The CNPDIA package - early diagenetic modelling of the C, N and P cycle

Karline Soetaert

20-June-2020

CNPDIA

The CNPDIA model is a 1-D model of Carbon, nitrogen, phosphorus and oxygen diagenesis in a marine sediment. It is based on the OMEXDIA model (Soetaert et al., 1996a, b), extended with simple P dynamics.

The model describes twelve state variables, in 100 layers:

- 2 fractions of organic carbon (FDET,SDET): fast and slow decaying, solid substance.
- Oxygen (O2), dissolved substance.
- Nitrate (NO3), dissolved substance.
- Nitrite (NO2), dissolved substance.
- Ammonium (NH3), dissolved substance.
- Oxygen demand unit (ODU), dissolved substance, as a lump sum of all reduced substances other than ammonium.
- Dissolved inorganic carbon (DIC), dissolved substance
- Phosphate (PO4), dissolved substance
- Iron-bound P (FeP), P bound to iron oxides, solid substance
- Ca-bound P (CaP), apatite, solid substance
- Adsorbed P (Pads), solid substance

Time is expressed in days, and space is expressed in centimeters.

Concentrations of liquids and solids are expressed in [nmol/cm3 liquid] and [nmol/cm3 solid] respectively (Note: this is the same as [mmol/m3 liquid] and [mmol/m3 solid]).

Compared to the OMEXDIA model, CNPDIA includes the following additions:

- simple phosphorus dynamics,
- allowing boundary conditions with water overlying sediment or exposure to the air,
- external conditions set either with time-variable forcings or as constant parameters,
- bottom water conditions either imposed or dynamically modeled,
- possibility to include sediment perturbation events,
- vertical profiles of porosity, irrigation, bioturbation either set with parameters or inputted as data.

See Soetaert et al., 1996 for further details of the original OMEXDIA model.

The model is implemented in fortran (for speed) and linked to R (for flexibility).

The package

```
require(CNPDIA)
```

The CNPDIA package contains functions to generate (a time series of) 1-D diagenetic profiles. It can either be run in dynamic mode, or the steady-state solution can be estimated. It contains several utility functions, e.g. to help in extracting information on the model output, or to estimate mass budgets. It contains functions to perturb sediment properties, e.g. mimicking resuspension or deposition events.

Main functions

The main functions allow to solve the model to steady state (*CNPDIAsolve*), to run it dynamically (*CNPDIAdyna*), or to add perturbations (*CNPDIAperturb*) to dynamic simulations (this is discussed in another vignette).

Steady-state solution, function CNPDIAsolve

Function *CNPDIAsolve* finds the steady-state solution of the CNPDIA model. Its arguments are:

```
args(CNPDIAsolve)
```

```
function (parms = list(), yini = NULL, gridtype = 1, Grid = NULL,
    porosity = NULL, bioturbation = NULL, irrigation = NULL,
    surface = NULL, diffusionfactor = NULL, dynamicbottomwater = FALSE,
    ratefactor = NULL, verbose = FALSE, ...)
NULL
```

here *parms* is a list with a subset of the CNPDIA parameters (see appendix for what they mean and their default values). If unspecified, then the default parameters are used.

The *gridtype* by default assumes a cartesian grid (*gridtype = 1*), but can be 1D cylindrical (*gridtype = 2*) or spherical (*gridtype = 3*). An irregular grid can be seleceted by specifying the surface areas at the interface through argument *surface*. In a cartesian grid the surface area remains constant.

The vertical profiles that can be imposed as a vector are: *porosity*, *bioturbation irrigation*, *surface* (surface areas of box interfaces) and *diffusionfactor* (multiplication factor to estimate effective sediment diffusion based on molecular diffusion).

dynamicbottomwater, when set to TRUE will also explicitly model the bottom water concentrations.

ratefac is a multiplication factor, that is multiplied with all biogeochemical rates. It is included here for consistency with *CNPDIAdyna*.

Dynamic run, function CNPDIAdyna

Function *CNPDIAdyna* runs the CNPDIA model for a specific time interval and produces output at requested times. Its arguments are:

```
args(CNPDIAdyna)
```

```
function (parms = list(), times = 0:365, spinup = NULL, yini = NULL,
    gridtype = 1, Grid = NULL, porosity = NULL, bioturbation = NULL,
    irrigation = NULL, surface = NULL, diffusionfactor = NULL,
    dynamicbottomwater = FALSE, CfluxForc = NULL, FePfluxForc = NULL,
    CaPfluxForc = NULL, O2bwForc = NULL, NO3bwForc = NULL, NO2bwForc = NULL,
    NH3bwForc = NULL, ODUbwForc = NULL, PO4bwForc = NULL, DICbwForc = NULL,
    wForc = NULL, biotForc = NULL, irrForc = NULL, rFastForc = NULL,
    rSlowForc = NULL, pFastForc = NULL, MPBprodForc = NULL, gasfluxForc = NULL,
    HwaterForc = NULL, ratefactor = NULL, verbose = FALSE, ...)
```

NULL

The functions to run the model dynamically also allow for several external conditions to be either constants or to vary in time. Thus, they can be set by a parameter or as a forcing function.

These conditions are:

- the flux of carbon, CaP and FeP (parameters *Cflux, CaPflux, FePflux*, forcings *CfluxForc, CaPfluxForc, FePfluxForc*),
- the bottom water concentrations (parameters *O2bw, NO3bw, NO2bw, NH3bw, ODUbw, PO4bw, DICbw*, forcings *O2bwForc, NO3bwForc, NO2bwForc, NH3bwForc, ODUbwForc, PO4bwForc, DICbwForc*)
- the sedimentation, bioturbation and bio-irrigation rates (*w, biot, irr*), (*wForc, biotForc, irrForc*)
- the decay rates of organic matter (parameters *rFast, rSlow*, forcings *rFastForc, rSlowForc*) and the fraction fast organic matter present in the flux (parameter *pFast*, forcing *pFastForc*)
- the air-sea exchange rate when exposed to the air (parameter *gasflux*, forcings *gasfluxForc*)
- the height of the overlying water (parameter *Hwater*, forcing function *HwaterForc*), effective only if *dynamicbottomwater* is *TRUE*.
- *ratefac* is a (time series or a constant) multiplication factor, that is multiplied with all biogeochemical rates. It can be used to impose temperature dependency.

These forcing functions are either prescribed as a list that contains a data series (*list (data = . . .)*) or as a list that specifies a periodic signal, defined by the amplitude (*amp*), *period*, *phase*, a coefficient that defines the strength of the periodic signal (*power*) and the minimum value (*min*) : the default settings are: *list(amp = 0, period = 365, phase = 0, pow = 1, min = 0)*. The mean value in the sine function is given by the corresponding parameter.

For instance, for the C flux, the seasonal signal would be defined as: $max(min, Cflux*(1+(amp*sin((times-phase)/period*2*pi))^pow)$.

Perturbation run, function CNPDIAperturb

```
head(args(CNPDIAperturb))
```

```
1 function (parms = list(), times = 0:365, spinup = NULL, yini = NULL,
2     gridtype = 1, Grid = NULL, porosity = NULL, bioturbation = NULL,
3     irrigation = NULL, surface = NULL, diffusionfactor = NULL,
4     dynamicbottomwater = FALSE, perturbType = "mix", perturbTimes = seq(from = 0,
5         to = max(times), by = 365), perturbDepth = 5, concfac = 1,
6     CfluxForc = NULL, FePfluxForc = NULL, CaPfluxForc = NULL,
```

Three types of perturbations are possible (argument *perturb*):

- *mixing* straightens the profiles over a certain depth
- *erosion* removes part of the surficial sediment
- *deposition* adds sediment on top.

These perturbations are implemented as events, and need input of the perturbation times (*perturbTimes*), and the depth (*perturbDepth*). For deposition events, the factor of increase/decrease of the solid fraction concentration can also be inputted (*concfac*).

Accessory functions

The default values of the parameters, and their units can be interrogated:

```
P  <- CNPDIAparms()
head(P)
```

```
##                parms       units                description
## Cflux   5.000000e+02 nmolC/cm2/d total organic C deposition
## pFast   9.000000e-01           -   part FDET in carbon flux
## FePflux 0.000000e+00 nmolP/cm2/d     deposition rate of FeP
## CaPflux 0.000000e+00 nmolP/cm2/d     deposition rate of CaP
## rFast   6.849315e-02          /d             decay rate FDET
## rSlow   1.369863e-04          /d             decay rate SDET
```

See the appendix for a complete list of the parameters.

Note: some parameters only apply if the bottom water concentration is modeled dynamically; they comprise the *dilution* of the bottom water (nudging to bottom water concentration), the height of the bottom water (*Hwater*), and the sinking rate of the solid constituents (C, FeP, CaP) (parameters *Cfall*, *CaPfall* and *FePfall*).

Budgets

Once the model is solved, it is possible to calculate budgets of the C, N, P and O2 cycle (*CNPDIAbudgetC*, *CNPDIAbudgetN*, *CNPDIAbudgetP*, *CNPDIAbudgetO2*).

```
std <- CNPDIAsolve(parms = list(Cflux = 1500))

print(CNPDIAbudgetC(std))
```

```
## $Fluxes
##                   FDET          SDET           DIC         CinCaP        Total
## surface  1.350000e+03  1.500000e+02 -1.499999e+03   0.000000e+00 8.420476e-04
## bottom  8.961902e-223 1.542884e-102  8.420476e-04  2.979111e-113 8.420476e-04
## perturb  0.000000e+00  0.000000e+00  0.000000e+00   0.000000e+00 0.000000e+00
## netin    1.350000e+03  1.500000e+02 -1.500000e+03 -2.979111e-113 1.887202e-11
##
## $Rates
##             OxicMineralisation Denitrification AnoxicMineralisation TotalMineralisation CaPprecipita
## nmolC/cm2/d           999.5993         41.5706             458.8301               1500
##             MPBDICuptake MPBResp MPBCdeath EPSproduction EPSmineralisation FDETprodMPBdeath DICprodM
## nmolC/cm2/d            0       0         0             0                 0                0
##
## $dC
##            DET            DIC            CaP           MPBC            sum
##   2.273737e-13   1.864464e-11 -1.038175e-112   0.000000e+00   1.887202e-11
##
## $Losses
## [1] 0.0008420476
##
## $Delta
## [1] 1.878721e-11
##
## $Fluxmat
##                   Ext  DET  DIC CaP EPS MPB        Burial
## Ext     0.000000e+00 1500    0   0   0   0  0.000000e+00
## DET     0.000000e+00    0 1500   0   0   0 1.542884e-102
## DIC     1.499999e+03    0    0   0   0   0  8.420476e-04
## CaP    2.979111e-113    0    0   0   0   0  0.000000e+00
```

```
## EPS     0.000000e+00    0    0   0   0   0  0.000000e+00
## MPB     0.000000e+00    0    0   0   0   0  0.000000e+00
## Burial  0.000000e+00    0    0   0   0   0  0.000000e+00
```

where all fluxes are in *nmol/cm2/day*.

Properties of solutions

There are functions to retrieve several properties of the solution:

- *CNPDIAdepth*, *CNPDIAdx*, *CNPDIAgrid* retrieve the sediment depths, layer thicknesses and grid of *CNPDIA* model solutions.
- *CNPDIAbiot*, *CNPDIApor*, *CNPDIAirr* retrieve the bioturbation, porosity, and irrigation profiles of *CNPDIA* model solutions.
- *CNPDIAparms* retrieves the parameter settings of *CNPDIA* model solutions.
- *CNPDIA0D* and *CNPDIA1D* return the output variables of the solution as a vector or data.frame. For dynamic runs, the output is averaged over the mean of the run. *CNPDIA1D* always returns the sediment depth and the porosity as the first two columns.

The modeled sedimentlayers increase in thickness from the surface (0.0099 cm) till the deepest layer (> 6 cm). The total modeled sediment depth is 100 cm. There are 100 sediment layers.

```
range(CNPDIAdx(std))
```

```
## [1] 0.00997846 6.31739978
```

```
range(CNPDIAgrid()$x.int)
```

```
## [1]   0 100
```

```
CNPDIAdepth(std)
```

```
##   [1]  0.00498923  0.01530360  0.02631242  0.03806243  0.05060354  0.06398901  0.07827570  0.0935242
##  [10]  0.12717048  0.14571102  0.16549985  0.18662101  0.20916421  0.23322519  0.25890613  0.2863161
##  [19]  0.34679666  0.38012410  0.41569539  0.45366161  0.49418400  0.53743466  0.58359729  0.6328679
##  [28]  0.74158438  0.80149191  0.86543287  0.93367882  1.00651960  1.08426458  1.16724393  1.2558100
##  [37]  1.45123268  1.55891911  1.67385579  1.79653089  1.92746540  2.06721541  2.21637445  2.3755760
##  [46]  2.72685673  2.92042781  3.12703156  3.34754544  3.58290599  3.83411278  4.10223272  4.3884045
##  [55]  5.01984708  5.36779962  5.73917895  6.13556236  6.55863329  7.01018855  7.49214594  8.0065523
##  [64]  9.14159813  9.76705813 10.43462883 11.14714543 11.90763405 12.71932451 13.58566411 14.5103322
##  [73] 16.55062701 17.67491888 18.87490658 20.15568653 21.52269828 22.98174760 24.53903116 26.2011628
##  [82] 29.86868243 31.88964653 34.04667723 36.34893555 38.80619932 41.42890467 44.22819038 47.2159451
##  [91] 53.80847306 57.44124497 61.31860256 65.45701319 69.87405294 74.58848121 79.62032045 84.9909411
## [100] 96.84130011
```

```
head(CNPDIA1D(std), n = 3)
```

```
##            x       por     FDET     SDET       O2      NO3       NO2      NH3      ODU      DIC
## 1 0.00498923 0.8934027 510902.4 598173.8 293.3768 10.03775 0.2707412 1.696420 1.076872 2214.633 0.67
## 2 0.01530360 0.8801069 467199.2 593183.7 279.6515 10.11785 0.8440844 3.115533 3.371520 2245.009 1.03
## 3 0.02631242 0.8664113 426823.5 588428.8 265.0068 10.20429 1.4695547 4.604036 5.905310 2277.503 1.41
##             CaP          Pads       TOC  DICprod  DINprod  DIPprod O2prod  Oxicmin Denitrific Anoxicr
## 1 7.577098e-106 7.577098e-106 1.0323566 4185.042 631.7044 39.48153      0 4180.146   3.596442  1.299
## 2 7.577089e-106 7.577089e-106 1.0089838 4370.281 659.6651 41.22907      0 4364.903   3.964532  1.413
## 3 7.577106e-106 7.577106e-106 0.9873211 4519.984 682.2617 42.64136      0 4514.096   4.355895  1.531
##      Nitri2   Anammox     ODUox ODUoxsurf    ODUdepo FePadsorp    FePdesorp CaPprod CaPdiss Padsorp
## 1  5.396431 0.1238844  21.46428         0 0.01371455 0.2016823 0.0009672296       0       0       0
## 2 16.821537 0.2785527  67.19013         0 0.01492027 0.3090809 0.0010144718       0       0       0
```

```
## 3 29.280604 0.4310702 117.66221         0 0.01616361 0.4239902 0.0010699513       0       0       0
##   MPBNO3uptake MPBNH3uptake MPBPO4uptake MPBDICuptake
## 1            0            0            0            0
## 2            0            0            0            0
## 3            0            0            0            0
```

```
CNPDIAparms(std, which = "Cflux")
```

```
##       parms       units                description
## Cflux  1500 nmolC/cm2/d total organic C deposition
```

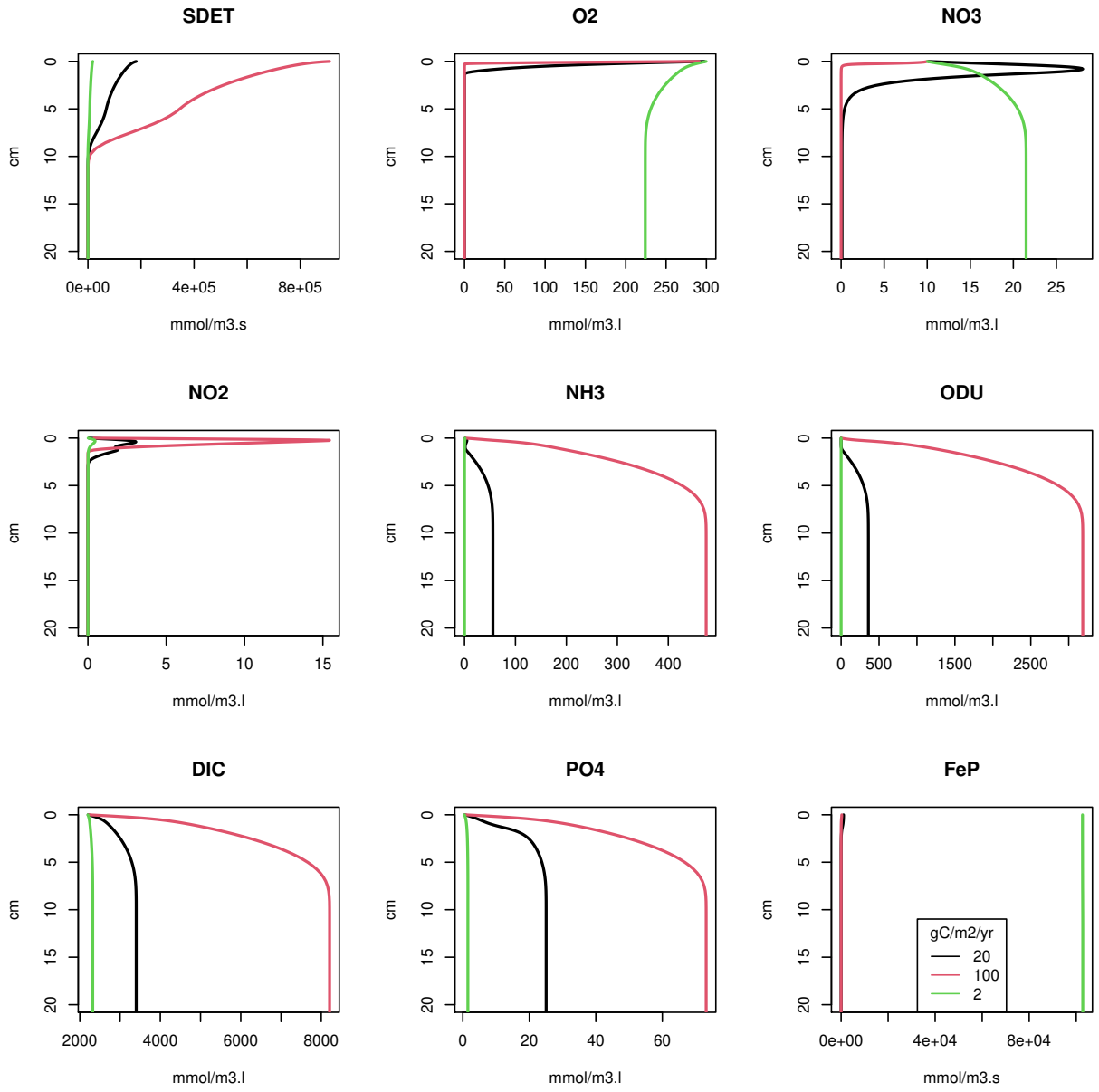
Steady-state applications

The function *CNPDIAsolve* solves for a steady-state condition.

Simple applications

In the frst example, we run the model for different carbon deposition rates (expressed in *nmolC/cm2/d*) and plot the results using *rootSolve*'s *plot* function.

```
convert <- 1e5/12/365
STD1 <- CNPDIAsolve (parms = list(Cflux = 20*convert))    # 20 gC/m2/yr
STD2 <- CNPDIAsolve (parms = list(Cflux = 100*convert))   # 100 gC/m2/yr
STD3 <- CNPDIAsolve (parms = list(Cflux = 2*convert))
plot(STD1, STD2, STD3,
     ylim = c(20, 0), lty = 1, lwd = 2, which = 2:10)
legend("bottom", legend = c(20, 100, 2), lty = 1, col = 1:3, title = "gC/m2/yr")
```



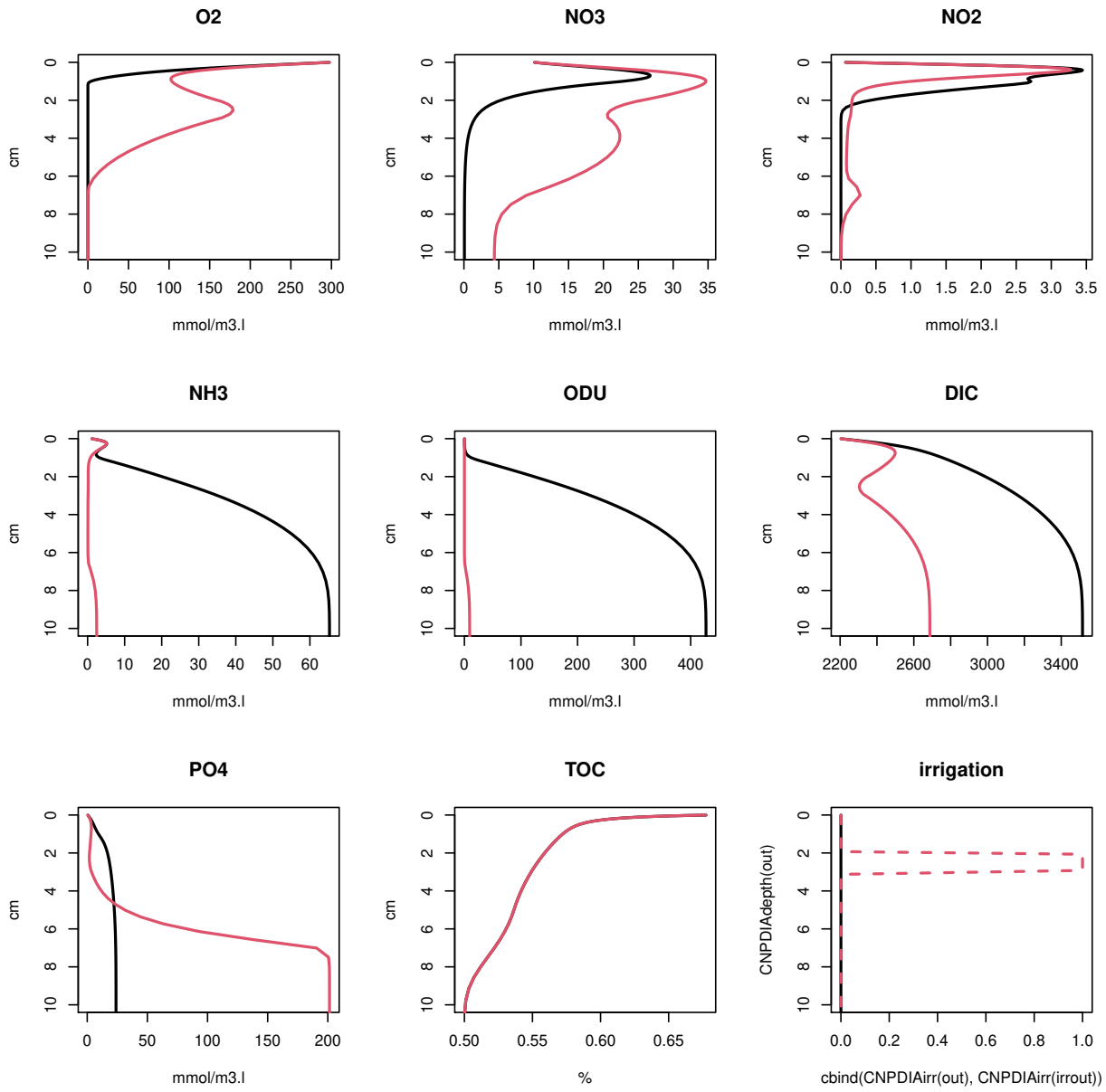
User-inputted profiles

By default porosity, bioturbation, and bio-irrigation profiles are generated based on parameter settings. However, it is possible to directly impose profiles for these quantities.

In the following example, an irrigation profile is generated where there is substantial irrigation only in a certain section of the sediment ([2-3 cm]).

```
Grid <- CNPDIAgrid()
Irr  <- rep(0, Grid$N)
Irr[Grid$x.mid > 2 &  Grid$x.mid < 3] <- 1
out <- CNPDIAsolve()
irrout <- CNPDIAsolve(irrigation = Irr)
plot(out, irrout,
```

```
     ylim = c(10, 0), lty = 1, lwd = 2, which = c(3:9))
plot(out, irrout,
     ylim = c(10, 0), lty = 1, lwd = 2, which = c("TOC"), mfrow = NULL)
matplot(x=cbind(CNPDIAirr(out), CNPDIAirr(irrout)), y = CNPDIAdepth(out),
     ylim = c(10, 0), type = "l", lty = 1:2, lwd = 2, main = "irrigation")
```



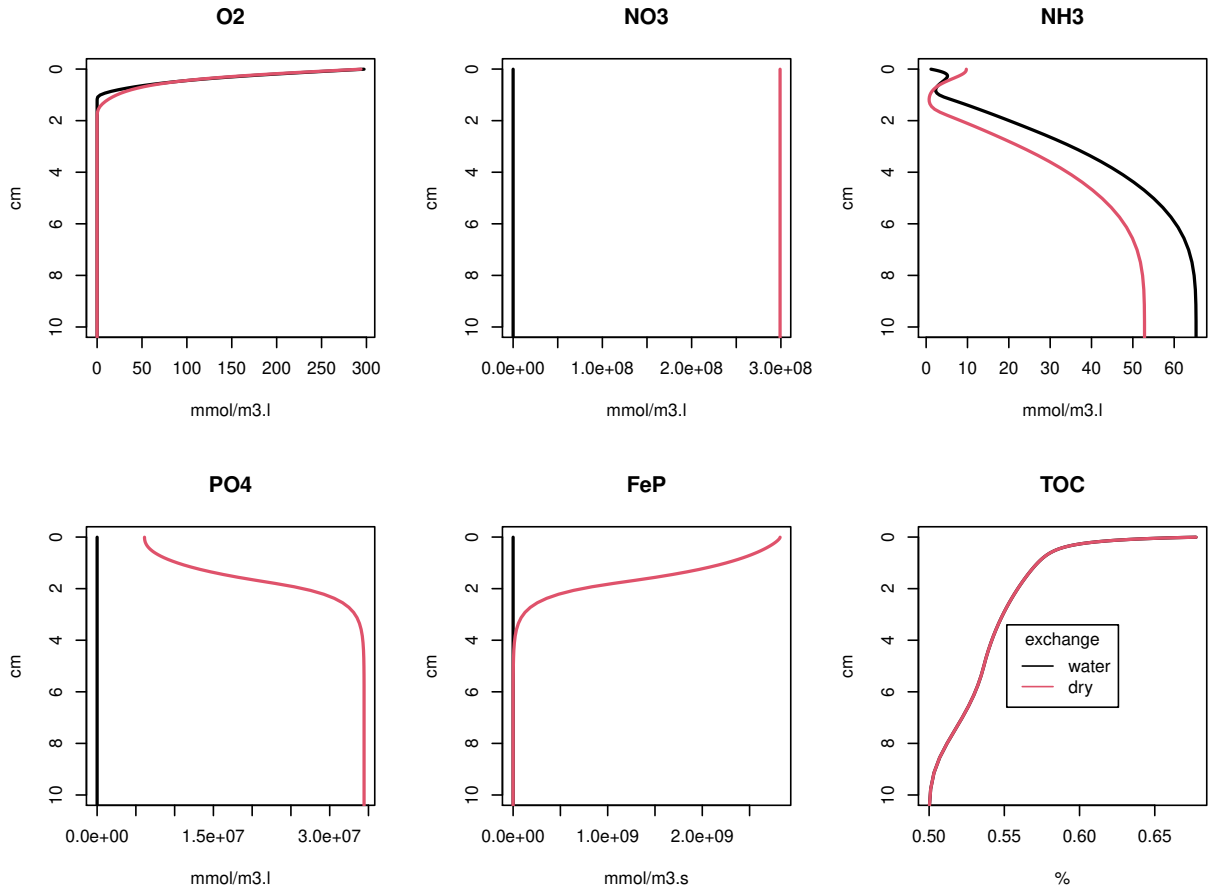
Dry flats (but moist sediment)

When flats are dry, the exchange is governed by a piston velocity. The exchange of substances at the upper interface can take on two modes: exchange with water overlying the sediment or exchange with the atmosphere.

When the parameter *gasflux*, or forcing function *gasfluxForc* is 0, this means that the sediment is submersed.

When they have a positive value, equal to the piston velocity, (units [cm/d]), this means that the sediment is exposed to the air. In that case, only oxygen and DIC are exchanged with the air at the upper interface, while there is no exchange for NH3, NO3, NO2, ODU, PO4. Deposition of the two carbon fractions and of FeP, CaP continues.

```
out    <- CNPDIAsolve()
outdry <- CNPDIAsolve(parms = list(gasflux = 1e2))


plot(out, outdry, ylim = c(10, 0), lty = 1, lwd = 2,
     which = c("O2","NO3","NH3","PO4","FeP","TOC"))
legend("center", col = 1:2, title = "exchange", legend = c("water","dry"), lty = 1)
```




```
print(CNPDIAbudgetO2(outdry))
```

```
## $Fluxes
##                    O2           ODU
## surface 6.078181e+02  0.000000e+00
## bottom  3.313266e-35  1.709773e-13
## perturb 0.000000e+00  0.000000e+00
## netin   6.078181e+02 -1.709773e-13
##
## $Rates
##              Nitrification ODUoxidation ODUoxid.dist OxicMineralisation MPBO2production MPBNO3reducti
## nmolO2/cm2/d      148.3675  1.34144e-07            0           459.4506               0
```

```
##              MPBO2respiration     Total
## nmolO2/cm2/d                0 -607.8181
##
## $Losses
## [1] 3.313266e-35
##
## $dC
## [1] 2.080469e-11
##
## $Delta
## [1] 2.080469e-11
##
## $Fluxmat
##         Ext       O2     NO2      NO3      DIC     Oxidant       Burial
## Ext       0 607.8181   0.000  0.00000   0.0000 0.00000e+00 0.000000e+00
## O2        0   0.0000 111.662 36.70549 459.4506 1.34144e-07 3.313266e-35
## NO2       0   0.0000   0.000  0.00000   0.0000 0.00000e+00 0.000000e+00
## NO3       0   0.0000   0.000  0.00000   0.0000 0.00000e+00 0.000000e+00
## DIC       0   0.0000   0.000  0.00000   0.0000 0.00000e+00 0.000000e+00
## Oxidant   0   0.0000   0.000  0.00000   0.0000 0.00000e+00 0.000000e+00
## Burial    0   0.0000   0.000  0.00000   0.0000 0.00000e+00 0.000000e+00
```

```
print(CNPDIAbudgetN(outdry))
```

```
## $Fluxes
##                FDET_N        SDET_N      NO3           NO2           NH3        N2        Total
## surface  6.792453e+01  7.547170e+00   0.0000  0.000000e+00  0.000000e+00 -34.50018 4.097152e+01
## bottom  4.509133e-224 7.762939e-104  40.9715  5.186522e-32  1.664411e-05   0.00000 4.097152e+01
## perturb  0.000000e+00  0.000000e+00   0.0000  0.000000e+00  0.000000e+00   0.00000 0.000000e+00
## netin    6.792453e+01  7.547170e+00 -40.9715 -5.186522e-32 -1.664411e-05 -34.50018 1.992727e-09
##
## $Rates
##             NH3production Denitrification MPBNO3consumption MPBNH3consumption Nitrification1 Nitrifi
## nmolN/cm2/d       75.4717        32.43948                 0                 0       74.44133       73
##             NH3adsorption N2production MPBNdeath NH3prodMPBdeath DETNprodMPBdeath
## nmolN/cm2/d   9.40811e-06     34.50018         0               0                0
##
## $Losses
## [1] 75.4717
##
## $dC
##           DET           NH3           NO3           NO2          MPBN           sum
##  1.421085e-14  1.005158e-09 -7.354117e-12  9.949115e-10  0.000000e+00  1.992729e-09
##
## $Delta
## [1] 1.992731e-09
##
## $Fluxmat
##        Ext     DET     NH3      NO3      NO2 MPB        N2        Burial
## Ext      0 75.4717  0.0000  0.00000  0.00000   0  0.000000  0.000000e+00
## DET      0  0.0000 75.4717  0.00000  0.00000   0  0.000000 7.762939e-104
## NH3      0  0.0000  0.0000  0.00000 74.44133   0  1.030348  1.664411e-05
## NO3      0  0.0000  0.0000  0.00000  0.00000   0 32.439484  4.097150e+01
## NO2      0  0.0000  0.0000 73.41098  0.00000   0  1.030348  5.186522e-32
## MPB      0  0.0000  0.0000  0.00000  0.00000   0  0.000000  0.000000e+00
```

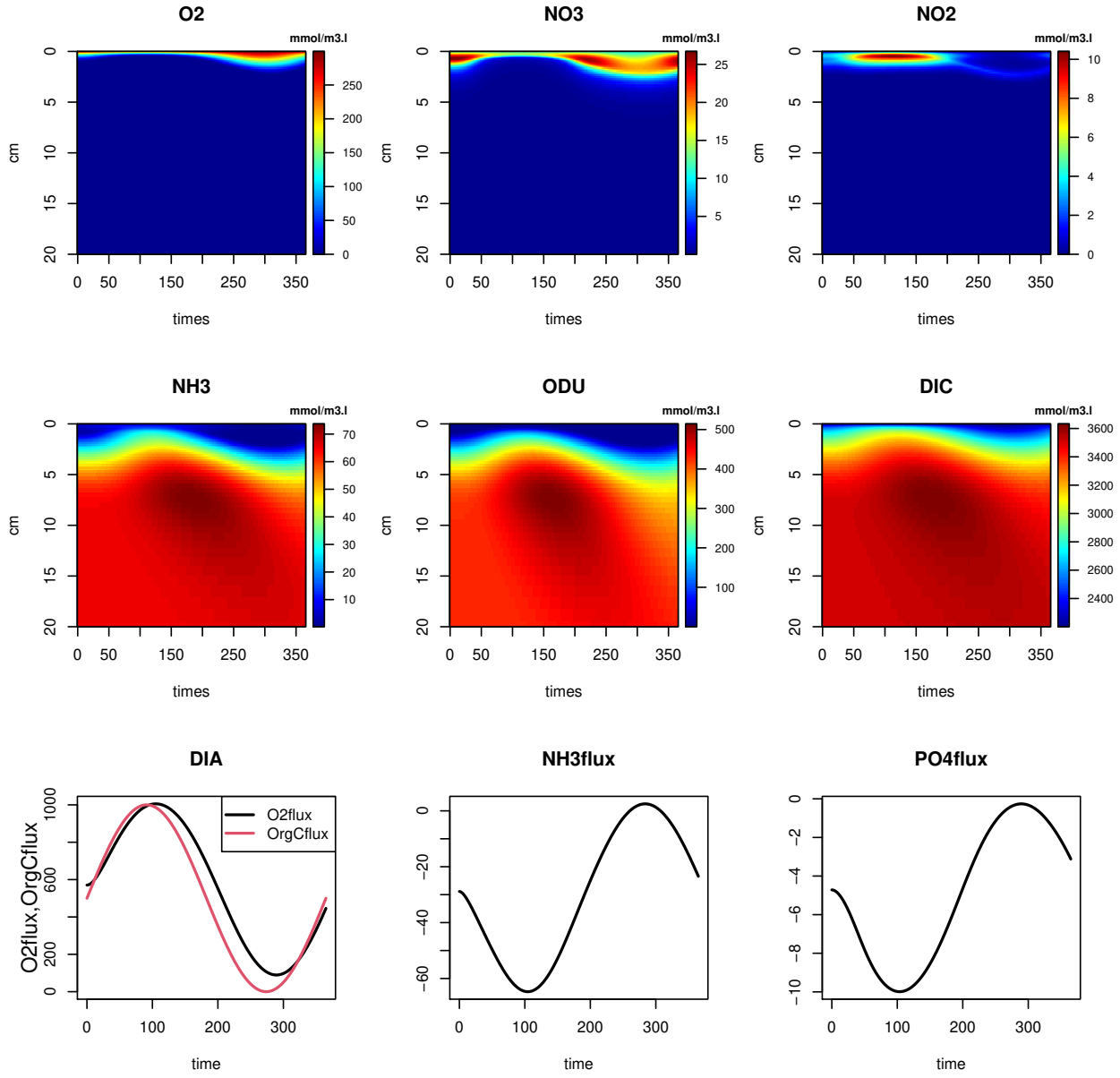
```
## N2       0  0.0000  0.0000  0.00000  0.00000   0  0.000000  0.000000e+00
## Burial   0  0.0000  0.0000  0.00000  0.00000   0  0.000000  0.000000e+00
```

Dynamic runs with sinusoidal forcing

Carbon input

In the first dynamic run, a sinusoidal variation in time is used for the C flux, with amplitude = 1, the other parameters are left equal to the default.

```
DIA <- CNPDIAdyna (CfluxForc = list(amp = 1))
image2D(DIA, ylim = c(20, 0), which = 3:8, mfrow = c(3,3))
matplot.0D(DIA, which = c("OrgCflux", "O2flux"), mfrow = NULL, lty = 1, lwd = 2)
plot(DIA, which = c("NH3flux", "PO4flux"), mfrow = NULL, lwd = 2)
```



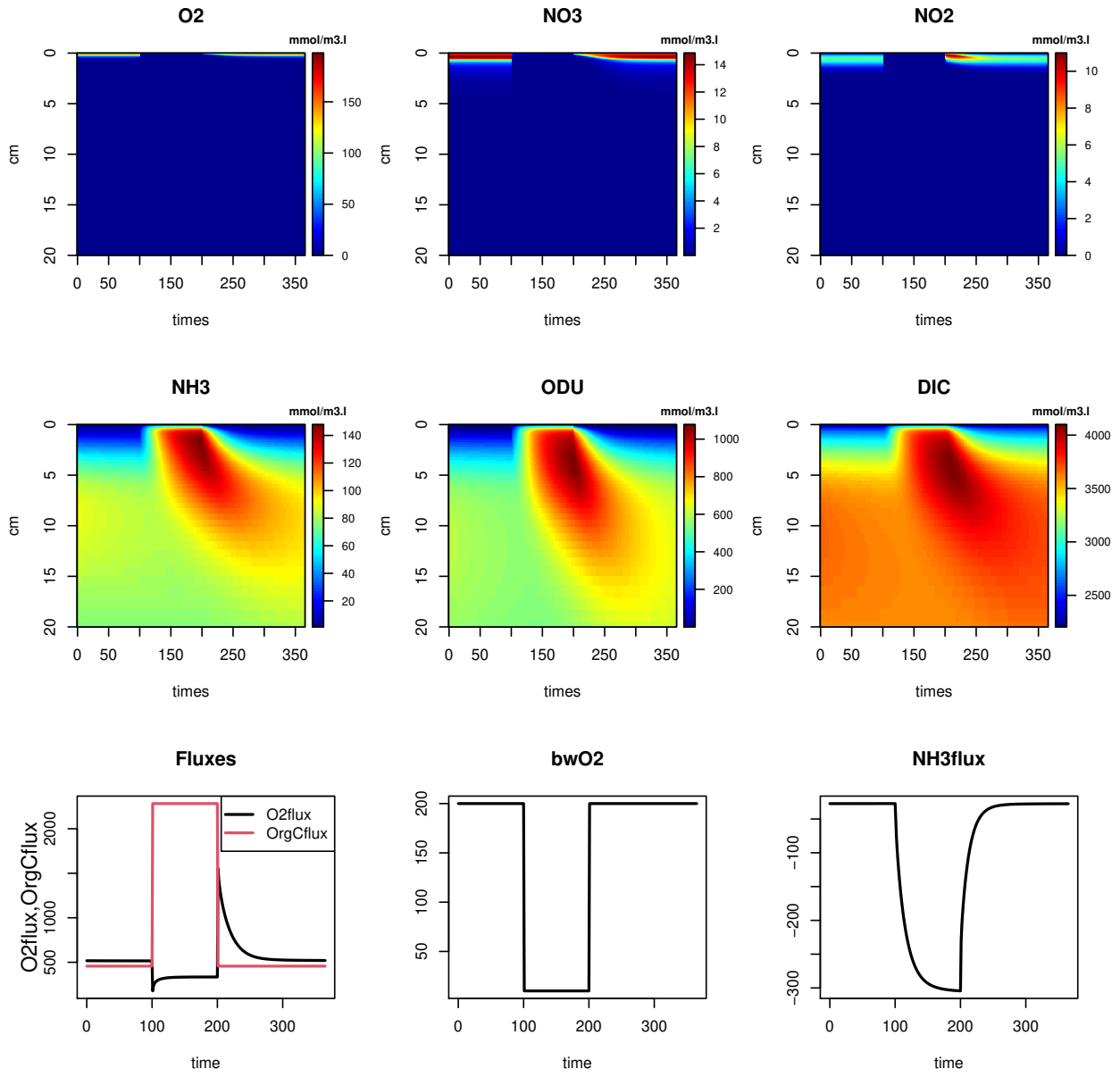
dynamic runs with imposed forcing function time series

Carbon flux and bottom water concentrations

We can also impose a time-series. Here we impose this for the carbon flux, and for the Oxygen bottom water concentration.

```
fluxforcdat <- data.frame(time = c(0, 100, 101, 200, 201, 365),
                          flux = c(20, 20, 100, 100, 20, 20)*1e5/12/365)
O2forcdat <- data.frame(time = c(0, 100, 101, 200, 201, 365),
                        conc = c(200, 200, 10, 10, 200, 200))
DIA <- CNPDIAdyna (CfluxForc = list(data = fluxforcdat),
```

```
                    O2bwForc = list(data = O2forcdat), spinup = 0:365)
image2D(DIA, ylim = c(20, 0), which = 3:8, mfrow = c(3,3))
matplot.0D(DIA, which = c("OrgCflux", "O2flux"), mfrow = NULL, lty = 1, lwd = 2, main = "Fluxes")
plot(DIA, which = c("bwO2","NH3flux"), mfrow = NULL, lwd = 2)
```

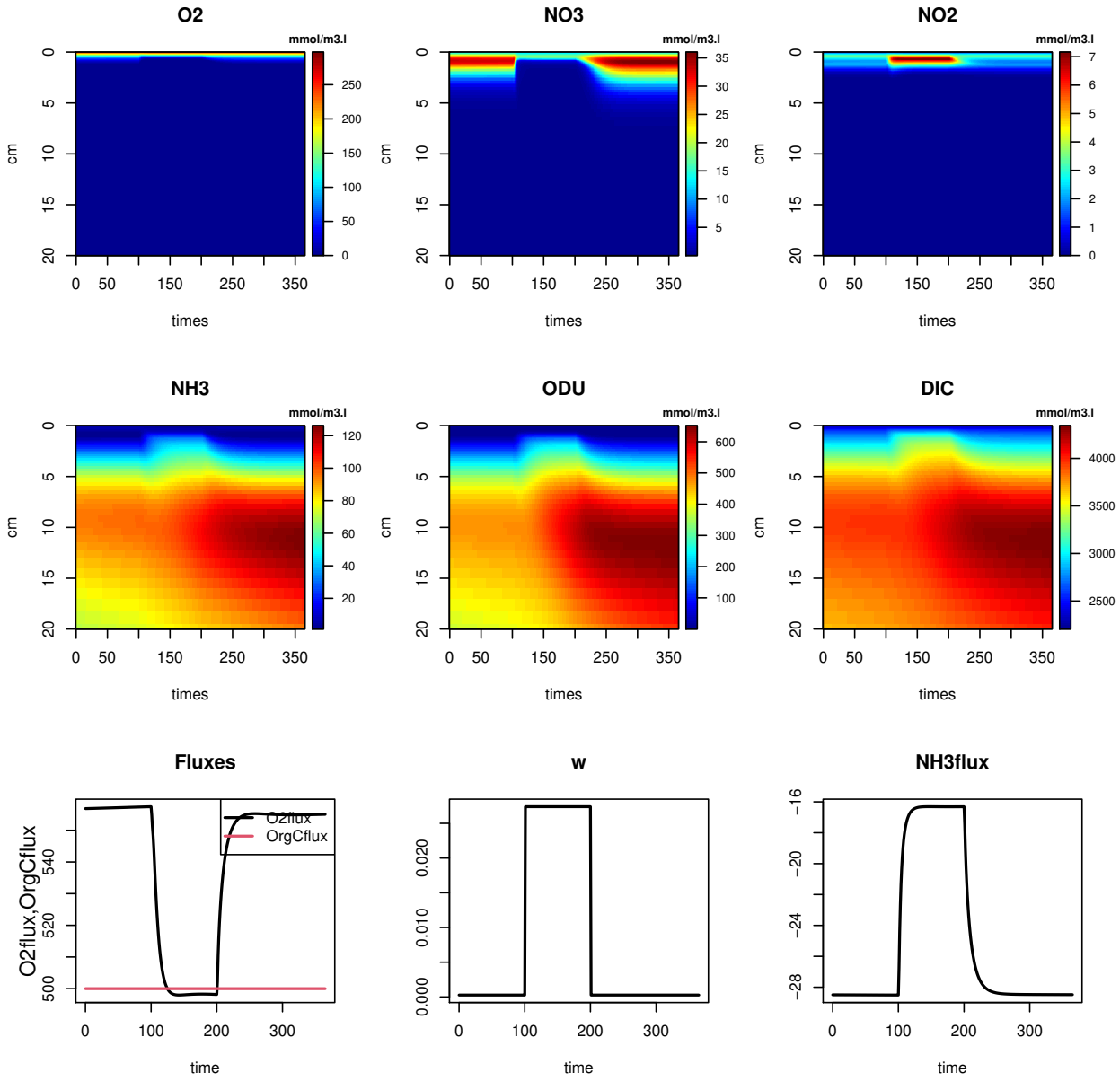


Flux and sedimentation rates

Other variables that are forced are *w*, *biot*, *irr* for the sedimentation rate, bioturbation rate and irrigation rates respectively, ...

```
fluxforcdat <- data.frame(time = c(0, 100, 101, 200, 201, 365),
                          flux = c(20, 20, 100, 100, 20, 20)*1e5/12/365)
seddat <- data.frame(time = c(0, 100, 101, 200, 201, 365),
```

```
                      w = c(0.1, 0.1, 10, 10, 0.1, 0.1)/365)  #cm/d
DIA <- CNPDIAdyna (CfluxfForc = list(data = fluxforcdat),
                   wForc = list(data = seddat),
                   spinup = 0:365)
image2D(DIA, ylim = c(20, 0), which = 3:8, mfrow = c(3,3))
matplot.0D(DIA, which = c("OrgCflux", "O2flux"), mfrow = NULL, lty = 1, lwd = 2, main = "Fluxes")
plot(DIA, which = c("w", "NH3flux"), mfrow = NULL, lwd = 2)
```



Deposition-erosion rates.

Particles often go through a repeated deposition-erosion cycle. In the first case, sedimentation rate, w is positive, and there is solid deposition; in the latter case, w is negative and there is no carbon deposition, $Cdepo$.

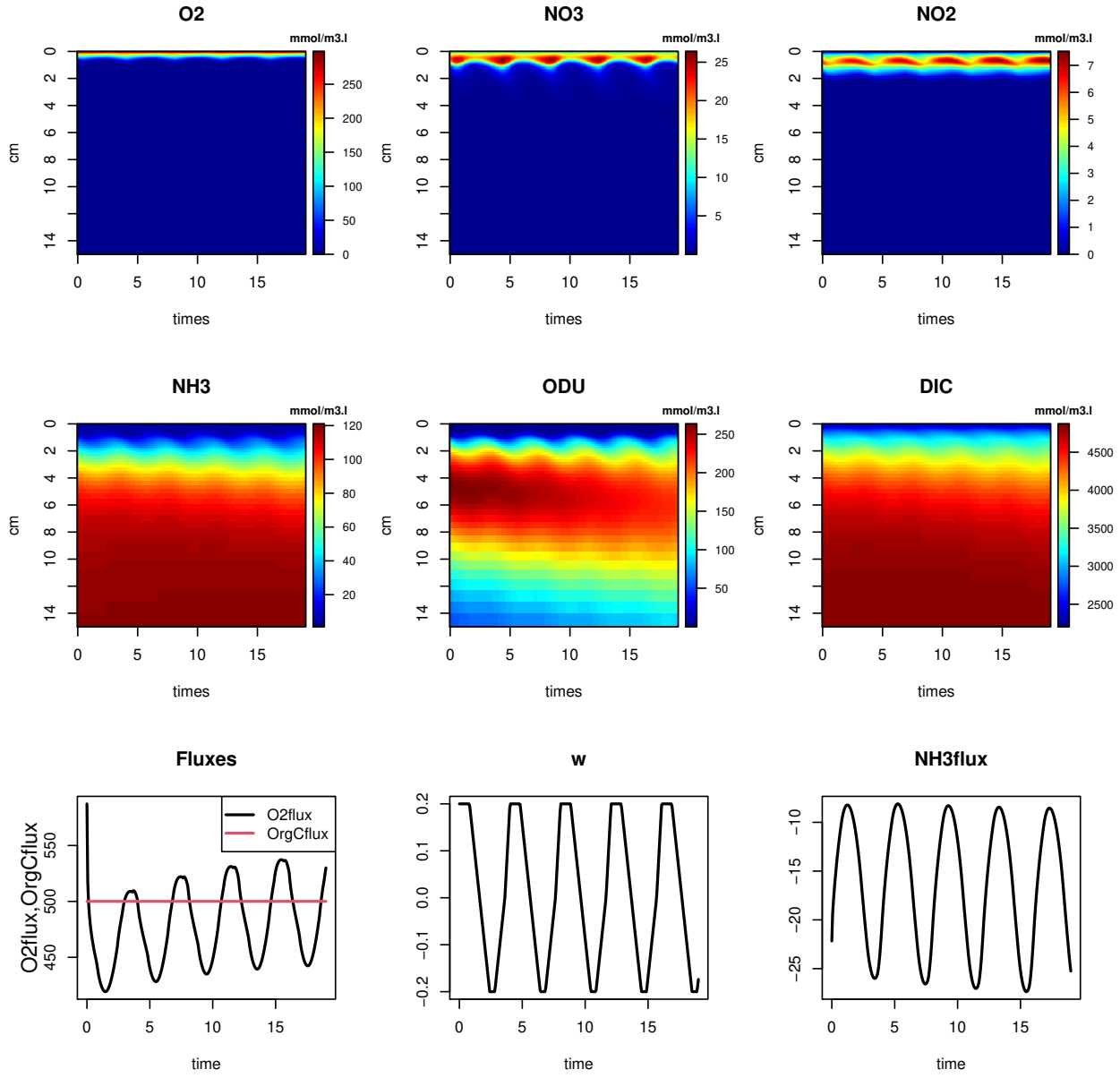
In the first run only the sedimentation rate fluctuates.

```
FF <- c(20, 30, 20, 10, 0, 0, 0, 0, 0, 0)*1e5/12/365
SS <- c(0.2, 0.2, 0.2, 0.1, 0.0,-0.1,-0.2,-0.2,-0.1, 0)  #cm/d
FF <- rep(FF, times = 10)
Fluxforcdat <- data.frame(time = seq(0, to = 39.8, length.out = length(FF)),
                          flux = FF)

SS <- rep(SS, times = 10)
Seddat <- data.frame(time = seq(0, to = 39.8, length.out = length(SS)),
                     w = SS)

times <- seq(0, 19, length.out = 300)

P <- list(Cflux = FF[1], w = SS[1])
std <- CNPDIAsolve(parms = P)
DIA <- CNPDIAdyna (wForc = list(data = Seddat), times = times, spinup = times,
                   yini = std$y)
CNPDIAbudgetC(std, DIA, which = "Rates")
```

```
## Warning in cbind(unlist(budgdC), budg$dC): number of rows of result is not a multiple of vector leng
```

```
##                            [,1]      [,2]
## OxicMineralisation   158.39998 345.05619
## Denitrification       21.31126  24.03150
## AnoxicMineralisation 234.26625  78.15896
## TotalMineralisation  413.97750 447.24665
## CaPprecipitation       0.00000   0.00000
## CaPdissolution         0.00000   0.00000
## MPBDICuptake           0.00000   0.00000
## MPBResp                0.00000   0.00000
## MPBCdeath              0.00000   0.00000
## EPSproduction          0.00000   0.00000
## EPSmineralisation      0.00000   0.00000
## FDETprodMPBdeath       0.00000   0.00000
## DICprodMPBdeath        0.00000   0.00000
```

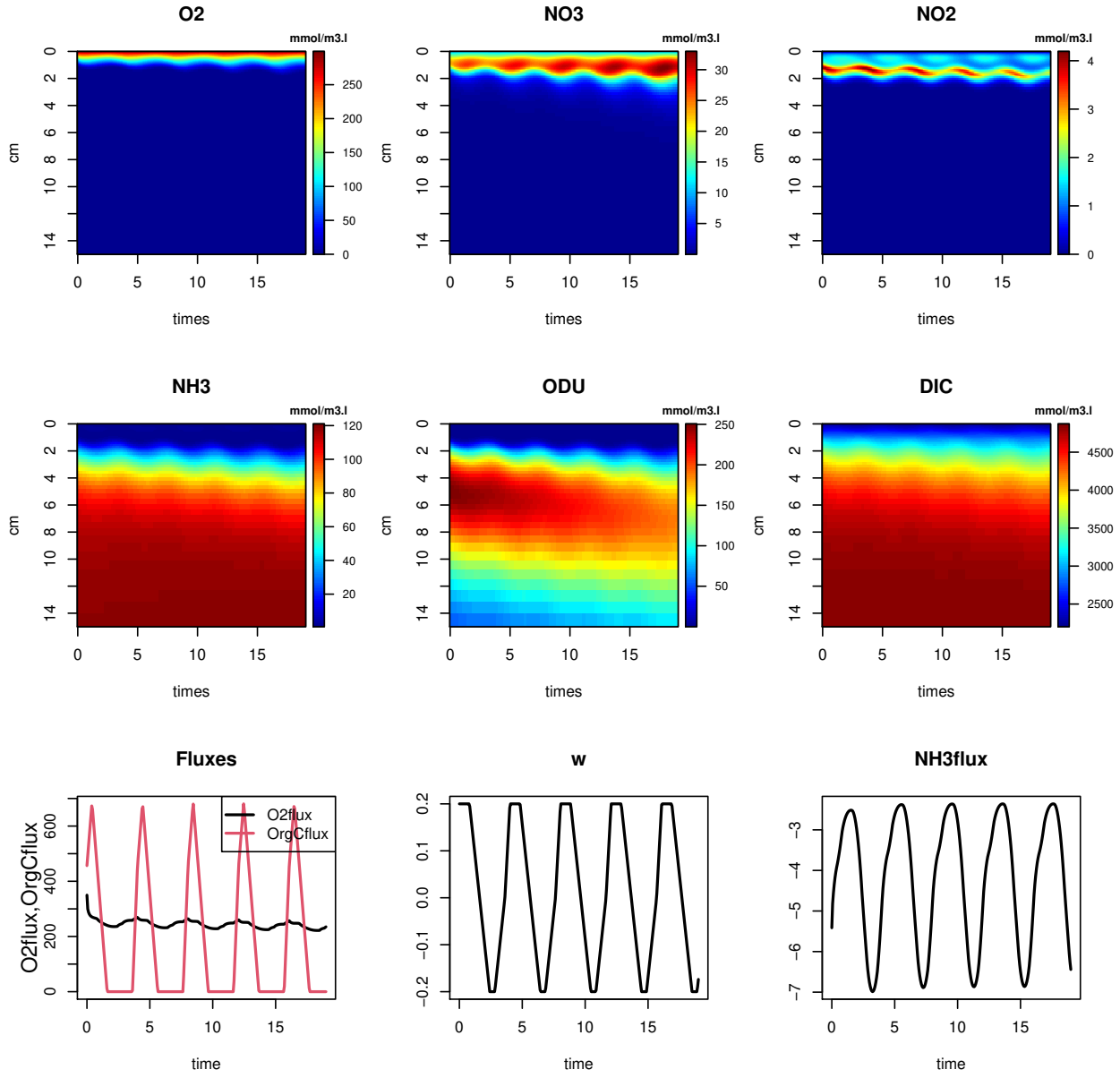
```
image2D(DIA, ylim = c(15, 0), which = 3:8, mfrow = c(3,3))
matplot.0D(DIA, which = c("OrgCflux", "O2flux"), mfrow = NULL, lty = 1, lwd = 2, main = "Fluxes")
plot(DIA, which = c("w","NH3flux"), mfrow = NULL, lwd = 2)
```



In the second run both the sedimentation rate and the carbon flux fluctuate.

```
DIA2 <- CNPDIAdyna (CfluxForc = list(data = Fluxforcdat), wForc = list(data = Seddat),
                    times = times, spinup = times, yini = std$y)
```

```
image2D(DIA2, ylim = c(15, 0), which = 3:8, mfrow = c(3,3))
matplot.0D(DIA2, which = c("OrgCflux", "O2flux"), mfrow = NULL, lty = 1, lwd = 2, main = "Fluxes")
plot(DIA2, which = c("w", "NH3flux"), mfrow = NULL, lwd = 2)
```




```
print(CNPDIAbudgetC(DIA, DIA2, which = "Rates"))
```

```
##                             [,1]      [,2]
## OxicMineralisation    345.05619 160.16106
## Denitrification        24.03150  12.03442
## AnoxicMineralisation   78.15896  38.38162
## TotalMineralisation   447.24665 210.57710
## CaPprecipitation        0.00000   0.00000
## CaPdissolution          0.00000   0.00000
## MPBDICuptake            0.00000   0.00000
## MPBResp                 0.00000   0.00000
## MPBCdeath               0.00000   0.00000
## EPSproduction           0.00000   0.00000
## EPSmineralisation       0.00000   0.00000
```

```
## FDETprodMPBdeath       0.00000   0.00000
## DICprodMPBdeath        0.00000   0.00000
```

Dynamic runs with explicitly modeled bottom water conditions

Incubation experiments

The simulation is initiated with the steady-state conditions, while keeping the bottom water conditions constant.

```
std <- CNPDIAsolve(dynamicbottomwater = FALSE, parms = list(Cflux = 100*1e5/12/365))
CNPDIAbudgetO2(std, which = "Fluxes")
```

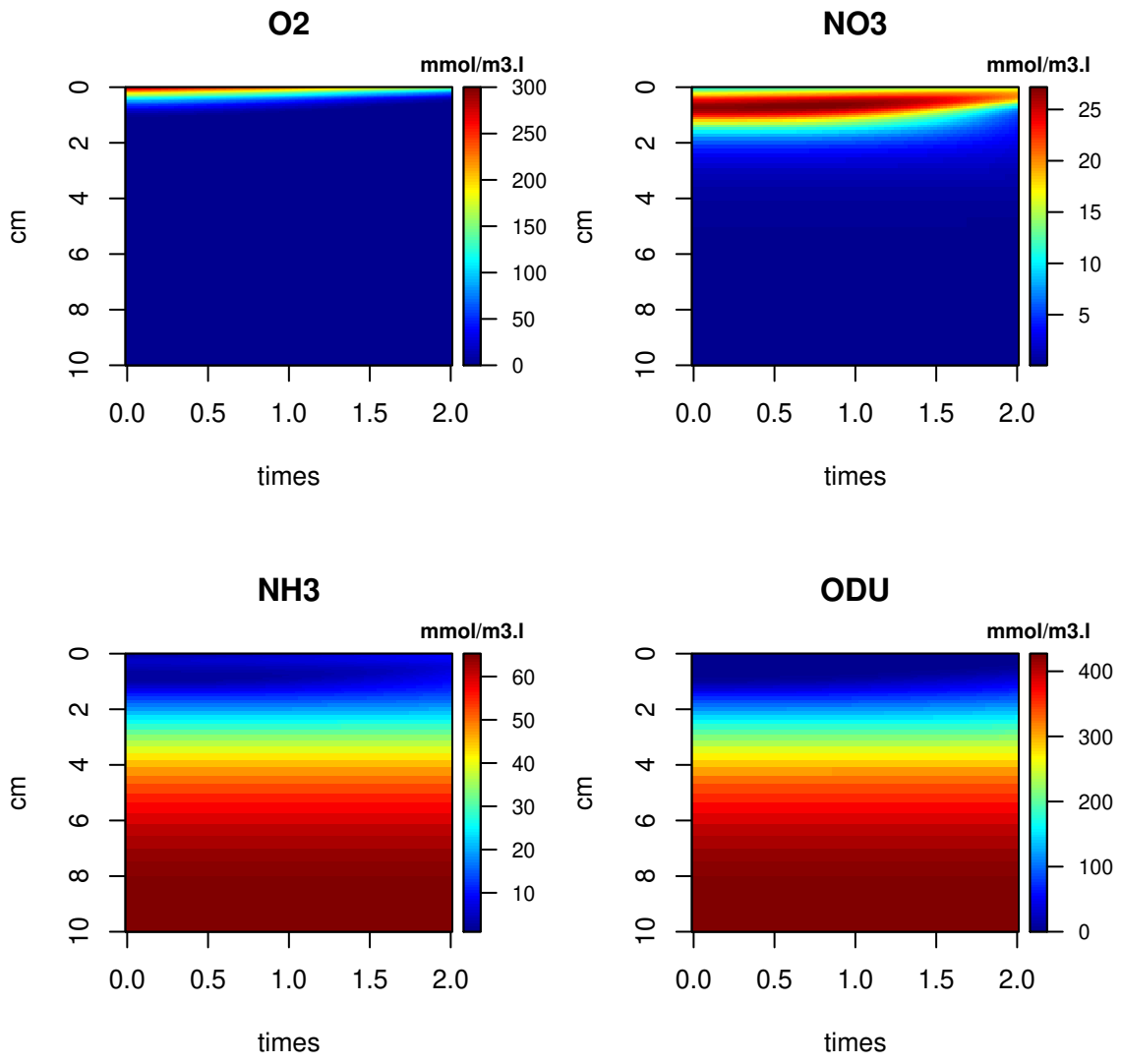
```
##                O2           ODU
## surface 1889.086 -5.246525e+02
## bottom     0.000  4.363768e-04
## perturb    0.000  0.000000e+00
## netin   1889.086 -5.246529e+02
```

The initial conditions for the dynamic bottom water concentration run needs to have the bottom water concentrations as the first row.

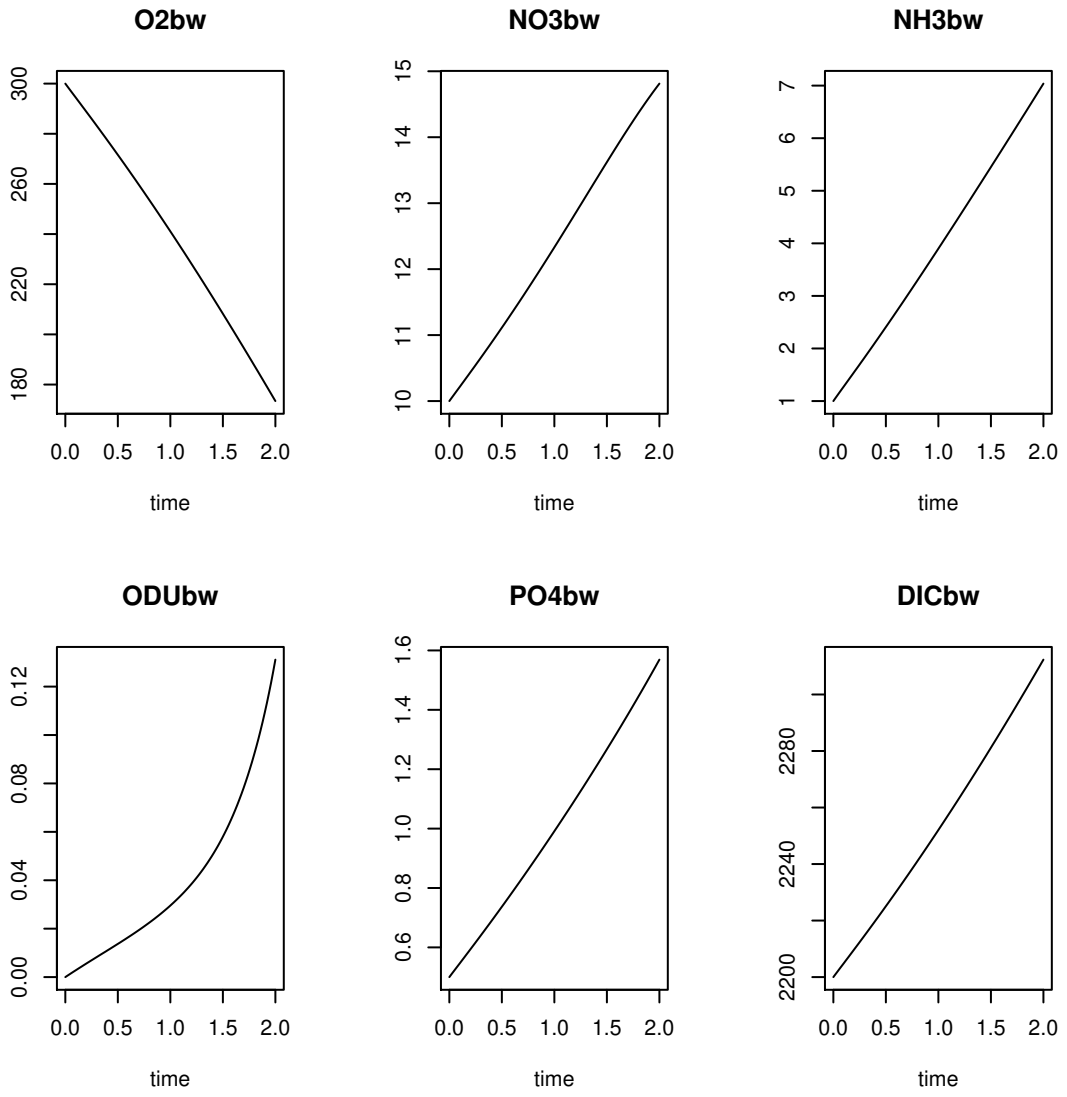
The model is run for two days.

```
std <- CNPDIAsolve()
P <- CNPDIAparms(std, as.vector = TRUE)[c("O2bw","NO3bw","NO2bw","NH3bw","ODUbw","DICbw","PO4bw")]
# order of state variables, FDET,SDET,O2,NO3,NH3,ODU,PO4,FeP,CaP,DIC
BW <- c(0, 0, P, 0, 0, 0)
dyn <- CNPDIAdyna(dynamicbottomwater = TRUE, yini = rbind(BW, std$y),
      parms = list(Cflux = 100*1e5/12/365), times = seq(0, 2, length.out = 100))
```

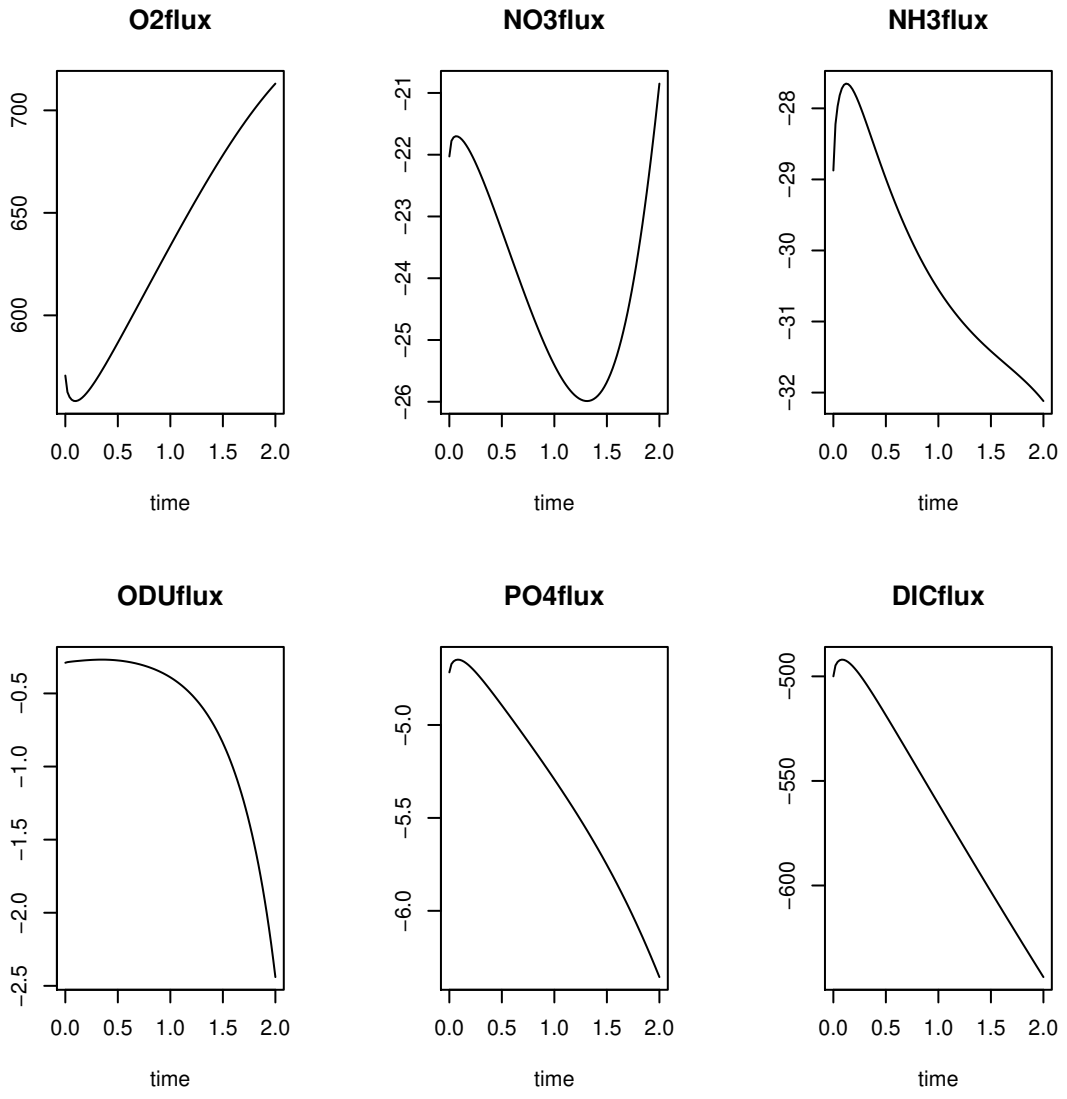
```
image2D(dyn, which = c("O2", "NO3", "NH3","ODU"), ylim = c(10,0))
```




```
plot(dyn, which = c("O2bw","NO3bw","NH3bw","ODUbw","PO4bw","DICbw"))
```




```
plot(dyn, which = c("O2flux","NO3flux","NH3flux","ODUflux","PO4flux","DICflux"))
```



Perturbation runs

See vignette ("CNPDIAperturb")

References

Soetaert K, PMJ Herman and JJ Middelburg, 1996a. A model of early diagenetic processes from the shelf to abyssal depths. Geochimica Cosmochimica Acta, 60(6):1019-1040.

Soetaert K, PMJ Herman and JJ Middelburg, 1996b. Dynamic response of deep-sea sediments to seasonal variation: a model. Limnol. Oceanogr. 41(8): 1651-1668.

APPENDIX

Parameters and default values.

```
knitr:::kable(CNPDIAparms())
```

	parms	units	description
Cflux	500.0000000	nmolC/cm2/d	total organic C deposition
pFast	0.9000000	-	part FDET in carbon flux
FePflux	0.0000000	nmolP/cm2/d	deposition rate of FeP
CaPflux	0.0000000	nmolP/cm2/d	deposition rate of CaP
rFast	0.0684932	/d	decay rate FDET
rSlow	0.0001370	/d	decay rate SDET
NCrFdet	0.1509434	molN/molC	NC ratio FDET
NCrSdet	0.1509434	molN/molC	NC ratio SDET
PCrFdet	0.0094340	molP/molC	PC ratio FDET
PCrSdet	0.0094340	molP/molC	PC ratio SDET
BCupLiq	2.0000000	-	upper boundary liq. 1:flux, 2:conc, 3:0-grad
BCdownLiq	3.0000000	-	lower boundary liq. 1:flux, 2:conc, 3:0-grad
O2bw	300.0000000	mmol/m3	upper boundary O2 -if BC=1: flux, 2:conc
NO3bw	10.0000000	mmol/m3	upper boundary NO3 -if BC=1: flux, 2:conc
NO2bw	0.0000000	mmol/m3	upper boundary NO2 -if BC=1: flux, 2:conc
NH3bw	1.0000000	mmol/m3	upper boundary NH3 -if BC=1: flux, 2:conc
ODUbw	0.0000000	mmol/m3	upper boundary ODU -if BC=1: flux, 2:conc
PO4bw	0.5000000	mmol/m3	upper boundary PO4 -if BC=1: flux, 2:conc
DICbw	2200.0000000	mmol/m3	upper boundary DIC -if BC=1: flux, 2:conc
O2dw	NA	mmol/m3	lower boundary O2 -if BC=1: flux, 2:conc
NO3dw	NA	mmol/m3	lower boundary NO3 -if BC=1: flux, 2:conc
NO2dw	NA	mmol/m3	lower boundary NO2 -if BC=1: flux, 2:conc
NH3dw	NA	mmol/m3	lower boundary NH3 -if BC=1: flux, 2:conc
ODUdw	NA	mmol/m3	lower boundary ODU -if BC=1: flux, 2:conc
PO4dw	NA	mmol/m3	lower boundary PO4 -if BC=1: flux, 2:conc
DICdw	NA	mmol/m3	lower boundary DIC -if BC=1: flux, 2:conc
w	0.0000003	cm/d	advection rate
biot	0.0027397	cm2/d	bioturbation coefficient
biotdepth	5.0000000	cm	depth of mixed layer
biotatt	1.0000000	cm	attenuation coeff below biotdepth
irr	0.0000000	/d	bio-irrigation rate
irrdepth	5.0000000	cm	depth of irrigated layer
irratt	1.0000000	cm	attenuation coeff below irrdepth
gasflux	0.0000000	cm/d	piston velocity for dry flats
NH3Ads	1.3000000	-	Adsorption coeff ammonium
rnitri1	20.0000000	/d	Max nitrification rate step1 (NH3ox)
rnitri2	20.0000000	/d	Max nitrification rate step2 (NO2ox)
ksO2nitri	1.0000000	mmolO2/m3	half-sat O2 in nitrification
ranammox	0.1000000	/d	Anammox rate
ksNO2anammox	0.1000000	mmolN/m3	half-sat NO2 in anammox
rODUox	20.0000000	/d	Max rate ODU oxidation in one layer
rSurfODUox	0.0000000	/d	Max rate ODU oxidation with BW O2
ODUoxdepth	5.0000000	cm	Max depth ODU oxidation with BW O2
ODUoxatt	1.0000000	cm	depth attenuation ODU oxidation
ksO2oduox	1.0000000	mmolO2/m3	half-sat O2 in oxidation of ODU

	parms	units	description
ksO2oxic	3.0000000	mmolO2/m3	half-sat O2 in oxic mineralisation
ksNO3denit	30.0000000	mmolNO3/m3	half-sat NO3 in denitrification
kinO2denit	1.0000000	mmolO2/m3	half-sat O2 inhib denitrification
kinNO3anox	1.0000000	mmolNO3/m3	half-sat NO3 inhib anoxic degr
kinO2anox	1.0000000	mmolO2/m3	half-sat O2 inhib anoxic min
pdepo	NA	-	part ODU prod lost (NA:estimated from w)
rdepo	0.0000000	/d	ODU removal rate
temperature	10.0000000	dgC	temperature
salinity	35.0000000	psu	salinity
TOC0	0.5000000	%	refractory Carbon conc
rFePdesorp	0.0100000	/d	rate FeP desorption
rFePadsorp	0.3000000	/d	rate FeP adsorption
rCaPprod	0.0000000	/d	rate CaP production
rCaPdiss	0.0000000	/d	rate CaP dissolution
CPrCaP	0.2869565	molC/molP	C:Pratio in CaP
rPads	0.0000000	/d	adsorption rate PO4
rPdes	0.0000000	/d	desorption rate of adsorbed P
maxPads	1000.0000000	mmolP/m3 solid	Max adsorbed P concentration
por0	0.9000000	-	surface porosity
pordeep	0.5000000	-	deep porosity
porcoeff	0.3000000	cm	porosity decay coefficient
formationtype	1.0000000	-	formationfactor, 1=sand,2=fine sand,3=general
dilution	0.0000000	/d	relaxation towards background conc
Hwater	10.0000000	cm	height of overlying water
Cfall	100.0000000	cm/d	fall speed of organic C (FDET, SDET)
FePfall	100.0000000	cm/d	fall speed of FeP
CaPfall	100.0000000	cm/d	fall speed of CaP
MPBprod	0.0000000	mmol/m3/d	maximal MPB production rate
kdSed	20.0000000	/cm	sedimentary light extinction coefficient
kNO3upt	3.0000000	mmolN/m3	NO3 limitation constant MPB
kNH3upt	3.0000000	mmolN/m3	NH3 limitation constant MPB
kPO4upt	0.1000000	mmolP/m3	P limitation constant MPB
kDICupt	1.0000000	mmolC/m3	C limitation constant MPB

State variables

```
knitr:::kable(CNPDIAsvar())
```

names	units	description
FDET	mmolC/m3 solid	Fast decaying Detritus (solid)
SDET	mmolC/m3 solid	Slow decaying Detritus (solid)
O2	mmolO/m3 liquid	Oxygen (liquid)
NO3	mmolN/m3 liquid	Nitrate (liquid)
NO2	mmolN/m3 liquid	Nitrite (liquid)
NH3	mmolN/m3 liquid	Ammonium/ammonia (liquid)
ODU	mmolO/m3 liquid	Oxygen Demand Units (liquid)
DIC	mmolC/m3 liquid	Dissolved Inorganic Carbon (liquid)
PO4	mmolP/m3 liquid	Phosphate (liquid)
FeP	mmolP/m3 solid	Iron-bound P (solid)
CaP	mmolP/m3 solid	Ca-bound P (solid)

names	units	description
Pads	mmolP/m3 solid	Adsorbed P (solid)

Zero-D ordinary variables

```
knitr:::kable(CNPDIAOD())
```

names	values	units	description
O2flux	NA	nmolO2/cm2/d	O2 influx sediment-water
O2deepflux	NA	nmolO2/cm2/d	O2 efflux lower boundary
NO3flux	NA	nmolN/cm2/d	NO3 influx sediment-water
NO3deepflux	NA	nmolN/cm2/d	NO3 efflux lower boundary
NO2flux	NA	nmolN/cm2/d	NO2 influx sediment-water
NO2deepflux	NA	nmolN/cm2/d	NO2 efflux lower boundary
NH3flux	NA	nmolN/cm2/d	NH3 influx sediment-water
NH3deepflux	NA	nmolN/cm2/d	NH3 efflux lower boundary
ODUflux	NA	nmolO2/cm2/d	ODU influx sediment-water
ODUdeepflux	NA	nmolO2/cm2/d	ODU efflux lower boundary
PO4flux	NA	nmolP/cm2/d	PO4 influx sediment-water
PO4deepflux	NA	nmolP/cm2/d	PO4 efflux lower boundary
DICflux	NA	nmolC/cm2/d	DIC influx sediment-water
DICdeepflux	NA	nmolC/cm2/d	DIC efflux lower boundary
FDETflux	NA	nmolC/cm2/d	FDET flux to sediment
FDETdeepflux	NA	nmolC/cm2/d	FDET efflux lower boundary
SDETflux	NA	nmolC/cm2/d	SDET flux to sediment
SDETdeepflux	NA	nmolC/cm2/d	SDET efflux lower boundary
FePdeepflux	NA	nmolP/cm2/d	FeP efflux lower boundary
CaPdeepflux	NA	nmolP/cm2/d	CaP efflux lower boundary
OrgCflux	NA	nmolC/cm2/d	OrgC influx to sediment
OrgNflux	NA	nmolN/cm2/d	OrgN influx to sediment
OrgPflux	NA	nmolP/cm2/d	OrgP influx to sediment
O_Cflux	NA	molO/molC	O2:DIC ratio flux sediment-water
ODUO_Cflux	NA	molO/molC	(O2+ODUflux): (DIC flux) sediment-water
DINDIPflux	NA	molN/molP	DIN:DIP ratio flux sediment-water
DINDIPmean	NA	molN/molP	DIN:DIP mean concentration
DINDIPdeep	NA	molN/molP	DIN:DIP deep concentration
TotMin	NA	nmolC/cm2/d	Vertically integrated Mineralisation
TotOxic	NA	nmolC/cm2/d	Vertically integrated oxic Mineralisation
TotDenit	NA	nmolC/cm2/d	Vertically integrated Denitrification
TotAnoxic	NA	nmolC/cm2/d	Vertically integrated anoxic Mineralisation
PartOxic	NA	-	Part of mineralisation by oxic min
PartDenit	NA	-	Part of mineralisation by denitrification
PartAnoxic	NA	-	Part of mineralisation by anoxic min
TotNitri1	NA	nmolN/cm2/d	Vertically integrated nitrification step 1 (NH3 oxidation)
TotNitri2	NA	nmolN/cm2/d	Vertically integrated nitrification step 2 (NO2 oxidation)
TotAnammox	NA	nmolN/cm2/d	Vertically integrated anammox
TotODUoxid	NA	nmolO2/cm2/d	Vertically integrated ODU oxidation
TotFePprod	NA	nmolP/cm2/d	Vertically integrated FeP production
TotCaPprod	NA	nmolP/cm2/d	Vertically integrated CaP production
TotFePdesorp	NA	nmolP/cm2/d	Vertically integrated FeP desorption
TotCaPdiss	NA	nmolP/cm2/d	Vertically integrated CaP dissolution

names	values	units	description
TotPads	NA	nmolP/cm2/d	Vertically integrated P adsorption
TotODUoxsurf	NA	nmolO2/cm2/d	Vertically integrated ODU oxid by BW O2
TotNH3ads	NA	nmolN/cm2/d	Vertically integrated NH3 adsorption
PartPremoved	NA	-	Part P removed
PartNremoved	NA	-	Part N removed
TotNH3prod	NA	nmolN/cm2/d	Vertically integrated NH3 production
TotPO4prod	NA	nmolP/cm2/d	Vertically integrated PO4 production
TotMPBNO3uptake	NA	nmolN/cm2/d	Vertically integrated MPB NO3 uptake
TotMPBNH3uptake	NA	nmolN/cm2/d	Vertically integrated MPB NH3 uptake
TotMPBPO4uptake	NA	nmolP/cm2/d	Vertically integrated MPB PO4 uptake
TotMPBDICuptake	NA	nmolC/cm2/d	Vertically integrated MPB DIC uptake
TotMPBO2prod	NA	nmolO2/cm2/d	Vertically integrated MPB O2 production
TotalFDET	NA	nmolC/cm2	Vertically integrated Fast decaying Detritus
TotalSDET	NA	nmolC/cm2	Vertically integrated Slow decaying Detritus
TotalO2	NA	nmolO/cm2	Vertically integrated Oxygen
TotalNO3	NA	nmolN/cm2	Vertically integrated Nitrate
TotalNO2	NA	nmolN/cm2	Vertically integrated Nitrite
TotalNH3	NA	nmolN/cm2	Vertically integrated Ammonium/ammonia
TotalODU	NA	nmolO/cm2	Vertically integrated Oxygen Demand Units
TotalDIC	NA	nmolC/cm2	Vertically integrated Dissolved Inorganic Carbon
TotalPO4	NA	nmolP/cm2	Vertically integrated Phosphate
TotalFeP	NA	nmolP/cm2	Vertically integrated Iron-bound P
TotalCaP	NA	nmolP/cm2	Vertically integrated Ca-bound P
TotalPads	NA	nmolP/cm2	Vertically integrated Adsorbed P
Cflux	NA	nmolC/cm2/d	Carbon flux to sediment
w	NA	cm/d	Sedimentation rate
biotfac	NA	-	Bioturbation multiplication factor
irrfac	NA	-	Irrigation multiplication factor
rFast	NA	/d	Decay rate FDET
rSlow	NA	/d	Decay rate SDET
pFast	NA	-	Part FDET in flux
MPBprod	NA	mmol/m3/d	MicroPhytoBenthos forcing
CaPflux	NA	nmolP/cm2/d	CaP deposition flux
FePflux	NA	nmolP/cm2/d	FeP deposition flux
gasflux	NA	cm/d	Gas exchange flux (piston velocity)
bwO2	NA	mmol/m3	Bottom water O2 concentration
bwNO3	NA	mmol/m3	Bottom water NO3 concentration
bwNO2	NA	mmol/m3	Bottom water NO2 concentration
bwNH3	NA	mmol/m3	Bottom water NH3 concentration
bwODU	NA	mmol/m3	Bottom water ODU concentration
bwPO4	NA	mmol/m3	Bottom water PO4 concentration
bwDIC	NA	mmol/m3	Bottom water DIC concentration
Hwater	NA	cm	Height of water above the sediment
ratefac	NA	-	Rate multiplication factor

One-D ordinary variables

```
knitr:::kable(CNPDIA1D())
```

names	units	description
FDET	mmolC/m3 solid	Fast decaying Detritus (solid)
SDET	mmolC/m3 solid	Slow decaying Detritus (solid)
O2	mmolO/m3 liquid	Oxygen (liquid)
NO3	mmolN/m3 liquid	Nitrate (liquid)
NO2	mmolN/m3 liquid	Nitrite (liquid)
NH3	mmolN/m3 liquid	Ammonium/ammonia (liquid)
ODU	mmolO/m3 liquid	Oxygen Demand Units (liquid)
DIC	mmolC/m3 liquid	Dissolved Inorganic Carbon (liquid)
PO4	mmolP/m3 liquid	Phosphate (liquid)
FeP	mmolP/m3 solid	Iron-bound P (solid)
CaP	mmolP/m3 solid	Ca-bound P (solid)
Pads	mmolP/m3 solid	Adsorbed P (solid)
TOC	%	Total Organic Carbon % profile
DICprod	nmolC/cm3 liquid/d	DIC production profile (C-mineralisation)
DINprod	nmolN/cm3 liquid/d	DIN production profile (N-mineralisation)
DIPprod	nmolP/cm3 liquid/d	DIP production profile (P-mineralisation)
O2prod	nmolO/cm3 liquid/d	O2 production profile (microphytobenthos)
Oxicmin	nmolC/cm3 liquid/d	Oxic mineralisation profile
Denitrific	nmolC/cm3 liquid/d	Denitrification profile
Anoxicmin	nmolC/cm3 liquid/d	Anoxic mineralisation profile
Nitri1	nmolN/cm3 liquid/d	Nitrification step 1 profile (NH3 oxidation)
Nitri2	nmolN/cm3 liquid/d	Nitrification step 2 profile (NO2 oxidation)
Anammox	nmolN/cm3 liquid/d	Anammox profile
ODUox	nmolO/cm3 liquid/d	ODU oxidation profile
ODUoxsurf	nmolO/cm3 liquid/d	ODU oxidation with surface O2 profile
ODUdepo	nmolO/cm3 liquid/d	ODU deposition profile
FePadsorp	nmolP/cm3 liquid/d	FeP adsorption profile
FePdesorp	nmolP/cm3 solid/d	FeP desorption profile
CaPprod	nmolP/cm3 liquid/d	CaP production profile
CaPdiss	nmolP/cm3 solid/d	CaP dissolution profile
Padsorp	nmolP/cm3 liquid/d	P adsorption profile
MPBproduction	nmolC/cm3 solid/d	MPB production profile
MPBNO3uptake	nmolN/cm3 liquid/d	MPB NO3 uptake profile
MPBNH3uptake	nmolN/cm3 liquid/d	MPB NH3 uptake profile
MPBPO4uptake	nmolP/cm3 liquid/d	MPB PO4 uptake profile
MPBDICuptake	nmolC/cm3 liquid/d	MPB DIC uptake profile