0.04
0
100
300
500
700
t

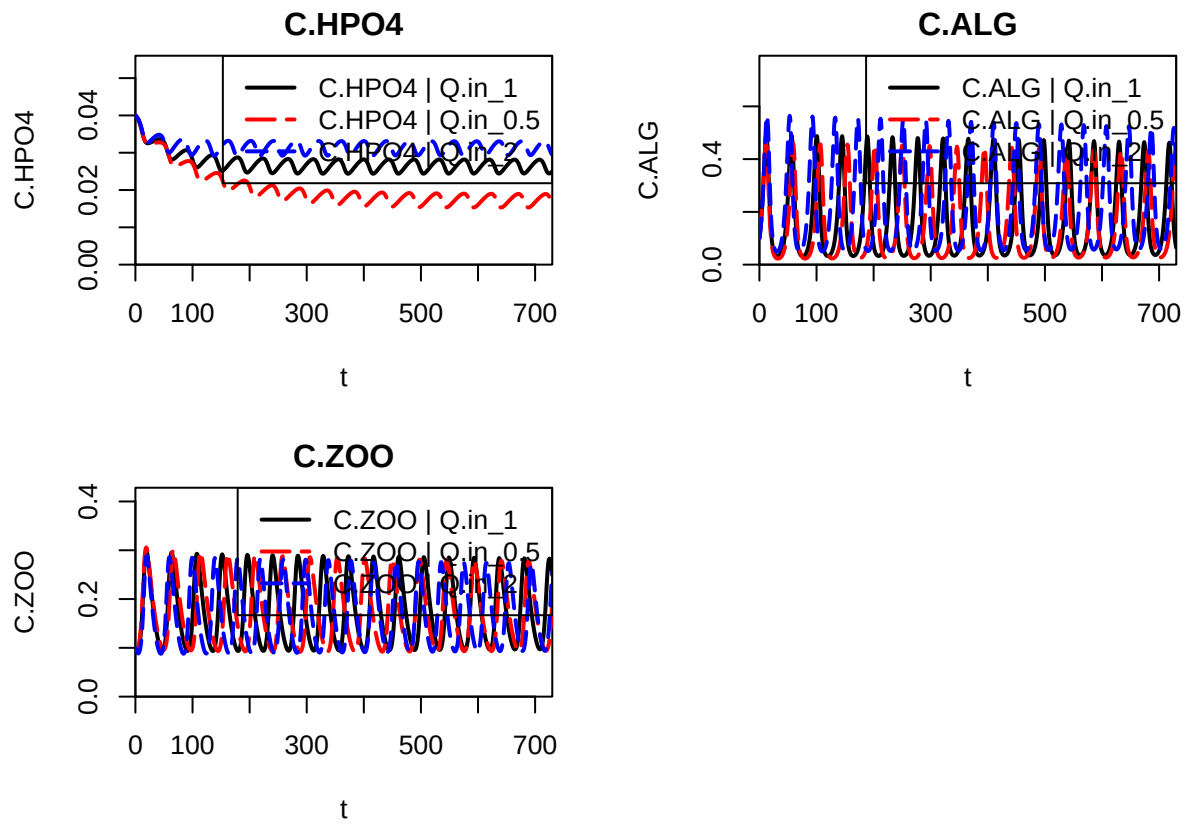

C.ALG
C.ALG | Y.ZOO_1
C.ALG | Y.ZOO_0.5
C.ALG | Y.ZOO_2
0.0
0.4
0.8
1.2
0
100
300
500
700
t


C.ZOO
C.ZOO | Y.ZOO_1
C.ZOO | Y.ZOO_0.5
C.ZOO | Y.ZOO_2
0.0
0.4
0.8
1.2
0
100
300
500
700
t




```
plotres(res      = res.11.2.a.sens,
        colnames = list("C.HPO4","C.ALG", "C.ZOO"),
        file     = "exercise_2_results_a_sens.pdf",
        width    = 10,
        height   = 8)
```

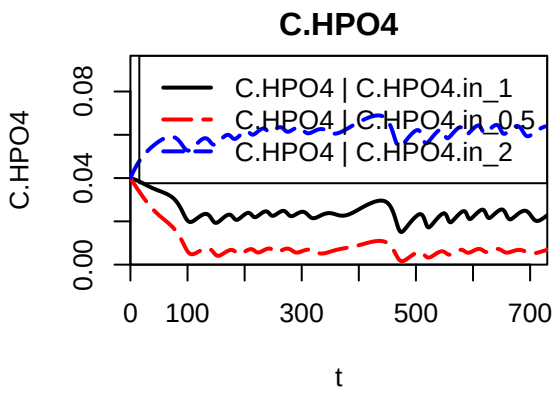
```
## pdf
##   2
```

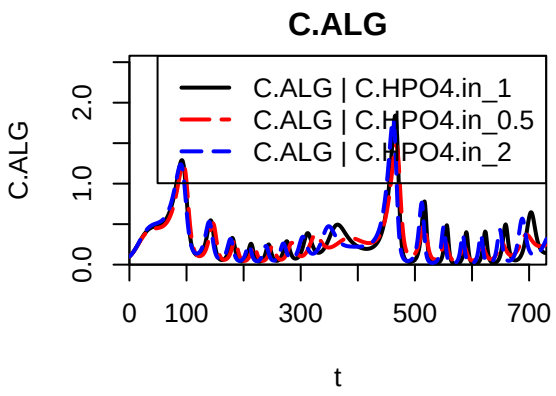


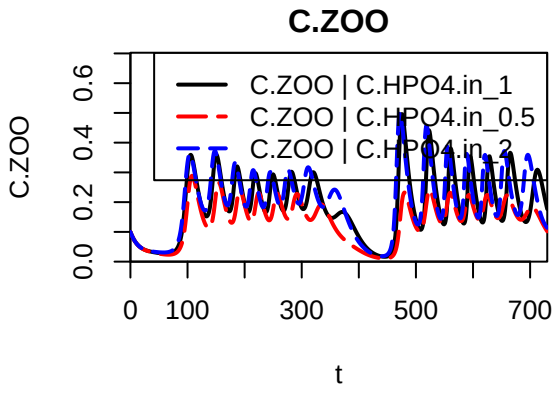
Task 4 - Homework: Sensitivity analyis for seasonally varying driving forces

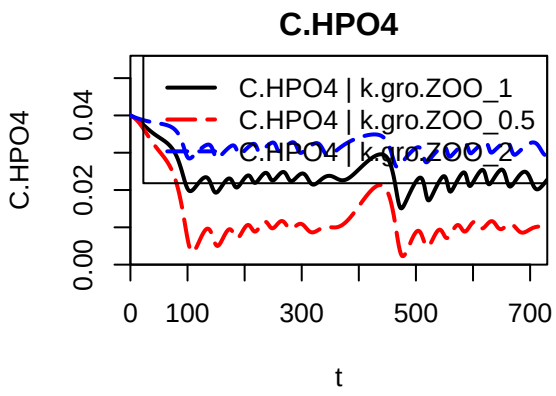
Do a sensitivity analysis for seasonally varying driving forces and discuss the differences to the case with constant driving forces.

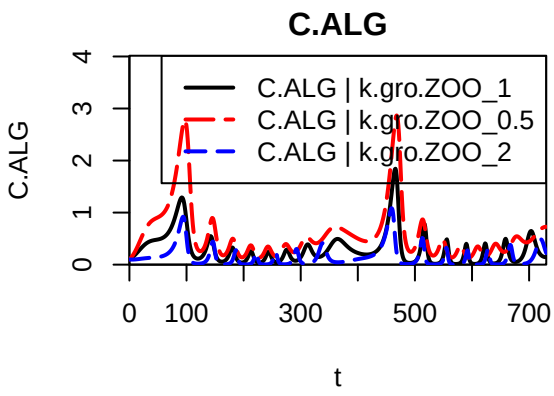
```
# calculate results of sensitivity analysis:

res.11.2.b.sens <- calcsens(system.11.2.b,
                            param.sens=c("C.HPO4.in",
                                         "k.gro.ZOO",
                                         "k.death.ZOO",
                                         "k.gro.ALG",
                                         "k.death.ALG",
                                         "K.HPO4",
                                         "Y.ZOO",
                                         "alpha.P.ALG",
                                         "Q.in"))

# plot results:

plotres(res      = res.11.2.b.sens,
        colnames = list("C.HPO4","C.ALG", "C.ZOO"))
```

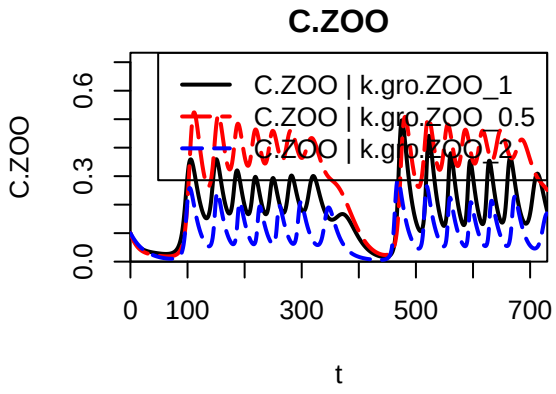

C.HPO4
C.HPO4 | C.HPO4.in_1
C.HPO4 | C.HPO4.in_0.5
C.HPO4 | C.HPO4.in_2
0.00
0.04
0.08
0
100
300
500
700
t

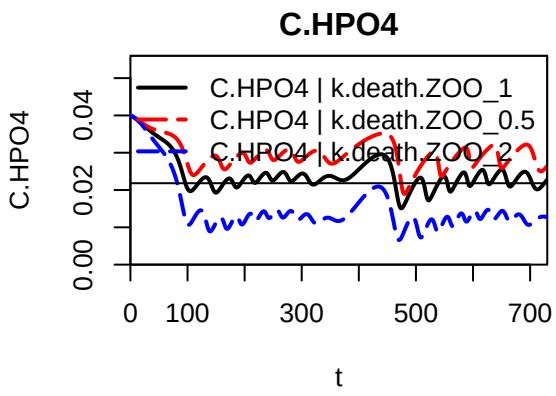

C.ALG
C.ALG | C.HPO4.in_1
C.ALG | C.HPO4.in_0.5
C.ALG | C.HPO4.in_2
0.0
1.0
2.0
0
100
300
500
700
t

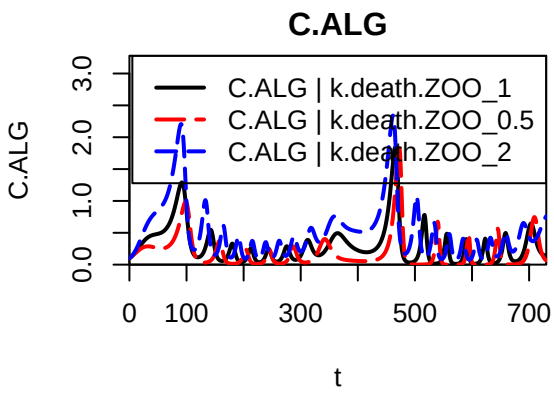

C.ZOO
C.ZOO | C.HPO4.in_1
C.ZOO | C.HPO4.in_0.5
C.ZOO | C.HPO4.in_2
0.0
0.2
0.4
0.6
0
100
300
500
700
t

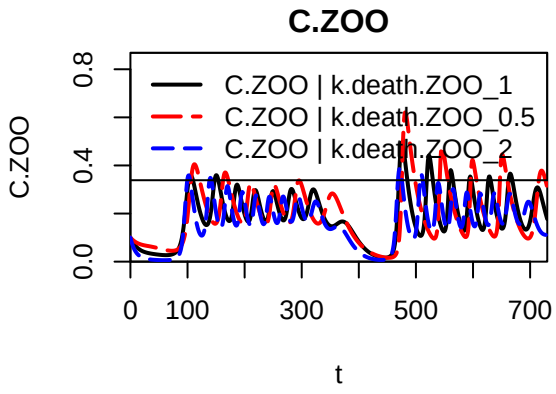

C.HPO4
C.HPO4 | k.gro.ZOO_1
C.HPO4 | k.gro.ZOO_0.5
C.HPO4 | k.gro.ZOO_2
0.00
0.02
0.04
0
100
300
500
700
t

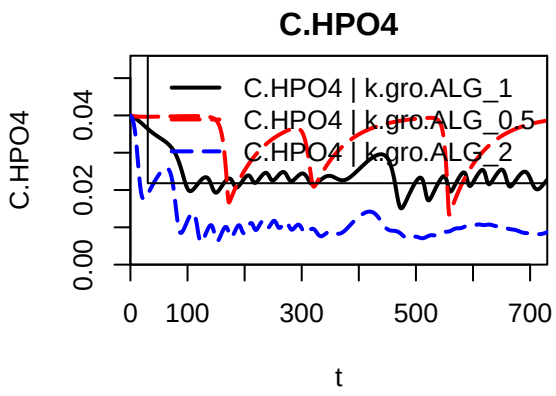

C.ALG
C.ALG | k.gro.ZOO_1
C.ALG | k.gro.ZOO_0.5
C.ALG | k.gro.ZOO_2
0
1
2
3
4
0
100
300
500
700
t

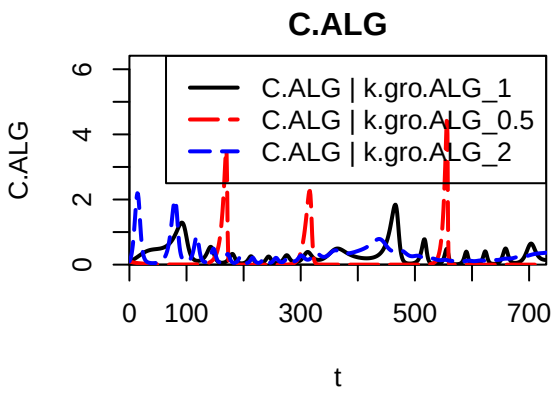

C.ZOO
C.ZOO | k.gro.ZOO_1
C.ZOO | k.gro.ZOO_0.5
C.ZOO | k.gro.ZOO_2
0.0
0.3
0.6
0
100
300
500
700
t

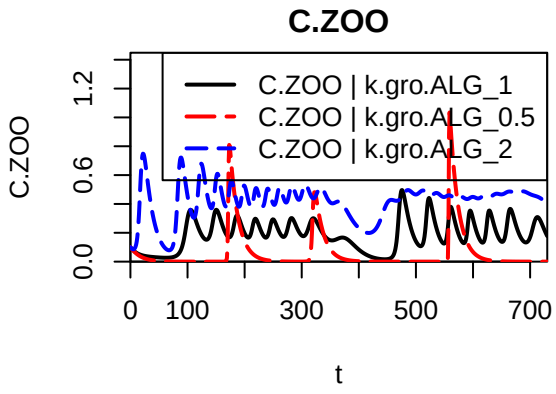

C.HPO4
C.HPO4 | k.death.ZOO_1
C.HPO4 | k.death.ZOO_0.5
C.HPO4 | k.death.ZOO_2
0.00
0.02
0.04
0
100
300
500
700
t

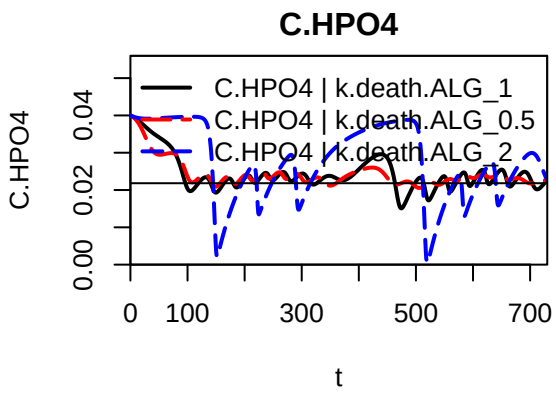

C.ALG
C.ALG | k.death.ZOO_1
C.ALG | k.death.ZOO_0.5
C.ALG | k.death.ZOO_2
0.0
1.0
2.0
3.0
0
100
300
500
700
t

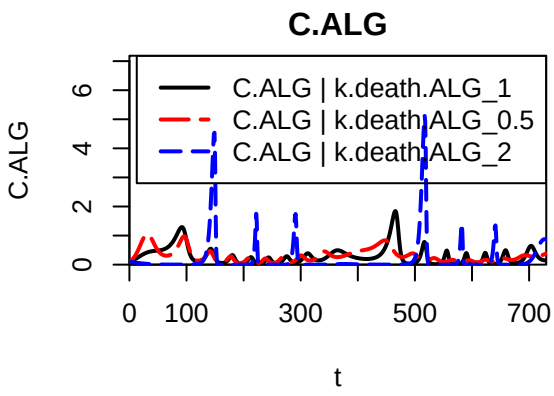

C.ZOO
C.ZOO | k.death.ZOO_1
C.ZOO | k.death.ZOO_0.5
C.ZOO | k.death.ZOO_2
0.0
0.4
0.8
0
100
300
500
700
t

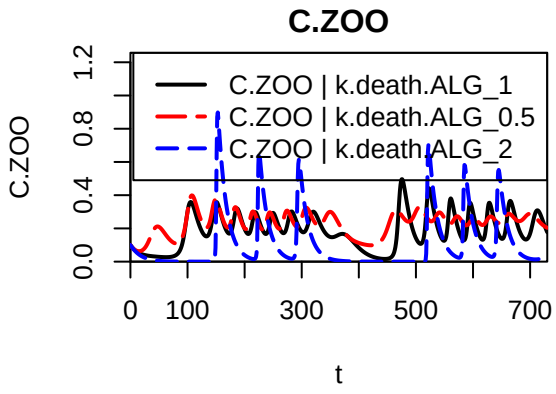

C.HPO4
C.HPO4 | k.gro.ALG_1
C.HPO4 | k.gro.ALG_0.5
C.HPO4 | k.gro.ALG_2
0.00
0.02
0.04
0
100
300
500
700
t

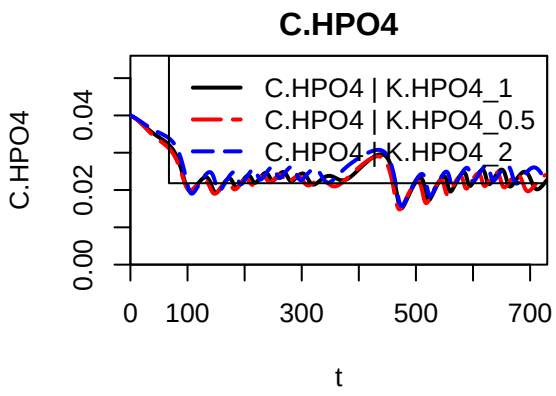

C.ALG
C.ALG | k.gro.ALG_1
C.ALG | k.gro.ALG_0.5
C.ALG | k.gro.ALG_2
0
2
4
6
0
100
300
500
700
t

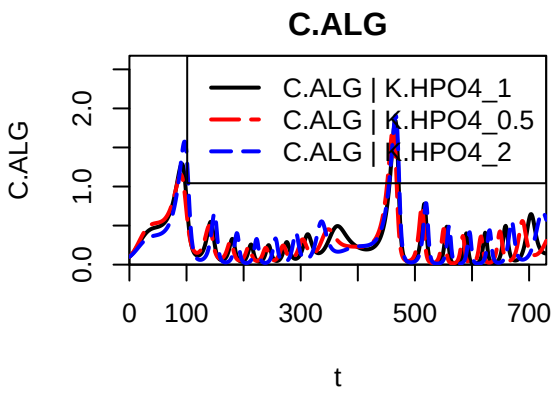

C.ZOO
C.ZOO | k.gro.ALG_1
C.ZOO | k.gro.ALG_0.5
C.ZOO | k.gro.ALG_2
0.0
0.6
1.2
0
100
300
500
700
t

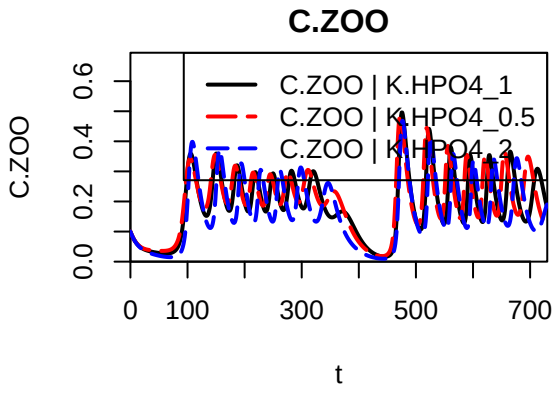

C.HPO4
C.HPO4 | k.death.ALG_1
C.HPO4 | k.death.ALG_0.5
C.HPO4 | k.death.ALG_2
C.HPO4
0.00
0.02
0.04
0
100
300
500
700
t

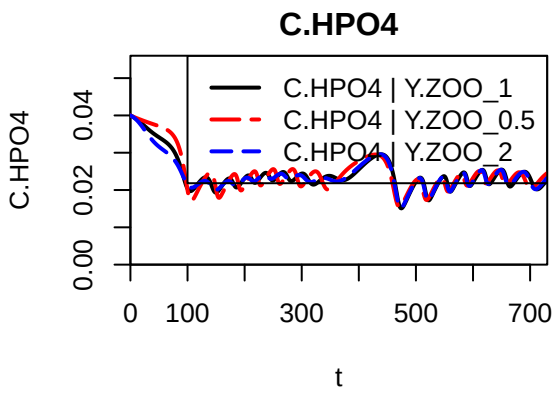

C.ALG
C.ALG | k.death.ALG_1
C.ALG | k.death.ALG_0.5
C.ALG | k.death.ALG_2
C.ALG
0
2
4
6
0
100
300
500
700
t

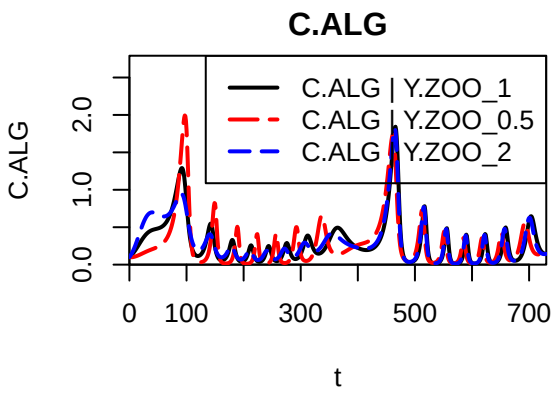

C.ZOO
C.ZOO | k.death.ALG_1
C.ZOO | k.death.ALG_0.5
C.ZOO | k.death.ALG_2
C.ZOO
0.0
0.4
0.8
1.2
0
100
300
500
700
t

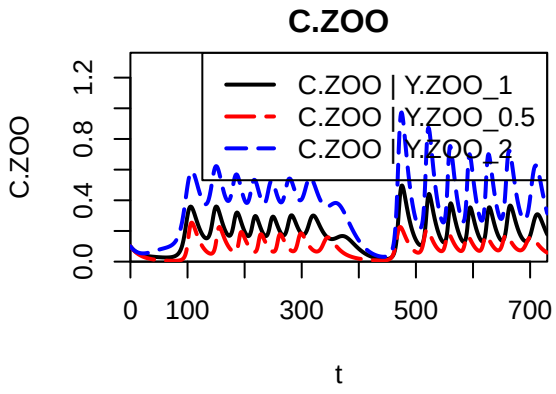

C.HPO4
C.HPO4 | K.HPO4_1
C.HPO4 | K.HPO4_0.5
C.HPO4 | K.HPO4_2
C.HPO4
0.00
0.02
0.04
0
100
300
500
700
t

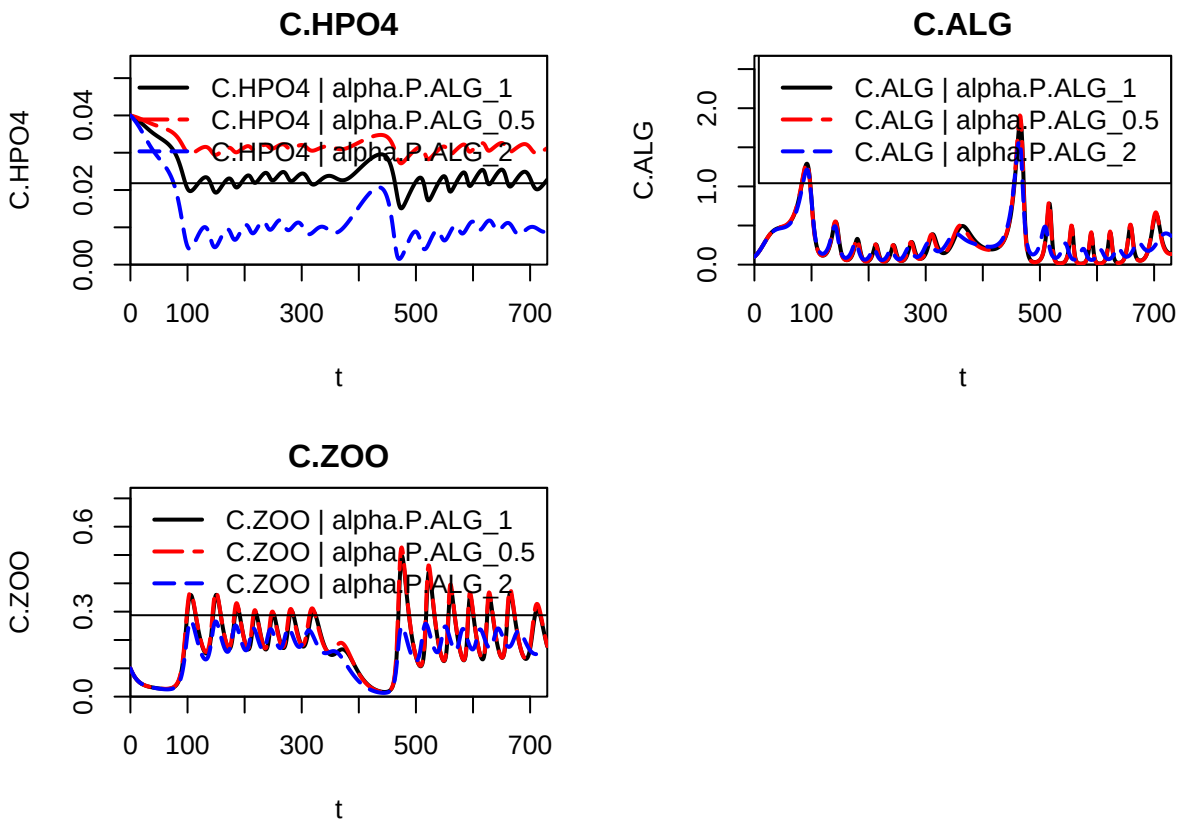

C.ALG
C.ALG | K.HPO4_1
C.ALG | K.HPO4_0.5
C.ALG | K.HPO4_2
C.ALG
0.0
1.0
2.0
0
100
300
500
700
t


C.ZOO
C.ZOO | K.HPO4_1
C.ZOO | K.HPO4_0.5
C.ZOO | K.HPO4_2
C.ZOO
0.0
0.2
0.4
0.6
0
100
300
500
700
t


C.HPO4
C.HPO4 | Y.ZOO_1
C.HPO4 | Y.ZOO_0.5
C.HPO4 | Y.ZOO_2
0.00
0.02
0.04
0
100
300
500
700
t

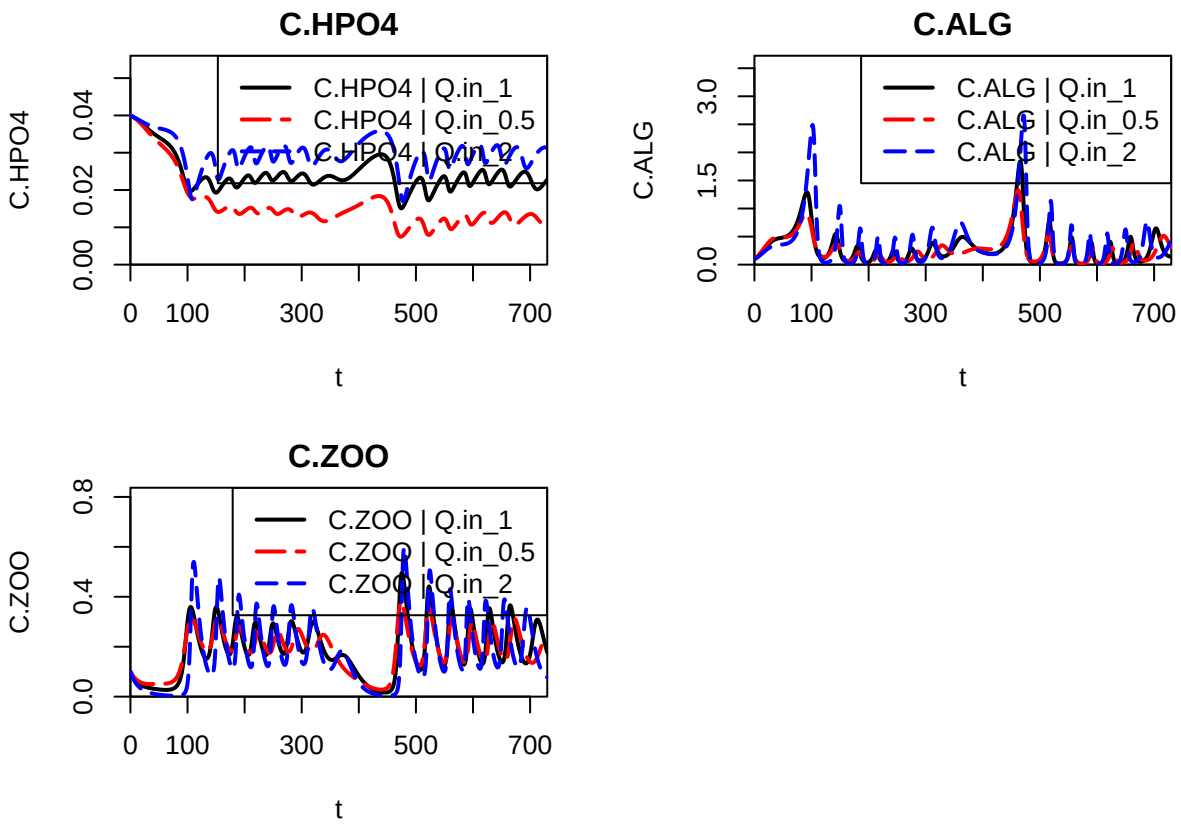

C.ALG
C.ALG | Y.ZOO_1
C.ALG | Y.ZOO_0.5
C.ALG | Y.ZOO_2
0.0
1.0
2.0
0
100
300
500
700
t


C.ZOO
C.ZOO | Y.ZOO_1
C.ZOO | Y.ZOO_0.5
C.ZOO | Y.ZOO_2
0.0
0.4
0.8
1.2
0
100
300
500
700
t




```
plotres(res      = res.11.2.b.sens,
        colnames = list("C.HPO4","C.ALG", "C.ZOO"),
        file     = "exercise_2_results_b_sens.pdf",
        width    = 10,
        height   = 8)
```

```
## pdf
##   2
```



Theory questions

1. Are the algae concentrations controlled bottom-up (by phosphate limitation) or top-down (by grazing of zooplankton)?
2. What is the reason for oscillating concentrations under constant driving forces? What happens when you introduce periodic driving forces?
3. What are the main deficits of the model compared to a real lake?
4. What is your expectation regarding the response of the model to the change in each parameter, does the result match your expectation and can you explain the observed changes?