EcoTiM

Ecological **Ti**dal **M**odel

User's and Programmer's Manual

Simulation model for the back barrier system
of the East Frisian islands Spiekeroog and Langeoog

Cora Kohlmeier

March 17, 2025

Institute for Chemistry and Biology of the Marine Environment (ICBM)
Carl von Ossietzky University of Oldenburg

Contact:
Cora Kohlmeier
Mathematical Modelling
ICBM, Carl von Ossietzky University of Oldenburg
Carl-von-Ossietzky-Strasse 9-11
Germany

eMail: kohlmeier@icbm.de
http://www.staff.uni-oldenburg.de/cora.kohlmeier

Contents

License

1 Introduction

The ecosystem model **EcoTiM** describes the cycling of carbon, nitrogen, phosphate and silicate within the back barrier system of the island Spiekeroog and/or Langeoog. The model describes the pelagic processes as well as the benthic and its coupling. The model description of the ecology bases on ERSEM (Baretta-Bekker, 1995; Baretta-Bekker & Baretta, 1997). The description of the water movement is described with a Lagrangian approach, while the benthic system is calculated as normal box model.

In the default setup for the back barrier system of Spiekeroog the number of simulated Lagrangian water bodies – called voxels[1] – is 100. In every voxel the pelagic system is simulated. The exchange between voxels is mainly determined by sedimentation and the benthic remineralization but also by diffusive mixing. The boundary conditions of the North Sea also influences the concentrations in the voxel. The number of geographical regions – called Eulerian boxes[2] – is seven, where box one is the North Sea boundary box. The benthic system is simulated in every box. The pelagic properties of a box is characterized by the voxels in this region at the considered time; the concentration of a pelagic variable within the box is assumed to be the mean of all voxels within the region. The benthic variables are treated as common variables in Eulerian sense. For further details see the model description (Kohlmeier, 2004; Kohlmeier & Ebenhoeh, 2007, 2009).

In the following the technical concept and implementation of **EcoTiM** is given. **EcoTiM** is built on the base of **CEMoS** (Hamberg, 1996). **CEMoS** is a model environment for convenient implementation of large models, especially adapted to the needs of ecosystem models. The implementation of a model in **CEMoS** is nearly C-Code, so that the code can be ported with minor effort. The differences to pure C and the special features of **CEMoS** are described in Chapter 3.

The implementation appears to be a little bit bumpy at some points. This is a consequence of recycling old code and of optimizing code in terms of simulation speed and readability. Especially no great store is set by memory management. Several defined variables are never used by the model, but the structured defi-

[1] The term "voxel" is borrowed from the computer world: Volume pixel or volume picture element. A three dimensional pixel; a concept used in three-dimensional modeling. The smallest division of a three dimensional space or image (source: http://www.computeruser.com, see also at wiki)

[2] To emphasized the fact that boxes are stationary while voxel are movable, the addition Eulerian is used. For simplicity this annex is omitted in the text.

nition in terms of the **CEMoS** notation increases the clearness.

2 Requirements

The here presented manual is of technical nature. The underlying model concept and the ecological processes are described in the model description (Kohlmeier, 2004).
The here described version of **EcoTiM** needs the **CEMoS** package. **CEMoS** runs on Linux machines providing a C-compiler. For convenient work with **CEMoS** the installation of the graphical user interface **CEMTK** is recommended. **CEMTK** requires the script language **Tcl/TK**.
To visualize the model results **TigerGraphics** provides the **Mo**del **Vi**sualization **E**nvironment **MoViE** (Kohlmeier & Hamberg, 2004)

For further details of download, installation and use see manuals of **CEMoS**, **CEMTK** and **MoViE** see (Hamberg & Kohlmeier, 2004; Kohlmeier & Hamberg, 2004).

It is strictly recommended to read
the model description and the CEMoS user manual
before working with the model!

3 CEMoS features

In principal **EcoTiM** could have been written in pure C, but **CEMoS** provides some features which allows an easy work with the model:

- **CEMoS** takes care of the integration of the model and allows operator splitting with several different integration schemes,
- recompilation is not necessary if only parameter values are modified,
- after recoding only the changed files must be compiled,
- the accuracy of the simulation can be changed without recoding,
- batch runs for sensitivity analyses are possible.

Such some non-C statements will be found in the code. In the C-files the most obvious difference is the data type `real` . Variables of this type are evaluated either as `float`, `double` or `long double` depending on the settings in the `%numeric` statement in the main definition file `model.def`.
The `*.def` files have **CEMoS**–specific coding. The included header files are written automatically by **CEMoS** evaluating the information from the `*.def` files. Additionally a reading function is generated for every `*.def` file for reading the changed parameter values (`%change` statement) during simulation.
The information of the main model definition file `model.def` can be found in the file `struct.h` resp. `struct_model.h`.
The needed **CEMoS** routines for the integration control and data handling are included during compilation.

Getting started

To start **CEMoS** goto the directory **EcoTiM**. If **CEMoS** is correctly installed you could start it by clicking on the file `cemos.par` in your file manger. Alternativly you can give the command cem in a coomand windoe in that directory. The model code will be compiled by clicking on compile or alternatively by the command compile main, the simulation will be started by clicking on `Start!` or alternatibvley by the command run. Model results are stored in the file `result.outc` by default and can be visualized by **MoViE**. To do this click on `TigerGraphics MoViE` or alternativly click on the `result.outc` file in your file manager.
The complete model code including all integration routines and all automatically generated code is visible in `main` if the model is compiled with the debug flag.

3.1 Operator splitting

The most important feature used with **EcoTiM** is the operator or mode splitting. In principal is **EcoTiM** an ordinary differential equation model coupled with a Lagrangian transport approach. Such processes described as differential equations have to be numerically treated in a more accurate way than the other processes. While transport processes are treated with a fixed time step (see 4.4) the differential equations are treated with a accurate method. This splitting leads to some technical features which must be understand to interpret model results correctly.

3.2 Storing of variables

3.2.1 Storing and operator splitting

Storing of variables takes normally place before processing all integrations of the next time step (`storestart=0`). Therefore the stored variables contain the values calculated within the last integration. For most of the variables, especially for all `states`-variables, this is no problem. State variables hold their values during the model run. Derived variables, `S`-variables and `real_derived_from_states`-variables are set to zero before entering an integration. This lead to misinterpretations. If such a variable is f.e. only set in the first integration it is set to zero before the next integration is called. Therefore the value zero will be stored. If values from integrations before the last one shall be stored, they have to be stored as `globals`-variables. These variables will never be zeroized by **CEMoS**. For derived variables a further type of variables exist, the `global_derived_from_states` variables. These variables will be zeroized after storing but not after every integration.

For `real_derived_from_states`-variables one exception exist:
`real_derived_from_states`-variables starting with the letter `eu` are not reset to zero. `eu`-variables hold the mean values for the Euler boxes. They are only diagnostic variables and are only calculated if at least one voxel is within the box. To avoid zeros in the output, the variables hold their last value until they are calculated again.

In **EcoTiM** most of the important variables are used in the ecological model.

Such the integration controlling this model part is the last one so that the values from the ecological model will be stored for all variables set to zero after integration.

3.2.2 Storing of states and fluxes

A further problem of storing global variables occurs if the integration method calls the model at intermediate interpolation points. Then the value calculated at the last interpolation point is stored. This is a well known problem with accurate integration methods. Even if the differences between the values at the end of the step and the values at some intermediate points is not serious, the results might be misinterpreted (f.e in budget computation where total mass conservation is expected). **CEMoS** provides the possibility to recalculate the values at the end of the step with the statement `recalc_globals=1;` in the file `cemos.par`. In this case the model is called once again to calculated the global variables at the sampling point but without changing state variable values. For further information see appendix C.
A similar more philosophical issue raises the question if the value of global variables especially fluxes at the beginning or at the end of a time step correspond to the value of a state variable. **CEMoS** provides both possibilities. The default is that states and fluxes are stored before integration (`storestart=0;` in the file `cemos.par`). In this case it is assumed that the value of the state variable before adding a flux corresponds to the flux. This is senseful if the calculation of the flux is time dependent (non-autonomous differential equation), such the time determine the flux corresponds to the value of the state variable at that time.
Setting `storestart` to any other value will force **CEMoS** to store after the integration. If in this case `recalc_globals=1;` is set, the values of all globals are calculated on the base of the new values of the state variables at the end of the time step.

More than one integration method

If more than one integration method is active, the stored results of global variables which are touched in both integrations may be misinterpreted. For a model with three integration methods the following schemes are valid: `storestart=0`:

storing

integration method 1
integration method 2
integration method 3

`storestart=1`:

integration method 1
storing
integration method 2
integration method 3

More complicated is the case if the store step `outdelt` is finer than the maximum of all maxdelts. The **CEMoS** stores at intermediate positions. To track the operation it is recommended to compile the model with the debug flag. In this case **CEMoS** prints additional messages which shows the time of storing.

In **EcoTiM** the simulation results are correct in all cases but the values of the derived variables holding the mean values for the boxes (eu-variables) may be misleading. If f.e storing takes place after the first integration, the tracer got new positions but still hold the forcing values of their old position.

To avoid such problems it is recommended to set `storestart=0` and outdelt to the maximum of all maxdelts.

3.2.3 Defining store variables

Store variables are defined in the file `cemos.par`. The easiest way to do this is to write them directly into the store block (`%store`). Because **EcoTiM** is very complex and hundreds of variables might be stored the storing has been structured. Variables to be stored are grouped in files located in the store directory. These files are included.in the `cemos.par`. There is one special file in store named `store.setup`. This file must be included as first file in the store block. It contains information about the indices to be stored for a special setup. These definitions are used in the other store files. Such the voxel and boxes to be stored are adapted to the actual setup.

3.3 Running a slave model

CEMoS provides the possibilty to run a slave model. This means that values between both models are exchanged beween both models after every time step. This option is experimental and not described. Only experienced **CEMoS** user should work with this option. Please contact the author for further information. In **EcoTiM** the switch `iswNORTHSEA` must be set to 1 to get the slave values as boundary values. At the moment no influences from **EcoTiM** to the slavemodel is implemented.

4 The Model structure

4.1 File structure

A detailed description of the structure of a **CEMoS** model is given in Hamberg (1996) and (Hamberg & Kohlmeier, 2004, URL). The complete code including all data files of the model can be found in the directory `ECOTIM`. This directory contains at least

`cemos.par`	contains all simulation control parameter (section)
`cemos-model`	the executable (if already compiled)
`main`	main model directory containing all sources and data files

The directory `main` is additionally structured. The directory itself contains at least the source code for general model control and subdirectories holding the source code and data files:

`model.c`	main model code file
`model.def`	main model definition file
`install.tcl`	install script
`cemos_com.c`*	interface for boundary model
`cemos_com.inc`*	include file for shared memory interface

* only needed if a slave model is called via shared memory

Before compilation the model code and the definition files (`*.def`) are linked from the subdirectories in `main` (except `initial`) into `main` itself. This is controlled by the install script `install.tcl` which is automatically called by **CEMTK** during compilation[1]. Additionally the preprocessing is done creating all header files (`*.h`) from the definition files (`*.def`). This subdivision of model code into different directories is mainly done to keep the survey of the code. In the case of self written `*.h` files it is strictly necessary because such files must not be placed in the main model directory of a **CEMoS** model[2].

The directories of main are (in alphabetical order)

[1] If **CEMoS** is used without **CEMTK** the file `install.csh` is evaluated.

[2] **CEMoS** will remove this files before the next compilation!

`benthic`	code and definition files for benthic model
`carbonate`	code and definition files for carbonate model
`diagnostic`	code and definition files of diagnostic functions
`floodgate`	code,data and definition files of input from floodgates
`forcing`	code,data and definition files of forcing functions
`initial`	files for the main model definition files incl. initial values in `model.def`
`manganese`	code and definition files for manganese and methane extension (optional)
`northsea`	code,data and definition files for North Sea boundary conditions
`pelagic`	code and definition files for pelagic model
`setup`	header and data files for setup control
`tide`	code, data and definition files for setup control
`tracer`	code, data and definition files for the transport model (velocity field Fourier series)
`util_bio`	utility code and for biological model
`util_setup`	utility code and definition for Lagrangian setup

Remark: In the following text the file names are given relative to `ECOTIM` or `ECOTIM/main`.

4.2 State variables

State variables are indexed variables. The index describes the voxel resp. box. All state variables have the same length. The highest index is given by definition in the string `MDIM` in the file `model.def`. In **EcoTiM** `MDIM` is the number of voxels. All state variables are indexed by `MDIM`. For state variables of the benthic submodel only the first `euboxes` array components are used. The variable `euboxes` is set in `model.def`. State variables are defined in `initial/initial.def`. In this file start values are given for every state variable in every voxel resp.box. For simplicity all voxels/boxes are set to the same value. The file `initial/initial.def`. is included in `cemos.par` The initial values can be overwritten in the `%change` statement in the file `cemos.par`. Most of the state-variables are state variables in the mathematical sense. This means that they are dynamic quantities described by differential equations.

The right hand side of these differential equations has to be given in a so called `S`-term. For every state variable define as `state` in `model.def` resp. `initial.def` **CEMoS** provides a `S`-variable holding the right hand side of the differential equation (`S`-term). Such the differential equation

$\frac{d}{dt}N = \alpha \cdot N$

must be given as

```
SN[1]= alpha*N[1];
```

in **CEMoS**. For further details concerning the notation see Hamberg & Kohlmeier (2004); Kohlmeier & Hamberg (2004).

Some technical state variables are added which are not dynamic states in the mathematical sense. Their values are set directly in the model. Their `S`-terms are not used. Normally these variable should have been defined as indexed global variables. For convenience they are defined as states. These are

`vol`	volume of voxels, actually equal for all voxels, calculated in `init_grid`
`xcoor`	actual x-coordinates of voxels within the grid, calculated in `calc_tracer`
`ycoor`	actual y-coordinates of voxels within the grid, calculated in `calc_tracer`

`vol` is defined as state so the derived variables (section 4.3) are available for budgeting volume fluxes).

`xcoor` and `ycoor` are defined as state to have the possibility to set initial values.

The real state variables which are integrated are defined in `initial.def` which is included in `model.def`. They are given in section A. The virtual states are defined in `model.def`.

4.2.1 itr-identifier

Origin of name: integer transport (ERSEM)

For every state variable an `itr`-identifier exist. The identifier are declared as `int-derived-from-state`-parameter in the file `model.def`.

`int-derived-from-state`-parameters are non-indexed quantities and are therefore voxel/box independent. The itr-parameter contains the information of a state variable whether it is pelagic or benthic, it is transported and information about the chemical identity (c,n,p,s). The values are set in the `change`-statement in the file `model.def`. Parameters which are not changed are set to zero. The **EcoTiM**-nomenclature is given as follows:

value	meaning
0	auxiliary state variables
1-9	benthic variables
≥ 10	= pelagic variables
≥ 100	not transported pelagic variables
2,22	carbon state variables
3,23	phosphate state variables
4,24	nitrogen state variables
5,25	silicate state variables
21	other state variables

Values between 10 and 20 are not used in **EcoTiM** (ERSEM compatibility). The itr-identifier are used in the following files:

`budget.c` `budget_rsw.c`
`calc_euboxes.c` `mixing.c`
`mixing_eubox1.c` `set_northsea.c`
`stdev.c` `trsptorates.c`

4.3 Derived-from-state-variables

4.3.1 eu-variables

Origin of name: Euler

For every state variable a derived-variable is defined in the file `model.def` starting with `eu`. These `eu`-variables are **CEMoS** `real-derived-from-states`-variables with the exception that their values are not set on zero before model evaluation (see section 3.2). The `eu`-variables are only used for pelagic quantities. While the state variables hold the concentration of the voxels, the `eu`-variables hold the mean concentration of all voxel in the considered box. Such only the first euboxes array components are used for the Euler boxes.The index 50 is used for the mean values within boxes 3 to 7, the index 40 for the mean values around the pole station (only senseful in the default `SPIEKEROOG`-setup, see section 5).

`eu`-variables do not affect the model results, they are only used for visualization and inspection.

They are calculated in `calc_euboxes` in `calc_euboxes.c`.

4.3.2 bTI- and bTO-variables

Origin of name: benthic transport in/out

bTI- and bTO-variables are real_derived_from_states-variables. They hold the exchange between the pelagic and the benthic system and allow budgeting between both systems. The regarded processes are sedimentation, remineralization and diffusion.bT-variables belonging to a pelagic state variable are voxel dependent, bT-variables belonging to a benthic state variable are voxel dependent. The variables contain all sedimentation fluxes, remineralization fluxes and diffusion fluxes

The values of the bTI- and bTO-variables are added in trsptorates to the S-terms. The trsp_to_rates-function avoids several loops over all voxel and all variables in local routines. The allocation to the S-terms is done once at the end for all derived variables which must be considered. The bTI- and bTO-variables are used in the following files:

ben.c	ben_advection.c (future)	ben_sink.c
ben_diff.c		sedimentation.c
ben_diff.c	ben_shortcut.c	trsptorates.c

4.3.3 wDI- and wDO-variables

Origin of name: water diffusion in/out (ERSEM)

wDI- and wDO-variables are real_derived_from_states-variables (origin of name: water diffusion in/out). They hold the values for diffusive mixing between the voxel (pelagic variables) and the particulate transport of benthic material (benthic variables, only wDI). The variables contain the change in concentration due to diffusive inflow into a voxel (wDI) resp. the change in concentration due to diffusive outflow out of a voxel (wDO). The index corresponds to the considered voxel. The values of the wDI- and wDO-variables are added in trsptorates to the S−terms. The wDI- and wDO-variables are used in the following files:

mixing.c trsptorates.c

4.3.4 sed-variables

Origin of name: sedimentation (ERSEM)

These variables hold the sedimentation fluxes of algae and detritus. They are calculated in the pelagic system. The transport into the benthic system is done in `sedimentation`.
The `sed`-variables are used in the following files:
`pelagial.c pprod.c sedimentation.c`

4.3.5 auxI- and auxO-variables

Origin of name: auxiliary in/out

`auxI`- and `auxO`-variables are `real_derived_from_states`- variables. They are diagnostic variables for budget computations (section 16.5) and do not influence the model results.
The `auxI`- and `auxO`-variables are used in the following files:
`ben_part_trans.c budget.c budget_rsw.c`

4.3.6 wRI-variable

Origin of name: water river in

The `wRI`-variables are `%global_derived_from_states`. Such they are not set to zero after the integration but after storing. Such these variables can be inspected even if they are not set in the last integration (see section 3.2). The `wRI`-variables contain all fluxes of substances which are transported with freshwater from the floodgate and aquifers. Additionally the variable `wRIX1x` contains the changes in salinity due to precipitation and evaporation. Because only a few of the `wRI`-variables are used their values are added directly to the `S`-terms (not in `trsptorates`).
`aquifer.c evaporation.c`
`floodgate.c precipitation.c`

4.4 Integration control

EcoTiM works with three separate integrations:

Integration	Method	Time step	Routines
Fixstep	1 or 99	0.01	Fig. 4.1 left
Fixstep	1 or 99	0.1	Fig. 4.1 middle
Runge-Kutta	2 or 3	0.1	Fig. 4.1 right

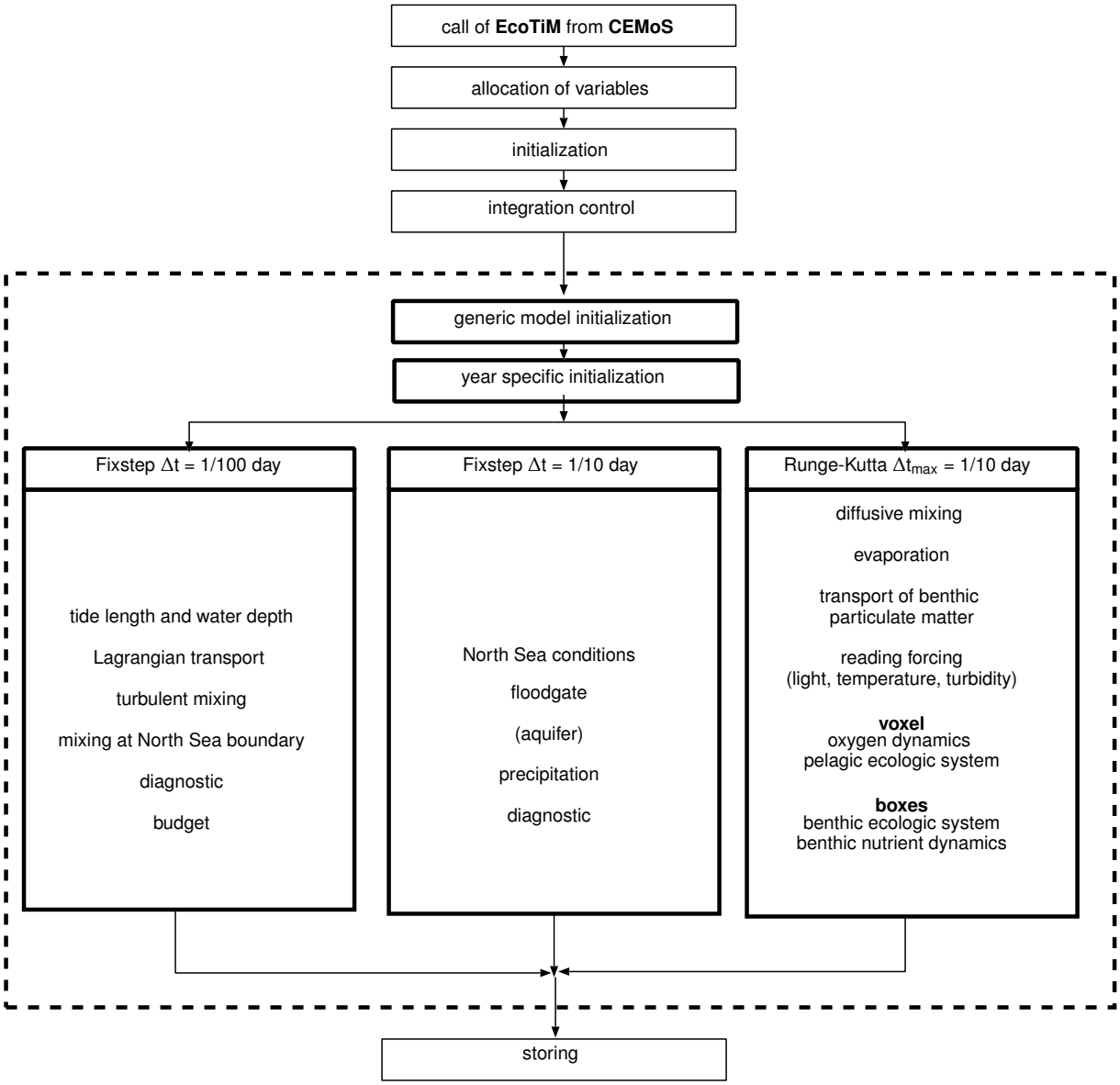


Figure 4.1: Flow chart of **EcoTiM**. The thin borders mark processes controlled by **CEMoS**. The dashed box marks the model code which can be modified by the user.

The fixstep routines can be either evaluated as real fix step integration (method

1) or by setting the state variables directly in the model evaluation (method 99, faster because no integration routine is called).
The Runge-Kutta integration is by default evaluated with embedded formulas of second order with a predictor method of third order and with adaptive time step control (method 2, see Kohlmeier, 1995) . This method takes special care that no state variable becomes negative. Other Runge-Kutta methods are also available (see Hamberg & Kohlmeier, 2004).
The sequence of the integrations is not arbitrary. The transport processes must be evaluated first .

4.5 Model run

It must be ensured that the model has been compiled. For further details see **CEMoS** manual. The model run is started either from the graphical user interface **CEMTK** by `Start!` or without using the graphical user interface from the directory `ECOTIM` by `run` (**CEMoS** shell script) or directly by `cemos.model`. Results are written by default in `result.outc`.
The first function which is called is `main`. This is an internal **CEMoS** function which cannot be modified by the user. The first model function which is in the user's responsibility is `model` in `model.c`. In this function the general structure of the model can be seen. `model` is called for every integration as often as the methods needs to do it.
The first call of model acts as declaration. The variable `assign` is set to -1 so that all functions are called once to initialize all variables.

4.5.1 Initialization of the model run

In the first call of `model` the initialization of the setup is done. These routines are called once at the beginning of a simulation. The routine names start with `init_`.

4.5.2 Initialization of a specific year

After every simulation year which is recognized by `cycle` in `cemos.par` the initialization for the next year is done. The function `setup` is called again and all files are set according to the following year. After that all initialization routines are called to open the files.

4.6 Model setup files

The function `setup` in `setup.c` contains nearly all information on files holding data for the boundaries, forcing precipitation, floodgate etc. The function is called once at the beginning of a new simulation year. The names of the actual data files corresponding to year and setup are reassigned. The setup function provides information for every setup and every model year for

4.6.1 Forcing

pointer	type	meaning	structure
`eir_file`	–	irradiance data	actually not used
`par_file`	`csv`	PAR[3] (time resolved)	time, irradiance in W/m²
`par_dm_file`	`csv`	PAR (daily mean)	time, irradiance in W/m²
`daylen_file`	`csv`	daylen	sunrise, daylen in fraction of the day
`etw_file`	`csv`	water temperature	time, temperature in °C
`etb_file`	`csv`	sediment temperature	time, temperature in °C
`seston_file`	`csv`	seston dynamic	time, value
`pre_file`	`csv`	precipitation	day, precipitation in mm
`esal_file`	`csv`	salinity	day, salinity in –

All `csv`-files file must have a header line. For further details see section 6.

4.6.2 Floodgates

pointer	type	meaning	structure
`gate_load_file_xxx`	`csv`	volumes from floodgate	time, volume in m^3
`gate_conc_file_xxx`	`csv`	concentrations	time,x1x,n1p,n3n,n4n,n5s
`gate_time_file`	`csv`	time stamps for releasing	time

`xxx` must be replaced by `nhs` for Neuharlingersiel, by `dor` for Dornumersiel and by `ben` for Bensersiel. All `csv`-files file must have a header line. For further details see section 7.1.

4.6.3 Aquifer

pointer	type	meaning	structure
`aqui_load_file_nhs`	`csv`	input from aquifer	time, volume in m^3
`aqui_conc_file_nhs`	`csv`	concentrations	time,x1x,n1p,n3n,n4n,n5s

All `csv`-files file must have a header line. For further details see section 7.2.

[3] photosynthetically available light

4.6.4 North Sea boundary

North Sea boundary from North Sea model

pointer	type	meaning	structure
northsea_result_file	_pk.outc	boundary values	spectral values

North Sea boundary from data

pointer	type	meaning	structure
northsea_n1p_file	csv	phosphate boundary values	time,value
northsea_n3n_file	csv	nitrate boundary values	time,value
northsea_n4n_file	csv	ammonium boundary values	time,value
northsea_n5s_file	csv	silicate boundary values	time,value
northsea_sal_file	csv	salinity boundary values	time,value
northsea_alk_file	csv	alkalinity boundary values	time,value
northsea_o3c_file	csv	DIC boundary values	time,value
northsea_pco2_file	csv	pCO_2 boundary values	time,value

For further details see section 8 .

4.6.5 Tracer movement

pointer	type	meaning	structure
trajec_result_file	outc	static trajectories	values for every voxel and every time step of integration 1
tracer_data_file	outc	spectral values for every grid point	see

All csv-files file must have a header line. For further details see section 10.1.1

4.6.6 Tide

pointer	type	meaning	structure
tidelen_data_file	csv	tide length and high tide time	time of high tide, tide length in fraction of the day
lowtide_data_file	csv	low tide time and daytime low tide	time of low tide and flag for low tide at day time
pegel_data_file	csv	gauge amplitude	time, deviation from mean in m

All csv-files file must have a header line. For further details see section 9 and section 17.2.

4.7 Voxel and box calculation

The pelagic biological model is calculated in every voxel in `model`. If a voxel reaches box 1 (`actbox[vox]=1`) no calculation takes place.

4.8 Initial conditions

Initial conditions were derived from a twenty year run using annually repeating forcing functions to reach a stable cycle. They are set in `cemos.par` in the `%change` section.

4.9 Benthic-Pelagic-Coupling

The model is treated as 2D-model. The total exchange of a box (Fig. 5.1) with the underlying sediment area is calculated by summing up the exchange of all single moving water bodies within the considered box at the actual time. It is assumed that voxel (with fixed volume) has a certain depth depending on the actual sea level at its position. This depth determines an estimation of the area of interaction which is related to the geographical area of the box. The exchange of nutrients due to diffusion, sinking resp sedimentation of suspended material and algae is calculated for every area of interaction.

The coupling to the sediment and benthic system is done by considering the system as two-dimensional. Each tracer is assumed to reach from the sea surface to the sediment's surface. The position of the tracer determines the actual water depth. The actual area contacting the sediment can be calculated under the assumption of a columnar shape (Fig. 4.2).

The actual contact areas of all tracers over a box are given by the actual positions and the water level at these positions. Unfortunately due to the discretization the sum of all areas may exceed the total area of the box. This is not acceptable if mass conservation is needed. Such the contact areas are recalculated from this first estimation: Let j be the number of tracers above a subregion at time t. For a mass conserving description of an exchange process between the pelagic and benthic system the sum of all contact areas A_j must equal the total area A of the subregion at any time,

$$A = \sum_{j \in \mathcal{J}(t)} A_j(t) \qquad \text{f.a. } t \geq 0\,, \tag{4.1}$$

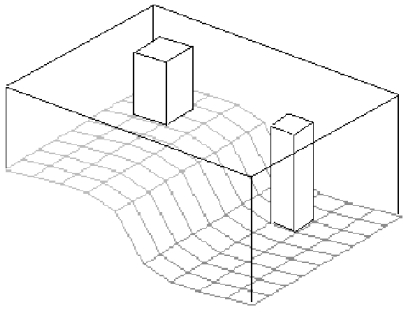


Figure 4.2: A subregion and the underlying grid. Two tracers are shown exemplarily. The depth of a tracer is determined by its actual position and the water depth at this position. All tracers have the same finite volume. The lower the water level the larger the exchange area to the sediment.

where $\mathcal{J}(t)$ is the index set of all voxel within the subregion at time t.

The property 4.1 can be fulfilled in different ways. Canonically, the areas are determined by water depth. Let $h_j(t)$ be the actual depth of the tracer j and let V be the fixed and finite volume of one tracer, then a first guess for the area is given by

$$A_j(t) = \frac{V}{h_j(t)} \; . \tag{4.2}$$

To fulfill the constrains of equation 4.1

$$A = V \cdot \sum_{j \in J(t)} \frac{1}{h_j(t)} \tag{4.3}$$

respectively

$$V = \frac{A}{\sum\limits_{j \in \mathcal{J}(t)} \frac{1}{h_j(t)}} \; , \tag{4.4}$$

must hold and utilizing equation 4.4 with equation 4.2 leads to

$$A_j(t) = A \frac{\frac{1}{h_j(t)}}{\sum\limits_{l \in \mathcal{J}(t)} \frac{1}{h_l(t)}} \; . \tag{4.5}$$

Now the sum of volumes of all tracers within the subregion equals the total

volume and the sum of the estimated areas equals the total area.
The change in concentration (in mmol/m^3) of a substance within a tracer due to an outflow of mass f_M is given by

$$\frac{d}{dt}c_j(t) = \frac{A_j(t)}{A} \cdot f_M \cdot \frac{1}{V} , \tag{4.6}$$

The change of concentration (in mmol/m^2) within the benthic subregion is given by

$$\frac{d}{dt}c^B(t) = -\frac{f_M(t)}{A} . \tag{4.7}$$

The calculation of sedimentation processes into the benthic system is determined by the sinking velocity of the considered substance. Let v be the sedimentation velocity of a substance with concentration c_j from a tracer j with the estimated depth $h_j^e(t) = \frac{V}{A_j(t)}$ into the sediment. Then the sedimentation rate is given by $\frac{v}{h_j^e(t)}$ and the resulting change of concentration (in mmol/m^3) within the tracer is

$$\frac{d}{dt}c_j(t) = -v \cdot \frac{1}{h_j^e(t)} \cdot c_j . \tag{4.8}$$

The corresponding change in concentration (in mmol/m^2) within the sediment is given by

$$\frac{d}{dt}c^B(t) = v \cdot \frac{1}{h_j^e(t)} \cdot c_j \cdot \frac{V}{A} . \tag{4.9}$$

From equation 4.9 it can be seen that impact of exchange processes between the pelagic and benthic system is as higher as lower the water level is.

5 Geographical setup

The coordinates of the model correspond to the 200 m resolution of the East Frisian back barrier area (Stanev *et al.* , 2003) which results in 324×88 grid points. The boundary for the model region are $[189, 252] \times [5, 88]$. The values are defined in `setup/setup_define.h`.

The depth profile of the whole region is stored in `setup/depth324.dat`. It is read in `init_grid.c`. Grid points with a depth lower than –3 m are defined as land points. Their depth value is set to -999.

The allocation of grid points and boxes is given in `main/setup/eulerboxes.dat`. It is read in `init_grid.c`. Grid points without box allocation or belonging to island or main land get the box index -999. Additionally all grid points which do not have a correct velocity field information are set to box -999 and therefore they are treated land points.

During simulation the actual depths at all grid points are calculated in `calc_griddepth.c`.

Actually 3 different setups are implemented:

1. `SPIEKEROOG` (default): Boxes 1-7 according to Fig. 5.1.
2. `LANGEOOG`: Boxes 1-8 according to Fig. 5.2.
3. `BOTH`: Boxes 1-14 boxes according to Fig. 5.2.

The control for the setup is done by a `%setup` statement in `model.def`. The number of voxel is independent on the set-up. It is recommended to run the model with at least 100 voxel. The setup BOTH should be run with at least 200 voxel. The number of voxel is set in `model.def` by a `#DEFINE` statement in `MDIM`. After changing the setup recompilation is necessary.

The statement `%setup <entry>` generates a `#define <entry>` in the automatically generated file `compiler_setup.h` which is re-included in the processing of the `model.def` itself. Such, references like `ifdef LANGEOOG` are evaluated in the `model.def` and in all other model files (even in `*.def`). With this construction the number of boxes and the parameterizations for the different setups are set automatically. No further changes are necessary.

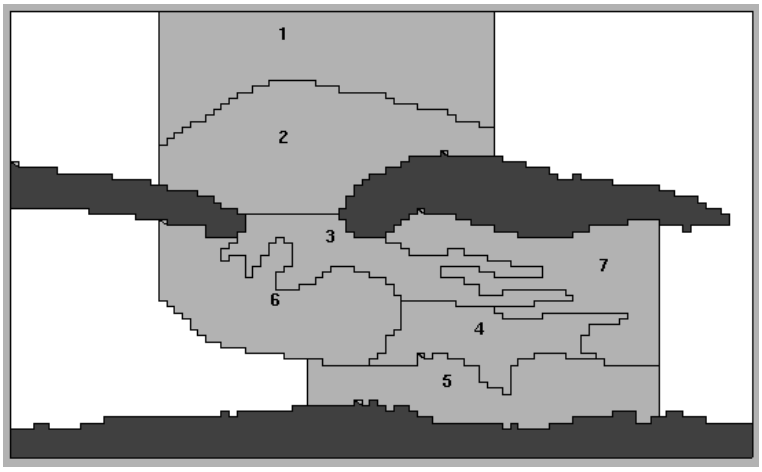


Figure 5.1: SPIEKEROOG-setup of the model. Box 1 corresponds to the North Sea. Dark grey ares mark island and the coast. White areas are not modeled.

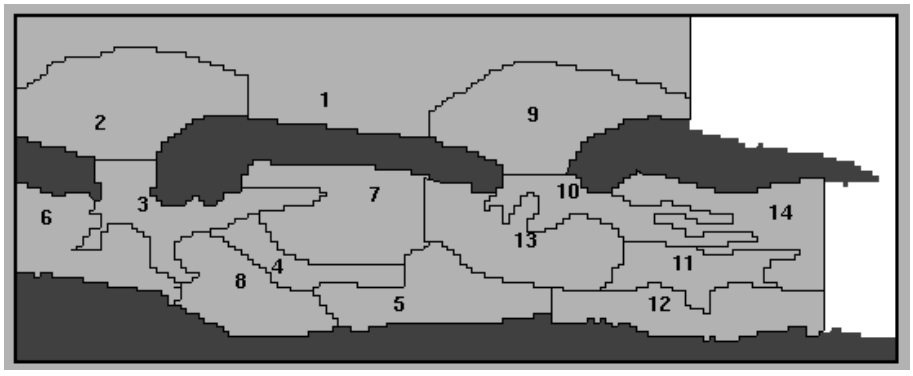


Figure 5.2: LANGEOOG- and SPIEKEROOG-setup of the model. Box 1 corresponds to the North Sea. If the model is run for Langeoog only the boxes 1-8 are active. Dark grey ares mark island and the coast. White areas are not modeled.

6 Forcing

File: `forcing.c`
Parameter file: `topography.def` and `forcing.def`

Forcing comprises all external driving forces. The assigment of the data files for different years and the definition of the string variables holding the file names is done in `setup.c`.

meaning	variable	code file
water temperature	`etw_file`	`etw.c`
sediment temperature	`etb_file`	`etb.c`
suspended matter conc. (turbidity)	`seston_file`	`ess.c`
salinity in water	`esal_file`	`esal.c`
photosynthetically available irradiance	`par_file`	`par.c`
daily mean photosynthetically available irradiance	`par_dm_file`	`parm.c`

Precipitation and evaporation are diagnostic quantities which do not affect the model results. Evaporation influences the modelled salinity `X1x`. This is also a dignostic quantity. The salinity `ESAL` used for the carbonate model **??** are read from a file.

The files are assigned by the string variables defined in `setup.c`. The files containing the data are located in the directory `forcing/data`.

The function call for precipitation is in integration method 2 (Fixstep). This makes sure that the masses coming into the system are correct. The other function calls are done in ntegration method 3 to get smooth curves for the biological functions.

The main routine of the forcing functions reading external data from data files is `forcing`. The forcing functions are firstly called for every box. The files containing the routines have all the same structure. They can be found `etw.c`, `etb.c`,`ess.c`, `esal.c`, `epar.c` , `eparm.c` . The first functions to be called are named `forc_euXXX`. In this routines the forcing model is chosen. There are three possibilities

iswXXX=0	a simple model for diagnostic purposes
iswXXX=1	reading of a time series
iswXXX=2	calculating a more or less sophisticated function (f.e Fourier series or derived values)

Not all possibilities are implemented for all forcing functions. Actually the following switches are available (the default is marked bold):

switch	values	meaning
iswETW	0,**1**	water temperature
iswETB	0,**1**	sediment temperature
iswESS	**1**	suspended matter concentration
iswESAL	**1**	salinity in water
iswEPAR	0, **1**,2	photosynthetically available irradiance
iswEPARm	0, **1**,2	mean photosynthetically available irradiance over photo period

The forcing values are read in variables named box_XXX for every box. In a second step the forcing for the voxels (PAR, ETW, ESS) is set.

6.1 Light conditions

6.1.1 Reading PAR from a file

File: epar.c
Switch: iswPAR =1

The photosynthetical available light PAR is read from a data file. Additionally the mean daily PAR is read which has also to be provided (must be externally calculated from the time resolved data). These mean values are recalculated to the mean over the photoperiod (PARm).

6.1.2 OLD: Reconstructing light from mean over photo period

File: par.c
Switch: iswPAR =2

The mean photoperiodic light conditions at the sea surface of the Southern North Sea `EIR` and the length of the photoperiod are used as forcing. The variable `EIR` is read by default from the COCOA model in `read_northsea`. `EIR` are daily values for irradiance as mean values over the photo period (sunset-sunrise) in W/m^2. These values are recalculated to time dependent values in `par.c`: Let `EIR` be the mean daily water surface irradiance within the photo period (sunset-sunrise) and q the photo period as fraction of the day, τ the time of the day mapped into the interval [$-\pi, \pi$]. An approximation of the time dependent irradiance is given by

$$I(t) = \max\left[0, EIR \cdot \left(\cos\left(\frac{t}{2q}\right) + \cos^2\left(\frac{t}{2q}\right)\right) \cdot \frac{2\pi}{4+\pi}\right] . \tag{6.1}$$

The factor $2\pi/(4+\pi)$ normalizes the equation so that

$$\frac{1}{2\pi} \int_{-\pi}^{\pi} I(t)\, dt = q \cdot EIR . \tag{6.2}$$

Such the global radiation energy reaching the sea surface is conserved. For further details see Ebenhöh *et al.* (1997).

The implementation of this function is a little bit tricky. Due to discretization errors the function I(t) becomes also positive at night and not only within the photo period. Such an additional check is necessary if the actual time is within the photo period or not.

The photo period as fraction of a day is read from a time series made by xtide (Flater, 2005).

The time dependent and the mean light is converted by a constant factor (`pEIR` in `forcing.def`) to photosynthetically available radiance `PAR` and `PARm` . Both build the base for the primary production (`pprod_light.c`, section 12.1) model. Additionally the variable `daylightfac` is stored which allows the recalculation of mean light over the photoperiod into time resolved values. This variable is also needed in `pprod_light.c`.

6.2 Water temperature ETW

File:	`etw.c`	
Switches:	`iswETW`	0 or 1 (default)
Parameter file:	`forcing.def`	
Parameter:	`light_file`	

Water temperature is given by default by a time series of daily values ICBM (2005). At the moment all voxel get the same water temperature.

The water temperature influences several metabolic processes. The model temperature as forcing function is a relict of the ERSEM philosophy. In the Wadden Sea this approach is questionable because the water bodies transport their heat quantity. In the recent model this is neglected because temperature influences the processes only implicitly. According to Van't Hoff's rule (Lampert & Sommer, 1993) leads an increase of temperature of 10 $^{\circ}$ to a reaction enhancement of factor Q_{10} (1.5 to 4 depending on the process). Such the dimensionless regulation factor of the temperature is given by

$$e_T(T) = Q_{10}^{\frac{T-10}{10}} \quad . \tag{6.3}$$

6.2.1 Modifying ETW

File:	`etw.c`	
Switches:	`iswHOT`	0 (default) or 1
	`iswICE`	0 (default) or 1
Parameter file:	`forcing.def`	
Parameters:	`hot_time`, `hot_days`, `hot_temp`	
	`ice_time`, `ice_days`, `ice_temp`	
	`ice_diff_red`	

If `iswHOT=1` a hot summer event is simulated starting at day `hot_time` for the duration of `hot_days`. In this period the temperature is set to `hot_temp`.

A similar scenario is possible if `iswICE=1` is set. In this case a cold period is simulated starting at day `ice_time` for the duration of `ice_days`. In this period the temperature is set to `ice_temp` (should be negative). The temperature `ice_temp` should be set to negative values. The diffusion coefficient for the diffusion between the pelagic and benthic system is modified by the factor `ice_diff_red`.

The mortality of benthic organisms is enhanced. The maximum additional mortalities of the functional groups due to ice stress are defined in `ben_zoo.def` (13.2.4). They are modified by the factor `ice_mort` determine the actual dry part of the box.

6.2.2 Simple calculation of ETW

File: `etw.c`
Switch: `iswETW` =0

As an alternative to the ETW-values from the time series station the values can be calculated by

$$\mathrm{ETW(t)} = 12.42 - 10 \cdot \cos(2\pi(\mathrm{t}-30)/365))$$

This formulation is not affected by the modification in 6.2.1.

6.3 Sediment temperature ETB

File: `etb.c`
Switch: `iswETB` =2

At the moment all boxes get the same sediment temperature as the water column.

6.4 Suspended matter, Seston

File: `derived_forc_vox.c`
Parameter files: `forcing.def`

The suspended matter concentration (seston) given in gram dry weight is determined as the sum of the dynamic silt concentration (section 6.5) and the concentrations of particulate organic states. It is assumed that the dry weight is approximately twice of the carbon weight Colijn (1982). The variable `seston` is only a diagnostic variable. It is calculated in `derived_forc_vox.c`.

6.5 Inorganic suspended matter, silt ESS

File: `forcing.c` and `ess.c`
Switch: `iswESS` =1

Silt (inorganic suspended matter, ESS) is one component of suspended matter (seston). The silt concentration is depth and velocity dependent and are therfore calculated for every voxel.
The annual mean values for the silt concentrations (inorganic suspended matter, ESS) are given for every box as parameters in topography.def as rough estimation in accordance to Colijn (1982). These mean values are multiplied by a time dependent factor. This factor is read by default from a file (pointer:seston_file) for every month. In between an linear interpolation is done. This factor reflects the annual dynamic of the silt concentration within the back barrier system and recalculated from seston measurements (Liebezeit *et al.* , 1996)).

6.6 Turbidity- the extinction coefficient xEPS

File:	derived_forc_vox.c
Parameter files:	topography.def
	forcing.def

The background attenuation. and scattering due to suspended matter leads to an extinction of light in the water column. The available part of the surface irradiance at depth d is given by $e^{-\sigma d}$, where σ is the attenuation or extinction coefficient (in 1/m
Colijn gives the following relation between background extinction σ_0 suspended matter concentration C (in mg/m^3) and extinction coefficient σ

$$\sigma = \sigma_0 + 0.04 \cdot 10^{-3} C_{seston} \ , \tag{6.4}$$

This linear correlation is also valid during the spring bloom at high phytoplankton densities (Colijn). In the Wadden Sea shadowing due to phytoplankton or detritus plays a minor role because of the high amount of inorganic matter.
The extinction coefficient in the model (xEPS) depends on the background extinction (parameter EPS0 in topography.def), the silt concentration ESS and the concentration of the most important organic states (phytoplankton and detritus):
The weighting factors for the different states are set in forcing.def. The extinction coefficient determines the light conditions for phytoplankton (section 12.1): xEPS is calculated in derived_forc_vox.c for every voxel.

6.7 Turbulence

File: `derived_forc_vox.c`
Parameter files: `forcing.def`

The turbulence factor (`turb`) depends on current velocity and the water depth. The turbulence determines the sedimentation velocities of detritus and phytoplankton (section 12.4 and section 12.1.9). Higher current velocities and lower water depth leads to higher turbulence. . The normed regulation factor e_{turb} is defined as follows :

$$e_{turb} = \left(1 - \frac{D}{D + D^h}\right) \cdot \frac{\parallel v_{tide} \parallel}{\parallel v_{tide} \parallel + v^h}, , \tag{6.5}$$

where D is the actual water depth, $\parallel v_{tide} \parallel$ the absolute current velocity of the tidal current of the considered voxel . D^h and v^h are half saturation values defined in `forcing.def`

7 Sources and sinks of the model

The sources of the model comprises the input from the floodgate, freshwater input due to precipitation (diagnostic), input from an aquifer (optional) and the atmospheric input into the water and into the sediment during dryfall. The assigment of the data files for different years and the definition of the string variables holding the file names is done in `setup.c`.

meaning	**variable**	**code file**
evaporation (diagnostic)	parameterized	`evaporation.c`
precipitation (diagnostic)	`pre_file`	`precipitation.c`
flood gate		
time	`gate_time_file`	`floodgate.c`
loadings	`gate_load_file_xxx`[1]	`floodgate.c`
concentrations	`gate_conc_file_xxx`[2]	`floodgate.c`
aquifer (optional)		
actually same as floodgate:		
loadings	`gate_load_file_xxx`	`aquifer.c`
concentrations	`gate_conc_file_xxx`	`aquifer.c`
loadings	`aqui_load_nhs`	prepared but not used
concentrations	`aqui_conc_file_nhs`	prepared but not used
atmospheric -> water		
nitrate	parameterized	`atm_intake.c`
ammonium	parameterized	`atm_intake.c`
phosphate	parameterized	`atm_intake.c`
CO_2	parameterized	`atm_intake.c`
atmospheric -> sediment		
nitrate	parameterized	`bennut.c`
ammonium	not implemented	-
phosphate	not implemented	-
CO_2	parameterized	`bennut.c`
O_2	parameterized	`bennut.c`

[1] `xxx` stands for `nhs`, `dor` resp `ben` for the floodgates Neuharlingesiel, Dornumersiel resp. Bensersiel.

[2] `gate_conc_file_xxx` contains data for salinity, phosphate, nitrate, ammoinium and silicate

7.1 Flood gate

File:	`floodgate.c`	
Parameter file:	`floodgate.def`	
	`gate_load_file_xxx`	loadings
	`gate_conc_file_xxx`	concentrations
	`gate_time_file`	time stamp for releasing
Switch:	`iswGATE`	0: no input by floodgate (default)
		1: input by floodgate (default)

Freshwater and nutrient input from the floodgate at Neuharlingersiel are given by time series for daily volume loadings and nutrient concentrations of freshwater. The files holding the data are set in setup (see 4.6.1). The volumes are estimations from Dellwig (n.d.). Nutrient concentrations are monthly measurements they are interpolated by linear interpolation (Kölsch *et al.* , 2003). Normally the water is released twice a day nearly at low tide. The file should be replaced by real data if existing.

The values for the floodgates Dornumersiel and Bensersiel (only used for `NORDERNEY` and `BOTH` setup) are estimations from Neuharlingersiel.

7.1.1 Modifying floodgate input

The floodgate values may be overruled. It is possible to define a dry period without input and a special event within this period. The concentrations can be modified by a factor. The switches and parameters can be found in `floodgate.def`.

7.2 Aquifer

File:	`aquifer.c`
Parameter file:	`aquifer.def`
Data files:	`gate_load_file_xxx` resp. `aqui_load_file_nhs (actually prep`
	`gate_conc_file_xxx` resp. `aqui_conc_file_nhs`
Switch:	`iswAQUI`

Freshwater input through an aquifer can be added. The input of freshwater is assumed as ground water input near the coast. For the back barrier system of Spiekeroog the water is released into box 5 (resp. 12), for Langeoog into box 8 and 5. The amount of freshwater and the nutrient loadings are read from files[3] . The files holding the data are set in setup (see 4.6.1). The amount can be modified by the factor `aqui_fac`, the nutrient loadings by `aqui_enh`.

7.3 Precipitation

File:	`precipitation.c`	
Parameter file:	`precipitation.def`	
Switch:	`iswPRE`	: 0: no precipitation
		1: precipitation (default)

Precipitation is treated as diagnostic quantity affecting only the salinity. It will not change the volumes of the voxel and has no impact on the model results for carbon and nutrient concentrations. Precipitation values are given as daily values from meteomedia (2004). The files are set in setup (see 4.6.1).

7.3.1 Modifying precipitation

The precipitation values may be overruled. It is possible to define a dry period and a special event within this period. The switches and parameters can be found in `precipitation.def`.

7.4 Evaporation

File:	`evaporation.c`	
Parameter file:	`evaporation.def`	
Switch:	`iswEVA`	: 0: no evaporation
		1: evaporation (default)

Evaporation is treated as diagnostic quantity affecting the modelled salinity `X1x`. It will not change the volumes of the voxel and has no impact on the model re-

[3] At the moment the same freshwater input an loadings of the aquifer are assumed to be equal to that of the floodgate

sults for carbon and nutrient concentrations. Evaporation is given by a parameter determine the mean evaporation at 10 °C. The temperature dependency is given by a factor according to Arrhenius formula

$$e_T = Q_{10}^{\frac{T-10}{10}}$$

7.5 Derived forcing quantities

7.5.1 Shear stress

The shear stress only affects the manganese model extension (section 15.2). It has no influence on the carbon and nutrient cycling. It is distinguished between the bottom shear stress vbot in (kgm/s^2)/m^2 (in a sense of TKE $\times \rho$, the turbulente kinetic energy times density) and a shear factor for free water in 1/s. The bottom shear velocity factor v_b is determined by the absolute tidal velocity of the voxel $\|v\|$ (see `calc_tracer.c`):

$$v_b = v_0 \|v\| , \tag{7.1}$$

where v_0 is a parameter (see `forcing.def`).
The bottom shear stress tau$_b$ in (kgm/s^2)/m^2 is per definition given by the product of the turbulent kinetic energy (TKE) and the density of the water (ρ = 1023 kg/m^3):

$$\tau_b = v_b^2 \rho . \tag{7.2}$$

The shear factor for free water τ can be approximated by the depth dependent gradient of the velocity and is set to

$$\tau = \frac{\|v\|}{\frac{D}{2}} \tag{7.3}$$

where D is the actual depth of the voxel.

7.5.2 Stickiness

The stickiness only affects the manganese model extension (section 15.2). It has no influence on the carbon and nutrient cycling. The stickiness factor st is

determined as normed abundance of phytoplankton

$$st = \frac{1}{P_{st}} \sum_{i} P_i \tag{7.4}$$

where i denotes for the different Phytoplankton groups and $P_s t$ is a parametrized reference abundance.

8 North Sea Boundary

The boundary conditions at the North Sea boundary are either read from a file or are calculated simultaneously by a slave model (`iswNORTHSEA=1`, section 3.3). This files have the structure of a **CEMoS**-pk file (Fourier data) and are evaluated in `read_northsea.c` (Fourier synthesis). This method avoids any discontinuities. The structure of the boundary file and how to create it is described in section 17.1.
These values can be overuled by field data in `set_northsea_add.c`. Overruling is a convenient way to use field data as boundary conditions without creating a new pk-file.
A further very simple way of modifying the boundary values can be done in `set_north_sea.c`. Factors, modyfiying the actual read values of the main state variables can be set in `northsea.def`.

8.0.1 Reading the boundary model result file

Files: `init_northsea.c`, `read_northsea.c` and `set_northsea.c`
Switch: `iswNORTHSEA` (1: holding initial values , 2: reading from file (default))

The model result file acting as boundary condition must have the structure of a **CEMoS** `_pk.outc` file (spectral values). Fourier series can be built with **MoViE** (Kohlmeier & Hamberg, 2004). The file must contain the results for every state variable that shall be used as boundary condition for exactly one box (corresponding to box 1 in **EcoTiM**). The names of the variables in the slave model result file and in **EcoTiM** must be equal. State variables which are not available in the slave model result file hold their initial value set in `initial.def`. The filename of the slave model result file must be set in `setup.c` for the specific year. The allocation is done in `set_northsea`. Preparations as opening the result file is done in `init_northsea`, reading of values is done in `read_northsea`.

8.0.2 Overruling the boundary model result file

File: `set_northsea_add.c`
Switch: `iswNSXXX` (0: no overruling, 1: overruling)
`XXX` = `N1p`, `N3n`, `N4n`, `N5s`, `SAL`, `ALK`, `O3c` resp. `pCO2`

Overruling is is prepared for several states resp. quantities. The assigment of the data files for different years and the definition of the string variables holding the file names including the path is done in `setup.c`.

meaning	**overr. variable**	**switch**	**string variable**
phosphate	`N1p`	?`iswNSN1p`	`northsea_n1p_file`
nitrate	`N3n`	?`iswNSN1p`	`northsea_n3n_file`
ammonium	`N4n`	?`iswNSN4n`	`northsea_n4n_file`
silicate	`N5s`	?`iswNSN5s`	`northsea_n5s_file`
salinity	`ESAL`[1]	?`iswNSSAL`	`northsea_sal_file`
alkalinity	`ALK`	?`iswNSALK`	`northsea_alk_file`
DIC	`O3c`	?`iswNSO3c`	`northsea_o3c_file`
pCO_2	`pCO2`[2]	?`iswNSpCO2`	`northsea_pco2_file`

The code can be found in `set_northsea_add.c`. Preparations as opening the files is done in `init_northsea_add`, reading of values is done in `set_northsea_ad`. The files containing the data must be given as csv-files, where the first row is a header. Each row must contain a time and a value. The data will be linearily interpolated.

8.0.3 Evaluating the boundary conditions

File: `mixing_eubox1.c`
Parameter file: `northsea.def`
Parameter: `mix1`

Voxels entering box 1 are "'mixed"' with North Sea water by a certain mount. This means that a mixing process on all transportable state variables takes place. This mixing is done in `mixing_eubox1`.
The parameter values can be interpreted as follows:
0.0: no mixing
1.0: full mixing within one day

[1] `ESAL` is not a state variable and different from `X1x`. `ESAL` is used as forcing and is used to calculate the density in the carbonate model.

[2] `pCO2` is not a state variable. It is used to calculate the atmospheric DIC input in the carbonate model.

100.0: full mixing within one time step=0.01day

9 Tide

The information about the tidal amplitude and the tidal length is needed for the movement of the voxel (10). The water level also determines the interchange area of the benthic and pelagic system (4.9). By default (`iswTIDE=1`) the normalized amplitude is read in `read_pegel` from a file .

9.1 High water and tidal length

The time stamps of high water and the tidal length of the corresponding tide is read in

9.2 Low water

The time stamps of low water are only used for the flood gate time stamps so that the phase shift is correctly for years without loading data where the default loadings are used. The time stamps are also stored in the specific loading files.

9.3 Tidal amplitude

By default (`iswTIDE=1`) the tidal amplitude is read from a file (see 4.6.6). The tidal amplitude is read in `read_pegel.c`. The values from xtide are normalized to [–1, 1] and are scaled by the tidal range during spring tide. This value is used to calculate the actual water level at every grid point in `calc_griddepth` (see 9.4).

Calculation of tidal amplitude
Alternatively (`iswTIDE=2`), the tidal amplitude is calculated in `calc_pegel.c`.

9.4 Calculation of the water level

The calculation of the water level for every grid point is done in `calc_griddepth.c`. The water level H at the position (x, y) at time t is calculated by:

$$H(t, x, y) = H_m(x, y) + \frac{H_T}{2} \cdot H_{amp}(t)$$

where H_m is the mean water level at position (x, y) read in `init_grid.c`. H_{amp} is the normalized amplitude according to section 9.3 and H_T is the tidal range at spring time (set in `tide.def`).

9.5 Dryfall area

For every box the part of the box which is covered by water is determined from the actual depth at every grid point (variable: `wet` in). Such the part which is fallen dry is given by `1-wet`. This values are needed for the benthic oxygen dynamics.

10 Transport model

10.1 Voxel setup

File:	`calc_tracer.c`	
Switch:	`iswTRACER`	1: predefined trajectory
		2: calculated by field (default)
Parameter file:	`model.def`	
Parameter:	`v_turb`	maximum displacement in m/s

The number of voxel `voxnum` is set in `model.def`. The number is arbitrary but must not exceed the maximum value `MDIM`. `MDIM` is also defined in `model.def` and is the model dimension. All state variables and globals are of this dimension.

The volume of all voxel is equal and calculated in `init_grid.c` from the total volume in the back barrier system. The initial position of all voxel is set as `%int_ind_par xcoor` and `ycoor` in `model.def`. Normally this setting is overruled in `cemos.par` with `initial/coordinates.dat`.

The actual position of each voxel is either read from `transport_data/trajec.outc` (`iswTRACER=1`) or calculated from the velocity field (`iswTRACER=2`). The recalculation from the velocity field has the effort that the velocity can be modified during simulation (f.e. to simulate a storm event), while reading from data file is faster.

10.1.1 Voxel movement

Due to the choice of the box boundaries corresponding to the water sheds the exchange at the west and east boundaries is negligible. Tracer on the verge of leaving the model area or moving on land are kept hold for that time step. The North Sea boundary conditions are taken into account by identifying box 1 with the North Sea conditions. Tracer entering box 1 are mixed with North Sea water by a certain rate.

Turbulent diffusion is considered by a randomized offset of water bodies. Diffusion between different water bodies is implemented by an exchange process. Transport of benthic material is separately treated as exchange process between adjacent sediment areas.

10.1.2 Calculating position from the velocity field

File:	`setup.c`	
Switch:	`iswTRACER`	=1
Parameter:	`tracer_data_file`	spectral values

The velocity field is given in form of spectral values. For each grid point spectral values are given for x– and y–direction.
The file holding the data is set in `setup.c`. The variable holding the name is `tracer_data_file`. The files has the structure of a **MoViE** result file (for the structure of see Kohlmeier & Hamberg (2004)). Such it is possible to have a look to these data by **MoViE**. The file must contain the following information and store variables:
For the result file header:
Store variables:
u1[1-324] - u88[1-324]: velocities in u-direction for the 88 lines and 324 columns
v1[1-324] - v88[1-324]: velocities in uv-direction for the 88 lines and 324 columns

The file must not contain any other store variable!!
The information are read in in `init_tracer.c`. The default file is `tracer_freq.outc`
The information for the spectral values (name of the data file, data structure and number of represented tidal cycles) are set in `init_tracer`.
Alternatively to the velocity field a file holding trajectories for all tracers can be evaluated (see 10.1.3).

10.1.3 Reading position from the trajectory file

File:	`setup.c`	
Switch:	`iswTRACER`	=2
Parameter:	`trajec_result_file`	trajectories

A trajectory file must have the structure of a **MoViE** result file. It must contain the following information and store variables:

```
xcoor[1-MDIM]
ycoor[1-MDIM100]
vx[1-MDIM100]
```

`vy[1-MDIM100]`

The values must be stored every 0.01 day (outdelt=0.01). Normally the trajectory file is written by **EcoTiM** itself and reused for further simulation runs.

Attention: The coordinates stored in the data file will be repeated if the actual simulation time exceeds the data. This may lead to inconsistent results if the tide and the trajectory do not match to each other.

10.1.4 Turbulent diffusion

File:	`calc_tracer.c`	
Parameter file:	`tracer.def`	
Parameter:	`v_turb`	maximum displacement in m/s

Turbulent diffusion is considered by a randomized offset of water bodies. The concentration of an arbritrary substance is considered as the number of particles within a volume. The change in concentration can then be described by a mean velocity of all particles and an individual stochastic movement of every single particle. Every voxel is subject to such a movement. Every voxel gets a displacement x at time τ ,

$$x = u\tau, \tag{10.1}$$

where u is given by a distribution φ. According to Einstein (1905) the turbulent diffusion coefficient is given by

$$K = \frac{1}{2\tau} \int_{-\infty}^{\infty} y^2 \varphi(y) dy \tag{10.2}$$

and depends only on the distribution φ. For an uniform distribution of u in [–U, U], describing the typical Monte-Carlo-Method, follows

$$K = \frac{1}{6} U^2 \tau. \tag{10.3}$$

In the model a two-dimensional distribution is considered. For a maximum velocity of 35 cm/s, corresponding to a displacement of 300 m or 1.5 grip points per time step (0.01 d), the diffusion coefficient amounts K = 35 m^2/s.

This value describes the turbulent diffusion coefficient of voxels but can also be interpreted as diffusion coefficient of the considered substance.

10.1.5 Diffusive Mixing

File:	`mixing.c`	
Switch:	`iswMIX`	1: switched on (default) , 0: switched off
Parameter file:	`mixing.def`	
Parameter:	`mix`	mixing parameter

Diffusion between adjacent voxels is implemented by an exchange process. This exchange process takes place between all voxels within one box[1]
For an arbitrary substance with concentration c in voxel i holds

$$c(i, t + \Delta t) = c(i, t) + \sigma \Delta t \sum_{j \in Box} (c(j, t) - c(i, t)). \tag{10.4}$$

For$\sigma\,\Delta t < 1$ this method is numerically stable. For $\Delta t = 0.01$ d is this condition therefore no constraint.The effective force length is assumed to be between 1 km and 8 km. The diffusion coefficient is assumed to be in the magnitude of 35 m^2/s. Then the mixing parameter σ is between 0.2 and 1.4 per day (Kohlmeier, 2004, appendix). The parameter values can be interpreted as follows:

0	no mixing
1	full mixing within one day
10	full mixing within one time step of 0.1 day

10.1.6 Transport of benthic material

File:	`ben_part_trans.c`	
Switch:	`iswPTP`	1: switched on (default) , 0: switched off
Parameter file:	`topography.def`	
Parameter:	`q_part[i]`	considered part of harmonic area per day

Transport of benthic material is treated as exchange process between adjacent sediment areas. It is assumed that benthic particulate matter is transported coastwards due to the tidal asymmetry.

[1] This is an constraint due to computation time.

For the calculation of this transport the harmonic mean A_h of the two adjacent boxes i, j with the areas A_i resp. A_j are considered.

$$A_h = \frac{A_i \cdot A_j}{A_i + A_j} \tag{10.5}$$

The transported mass M_{ij} from box i to box j per day is given by

$$M_{ij} = r_i \cdot A_h \cdot Q_i \ , \tag{10.6}$$

where r_i is the part of the harmonic area A_h considered per day and Q_i the concentration of detritus[2] within the i-th box.
The new detritus distribution is calculated for the benthic model.

[2]The C-,N-,P-,Si-parts are treated separately

11 The ecosystem model

The ecological model of **EcoTiM** is built on the base of ERSEM (Baretta-Bekker, 1995; Baretta-Bekker & Baretta, 1997; Vichi, 2002). ERSEM is a biomass based differential equation model, which describes the cycling of carbon and nutrients within an ecosystem. According to the ERSEM-philosophy of a top-down approach, the food web consists of functional groups and not of single species. The considered food web is built in respect to trophic position, size, and function. The model describes the main metabolism processes within the functional groups and the predation processes between the functional groups. These processes are given by carbon assimilation, nutrient uptake, and lysis processes of primary producers, grazing processes of secondary producers, and respiration, mortality excretion, and exudation of all organisms. The considered food web is given in Fig. 11.1.

The ecological model of **EcoTiM** is not identical to ERSEM. Some parts of the model base on ERSEM II (Baretta-Bekker & Baretta, 1997) other on ERSEM III (Vichi, 2002). Some changes in the formulation were necessary to adapt the model to the Wadden Sea system.

The ecological part of the model can be understood as ordinary differential equation system where temporal changes of state variables are defined. The pelagic system is simulated in every voxel, the benthic system in every box. The model equations are given in the form

$$\frac{d}{dt}Z = r(Z, ...) \cdot Z \; . \tag{11.1}$$

or expressed in **CEMoS**-notation

```
SZ[i] += ... \cdot Z[i]\; .
```

The multiplication of the dynamic change rate with the state variable will be omitted below.

The description of the processes must be seen as an overview. For clearness not every process is described in form of an equation. Additionally, the identifier used in the description are not unique. The meaning should be clear from the particular context. A nearly complete description can be found by Vichi (2002).

The nomenclature of the model follows as far as possible the ERSEM naming convention (Blackford & Radford, 1995). The names according to the ERSEM naming convention are given in brackets in `typewriter` . Tables of all states variables

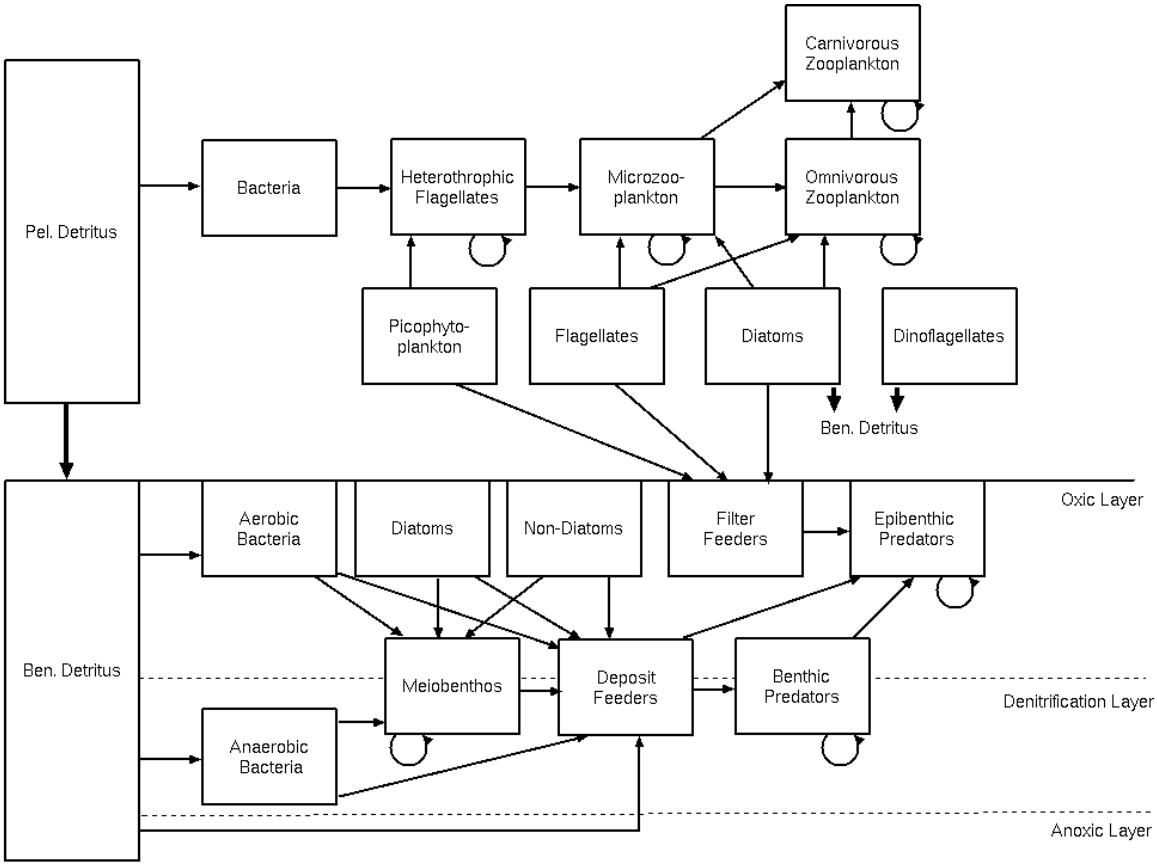


Figure 11.1: Conceptual diagram of the pelagic and benthic food web showing functional groups and fluxes. The position of the functional groups within the food web reflect the size of the organisms and their trophic position. The trophic position increases from bottom to top, the sizes of organisms increase from the left to the right. Narrow arrow symbolize biomass fluxes. The circles symbolize pseudo-cannibalism (Kohlmeier & Ebenhöh, 1995). Bold arrows symbolize sedimentation fluxes. The thicknesses of the benthic layers in the diagram do not reflect their real thicknesses.

and fluxes including their meanings and units are given in section A.1.

12 The pelagic model

Pelagic state variables are given in mg C m^{-3} resp. mmol m^{-3}. The pelagic nutrients are given by

- ammonium (`N4n`) in mmol m^{-3},
- nitrate in (`N3n`) mmol m^{-3},
- phosphate (`N1p`) in mmol m^{-3},
- silicate in (`N5s`) mmol m^{-3},

The pelagic gases are

- oxygen (`O2o`) in mmol m^{-3}
- carbon dioxide (`O3c`) in mg C m^{-3} (exception because of carbon content)
- nitrogen (`O4n`) in mmol m^{-3}.

The pelagic food web (Fig. 11.1) consists of four functional groups for primary producers of four groups of secondary producers as well as one group for pelagic bacteria.
The primary producers are subdivided as follows

- diatoms (`P1c, P1n, P1p, P1s`): 20 —m - 200 —m ESD (equivalent spherical diameter), procaryotic eukaryote with cell walls containing silicate
- flagellates (`P2c, P2n, P2p`): 2 —m - 20 —m ESD, mobile procaryotic eukaryote, f.e. *phaeocystis*
- picophytoplankton (`P3c, P3n, P3p`): 0.2 —m - 2 —m ESD, small autotrophic procaryote with preference for ammonium as nitrogen source
- dinoflagellates (`P4c, P4n, P4p`): 20 —m - 200 —m ESD, larger phytoplankton, inclusive inedible species

Every of these functional group is described by three resp. four state variables for the C, N, and P content. For the diatom group additionally the state for silicate.
The secondary producers are subdivided into

- heterotrophic nanoflagellates (`Z6c,Z6n,Z6p`): 2 —m - 20 —m ESD, f.e. protozoa

- microzooplankton (`Z5c, Z5n, Z5p`): 20 —m - 200 —m ESD, heterotrophic microzooplankton without flagellates
- omnivorous mesozooplankton (`Z4c`): 200 —m - centimeters, f.e. copepodae *acartia* and meroplankton
- carnivorous mesozooplankton (`Z3c`): 200 —m - centimeters, f.e.annelidae

The larval stages of multicellular zooplankton are assigned to the particular functional group. (Broekhuizen *et al.* , 1995). Predation of species within a functional group is described by pseudo cannibalism. This process is a stabilizing factor for the total system (Kohlmeier & Ebenhöh, 1995).
Primary producers, microzooplankton and heterotrophic flagellates are modeled with a variable internal content of nutrients. The mesozooplankton groups are modeled with a fixed C:N:P ratio according to Redfield *et al.* (1963).

The microbial loop contains state variables for pelagic bacteria (`B1c, B1n, B1p`), and detritus (`R6c`, `R6n`, `R6p`, `R6s`). In contrast to ERSEM dissolved organic matter (DOM, `R1c`, `R1n`, `R1p`) is also modeled dynamically. All groups are given as state variables for C,N and P., detritus also for Si.
In this model is carnivorous mesozooplankton the top predator of the system. Losses due to predation by carnivorous and the mortality of carnivorous itself contain also losses due to mammals, birds and fishery.

12.1 Primary producer

File:	`pprod.c`	
Parameter file:	`pprod.def`	
Switches:	`iswP1`	=1
	`iswP2`	=1
	`iswP3`	=1
	`iswP4`	=1

It is assumed that the structural part of the cells has a fixed C:N:P-ratio according to Redfield (Redfield *et al.* , 1963) while the storage capacity for nutrients (except silicate) follows Droop's kinetic (Droop, 1973) with a variable C:N:P ratio. Therfore, for each group exist at least 3 (4 for diatoms) state variables representing the C-, N-, P-, Si-amounts. The assimilation of carbon is modeled

independently of the nitrogen and phosphate uptake but depends on the external light conditions and the light status of the cells (Ebenhöh *et al.* , 1997). The adaptive process of phytoplankton according to the actual light conditions is described by a state variable representing the light history for all primary producers (see 12.1). The four functional groups of phytoplankton are modeled as unique as possible. Every group is described by each one state for carbon, nitrogen and phosphate. The diatom group has additionally a state for silicate. The dynamic of the primary producers is described by the following processes (Fig. 12.1):

- assimilation
- exudation
- respiration
- nutrient uptake
- lysis
- sedimentation losses
- predation losses

In the following squared brackets assign terms corresponding only to some of all functional groups

12.1.1 Nutrient dependent regulation factors

Several processes within the cell depend on the actual nutrient to carbon ratio.This ratio builds the base of the regulation factors. If not explicitly mentioned no distinction between the different nutrient is made in the following description. To avoid conflicts in the notation an arbitrary nutrient is assigned to **N** while N describes nitrogen. If the specific nutrient is relevant to describe the regulation factor it is indexed by p for phosphate , by n for nitrogen and by s for silicate.
The minimum of nutrient to carbon ratio is bounded below due to the nearly constant ratio of the structural parts of the cell. This is assumed to be half of the Redfield value ($n^{min} = 0.5\ n^{R}$) according to Sommer (1994).[1]. The maximum ratios are limited by the storage capacity of the cell and amount about the double of the Redfield values.
The regulation factors of phosphate and nitrogen limitation are given by

$$e_p = \min\left(1, \max\left(0, \frac{n_p - n_p^{min}}{n_p^R - n_p^{min}}\right)\right), \quad e_n = \min\left(1, \max\left(0, \frac{n_n - n_n^{min}}{n_n^R - n_n^{min}}\right)\right).$$

[1] The atomic ratios according to Redfield: C:N:P:Si = 106:15:1:15.

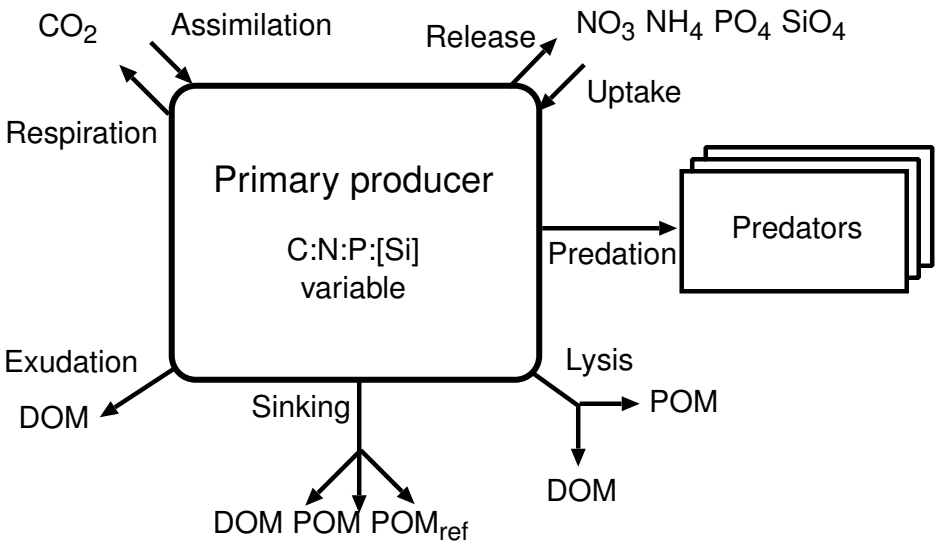


Figure 12.1: Diagram of the modeled primary producer. The processes and their target quantities are shown. Primary production and respiration modify only the carbon content of the functional group. Nutrient uptake and release are decoupled from carbon uptake. The exudation products are assumed to be dissolved (DOM). Lysis and sedimentation products are partly particulate (POM) and partly dissolved (DOM) depending on the actual cell status. The partitioning follows the differentiation between structural parts of the cells and cytoplasmatic content. Sedimentation products are partly refractory (POM_{ref}).

(12.1)

The regulation factor of silicate limitation e_s only depends on the external silicate concentration N_s. Actually no storage capacity for silicate is implemented in the model.

$$e^s = \frac{N_s}{2h_s} , \tag{12.2}$$

where h_s is the concentration leading to half of the maximum assimilation rate. It is assumed that all processes underly a limitation according to Liebig . Such the nutrient status determine the processes is given by

$$e^N = \min(e_p, e_n) \qquad \text{resp. for diatoms} \qquad e^N = \min(e_p, e_n, e_s) \tag{12.3}$$

gilt.

12.1.2 Assimilation

The assimilation rate of carbon r_{ass} depends on the maximum assimilation rate constant r_{ass_0} at 10 $^\circ$ and regulation factors for light Licht (e_I) and temperature (e_T, see 6.2). The gross assimilation of diatoms also depends on the silicate concentration of the water (e_{Si}).

$$r_{ass} = r_{ass_0} \cdot e_I \cdot e_T \cdot \left[e_{Si}\right] \tag{12.4}$$

The assimilation rate of flagellates, mainly represented by *Phaeocystis*, sinks at to high temperature.[2]. The temperature factor e_T for flagellates is therfore modified (see Kohlmeier (2004)).

12.1.3 Light dependency

File: `pprod_light.c`
Parameter file: `pprod_light.def`

The light regulation factor e_I depends on the actual photosynthetic available irradiance I_0 which is the surface irradiance multiplied by a factor determine the P-synthetically available radiance `pEIR` (see also 6.1) and on the actual optimal utilizable irradiance I_{opt} the so called optimal light (section 12.1.4).
The light dependency is modeled according to Ebenhöh *et al.* (1997). The averaging over the day as used in ERSEM is omitted.
The irradiance I as function of the water depth z is given by

$$I(z) = I_0 \, e^{-\sigma z} \tag{12.5}$$

with the extinction coefficient σ. The mean light dependent productivity within the water column [0, D] is given by

$$prod(I_0) = \frac{1}{D} \int_0^D p(I(z)) \, dz \; . \tag{12.6}$$

where p(I(z)) is the productivity per cubic meter with the productivity curve

$$p(I) = p_0 \frac{2x}{1 + x^2} \; , \qquad x = \frac{I}{I_{opt}} \tag{12.7}$$

[2] In culture the assimilation of *Phaeocystis* is inhibited above 20 $^\circ$ and stops above 25 $^\circ$ (Elbrächter *et al.* , 1994)

according to Ebenhöh *et al.* (1997) with the maximum productivity p_0 .

The light dependent regulation factor results in

$$e_I = \frac{\text{prod}(I_0)}{p_0} \ . \tag{12.8}$$

Assuming p_0 as depth independent, e_I becomes independent of the choice of p_0. The dependence of the productivity on the optimal light leads to an initial decrease of productivity at increasing light until the cells are adapted again.

12.1.4 Light adaptation

File: `pprod_light.c`
Parameter file: `pprod_light.def`

Algal cells are able to adapt to changing light conditions. The adaptation process normally takes about several days. Such, short term fluctuations have no impact. It is assumed that the cells adapt to the mean daily photosynthetically available irradiance $\bar{I}_0$ and not to the day night rhythm. The adaption is modeled as relaxation process with rate constant r_I and the equilibrium value $\tilde{I}_{opt}$. The dynamic of the so called optimal light I_{opt} (`PIi`) is given by

$$\frac{d}{dt} I_{opt} = r_I \left(\tilde{I}_{opt} - I_{opt} \right) \ , \tag{12.9}$$

where

$$\tilde{I}_{opt} = \max \left(I_{opt}^{min}, \overline{I(D_a)} \right) \tag{12.10}$$

holds. I_{opt}^{min} is a free parameter describing the minimum value of I_{opt}. It is assumed that the adaptation takes place in the water depth D_a which is set to the half of the actual water depth at most 5 m. The mean irradiance for adaption is then given by

$$\overline{I(D_a)} = \bar{I}_0 \cdot e^{-\sigma D_a} \ . \tag{12.11}$$

The adaption process is assumed to be equal for all pelagic primary producer.

12.1.5 Exudation

The exudation describes the excretion of dissolved constituents and is assumed to be proportional to the assimilation. The amount depends on the cell's nutrient status. The exudation rate r_{exu} is given by

$$r_{exu} = \left(q_{exu_0} + (1 - q_{exu_0})(1 - e^{N})\right) \cdot r_{ass} \ , \tag{12.12}$$

where q_{exu_0} is the part of assimilated material which is exudated in the case of maximum nutrient storage. If the cells are totally nutrient limited, the complete gross assimilation is exudated.

12.1.6 Respiration

The respiration is composed of the basal respiration and the activity respiration. The basal respiration rate depends only on the temperature. The activity respiration is given by a part of the assimilated carbon less the carbon loss due to lysis.

$$r_{res} = r_{res_0} \cdot e_T + q_{res}\,(r_{ass} - r_{exu}) \ , \tag{12.13}$$

r_{res_0} is the basal respiration rate at 10 $^\circ$.

12.1.7 Lysis

Lysis comprises all processes which lead to the destruction of the cell's membrane resp. wall. In contrast to the exudation process also the structural parts of the cell are released. It is assumed that the nutrient content of the structural parts is consistent with the minimum possible value. The structural part become particulate organic material under lysis. The cytoplasm with its variable nutrient content becomes mainly dissolved material. If the cell's are totally nutrient limited all lysis products become particulate material. Such the part turning to particulate is given by

$$q_{part} = \min(1, \frac{n_p^{min}}{n_p}, \frac{n_n^{min}}{n_n}) \ , \tag{12.14}$$

while the remaining part becomes dissolved material.

The lysis rate increases with increasing stress due to nutrient limitation and

amounts

$$r_{lys} = r_{lys_0} \cdot \frac{h_N}{e_N + h_N} , \tag{12.15}$$

where r_{lys_0} is the specific maximum lysis rate constant and h_N the half saturation value. Dinoflagellates have an additional density dependent mortality due to local nutrient limitation and shadowing effects.
The nutrient dynamic of the lysis process is similar to the carbon dynamic. The nutrients within the lysis products turns to dissolved organic matter or detritus depending on the cell status. Silicate components always turns to detritus.

12.1.8 Nutrient uptake

The growth rate of phytoplankton depends on the internal nutrient concentration of the cell (Droop, 1973). Therefore the uptake of nutrients is independent of the carbon assimilation and the photo synthetical activity of the cell. The nutrient uptake depends on the internal cell status as well as on the external nutrient concentration. To take both factors into account two independent uptake rates $f_{up_{ext}}$ and $f_{up_{int}}$ are determined where the effective value is given by the smaller one.
Assuming a cell which is nearly "'empty"' and consists only of structural parts. Such a cell has a minimum **N**:C ratio and its uptake rate is proportional to the external concentration of the nutrient[3] with a rate constant λ reflecting the permeability of the cell membrane (Aksnes & Egge, 1991):

$$r_{up_{ext}} = \lambda \cdot \mathbf{N} . \tag{12.16}$$

The internal uptake rate depends on the actual net carbon production and the potential uptake to fill the nutrient storage

$$r_{up_{int}} = r_{ass_{net}} n_{max} + r\,(n_{max} - n) , \tag{12.17}$$

with the maximum regeneration rate constant r, the maximum **N**:C ratio n_{max}, the actual **N**:C ratio n and the specific net carbon assimilation rate

$$r_{ass_{net}} = r_{ass} - r_{resp} - r_{exu} - r_{lys} . \tag{12.18}$$

[3]Nitrogen is available in form of ammonium and nitrate. The preference depends on the permeability of the cell membrane.

The uptake of silicate by diatoms depends directly on the specific net assimilation. No storage capability of silicate is actually assumed in the model.

12.1.9 Sedimentation losses

Sedimentation losses occur if nutrient stress leads to sinking of phytoplankton. This occurs especially during silicate stress phases in case of diatoms and during nitrate stress phases in case of dinoflagellates (Varela *et al.* , 1995). Therefore additionally to a base sinking velocity a stress dependent enhancement of the sinking velocity is assumed which depends on the nutrient status of the cells. Such the sedimentation rate depends on the actual water depth. Low water depth leads to a relatively high sedimentation rate while the impact of the sedimentation in during deep water phases is relatively low. The sinking depends also on the turbulence. Such the sinking velocity v is a function of the nutrient status and the turbulence (section 6.7):

$$v = (v_0 + v_{max} \cdot \max(0, n_{low} - n)) \cdot (1 - e_{turb}) \ , \tag{12.19}$$

where v_0 is the base sedimentation velocity, v_{max} the maximum sinking velocity during total nutrient stress and e_{turb} a regulation factor describing the degree of turbulence (section 6.7). The actual **N**:C ratio is given by n, while n_{low} describes the ratio underneath sedimentation is enhanced. The sedimentated material turns mainly to fast degradable benthic material (`Q1`), a small part turns to benthic detritus (`Q6`) resp. refractory material (`Q7`).

12.1.10 Predation losses

The losses due to predation are given by the predation rates of the several functional groups of secondary producers. The are given in the appendix A.11 and are described in section 12.2.

12.1.11 Chlorophyll

Actually chlorophyll is derived from the carbon content of the phytoplankton groups. It is assumed that the chlorophyll to carbon ratio is constant (diatoms: 1:25, other: 1:50).

12.2 Secondary producers

Files:	`microzoo.c`	
	`mesozoo.c`	
Parameter files:	`microzoo.def`	
	`mesozoo.def`	
Switches:	`iswZ5`	=1
	`iswZ6`	=1
	`iswZ3`	=1
	`iswZ4`	=1

The secondary producers are described by two functional groups of mesozooplankton and two groups of mikrozooplankton as part of the microbial loop. The main difference in the formulation is of technical nature. While mesozooplankton is assumed to have fixed nutrient to carbon ratios, microzooplankton is modeled similar to phytoplankton with variable ratios varying due to food composition. The secondary producers are described by the following processes (Fig. 12.2):

- predation
- assimilation, excretion, exudation and mortality
- respiration
- exudation of nutrient surplus

12.2.1 Microzooplankton

12.2.2 Predation

The predation rate of microzooplankton depends on the water temperature and the available amount of food. The dependence on the food availability is described by a Holling type II response (Lampert & Sommer, 1993). The predation of the carbon component is given by

$$r_{pred} = r_{pred_0} \cdot e_T \cdot \frac{F_c}{F_c + F_c^h} \tag{12.20}$$

where r_{pred_0} is the maximum uptake rate at 10 °C, e_T the temperature factor according to (6.2), F_c the total available carbon content of food and F_c^h the half saturation value.

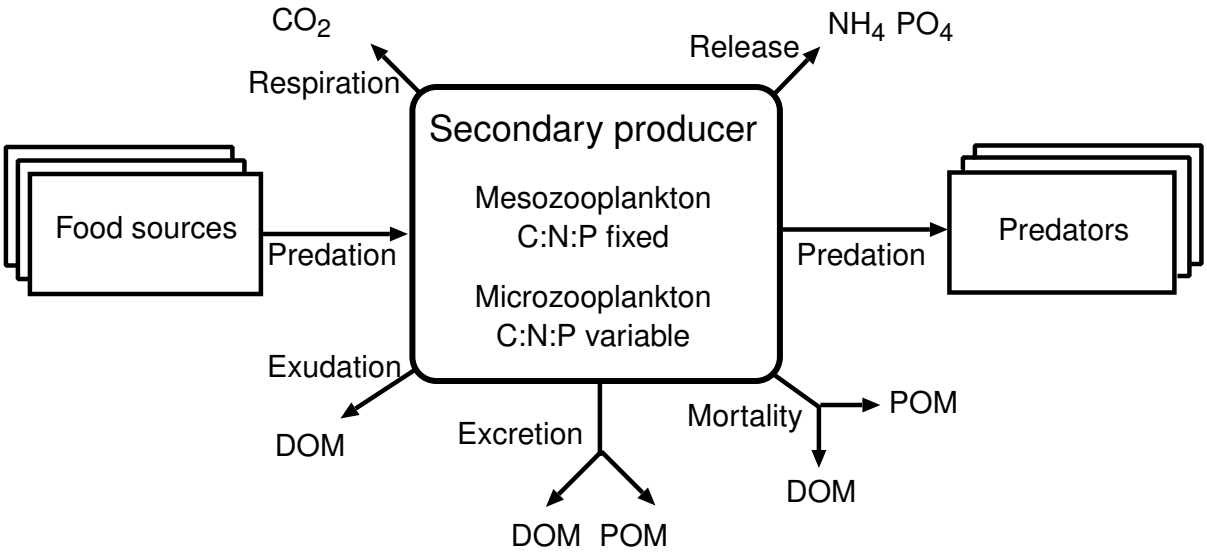


Figure 12.2: Diagram of secondary producer. The C:N:P ratio of microzooplankton is variable, while mesozooplankton is assumed to have a fixed ratio. The processes and their target quantities are shown. Respiration modify only the carbon content of the functional group. The exudation products are assumed to be dissolved (DOM). Defecation and mortality products are partly (with a fixed ratio) particulate (POM) and partly dissolved (DOM) depending on the actual cell status. The release of nutrients preserves the fixed C:N:P ratio of mesozooplankton

The food amount F_c comprises the carbon contents of the several food sources according to Tab. A.11:

$$F_c = \sum_i q_i \cdot e_i \cdot X_i \tag{12.21}$$

where X_i denotes the carbon content of the i-th food source and q_i the preference factor of zooplankton for the i-th food source according to Tab. A.11[4]. In case of microzooplankton it is assumed that food sources of higher abundance are disproportionally high favored. The regulation factor e_i for the food density is given in form of a Michaelis-Menten response. At a food density of X_i^h the preference is halved.

$$e_i = \frac{X_i}{X_i + X_i^h} \ . \tag{12.22}$$

[4]Further indices are omitted for clearness.

The nitrogen, phosphate and silicate amount[5] of the food are incorporated according to the actual nutrient to carbon ratio of the food.

12.2.3 Assimilation, excretion, exudation and mortality

It is assumed that a constant part of the food is assimilated while the remaining part is excreted resp. respirated. The net assimilation rate is given by

$$r_{pred_{net}} = q_{pred} \cdot r_{pred} \ , \tag{12.23}$$

and the excretion rate is

$$r_{excr} = q_{excr} \cdot (1 - q_{pred}) \cdot r_{pred} \ , \tag{12.24}$$

where q_{excr} denotes the part which is excreted and $(1 - q_{excr})$ the part which is respired (equation 12.27). The excretion products turns with a fixed ratio to dissolved and particulate material. The mortality rate is composed of a natural mortality rate constantr_{mort_0} and a part depending on the oxygen saturation of the water (equation 12.5):

$$r_{mort} = r_{mort_0} + r_{mort_{O_2}} (1 - e_{O_2}) \ , \tag{12.25}$$

The regulation factor e_{O_2} depends on the relative oxygen saturation O_{SAT}:

$$e_{O_2} = \frac{O_{SAT}}{O_{SAT} + O^h_{SAT}} \ . \tag{12.26}$$

where O^h_{SAT} is the half saturation value, where the oxygen dependent mortality reaches half of its maximum value at total oxygen depletion.
Nutrients are exudated proportionally to the carbon exudation according to the actual **N**:C ratio of the considered microzooplankton group.

12.2.4 Respiration

The respiration is composed of the basal respiration and the activity respiration. The basal respiration rate depends only on the temperature with a basal respiration rate constant r_{resp_0} at 10 $^\circ$. The activity respiration is derived from the

[5] The silicate uptake is only a technical process for unification and all silicate is excreted immediately.

excretion rate (12.24)

$$r_{resp} = r_{resp_0} \cdot e_T + (1 - q_{excr}) \cdot (1 - q_{pred}) \cdot r_{pred} \,. \tag{12.27}$$

12.2.5 Predation losses

The losses due to predation are given by the predation rates of the several functional groups of secondary producers. The are given in the appendix A.11.

12.2.6 Mesozooplankton

12.2.7 Predation

The predation rate of mesozooplankton depends on the water temperature and the available amount of food. The amount of food is determined by a search volumev per time step (in m^3/(mg C.d)) and the handling time. The handling time and the search volume are temperature and season dependent (Broekhuizen *et al.* , 1995). In the model only a temperature dependency is implemented. For details see Broekhuizen & Bryant (1996). The maximum uptake rate at 10 $^\circ$Cis then given by

$$r_{pred_0} = \frac{1}{\tau} \,, \tag{12.28}$$

and F_c^h the half saturation value for the Holling type II response

$$F_c^h = \frac{r_{pred_0}}{v} \,. \tag{12.29}$$

The predation rate equals to (compare equation 12.20)

$$r_{pred} = r_{pred_0} \cdot e_T \cdot \frac{F_c}{F_c + F_c^h} \,, \tag{12.30}$$

where e_T is the temperature factor according to equation **??**and F_c the total available food. The half saturation value for the Holling type II response F_c^h is given by

$$F_c^h = \frac{r_{pred_0}}{v} \,. \tag{12.31}$$

The index c denotes the carbon content of the food. The total amount of fF_c comprises the carbon contents of the several food sources according to Tab. A.11:

$$F_c = \sum_i q_i \cdot X_i \tag{12.32}$$

where X_i is the carbon content of the i-th food source and q_i the preference factor of zooplankton for the i-th food source according to Tab. A.11 In case of mesozooplankton it is assumed that the food uptake is proportional to the density of the considered source. This assumption is made in respect to the high diversity of species (Broekhuizen *et al.* , 1995).

The nitrogen, phosphate and silicate amount[6] of the food are incorporated according to the actual nutrient to carbon ratio of the food.

12.2.8 Excretion, exudation and mortality

It is assumed that a part of the food is exudated, a part is excreted in form of fecal pellets and a part is respirated. The difference to the formulation of the microzooplankton is that the excretion rate and the respiration rate are independently of each other.

The excretion rate is given by

$$r_{excr} = q_{dil} \cdot r_{pred} + q_{fec} \cdot r_{pred} \ , \tag{12.33}$$

where q_{dil} is the part exudated in dissolved form and $q_{fec}$the part excreted as fecal pellets. The parts q_{dil} and q_{fec} depend on the considered element (C,N,P,Si).

The temperature dependent mortality rate is given by

$$r_{mort} = r_{mort_0} \cdot e_T \ . \tag{12.34}$$

where r_{mort_0} is the mortality rate constant at 10 $^\circ$Cwater temperature and e_T the temperature regulation factor according to equation 6.2.

Nutrient losses due to mortality are proportionally to the carbon losses according to the fixed **N**:C ratio of the considered mesozooplankton group.

[6] see footnote 5 on page 76.

12.2.9 Respiration

The respiration is composed of the basal respiration and the activity respiration. The basal respiration rate depends only on the temperature with a basal respiration rate constant r_{resp_0} at 10 °. The activity respiration rate depends on the gross uptake rate r_{pred} where the fixed part q_{resp} is respired

$$r_{resp} = r_{resp_0} \cdot e_T + q_{resp} \cdot r_{pred} \; . \tag{12.35}$$

12.2.10 Predation losses

The losses due to predation are given by the predation rates of the several functional groups of secondary producers. The are given in the appendix A.11.

12.2.11 Exudation of surplus nutrients

Due to the definition of mesozooplankton with a fixed **N**:C ratio surplus nutrients violating this ratio are exudated. This process is of technical nature and is described in detail in Vichi (2002).

12.3 The microbial loop- Bacteria, detritus and dissolved matter

The functional group of pelagic bacteria (destruents) covers all heterotrophic species with the capability to mineralize dissolved organic matter and detritus. This decomposition processes can be either aerobic or anaerobic. In the actual model version no differentiation between aerobic or anaerobic bacteria and free living bacteria and aggregate-attached bacteria is made. Bacteria (`B1`), dissolved organic matter (DOM, `R1`) and detritus (`R6`) are assumed to have variable C:N:Pratios. Bacteria are described by the following processes (Fig. 12.3):

- uptake of DOM and detritus
- respiration
- uptake and exudation of nutrients

12.3.1 Assimilation

The ingestion of substrate depends either on the maximum possible uptake rate at the actual environmental conditions or on the actual supply of substrate. The

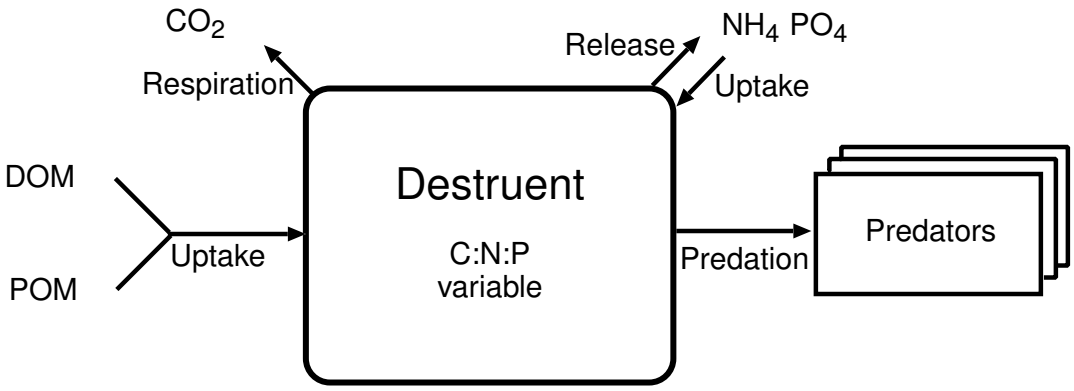


Figure 12.3: Diagram of bacteria. The C:N:P ratio of bacteria is variable. The processes and their target quantities are shown. Respiration modify only the carbon content of the functional group. Destruents ingest dissolved and particulate substrate, but the ingestion rate for dissolved material is higher. In case of nutrient depleted substrate bacteria ingest inorganic nutrients.

ingestion rate in case of a surplus supply amounts

$$r_{upt_{int}} = r_{upt_0} \cdot e_{O_2} \cdot e_T \ , \tag{12.36}$$

where r_{upt_0} is the maximum uptake rate at 10 $^\circ$ and e_T the temperature regulation factor according to equation 6.2. The regulation factor e_{O_2} describes the dependence of the activity on the actual relative oxygen saturation O_{SAT} according to equation 12.26. This ingestion rate is only reached if enough substrate is available. It is assumed that the total available dissolved organic matter and a part of the available detritus POC^{up} can be ingested. This part depends on the nutrient content of the detritus. The substrate dependent uptake rate is given by

$$r_{upt_{ext}} = \frac{DOC + POC^{up}}{B_c} \ . \tag{12.37}$$

where B_c is the carbon content of the actual bacterial biomass.

The effective uptake rate is given by the smaller one of the both rates $r_{upt_{int}}$ and $r_{upt_{ext}}$:

$$r_{upt} = \min(r_{upt_{int}}, r_{upt_{ext}}) \ . \tag{12.38}$$

12.3.2 Respiration

The respiration is composed of the basal respiration and the activity respiration. The basal respiration rate depends only on the temperature with a basal respiration rate constant r_{resp_0} at 10 $^{\circ}$. The activity respiration is derived from the uptake rate r_{upt}. It is differentiated between an aerobic or anaerobic bacterial activity. Dependent on the oxygen saturation of the water the composition of the functional group is interpreted more or less aerobic. It is taken into account that the anaerobic decomposition of organic material costs more energy than the aerobic mineralization. Such, the anaerobic decomposition is assumed to be less effective resulting in a higher respiration The respiration rate is given by

$$r_{resp} = r_{resp_0} \cdot e_T + \left(1 - q_{ox} \cdot O_{SAT} - q_{anox} \cdot (1 - O_{SAT})\right) \cdot r_{upt} \,. \qquad (12.39)$$

with the bacterial efficency q_{ox} at high and q_{anox} at low oxygen saturation.

12.3.3 Nutrient uptake

The uptake of nutrients from detritus is proportional to the carbon uptake flux. If the actual cellular nutrient to carbon ratio is higher than the maximum ratio, surplus nutrients are released into the water. If the actual cellular nutrient to carbon ratio is lower than the physiologic value, inorganic nutrients are ingested from water. This ingestion is regulated by a Michaelis-Menten kinetic. Bacteria and phytoplankton compete in this case for nutrients.

12.3.4 Predation losses

The losses due to predation are given by the predation rates of the several functional groups of secondary producers. The are given in the appendix A.11 and are described in section 12.2.

12.3.5 Silicate regeneration

The regeneration of silicate in the pelagic system is modeled as first order process proportional to the silicate content of detritus.

12.3.6 Nitrification

The transformation of ammonium to nitrate under oxygen bonding is chemically given by

$$(NH_4^+ + OH^-) + 2\,O_2 \longrightarrow (NO_3^- + H^+) + 2\,H_2O\,. \tag{12.40}$$

In the model the nitrification is modeled as first order process with a temperature dependent (equation 6.2) nitrification rate.

12.4 Sedimentation of detritus

The detritus sedimentation velocity is given by

$$v_{sed} = v_{sed_0} \cdot (1 - e_{turb})\,, \tag{12.41}$$

with the base sedimentation velocity v_{sed_0} and the normed turbulence factor e_{turb}. (section 6.7). The regulation factor increases with increasing current velocity and decreasing water depth, so that the sedimentation velocity is highest at low current velocities an high water level.

12.5 Oxygen saturation

The oxygen saturation $O_{100\%}$ in mmol O_2/m^3, where the saturation of the water is 100 % is calculated by (Baretta & Ruardij, 1988)

$$O_{100\%} = \frac{475.0 - 2.65 \cdot C_{Salt}}{33.5 + T}. \tag{12.42}$$

where C_{Salt} is the salt concentrationand und T the temperature in $^\circ$. The paramters are empirical values The relative saturation O_{SAT} for at the actual concentration C_{O_2} is given by

$$O_{SAT} = \frac{C_{O_2}}{O_{100\%}}. \tag{12.43}$$

12.6 Salinity

Salt influences only the calculation of oxygen saturation. The salt concentration of the back barrier areas depends on the input of fresh water from the flood gate, precipitation and evaporation. The salt concentration is only for diagnostic purposes and do nearly not influence the results.

13 The benthic bilogical model

Main file:	`ben.c`
files:	`ben_*.c`
Parameter files:	`ben_*.def`
Switches:	`iswBEN` 1

The benthic system with its storage and remineralization capacity constitutes the main nutrient source of the Wadden Sea. The importance of the exchange processes as sedimentation, advection and diffusion between the pelagic and benthic system increases with decreasing water depth (Ebenhöh *et al.* , 2004). The essential processes within the benthic system which are considered in the model are the intake of organic material into the benthic system and the nutrient cycling of carbon, phosphate, nitrogen and silicate within the benthic food web. Further the diagenetic processes within the pore water are considered. To describe remineralization of detritus to inorganic nutrients correctly, the vertical structure of the sediment is considered implicitly. The upper 30 cm of the sediment is differentiated into 3 layers. The penetration depths of oxygen (`D1m` in m) and nitrate (`D2m` in m) are modeled dynamically (Fig. 13.1):

- oxic layer (0-`D1m`): upper layer of few millimeter containing free oxygen where oxygen respiration and nitrification is possible, the
- denitrification layer (`D1m` - `D2m`): middle layer of few millimeter up to some centimeters where oxygen is only in form of nitrate available and energy production is only possible by denitrification and the
- anoxic layer (`D2m` -`d_tot`): lower layer of few centimeters where sulphate reduction occurs[1].

Benthic state variables are given in mg C m^{-2} resp. mmol m^{-2}. The benthic nutrients are given by

- ammonium (`K4n`) in mmol m^{-2},
- nitrate in (`K3n`) mmol m^{-2},
- phosphate (`K1p`) in mmol m^{-2},
- silicate in (`K5s`) mmol m^{-2},

The benthic gases are

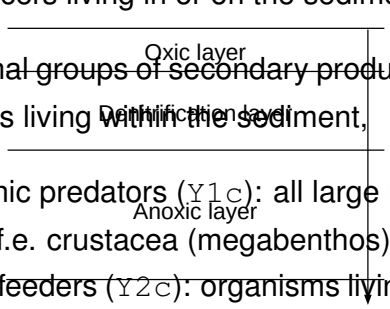
[1] strictly speaking also the denitrification layer is anoxic. In the ERSEM model only the nitrate free part of the sediment is named anoxic.

Figure 13.1: Layers of the sediment. The thicknesses of the layers are not drawn to scale. In the model the thicknesses are represented by dynamic state variables and therefore variable in time. The oxic layer measures normally about a few millimeter, the other a few centimeters. The total thickness of the model sediment is set to 30 cm.

- oxygen (`G2o`) in mmol m^{-2}
- carbon dioxide (`G3c`) in mg C m^{-2} (exception because of carbon content)
- nitrogen (`Gn`) in mmol m^{-2}.

All exchange processes between the pelagic and benthic system are defined as benthic processes within the model. In the recent stage the benthic model of **EcoTiM** bases on the so called "'Oldenburg nutrient model"' of ERSEM **?**. Such, processes as pore water advection and erosion of benthic material are missing. The benthic food web (Fig. 11.1) consist of two functional groups of benthic primary producers,

- benthic diatoms (`A1c, A1n, A1p, A1s`): all silicate dependent primary producers living in or on the sediment[2] and
- benthic non-diatoms (`A2c, A2n, A2p`): all silicate independent primary producers living in or on the sediment, especially cyano bacteria,



two functional groups of secondary producers living at the sediment surface and three groups living within the sediment,

- epibenthic predators (`Y1c`): all large mobile organisms at the sediment surface, f.e. crustacea (megabenthos),
- deposit feeders (`Y2c`): organisms living within the sediment feeding on detritus and smaller organisms of the meiobenthos (f.e. polychaetae as *arenicola marina* and *lanice conchilega*),

[2] Erosion of benthic algae and settling of pelagic species is neglected in the model

- filter feeders (`Y3c`): all nearly immobile organisms at the sediment surface filtering the surrounding water (f.e. molluscae),
- meiobenthos (`Y4c`): small organisms (< 1 mm), feeding mainly on bacteria,
- benthic predators (`Y5c`): carnivorous living organisms of mean size living in the oxidized layer of the sediment.,

Benthic primary producers are assumed two have a variable C:N:P ratio (diatoms have also a variable Si:C ratio). All secondary producers are assumed to have a fixed C:N:P ratio.
The microbial loop of the benthic model comprises dissolved organic matter (DOM, `Q1c, Q1n, Q1p`), fast degradable (`Q6c, Q6n, Q6p, Q6s`) and refractory (`Q7c, Q7n, Q7p`) detritus as well as aerobic (`H1c`) and anaerobic (`H2c`) bacteria (destruents). The organic material is assumed two have variable nutrient to carbon ratios while the ratio of bacteria is fixed
The destruents live in different layers:

- aerobic bacteria bacteria (`H1c`, oxic layer) ,
- anaerobic nitrate reducing bacteria (nitrate reducing layer) ,
- anaerobic sulphate reducing and methanogenic bacteria (nitrate reducing layer) .

The last two groups are combined in the model to anaerobic bacteria (`H2c`).
The horizons of the sediment layers (`D1m` and `D2m`) are affected by *bioturbation* and *bioirrigation*: Biologic activity ("'digging"' the sediment, excretion into deeper layers) transports detritus in deeper layers and modifies the actual diffusivity within the pore water. Releasing of nutrients into the free water and permeation of oxygen into the sediment become enhanced.

The process description of the functional groups and of the microbial loop are nearly identical with the formulation of ERSEM Ebenhöh *et al.* (1995); Blackford (1997). Two additional groups for benthic diatoms and benthic non-diatoms as primary producers were added to adapt the model to the constraints of the Wadden Sea. The process description of these primary producers is oriented on the formulation of the pelagic groups and the formulation suggested by Blackford (2002). The nutrient cycling and the dynamic variation of the sediment horizons are described by the so called "'Oldenburg nutrient model"' of ERSEM (Ebenhöh *et al.* , 1996).

The nomenclature of the model follows as far as possible the ERSEM naming convention (Blackford & Radford, 1995). Tables of all states variables and fluxes including their meanings and units are given in section A.3.
The benthic organisms are subdivided in modules for primary producers, secondary producer and destruents. The functional groups within a specific modules differ principally only in their parameterization.

13.1 Benthic primary producers

File:	`ben_pprod.c`	
Parameter file:	`ben_pprod.def`	
Switches:	`iswA1`	=1
	`iswA2`	=1

The benthic primary producers are modeled similar to the pelagic phytoplankton The two groups for benthic diatoms (`A1c, A1n, A1p, A1s`) and benthic non-diatoms (`A1c, A1n, A1p`) are described by each one state for carbon, nitrogen and phosphate. The diatom group has additionally a state for silicate. The dynamic of the primary producers is described by the following processes (Fig. 12.1):

- Assimilation
- Exudation
- Respiration
- Nutrient uptake
- Lysis
- Predation losses

Both groups are combined in the food web to benthic algae because no differentiation in predation on these groups is made.

13.1.1 Assimilation

The assimilation rate for carbon r_{ass} depends on the maximum assimilation rate constant r_{ass_0} at 10 $^\circ$, on the regulation factors for light e_I, temperature e_T (equation 6.2) and silicate concentration within the pore water e_{Si} (section .

12.1.1) ab:

$$r_{ass} = r_{ass_0} \cdot e_I \cdot e_T \cdot e_{Si}. \tag{13.1}$$

Watermann *et al.* (1999) showed in experiments concerning the competition of benthic cyano bacteria and diatoms that at temperatures of 10 $^\circ$ and 20 $^\circ$ benthic diatoms dominate (*Nitzschia sp.*), while at 25 $^\circ$ cyano bacteria (*M. chthonoplastes*) dominate. It is assumed in the model that high water temperatures inhibit the photosynthetic activity. Therfore the temperature regulation factor is modified correspondingly (see 12.1.2).

13.1.2 Light dependency

File: `ben_pprod_light.c`
Parameter file: `pprod_light.def`

The light dependency is modeled accordingly to the pelagic description (12.1.2). It is assumed that benthic primary producers are at the sediment's surface during the dry fall period. During flood the algae dig into the sediment for a few centimeters Asmus *et al.* (1994). The productivity on the dry sediment (the part $(1 - q_{wet})$ of the box) depends on the photosynthetic available irradiance at the sediment's surface I_0 (section 6.1). On the flooded part of the box q_{wet} the depth of the water column above z and its extinction coefficient sigma (section **??**) determines the available light at the surface. Additionally the extinction coefficient within the sediment σ_D determines the irradiance at a certain depth D within the sediment. The depth D and the extinction coefficient of the sediment σ_D are parameters given in (Tab. A.13).
The productivity curve p is assumed to be equal to the pelagic one (section 12.1.2)

$$p(I) = p_0 \frac{2x}{1 + x^2}, \qquad x = \frac{I}{I_{opt}} \tag{13.2}$$

where p_0 describes the maximum productivity.
The mean productivity within the box $prod(I_0)$ is determined by the weighted sum of the productivity at dry fall and the productivity at the actual mean water depth z:

$$prod(I_0) = p_0 \cdot \left(\frac{2x}{1 + x^2} \cdot (1 - q_{wet}) + \frac{2y}{1 + y^2} \cdot q_{wet} \right) \tag{13.3}$$

with

$$x = \frac{I_0}{I_{opt}} \qquad \text{and} \qquad y = \frac{I_0 \cdot e^{-\sigma z} \cdot e^{-\sigma_D D}}{I_{opt}}.$$

The light dependent regulation factor is given by

$$e_I = \frac{prod(I_0)}{p_0}. \tag{13.4}$$

13.1.3 Light adaptation

File: `ben_pprod_light.c`
Parameter file: `ben_pprod_light.def`

As pelagic primary producers benthic algae adapts to the light conditions. The adaptation depth according to section 12.1.4 is the mean actual water depth within the box. It is taken into account that the part $(1-q_{wet})$ (**??**) of the box falls dry.
The mean irradiance determine the so called benthic optimal light (`AIi`, see also 12.11) of benthic diatoms is given by

$$\overline{I(D_a)} = \bar{I}_0 \cdot \left(e^{-\sigma D_a} \cdot q_{wet} + (1 - q_{wet}) \right). \tag{13.5}$$

13.1.4 Exudation

Exudation comprises the release of dissolved part of the cells. It is assumed that the exudation is proportional to the assimilation. The rate depends on the actual nutrient cell status. The exudation rate amounts

$$r_{exu} = \left(q_{exu_0} + (1 - q_{exu_0})(1 - e^N) \right) \cdot r_{ass}, \tag{13.6}$$

where q_{exu_0} is the part of assimilated material which is exudated under optimal nutrient conditions. In case of total nutrient limitation the total gross assimilation is exudated.

13.1.5 Respiration

The respiration is composed of the basal respiration and the activity respiration. The basal respiration rate depends only on the temperature. The activity respi-

ration is given by a part of the assimilated carbon less the carbon loss due to lysis.

$$r_{res} = r_{res_0} \cdot e_T + q_{res} \left(r_{ass} - r_{exu}\right) . \tag{13.7}$$

r_{res_0} is the basal respirations rate at 10 $^\circ$.

13.1.6 Lysis

Lysis comprises all processes which lead to the destruction of the cell's membrane resp. wall. In contrast to the exudation process also the structural parts of the cell are released. It is assumed that the nutrient content of the structural parts is consistent with the minimum possible value. The structural part become particulate organic material under lysis. The cytoplasm with its variable nutrient content becomes mainly dissolved material. If the cell's are totally nutrient limited all lysis products become particulate material Such the part turning to particulate is given by

$$q_{part} = \min(1, \frac{n_p^{min}}{n_p}, \frac{n_n^{min}}{n_n}) , \tag{13.8}$$

while the remaining part becomes dissolved material.
The lysis rate increases with increasing stress due to nutrient limitation and amounts

$$r_{lys} = r_{lys_0} \cdot \frac{h_N}{e_N + h_N} \tag{13.9}$$

where r_{lys_0} is the specific maximum lysis rate constant and h_N the half saturation value. Benthic diatoms have an additional density dependent mortality due to local nutrient limitation and shadowing effects.
The nutrient dynamic of the lysis process is similar to the carbon dynamic. The nutrients within the lysis products turns to dissolved organic matter or detritus depending on the cell status. Silicate components always turns to detritus.

13.1.7 Nutrient uptake

The growth rate of phytoplankton depends on the internal nutrient concentration of the cell (Droop, 1973). Therefore the uptake of nutrients is independent of the carbon assimilation and the photo synthetical activity of the cell. The nutrient

uptake depends on the internal cell status as well as on the external nutrient concentration. To take both factors into account two independent uptake rates $f_{up_{ext}}$ and $f_{up_{int}}$ are determined where the effective value is given by the smaller one.

Assuming a cell which is nearly "'empty"' and consists only of structural parts. Such a cell has a minimum **N**:C ratio and its uptake rate is proportional to the external concentration of the nutrient[3] with a rate constant λ reflecting the permeability of the cell membrane (Aksnes & Egge, 1991):

$$r_{up_{ext}} = \lambda \cdot N. \tag{13.10}$$

The internal uptake rate depends on the actual net carbon production and the potential uptake to fill the nutrient storage:

$$r_{up_{int}} = r_{ass_{net}} n_{max} + r\,(n_{max} - n)\,, \tag{13.11}$$

with the maximum regeneration rate constant r, the maximum **N**:C ratio n_{max}, the actual **N**:C ratio n and the specific net carbon assimilation rate

$$r_{ass_{net}} = r_{ass} - r_{resp} - r_{exu} - r_{lys}\ . \tag{13.12}$$

The uptake of silicate by diatoms depends directly on the specific net assimilation. No storage capability of silicate is actually assumed in the model.

13.1.8 Predation losses

The losses due to predation are given by the predation rates of the several functional groups of secondary producers. The are given in the appendix A.24 and are described in section 13.2.

13.1.9 Chlorophyll

Actually chlorophyll is derived from the carbon content of the phytoplankton groups. It is assumed that the chlorophyll to carbon ratio is constant (diatoms: 1:25, non diatoms: 1:50).

[3] Nitrogen is available in form of ammonium and nitrate. The preference depends on the permeability of the cell membrane.

13.2 Benthic secondary producers

Files:	`ben_zoo.c`	
Parameter files:	`ben_zoo.def`	
Switches:	`iswY1`	=1
	`iswY2`	=1
	`iswY3`	=1
	`iswY4`	=1
	`iswY5`	=1

The benthic secondary producers are described by the following processes (Fig. 12.2):

- predation resp. filtration
- respiration
- mortality
- excretion
- predation losses
- exudation of nutrient surplus

13.2.1 Predation

The predation rate depends on the maximum predation rate r_{pred_0} at 10 °C, the temperature, the availability of oxygen and food:

$$r_{pred} = r_{pred_0} \cdot e_T \cdot e_O \cdot \frac{F_c}{F_c + F_c^h} \tag{13.13}$$

where e_T is the temperature regulation factor according to equation 6.2), e_O the oxygen limitation factor, F_c the carbon content of the available food and F_c^h the half saturation value of the Holling type two response of food uptake. It is assumed that a low oxygen concentration within the pore water C_{O_2} influences the uptake rate directly. Additionally an enhanced mortality due to oxygen limitation is assumed. The dimensionless oxygen limitation factor is given by

$$e_O = \frac{(C_{O_2} - C_{O_2}^{min})^3}{(C_{O_2} - C_{O_2}^{min})^3 + {C_{O_2}^h}^3} \tag{13.14}$$

the cubic shape takes care that organisms which are relatively tolerant against oxygen stress live well until the critical boundary concentration $C^h_{O_2}$ is reached. Below the minimum concentration $C^{min}_{O_2}$ living is impossible.
The food amount F_c comprises the carbon contents of the several food sources according to Tab. A.24:

$$F_c = \sum_i q_i \cdot e_i \cdot X_i \tag{13.15}$$

where X_i denotes the carbon content of the i-th food source and q_i the preference factor of the considered functional group for the i-th food source according to Tab. A.24[4]. It is assumed that food sources of higher abundance are disproportion high favored. The regulation factor e_i for the food density is given in form of a Michaelis-Menten response. At a food density of X^h_i the preference is halved.

$$e_i = \frac{q_i \cdot X_i}{q_i \cdot X_i + X^h_i}. \tag{13.16}$$

The nitrogen, phosphate and silicate amount[5] of the food are incorporated according to the actual nutrient to carbon ratio of the food.
If the food source is detritus the vertical distribution is implicitly considered. The potential available food depends on the depth where the organisms of the considered functional group typically lives. Let the habitat given by the layer $[D_{min}, D_{max}]$ and let ρ be the vertical distribution of the total detritus amount Q (see section 13.4). The available detritus as for the functional group $\tilde{Q}$ sums up to

$$\tilde{Q} = Q \cdot \int_{D_{min}}^{D_{max}} \frac{\rho(z)}{\rho_0} dz. \tag{13.17}$$

The nutrient components of detritus are determined correspondingly.

13.2.2 Filtration

The carbon uptake by filtration is different from the uptake by predation. It is assumed that the filtration rate depends on the water volume which can be filtered

[4] Further indices are omitted for clearness.

[5] The silicate uptake is only a technical process for unification and all silicate is excreted immediately.

in a certain time step. Additionally is assumed that particulate organic matter is not distributed homogeneously within the water column and the concentration increases near the sediment's surface. The parameter d_w (m) describes this fact. Multiplied with the available food it can be interpreted as enhancing factor due to higher matter concentrations near the sediment's surface. It can also be interpreted as enhancing factor for the search volume at fixed concentration. During ebb the available volume which potentially can be filtered decreases resp.the amount of food decreases. Such a depth dependent regulation factor e_D is assumed considering this aspect:

$$e_D = \min(d_w, D) \ , \tag{13.18}$$

where D is the depth of the voxel. The potential available food amount under the assumption that filter feeders only feed on pelagic material is given by

$$F_c = e_D \cdot \sum_i q_i \cdot e_i \cdot X_i \ , \tag{13.19}$$

where q_i and e_i are given according to equation 13.15).

The filter capability depends on the density of the filter feeders. High abundances lead to intraspecific competition Lohse (2002). In the model a regulation e_C depending on the density of filter feeders Y is assumed:

$$X = \frac{(Y - Y_l)^2}{Y - Y_l + Y^h} \ . \tag{13.20}$$

X can be interpreted as measure for the effective biomass leading to intraspecific competition. The density where competition starts is given by Y_l and Y^h is a kind of half saturation density for the competition. The effective biomass X determines the normed regulation factor:

$$e_C = 1 - \frac{X}{X + X^h} \ , \tag{13.21}$$

with its half saturation constant X^h.

The graph of e_C is given in Fig. 13.2 for the parameter values in Tab. A.18.. The filtration rate r_{filt} is given according to equation 13.13:

$$r_{filt} = r_{filt_0} \cdot e_T \cdot e_O \cdot e_C \cdot \frac{F_c}{F_c + F_c^h} \ . \tag{13.22}$$

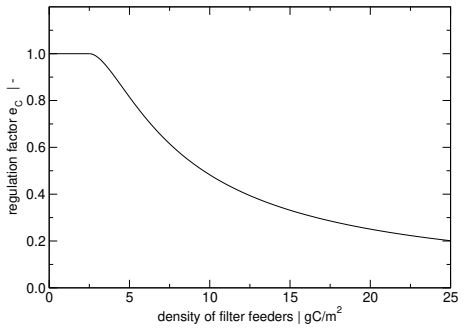


Figure 13.2: [Regulation factor for the density dependent filtration rate. The graph is shown for the parameter values in Tab. A.18 .

The technical realization of the mass fluxes between the Lagrangian water bodies (voxel) and the Eulerian benthic regions (box) is described in section **??**.

13.2.3 Respiration

The respiration is composed of the basal respiration and the activity respiration. The basal respiration rate depends only on the temperature with a basal respiration rate constant r_{resp_0} at 10 $^\circ$. The activity respiration is derived from the gross uptake rate r_{pred} (13.13):

$$r_{resp} = r_{resp_0} \cdot e_T + q \cdot r_{pred} \, . \tag{13.23}$$

13.2.4 Mortality

The temperature dependent mortality rate with its mortality rate constant r_{mort_0} at 10 $^\circ$ increases with increasing oxygen limitation (13.14) from r_{mort_0} to $r_{mort_{0_2}}$:

$$r_{mort} = \max(r_{mort_0}, r_{mort_{0_2}} \cdot (1 - e_O)) \cdot e_T. \tag{13.24}$$

13.2.5 Excretion

A part of the assimilated food q_{excr} is directly excreted and turns to particulate detritus. The excretion rate is given by

$$r_{excr} = q_{excr} \cdot r_{pred} \, , \tag{13.25}$$

where q_{excr} is the actual uptake rate.

Assimilated nutrients turns partly to detritus. If the nutrient content of the food exceeds the fixed nutrient to carbon ratio of the considered functional group the surplus nutrients are exudated in dissolved form.
If the food is nutrient depleted carbon is excreted in particulate form until the fixed nutrient to carbon ratio is reached.

13.2.6 Predation losses

The losses due to predation are given by the predation rates of the several functional groups of secondary producers. The are given in the appendix A.24.

13.2.7 Exudation of surplus nutrients

Due to the definition of secondary producers with a fixed **N**:C ratio, surplus nutrients violating this ratio are exudated. This process is of technical nature and is described in detail in Vichi (2002).

13.3 Benthic bacteria and detritus

In the model it is differentiated between aerobic bacteria living in the upper part of the sediment (comp. section 13.5), and nitrate and sulphate reducers. The dynamic of benthic bacteria is described by the following processes (Fig. 12.3):

- assimilation
- respiration
- mortality
- uptake and exudation of nutrients
- predation losses

13.3.1 Aerobic bacteria

13.3.2 Assimilation

The ingestion of substrate depends either on the maximum possible uptake rate at the actual environmental conditions or on the actual supply of substrate. The ingestion rate in case of a surplus supply amounts The degradation of organic material by aerobic bacteria is assumed to be partly a fast degradation of detritus, partly a slow degradation of detritus and partly a fast degradation of dissolved organic material (DOM). The fast degradation process depends on the

nutrient content of detritus and slows down at low nutrient content. Refractory material is degradated with a very low rate.

The detritus available as source for aerobic bacteria is located within the oxidized layer $[0, D_1]$. Let ρ be the vertical distribution of the total detritus amount Q according to section 13.4 . So the available amount $\tilde{Q}$ is given by:

$$\tilde{Q} = Q \cdot \int_0^{D_1} \frac{\rho(z)}{\rho_0} dz. \tag{13.26}$$

The C-,N-,P-,Si-components of the available detritus are determined separately (The distributions might be different).

The ingestion rate is composed of a fast and a slow part. The slow part depends on temperature and oxygen availability, the fast part depends also on the quality (nutrient richness) of the detritus. The carbon ingestion rate is given by

$$r_{upt} = \left(r_{upt_{slow}} + r_{upt_{fast}} \cdot e_O \cdot e_Q\right) \cdot e_T \cdot \tilde{Q_c}\,, \tag{13.27}$$

where r_{upt_0} is the maximum uptake rate, $\tilde{Q_c}$ the concentration of the carbon content of the available detritus and e_Q, e_T, e_O the regulation factors concerning detritus quality, temperature (equation 6.2) and oxygen.

The oxygen regulation factor e_O results from the actual penetration depth of oxygen, the oxygen horizon (D_1), with a half saturation value D_1^h:

$$e_O = \frac{D_1}{D_1 + D_1^h}. \tag{13.28}$$

The regulation factor e_Q for detritus quality is given as Liebig limitation by

$$e_Q = \min(1, \frac{n_n^{\tilde{Q}}}{n_n^H}, \frac{n_p^{\tilde{Q}}}{n_p^H}) \tag{13.29}$$

where $n_n^{\tilde{Q}}$ resp. $n_p^{\tilde{Q}}$ describes the N:C resp. P:C ratio of the available detritus and n_n^H and n_p^H the fixed corresponding ratios of bacteria.

The decomposition of refractory material ($r_{upt_{refr}}$) is similar to the slow degradation of detritus but with a lower maximum value. The available amount of refractory material is determined analogously to equation (13.26).

The decomposition of dissolved material ($r_{upt_{diss}}$) is faster with an enhanced maximum value. The total amount of dissolved matter is available as source.

The total uptake rate is given by

$$r_{upt} = r_{upt_{part}} + r_{upt_{diss}} + r_{upt_{refr}} \tag{13.30}$$

The uptake of nutrients is modeled analogously to the carbon uptake. The uptake of nutrients from detritus is enhanced and a part of the ingested detritus is released in dissolved form.

13.3.3 Respiration

The respiration is composed of the basal respiration and the activity respiration. The basal respiration rate depends only on the temperature with a basal respiration rate constant r_{resp_0} at 10 $^\circ$. The activity respiration is derived from the gross uptake rate r_{upt} as a constant part q_{upt}:
The respiration rate is given by

$$r_{resp} = r_{resp_0} \cdot e_T + q_{upt} \cdot r_{upt} \ . \tag{13.31}$$

13.3.4 Mortality

The mortality rate r_{mort} of bacteria increases due to oxygen limitation up to the maximum values r_{mort_0}:

$$r_{mort} = r_{mort_0} \cdot (1 - e_O) \ . \tag{13.32}$$

The resulting dead organic matter turns partly to detritus and dissolved matter in a fixed ratio.

13.3.5 Nutrient uptake and release

Bacteria can assimilate if necessary inorganic nutrient from the pore water. This is the case if the nutrients within the detritus and DOM source cannot supply the need given by the fixed nutrient to carbon ratio of bacteria. In this case if temporary the quality of organic matter is too bad nutrients are ingested directly from the pore water. The uptake rate depends on the availability of nutrients within the pore water. If the nutrient to carbon ratio of the ingested material exceeds the bacterial ratio the surplus nutrients are released.

13.3.6 Predation losses

The losses due to predation are given by the predation rates of the several functional groups of secondary producers. The are given in the appendix A.24 and are described in section 12.2.

13.3.7 Anaerobic bacteria

the group of anaerobic bacteria is modeled similar to the aerobic bacteria, but live below the oxidized zone in the denitrification layer $[D_1, d_{tot}]$ (Fig. 13.1). The available detritus in this layer $\tilde{Q}$ is given by (see equation 13.26):

$$\tilde{Q} = Q \cdot \int_{D_1}^{d_{tot}} \frac{\rho(z)}{\rho_0} dz \ . \tag{13.33}$$

Dissolved organic material with its small penetration depth is not available in the denitrification layer. For simplicity it is assumed that organic material resulting from dead anaerobic bacteria turns only to detritus.
It is assumed that anaerobic bacteria satisfies their demand on oxygen exclusively by denitrification. The oxygen regulation factor e_O is therfore determined by the thickness of the layer containing nitrate $[D_2 - D_1]$:

$$e_O = \frac{D_2 - D_1}{D_2 - D_1 + D_2^h} \tag{13.34}$$

where D_2^h is the half saturation value.

13.4 Detritus distribution and bioturbation

The distribution of detritus determines definitively the available food of the detritus consuming organisms. Therefore it is necessary to determine the availability in the different habitats (Ebenhöh *et al.* , 1995). Assuming that the detritus component (C, N, P, Si) has the mean penetration depth D, the density of detritus q can be given as function of depth x by the normed distribution function ρ:

$$q(x) = q_0 \cdot \rho(x, D) \ , \tag{13.35}$$

where$\rho(0, D) = 1$ holds and q_0 the detritus density at the sediment's surface denotes.d

Let Q be the total detritus amount below 1 m^2 sediment and d_{tot} the thickness of sediment, then

$$Q = \int_0^{d_{tot}} q(x)\, dx \tag{13.36}$$

holds and the value q_0 is determined.

$$DQ = \int_0^{d_{tot}} x \cdot q(x)\, dx \tag{13.37}$$

defines D as mean penetration depth. These penetration depths [6] vary due to production and predation in certain depths , due to sinking of phytoplankton and detritus onto the sediment's surface and due to vertical transport (bioturbation) within the sediment . .

If the production rates f_i (consumption, if negative) in the depths D_i are given, the variation of the mean penetration depth is given by

$$\frac{d}{dt}D = \sum_i (D_i - D)\,\frac{f_i}{Q} \; + \; \gamma \tag{13.38}$$

for every distribution ρ. Here γ denotes the variation due to vertical transport.

The vertical transport is caused by physical processes and bioturbation. For simplicity it is assumed that a layer Δ (m/d) between the sediments surface and the depth δ is exchanged. The the equation for γ is given by

$$\gamma = \frac{\tau}{\delta} \cdot \frac{1}{Q} \cdot (q(0) - q(\delta))\,, \tag{13.39}$$

where$\tau = \Delta \cdot \delta$ (m^2/d) is in analogy to a diffusion constant a free parameter and $q = q_0 \cdot \rho(x, D)$ a function of depth and of the mean penetration depth.

Assuming an exponentially decaying detritus concentration within the sediment

$$\rho(x, D) = e^{-\frac{x}{T(D)}} \tag{13.40}$$

holds, whereby the "'extinction parameter"' only on D depends. This choice is obvious because T(D) = D holds for $d_{tot} \to \infty$. For small penetration depth the approximation T = D is acceptable. The transport term γ in equation 13.38

[6] The penetration depths are different for the components C ,N ,P ,St.

results in

$$\gamma = \frac{\tau}{D} \cdot \frac{\left(1 - e^{\frac{-\delta}{D}}\right)}{\left(1 - e^{\frac{-d_{tot}}{D}}\right)} \ . \tag{13.41}$$

The approximation of T by D overestimates the transport for large T. The denominator in equation 13.41 enhances the effect. Such, the simplification

$$\gamma = \frac{\tau}{D}\left(1 - e^{\frac{-\delta}{D}}\right) \tag{13.42}$$

is assumed.
The parameter τ depends on physical quantities and on the macrobenthic biomass. The value τ_0 in absence of macrobenthic organisms, in enhanced by μ at maximum. The enhancement depends on the weighted sum of macrobenthic biomass Y_{tur} with a Michaelis-Menten response:

$$\tau = \tau_0 \cdot \left(1 + \mu \cdot \frac{Y_{tur}}{Y_{tur} + Y_{tur}^h}\right) \tag{13.43}$$

with

$$Y_{tur} = \sum_i \beta_i Y_i \ , \tag{13.44}$$

where Y_i denotes the biomass of the i-th functional group and β_i the weighting factor of that group. If only bioturbation acts on the vertical distribution and no production terms are considered, the penetration of detritus increases very slowly (D $\sim \sqrt[3]{t}$ für D >> δ).

13.5 Benthic nutrients

The benthic nutrient model base on the so called "'Oldenburg nutrient model"' of ERSEM (Radford, Internal Report). It is a simplification of the "'NIOZ nutrient model"' of Ruardij & van Raaphorst (1995). For convenience and for saving computation time this alternative model is implemented basing on the same ideas and assumptions as the original but dealing with some more approximations and pre-calculations. An advantage of this implementation is that the model works

mass conserving for all treated variables so that it is recommendable for diagnostic work.The model describes the vertical distribution of ammonium, nitrate, phosphate and silicate as well as of carbon dioxide, nitrogen gas and oxygen within the sediment. The model embraces the description of the processes nitrification and denitrification, silicate regeneration as well as adsorption and desorption.

This section describes the basic principles of the nutrient model. A detailed description of the technical implementation is given in section 13.6.

The sediment is vertically subdivided into the oxic layer the denitrification layer and the anoxic layer(Fig. 13.1).

The oxic layer comprises normally only a few millimeter, the denitrification layer with nitrate as oxygen donator for bacterial respiration, a few centimeter. The total thickness of the sediment is bounded to d_{tot} = 30 cm. The horizons of the layers are state variables and dynamically modeled.

The other state variables within the benthic system are given as mass per square meter (mg C/m^2 resp. mmol/m^2). To allow an adequate description of nutrient profiles the masses are converted to pore water concentrations in (mg C/m^3 PW resp. mmol/m^3 PW).

The basic idea of the model is to describe pore water profiles which gradients at the sediment's surface determine the exchange fluxes between the pore water an the pelagic water.

In a first step an equilibrium profile is calculated successively layer by layer from sediment's top to bottom starting with the given surface concentration. Each upper profile determines the boundary and continuity conditions for the next lower profile. In a second step the difference between equilibrium mass (integral over the equilibrium profiles) and actual mass (from the prognostic quantities) determines the flux through the sedimet's surface. In contrast to the model of Ruardij & van Raaphorst (1995) the resulting profiles are not explicitly calculated but may be reconstructed for diagnostic purposes.

The assumed form of the generic equation for a dissolved nutrient $C(\vec{x}, t)$ is:

$$(p+1)\frac{\partial C}{\partial t} = \sigma\frac{\partial^2 C}{\partial z^2} + \frac{P}{D} \tag{13.45}$$

The formula describes the vertical diffusion process with the diffusion constant σ and the source and sink term P/D. The sources and sinks are assumed to be vertically uniform in the considered layer of thickness D. The dimensionless

parameter p (section 13.5.7) is the non-dimensional adsorption coefficient depending on the porosity Φ of the sediment Ruardij & van Raaphorst (1995). The sorption process is assumed as a fast process where the pore water concentration and the sorbent phase are in equilibrium.
The changes of the concentration C in time is determined as follows:

1. A steady-state analytical solution $C_{eq}(z)$ is calculated for every considered layer. Beginning at the sediment's surface the equation

$$0 = \sigma \frac{\partial^2 C}{\partial z^2} + \frac{P}{D} \tag{13.46}$$

is solved for the oxic layer. The general solution with constant P and σ and the layer thickness D is

$$C_{eq}(z) = -\frac{P}{2\sigma D} z^2 + az + b \tag{13.47}$$

The first considered boundary condition is the surface concentration $C_{eq}(0) = C_0$. Such $b = C_0$. The second condition to fix a is given by the fluxes at the boundaries. The flux j(z) is given by

$$j(z) = \sigma \frac{\partial C}{\partial z} = -\frac{P}{D} z + \sigma a \tag{13.48}$$

Under equilibrium conditions the flux j(D) at the lower boundary D must compensate the production P_L below D and the flux at the sediment's surface j(0) must compensate the total production in the sediment $P + P_L$:

$$j(0) = P + P_L \tag{13.49a}$$
$$j(D) = P_L \tag{13.49b}$$

With these conditions the unique solution has the form

$$C_{eq}(z) = -\frac{P}{2\sigma D} z^2 + \frac{P + P_L}{\sigma} z + C_0 \tag{13.50}$$

For the denitrification layer and the anoxic layer this process is repeated. It is assumed that the production of the anoxic layer and below is zero but a potential flux out of the deeper sediment may be considered. The

resulting profile for a dissolved nutrient consists of three parabola pieces:

$$C_{eq}^{ox}(z) \qquad 0 \leq z \leq D_1 \tag{13.51a}$$

$$C_{eq}^{nit}(z) \qquad D_1 < z \leq D_2 \tag{13.51b}$$

$$C_{eq}^{anox}(z) \qquad D_2 < z \leq d_{tot} \tag{13.51c}$$

where the thickness D and the productivities P and P_L must be substituted by the actual layer thickness resp. productivities in (13.50).

2. Basing on these profiles the total mass per square meter K_{eq} in the sediment under equilibrium conditions is

$$K_{eq} = (p+1)\Phi \left(\int_0^{D_1} C_{eq}^{ox}(z)dz + \int_{D_1}^{D_2} C_{eq}^{nit}(z)dz + \int_{D_2}^{D_{tot}} C_{eq}^{anox}(z)dz \right) \tag{13.52}$$

It is proportional to the porosity Φ. For keeping the formulas simple here the adsorption coefficient p is taken to be the same for each layer. In reality and in the model coefficients for phosphate differ from layer to layer depending on the oxygenation state (see 13.5.7)

3. The total diffusion flux is determined by the total production in the sediment P_{tot} and the excess mass due to non-equilibrium conditions $K_\Delta = K_{eq} - K$. This is released with an appropriate adaptation time τ

$$\frac{\partial K}{\partial t} = -\left(P_{tot} - \frac{1}{\tau} K_\Delta \right) \tag{13.53}$$

The adaptation times are assumed to be parameters in the cases oxygen and nitrate. For all other substances the additional flux is calculated as follows: It is assumed that the excess mass is distributed over the sediment according to Fig. 13.3. This additional mass induces additional concentrations $\widetilde{C}_{eq}$ in every layer which again have parabolic shapes:

$$K_\Delta = (p+1) \left(\int_0^{D_1} \widetilde{C}_{eq}^{ox}(z)dz + \int_{D_1}^{D_2} \widetilde{C}_{eq}^{nit}(z)dz + \int_{D_2}^{D_{tot}} \widetilde{C}_{eq}^{anox}(z)dz \right) \tag{13.54}$$

The gradient at the surface determines the additional flux:

$$j(0) = \sigma \frac{\partial \widetilde{C}_{eq}^{ox}}{\partial z}(0) \tag{13.55}$$

To determine the shape of the parabola the same algorithm as described above is used. The following assumptions and boundary conditions have to be fulfilled: The concentration of the additional mass is zero at the sediments surface. The flux at the sediment's bottom is zero for mass conservation. The parabola pieces fit continously at the layer boundaries.

$$\widetilde{C}_{eq}^{ox}(0) = 0 \tag{13.56a}$$

$$\widetilde{C}_{eq}^{nit}(D_1) = \widetilde{C}_{eq}^{ox}(D_1) \tag{13.56b}$$

$$\widetilde{C}_{eq}^{anox}(D_2) = \widetilde{C}_{eq}^{nit}(D_2) \tag{13.56c}$$

$$D \frac{\partial \widetilde{C}_{eq}^{anox}}{\partial z}(D_{tot}) = 0 \tag{13.56d}$$

Furthermore it is assumed that the productions resp. consumptions resulting into the excess masses in the particular layers are proportional to the layer thicknesses. This approach ensures a dynamic adaptation of the process velocity.

An illustration of the typical profiles and the assumed distribution of the excess mass is shown in 13.3.

13.5.1 The dynamics of the oxygen and nitrate horizons

To determine the equilibrium solution for oxygen (equation 13.47) the following boundary conditions must be fulfilled: The oxygen concentration at the sediment's surface must equal the pelagic concentration (equation 13.71) and the oxygen concentration and the gradient must be zero at D_1. In combination with the flux condition 13.49 the system is overestimated. The equilibrium penetration depth can be calculated if the the layer horizon D_1 is treated as independent variable. It is assumed that the equilibrium is not reached instantaneously but will be reached after the relaxation time. The following differential equation for the penetration depth results:

$$\frac{\partial D_1}{\partial t} = \frac{1}{\tau} \left(D_1^{eq} - D_1 \right) . \tag{13.57}$$

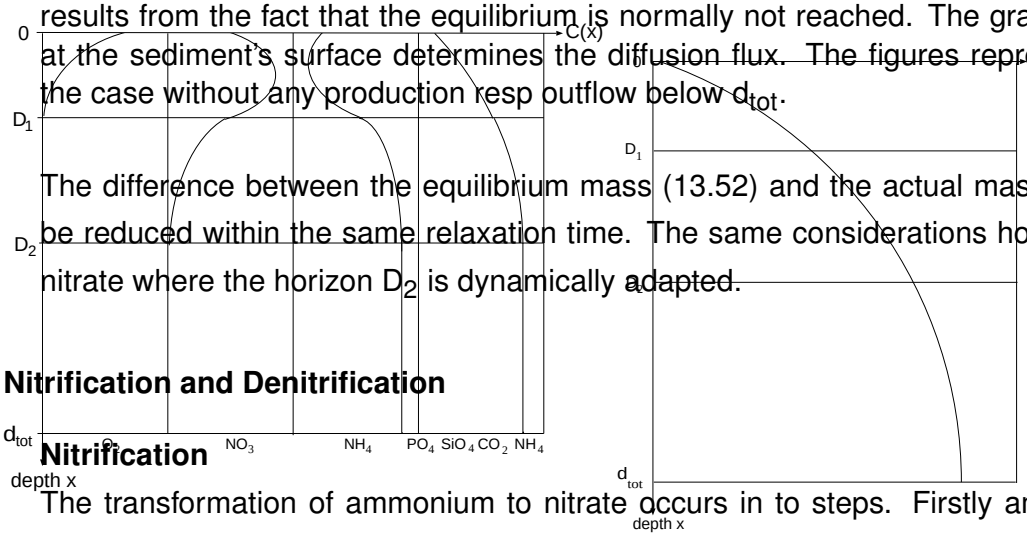


Figure 13.3: Typical profiles of nutrients and gases in the sediment (left). Assumed distribution of the excess mass in the sediment (right). The excess mass results from the fact that the equilibrium is normally not reached. The gradient at the sediment's surface determines the diffusion flux. The figures represent the case without any production resp outflow below d_{tot}.

The difference between the equilibrium mass (13.52) and the actual mass will be reduced within the same relaxation time. The same considerations hold for nitrate where the horizon D_2 is dynamically adapted.

13.5.2 Nitrification and Denitrification

Nitrification

The transformation of ammonium to nitrate occurs in to steps. Firstly ammonium is transformed to nitrite under oxygen consumption (f.e. by *Nitrosomonas*) and secondly nitrite is oxidized to nitrate (f.e. by *Nitrobacter*). The complete processes is called denitrification. The simplified chemical reaction equation is given by:

$$(NH_4^+ + OH^-) + 2\,O_2 \longrightarrow (NO_3^- + H^+) + 2\,H_2O\,. \tag{13.58}$$

the nitrification is modeled as first order process. The nitrification rate r_{nit} depends on the temperature of the pore water and of the availability of ammonium and oxygen.It is assumed that the nitrification is determined by the mean ammonium concentration within the oxic layer and the reaction only takes place in this layer. The oxygen dependency is expressed by the relative thickness of the oxidized layer D_1 related to the total thickness of the Sediments d_{tot}. Additionally

the nitrification is inhibited at too high nitrate concentration.

$$r_{nit} = r_{nit_0} \cdot e_T \cdot e_n \cdot \frac{D_1}{d_{tot}} \ . \tag{13.59}$$

where r_{nit_0} is the base nitrification rate at 10 °C, e_T the temperature regulation factor (equation 6.2) and e_n the nitrate limitation facto:

$$e_n = \frac{C^h_{NO_3}}{C^h_{NO_3} + \overline{C_{NO_3}}} \ , \tag{13.60}$$

where $\overline{C_{NO_3}}$ is the mean nitrate concentration:

$$\overline{C_{NO_3}} = \frac{K_{NO_3}}{D_1 + \frac{(D_2 - D_1)}{3}} \ , \tag{13.61}$$

resulting from the total nitrate mass K_{NO_3}, the thickness of the oxidized layer D_1 and the denitrification layer $D_2 - D_1$. It is assumed that nitrate penetrates into the denitrification layer for a short time at high concentrations. This leads to lower concentrations as expected from the equilibrium solution. If the mean nitrate concentration amounts $C^h_{NO_3}$ the nitrification rate is half of its possible maximum value.

Denitrification

The oxygen demand within the denitrification layer due to respiration of anaerobic bacteria (section 13.5.4) is partly supplied by nitrate[7].

Two possible pathways are possible for denitrification. The reaction is possible as ammonification

$$(NO_3^- + H^+) + 2\,H_2O \longrightarrow (NH_4^+ + OH^-) + 2\,O_2 \tag{13.62}$$

oder as dissimilatoric denitrification

$$4\,NO_3^- + 4\,H^+ \longrightarrow 2\,N_2 + 5\,O_2 + 2\,H_2O \tag{13.63}$$

The oxygen demand M^2_O by anaerobic bacteria (equation 13.68) expressed as

[7] In the model the reduction of sulphate, iron oxides and manganese oxides can be treated implicitly (parameter). In this case only a part of the needed oxygen is taken from nitrate. The mass conservation for oxygen will be violated.

nitrate reduction $\mathcal{N}$ (in mmol/m^2) is given by (in mmol/m^2)

$$M_O^2 = \frac{\mathcal{N}}{\phi(D_2 - D_1)} \cdot \left((1-\eta)\cdot\Omega_{N_2} + \eta\cdot\Omega_{NO_3}\right) \ . \tag{13.64}$$

where η is the part respired to nitrogen gas. The denominator with the porosity of the sediment ϕ and the actual thickness of the layer $D_2 - D_1$ provides the recalculation of the nitrate reduction from mmol/m^2 to mmol/m^3 pore water. The stoichiometric coefficients $\Omega_{N_2} = 2$ and $\Omega_{NO_3} = 5/4$ result from the chemical reaction equations.

13.5.3 Input to the benthic nutrient model

13.5.4 Production and consumption terms

The production and consumption terms changing the pore water concentration of ammonium and phosphate result from the fluxes described in section 13.1, 13.2 and 13.3,the nitrification and denitrification in section 13.5.2 and advection section 15.1 . The pore water concentrations within the oxic layer $[0, D_1]$ are determined by benthic diatoms, secondary producers and aerobic bacteria including nitrification and by advection. The pore water concentrations within the denitrification layer $[D_1, D_2]$ are determined by anaerobic bacteria including denitrification.

It is assumed that the nutrients are homogeneously distributed within the particulate layer. The production fluxes M_N^1 in the aerobic layer and M_N^2 in the anaerobic layer (both in mmol/(m^3.d) PW) are given by

$$M_N^1 = \frac{1}{\phi D_1} \cdot f^1_{prod_N} \qquad N = n, p \tag{13.65}$$

$$M_N^2 = \frac{1}{\phi(D_2 - D_1)} \cdot f^2_{prod_N} \qquad N = n, p\,, \tag{13.66}$$

where $f^i_{prod_N}$ $(i = 1, 2)$ are production and consumption fluxes of ammonium resp. phosphate within the oxic resp. denitrification layer in mmol/(m^2.d)[8] The porosity of the sediment is given by ϕ.

The oxygen dynamic is derived from the carbon dynamic of benthic organisms. It is determined by the assimilation of benthic algae and the respiration of ben-

[8] Names of $f^1_{prod_N}$ resp.$f^2_{prod_N}$ in the model code:
function `benthos`: `fD1` resp. `jD1` , function `Benthic_Nutrients`: `fD2` resp. `jD2`

thic algae, secondary producer an aerobic bacteria. For a total carbon consumption of $f^1_{prod_c}$ (production, if negative) the oxygen consumption within the oxic layer is given by

$$M^1_O = \frac{\Omega_c}{\phi\, D_1} \cdot f^1_{prod_c} \tag{13.67}$$

in mmol O_2/m^3 PW. The porosity of the sediment is given by ϕ, D_1 denotes the thickness of the oxic layer and $\Omega_c = 1/12$, is the reciprocal of the molar mass of carbon.
The "'oxygen consumption"' in the denitrification layer in form of nitrate is quite similar

$$M^2_O = \frac{1}{\phi(D_2 - D_1)} \cdot f_{prod^2_c}\ , \tag{13.68}$$

where $f^2_{prod_c}$ is the consumption of nitrate by anaerobic bacteria (section 13.5.2).

13.5.5 Diffusion

The diffusion coefficient D in equation 13.45 is assumed to be equal for all dissolved substances. In the model the diffusion coefficient is combined of a constant part D_0 describing the molecular diffusion, the temperature regulation factor e_T(equation 6.2) and an enhancement factor e_{irr}. This enhancement describes the bioirrigation and depends similar to the bioturbation (equation 13.43) on the weighted macrobenthic biomass Y_{irr}:

$$D = D_0 \cdot e_T \cdot e_{irr} \tag{13.69}$$

$$e_{irr} = \mu_{min} + \mu_{irr} \cdot \frac{Y_{irr}}{Y_{irr} + Y^h_{irr}} \qquad \text{with} \qquad Y_{irr} = \sum_i \beta_i Y_i\ , \tag{13.70}$$

where μ_{min} is the minimum enhancement and μ_{irr} the maximum additional enhancement The biomass of the i-th macrobenthic functional group Y_i i and its weighting factor β_i are summed up to the macrobenthic biomass.

13.5.6 Surface concentration

The concentration of the pelagic substances near the sediment's surface in equation 13.47 differ from the mean pelagic concentrations due to a gradient

between pore water and pelagic water. Therefore a surface concentration C_0 is assumed depending on the mean pelagic concentration C, the actual total production $\mathcal{P}$ and a mixing coefficient ζ:

$$C_0 = \begin{cases} C + \zeta \mathcal{P} & \mathcal{P} \geq 0 \\ \frac{C^2}{C - \zeta \mathcal{P}} & \mathcal{P} < 0 \end{cases} . \tag{13.71}$$

13.5.7 Porosity and adsorption

The sediment of the back barrier area are subdivided into three classes depending on their mud content:

- muddy: consist to more than 50 % of particles with a grain size smaller 63 —m
- mixed: consist of 5 % - 50 % of particles with a grain size smaller 63 —m
- sandy: consist to less than 5 % f particles with a grain size smaller 63 —m

The back barrier area of Spiekeroog has only small isolated regions of muddy flats according to this classification. The main part is mixed and sandy. The central part is dominated by sandy tideland (Flemming & Davis, 1994). In the model boxes two, three and four (Spiekeroog setup, Fig. 5.1) are assumed to be sandy boxes five to seven mixed the porosity of the sediment describing the part of the interstitial volume of the total volume of the sediment increases from sand (> 63 —m) to clay (< 63 —m). Therefore the values near the coast are slightly higher than in the central area. the parameterization(see `topography.def`) is similar to the values suggested by Ruardij & van Raaphorst (1995).
The capability of the sediment to bind substances resulting in lower pore water concentrations depends on the substance itself, the grain size and the oxidation state of the sediment. The formation of vertical gradients especially for phosphate depends on the actual adsorption coefficient. Slomp & van Raaphorst (1993) showed that a linear correlation between the porosity and the adsorption coefficient for phosphateK_a exist. The model parameter p for adsorption is given by Ruardij & van Raaphorst (1995):

$$p = K_a \frac{1 - \phi}{\phi} \rho , \tag{13.72}$$

where ϕ denotes the porosity of the sediment and ρ the density of the particles. Phosphate bounds to Fe(III) . Fe(III) is available within the oxic and denitrification layer at high concentrations – the adsorption coefficient within these layers are

accordingly high ($\approx$ 350 : 1 – 450 : 1). In the anoxic layer Fe(II) is released due to the reduction of Fe(III) to Fe(II). The proportion of adsorbed to dissolved phosphate is correspondingly low ($\approx$ 2 : 1). . The adsorbtion coefficient of the anoxic layer is assumed to be equal for all regions of the modeled area. The adsorption coefficient of ammonium is assumed also to be equal in all layers.

13.5.8 Silicate regeneration

The regeneration of silicate is described as first order process with a constant regeneration rate transforming the silicate component of detritus to inorganic silicate. The regeneration in the oxic and the denitrification layer is calculated separately. The available organic silicate in form of detritus is given according to section 13.4. The variation of the mean penetration depth of detritus due to the regeneration process is considered according to equation 13.38.

13.5.9 Determine the pore water concentrations

The concentration of a substance with in the pore water C (in mmol/m^3 PW) is recalculated from the mass of the substance within the sediment M (in mmol/m^2). The calculation for the particular layer is done in respect to the thickness of the layer D, the porosity ϕ and in case of phosphate and ammonium to the adsorbtion coefficient p (13.5.7) of the layer:

$$C = \frac{M}{\phi \; D \; (p + 1)} \; . \tag{13.73}$$

For nitrate and silicate p is set to zero.

13.5.10 Oxygen dynamics during the dry fall period

The exchange processes between the water column and the pore water are described in section 13.5.1. The exchange of oxygen during the dry fall period occurs also at the air-sediment-boundary-layer. For the part of the box falling dry ($1 - q_{wet}$) (see 9.5) the surface concentration (section 13.5.6) for oxygen is set to the saturation value of oxygen in water. Starting from this boundary condition the equilibrium profile is calculated. For the remaining part of the box q_{wet} the boundary concentrations are set to the voxel values.

13.5.11 DIC dynamics during the dry fall period

Under construction. Actually same as oxygen with an air saturation value for CO_2.

13.6 Implementation of benthic profile dynamic

This section describes the technical implementation of the benthic profiles. The basic principles are described in section 13.5.

The nutrient dynamics is implemented in the file `ben_nut.c`. The function `benthic_nutrients` is called in `benthos` in `ben.c`.

As described the penetration depth of oxygen and nitrate and the diffusion fluxes for nutrients are determined by the benthic profile routines. The calculation of the benthic profiles is done in two steps. First the new equilibrium oxygen and nitrate horizons are determined. Second, on the base of these new horizons the equilibrium profiles for the nutrients are calculated. The change of the real horizons are calculated from the equilibrium horizons by assuming a parametrized relaxation time.

It is assumed that there is an exchange between a benthic box and all voxels actually within the box. The calculation of the exchange area is described in section 4.9. . If parts of a box are fallen dry the resulting diffusion flux is weighted by the fractional area which is still wet. For oxygen and carbon dioxide an exchange is also assumed if the sediment is dry. For all other substances only if at least one voxel is within the box.

For simplicity all quantities affecting the dynamic of a substance (gases or nutrients) are combined in form of a structure called `profile` (defined in `structures.inc`). These structures become refilled by the call of `prof_parameter` in `benthic_nutrients`. A structure of type `profile` contains the following values

The general routines describing the benthic profiles are

`end_profile`	calculates a new penetration horizon
`equ_profile`	calculates the difference between the real mass and the equilibrium mass

Both functions needs the function `arc` which calculates a parabolic profile within a layer. `end_profile` also uses the function `endarc`.

p1	production in oxygenated layer
p2	production in denitrification layer
p3	production in anoxic layer
c0	pore water concentration at sediment surface
c1	pore water concentration at d_1
c2	pore water concentration at d_2
c3	pore water concentration at $d_t$ot
v1	adsorption coefficient in oxygenated layer
v2	adsorption coefficient in denitrification layer
v3	adsorption coefficient in anoxic layer
m1	equilibrium mass in in oxygenated layer
m2	equilibrium mass in in denitrification layer
m3	equilibrium mass in in anoxic layer
mdel	difference from equilibrium mass to real mass
d	new penetration horizon

13.6.1 The function `arc`

The function arc determines the parameters of a parabola between an arbitrary depth d0 and d1 with $d_0 < d_1$ from the boundary condition at d0 and the flux conditions at d0 and d1.The resulting concentration at d_1 and the mass (integral) in $[d_0, d_1]$ is calculated.

Input:

upper layer boundary: d_0

lower layer boundary:d_1

concentration at d0 : c_0

production in the layer d_0, d_1: p

production below d1: p_{low}

diffusion constant: σ

Output:

mass in $[d_0, d_1]$: m_1

d_1 is input variable but must be modified under some circumstances.

Similar to equation 13.50 the parabola is given by

$$c(z) = -\frac{p}{2\sigma d} z^2 + \frac{p + p_{low}}{\sigma} z + c_0 \tag{13.74}$$

with $d := d_1 - d_0$, the layer thickness. This parabola reaches from 0 to d. Such c(0) describes the concentration at d_0 and c(d) the concentration at d_1.
Depending on the production/consumption terms (sources and sinks) different

situations occur. The possible shapes are shown in Fig. 13.4.

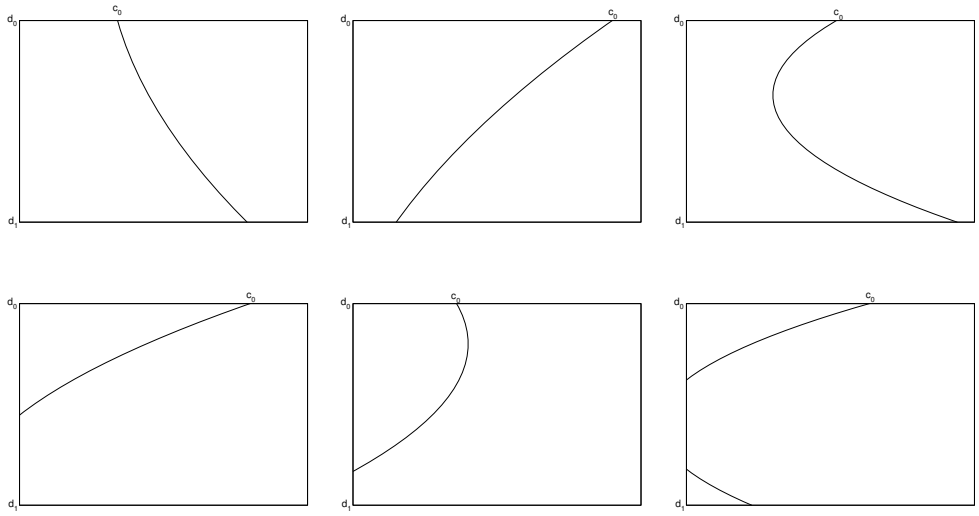


Figure 13.4: The possible shapes of the profiles according to equation 13.86. Depending on the production flows the equilibrium profiles exist in $[d_0, d_1]$ (upper row) or not (lower row).

The decision which case occurs is done by determing
the concentration at d_1

$$c(d) = c_0 + \frac{\frac{1}{2}p + p_{low}}{\sigma} d \ , \tag{13.75}$$

the position of the vertex

$$z_s = d\frac{p + p_{low}}{p} \ , \tag{13.76}$$

and the concentration at the vertex

$$c(z_s) = c_0 + \frac{d}{2\sigma}\frac{(p + p_{low})^2}{p} \ . \tag{13.77}$$

If the production in $[d_0, d_1]$ is zero, no parabola exist. To get the calculations right z_s is set to $d + 1$ and $c(z_s)$ is set to -1.

1st case: normal situation

See Fig. 13.4 upper row.

This case occurs if $c(d) \geq 0$ and the concentration at the vertex is either positive or the vertex is outside of $[d_0, d_1]$. Under this constraints everything is ok, the concentration at d_1 is given by

$$c(d) = c_0 + \frac{\frac{1}{2}p + p_{low}}{\sigma} d \,, \tag{13.78}$$

and the equilibrium mass is given by the intergal

$$\int_0^d c(z)\, dz = c_0 d + \frac{\frac{2}{3}p + p_{low}}{2\sigma} d^2; \tag{13.79}$$

2nd case: d_1 cannot be reached

See Fig. 13.4 lower row, left and mid.

In this case the parabola cannot reach d_1 under the given conditions. The solution is modified such that it fulfils the flux boundary conditions for a thinner layer $\tilde{d}$:

$$\tilde{c}(z) = -\frac{p}{2\sigma\tilde{d}} z^2 + \frac{p + p_{low}}{\sigma} z + c0 \,. \tag{13.80}$$

Such $\sigma\tilde{c}'(\tilde{d}) = p_{low}$. It is assumed that the concentration is zero at $\tilde{d}$, so that from $c(\tilde{d}) = 0$ follows

$$\tilde{d} = -\frac{\sigma c_0}{\frac{1}{2}p + p_{low}} \tag{13.81}$$

Unfortunately is this solution physically not senseful because an analytical continuation in the next layer cannot be calculated. Due to the negative slope at $\tilde{d}$ under the given conditions the concentration would become negative. Therefore the calculation of the equlibrium profile and the equilibrium mass is neglected for the calculation of the diffusion flux (see `equ_profile` in `ben_profiles.c`).

3rd case: profile with gap

See Fig. 13.4 lower row, right.

This case occurs if the concentration a the vertex is negative and the vertex is

positioned within the considered layer. The technical solution how to deal with this case is shown in Fig. 13.5.

First a preliminary penetration depth $\hat{d}$ is assumed so that the new concentration is given by

$$\hat{c}(z) = -\frac{p}{2\sigma\hat{d}}z^2 + \frac{p + p_{low}}{\sigma}z + c_0 \tag{13.82}$$

It is assumed that the concentration$\hat{c}$ at its vertex $\hat{z}_s$ becomes zero. The vertex is given by

$$\hat{z}_s = \frac{p + p_{low}}{p}\,\hat{d}\,, \tag{13.83}$$

and

$$\hat{c}(\hat{z}_s) = -\frac{p}{2\sigma\hat{d}}z^2 + \frac{p + p_{low}}{\sigma}z + c_0 =: 0 \tag{13.84}$$

leads to

$$\hat{d} = -\frac{2\sigma c_0 p}{(p + p_{low})^2} \tag{13.85}$$

Now the profile is split (Fig. 13.5) so that the part of the parabola below the vertex reaches the original penetration depth.

The equilibrium mass however can be calculated by integrating **??** in $[0, \hat{d}]$.

13.6.2 The function `endarc`

The function `endarc` calculates the shape of a parabola in contrast to the function arc (13.6.1) no penetration depth is given but it is calculated from the surface concentration c_0 at d_0 and the production/consumption flux p within the layer. In the model the flux below d_1 is always zero ($p_{low} = 0$). Such the gradient at d1 has to be zero. The resulting quantities are the penetration depth d_1 and the mass (integral).

Similar to equation 13.50 and 13.86 the parabola is given by

$$c(z) = -\frac{p}{2\sigma d}z^2 + \frac{p + p_{low}}{\sigma}z + c_0 \tag{13.86}$$

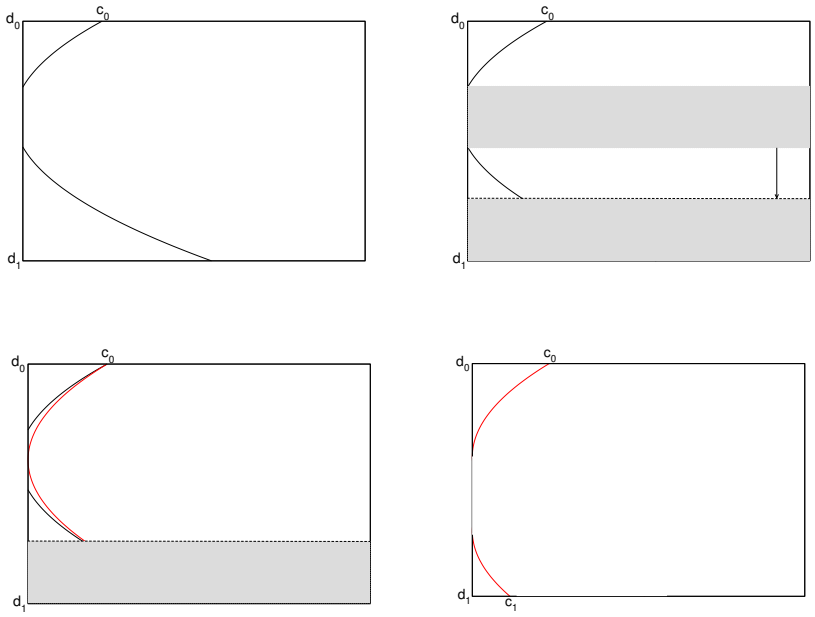


Figure 13.5: Technical solution if the profile has a gap (top left). The length of the gap is determined and the total penetration depth is temporarily deminished by this amount. (top right). A new parabola is determined with concentration zero at its vertex (bottom left). At last the parabola is split at its vertex and shifted towards the original penetration depth (bottom right). The concentration $c(d_1)$ is now determined.

With $p_{low} = 0$ and $c(d) = 0$ follows

$$d = -\frac{2\sigma c_0}{p} \tag{13.87}$$

Such the new penetration depth is given by $d_1 = d_0 + d$ and the concentration at d_1 is zero. The mass is given by

$$\int_0^d c(z)\, dz = \frac{1}{3} c_0 d \tag{13.88}$$

If the flux p becomes zero the equilibrium profile is a staight line and the concentration is c_0. If the calculated new penetration depth exceeds the total depth d_{tot}, the depth d_1 is set to d_{tot}. In this case the concentration at d_tot is greater zero. Both cases are calculated by a function call of `arc`.

13.6.3 The functions `end_profileO2` and `end_profileNO3`

The functions `end_profileO2` and `end_profileNO3` determine the penetration depth of oxygen resp. nitrate. Oxygen in form of O_2 only occurs within

the oxygenated layer (see (Fig. 13.1)).

For the calculation of the oxygen equilibrium profile it is assumed that the oxygen concentration at d_1 is zero and the flux below d_1 is zero. The equilibrium profile is calculated by one function call of `endarc`. The output is the equilibrium penetration depth of oxygen and the equilibrium mass within the oxygenated layer.

From these values the change of the oxygen horizon and the diffusion flux are calculated in `benthic_nutrients` in `ben_nut.c`.

Under some circumstances if the oxygen production exceeds the consumption a negative value for the penetration depth and mass is calculated resulting in an increase of the penetration depth and an additional outflow of oxygen.

Nitrate is generally produced in the oxygenated layer by nitrification and consumed in the denitrification layer. For the calculation of the nitrate equilibrium profile it is assumed that the nitrate concentration at d_2 is zero and the flux below d_2 is zero. The equilibrium profile is calculated by one function call of `arc` followe by the call of `endarc`. The output is the equilibrium penetration depth of nitrate and the equilibrium mass within the oxygenated layer and denitrification layer.

From these values the change of the nitrate horizon and the diffusion flux are calculated in `benthic_nutrients` in `ben_nut.c`. Under some circumstances if the nitrate production exceeds the consumption in the denitrification layer a negative value for the penetration depth and mass is calculated resulting in an increase of the penetration depth and an additional outflow of nitrate.

13.6.4 The function `equ_profile`

`equ_profile` calculates the equilibrium profiles in the three layers, oxic, denitrification and anoxic. For every nutrient a function call is done in `bennut.c`. After modifiying the surface concentration according to via modconc the profiles are calculted successivley from top to bottom. by three function calls of `arc`. Every call of arc gives back the equilibrium mass within the considered layer (without considering any adsorption coefficents). From these masses the difference to the effective mass given by the value of the state variables (`K1p`, `K3n`, `K4n`, `K5s`) is calculated (delta mass). For this the equilibrium masses are first multiplied by the adsorption coefficent of the particular layer. The delta mass determines the second part of the diffusion flux. The first part is given by the production/consumption terms which must in any cases leave the system (see

non_equ_flux 13.6.5).

13.6.5 The function `non_equ_flux`

non_equ_flux determines for every nutrient the final diffusion flux. Every flux consists of two parts. The first part is determined by the production/consumption within the sediment. The second results from the delta mass (see equ_profile 13.6.4). This additional mass is portioned to the three layers according to the actual thicknesses (Fig. 13.3). It is assumed that the delta mass also fulfills some kind of equilibrium property. The concentration at the sediments surface is assumed to be zero while the fluxes are assumed to be proportional to the layers. This is realized by initializing a structure profD of type profile in bennut.c. profD has to be interpreted as the proportions of a normed mass to the layers. The resulting norming factors are multiplied with the delta mass. By considering the particular adsorption coefficent it is ensured that especially for phosphate with varying adsortion coefficents the resulting diffusion flux is more or less realistic.

14 Carbonate system

14.1 Alkalinity-Under construction

Files: `carbonate.c calcite.c ben_calcite.c`
Parameter file: `carbonate.def`
Switch: `iswCS`

Total alkalinity (TA) within the model **EcoTiM** is defined as state variables (prognostic variables). It is distinguished between the alkalinity in the water (`ALK` in mmol m^{-3}) and within the sediment (`BALK` in mmol m^{-2} – "`B`" for benthic). Alkalinity is treated as a " normal" pelagic resp. benthic state variable. This means that `ALK` is mixed at the North Sea boundary and is subject to all pelagic transport processes. For better comparison, the pore water concentration is treated as the diagnostic variable `MALK` in mmol/m^3 PW. It is calculated in `ben_nut.c` (see also section 13.5.9)

$$\mathtt{MALK} = \frac{\mathtt{BALK}}{d_{tot} \cdot \phi}$$

where d_{tot} is the total sediment depth and ϕ the porosity of the sediment.
The alkalinity changes due to the following processes (Fig. 14.1):

1. exchange with the North Sea in box 1
2. diffusion between water bodies
3. dissolution of calcium carbonate
4. precipitation of calcium carbonate (neglected)
5. assimilation of inorganic nutrients by algae
6. release of inorganic nutrients by plankton
7. release of inorganic nutrients by bacteria (remineralization)
8. nitrification and dentrification
9. inflow of nutrients from floodgate
10. inflow of nutrients from atmosphere
11. in and outflow of inorganic nutrients into resp. out of the sediment
12. uptake and release of sulfur
13. sulfate reduction (missing)

The physical processes (1) and (2) are considered according to the transport affecting all pelagic states. Process (11) is treated by considering the fluxes of

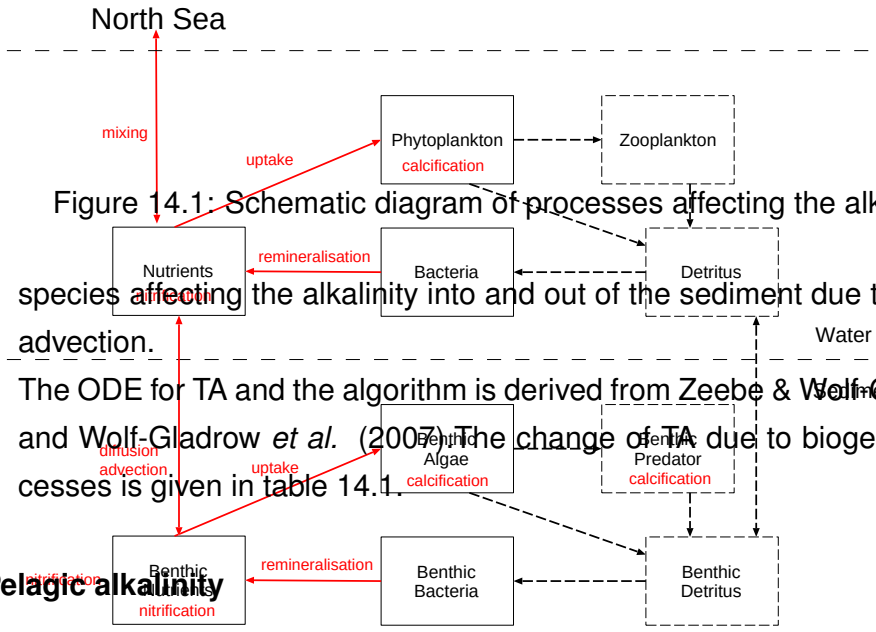


Figure 14.1: Schematic diagram of processes affecting the alkalinity (red).

species affecting the alkalinity into and out of the sediment due to diffusion and advection.

The ODE for TA and the algorithm is derived from Zeebe & Wolf-Gladrow (2001) and Wolf-Gladrow *et al.* (2007). The change of TA due to biogeochemical processes is given in table 14.1.

14.1.1 Pelagic alkalinity

The total change of alkalinity in water is given by the inflow due to calcite dynamic called `fO3ALK`, by nitrogen dynamic called `fNIALK` and by phosphorous dynamic (including sulfur) `fN1ALK` (`carbonate.def`).

$$\frac{\partial \mathtt{ALK}}{\partial t} = \mathtt{fO3ALK} + \mathtt{fNIALK} + \mathtt{fN1ALK}$$

where `fO3ALK` consideres the calcite production of flagellates `fO3P2_ca` and the dissolution of calcite `fR6O3_ca`.

Calcite

It is assumed that a part $pP2_{ca}$ of species belonging to the functional group "flagellates" are calcite builders and that the ratio of carbon in soft tissue to

process	Δ mole in water	Δ TA in mole
Precipipitation of calcium carbonate	-1	-2
Dissolution of calcium carbonate	+1	+2
Uptake of nitrate by phytoplankton	-1	+1
Uptake of ammonium by phytoplankton	-1	-1
Uptake of phosphate by phytoplankton	-1	+1
Uptake of sulfur by phytoplankton	-1	+2
Remineralization of organic nitrate	+1	-1
Remineralization of organic ammonium	+1	+1
Remineralization of organic phosphate	+1	-1
Remineralization of organic sulfur	+1	-2
Nitrification of ammonium (to nitrate)	-1	-2
Denitrification of nitrate (to N_2-gas resp. ammonium)	-1	+1
Reduction of sulphate (to sulphide)	-1	+2
Inflow of nitrate	+1	-1
Inflow of ammonium	+1	+1
Inflow of phosphate	+1	-1
Outflow of nitrate	+1	+1
Outflow of ammonium	+1	-1
Outflow of phosphate	+1	+1

Table 14.1: Processes affecting the alkalinity and their quantitative effect on to the alkalinity.

carbon in skeleton is given by $qP2_{ca}$. The calcite production of flagellates in mmol m^{-3} as part of the carbon assimilation `fO3P2c-fP2O3c` (in mg C m^{-3}) is given by

$$\texttt{fO3P2\_ca} = (\texttt{fO3P2c-fP2O3c}) \cdot \frac{pP2_{ca}}{qP2_{ca}} \cdot u_{g2mol}$$

according to Kühn (2007). $u_{g2mol} = \frac{1}{12}$ is the unit conversion factor from gram carbon to mole. The dissolution of calcite `R6_ca` (calcite content of detrius) depends on the carbonate ion CO_3^{2-} concentration in water. The saturation status Δ is calculated from the actual concentration and the saturation concentration CO_{3sat}^{2-}

$$\Delta = \max(0, CO_3^{2-} - CO_{3sat}^{2-})$$

The solubility of calcite (`aksp` depends on temperature, salinity and pressure and is calculated in `chemie.c` (CO_{3sat}^{2-}=`aksp/ca`, `ca` fixed).
The maximum dissolution rate r_{max} is reached if the system is undersaturated. If the system is oversaturated, dissolution declines with increasing degree of oversaturation according a Monod kinetic with the half saturation constant h_Δ. This results in the dissolution flux

$$\texttt{fR6O3\_ca} = r_{max} \cdot \left(1 - \frac{\Delta}{\Delta + h_\Delta}\right) \cdot R6_{ca}$$

Nitrogen

The change of pelagic alkalinity due to changes in nitrogen `fNIALK` is given by the uptake(5) and release(6) of nitrogen (ammonium resp. nitrate) by plankton, the release of nitrogen by bacteria (7), by nitrification and dentrification (8), the inflow of nitrogen from floodgate (9) and atmosphere (10) and the flow of nitrogen into resp. out of the sediment(11). The changes are considered according Tab. 14.1. The changes due to diffusion, advection (incl. North Sea boundary) are directly calculated by transporting alkalinity.

Phosphate

The change of pelagic alkalinity due to changes in phosphate `fN1ALK` is given by the uptake(5) and release(6) of phosphate by plankton, the release of phosphate by bacteria (7), the inflow of phosphate from floodgate (9) and atmosphere (10) and the flow of phosphate into resp. out of the sediment(11). The changes are considered according Tab. 14.1. The changes due to diffusion, advection (incl. North Sea boundary) are directly calculated by transporting alkalinity.

Sulfur

EcoTiM does not treat explicitly sulfur. Therefore the change in TA due to changes in sulfur concentration is derived from the phosphorous to sulfur ratio of marine organic matter (Wolf-Gladrow *et al.* (2007) sec. 5.6)). It is assumed that the S:P is approximatley 2.4:1. This means that f.e. uptake of 1 mole phosphate by algae leads to an uptake of 2.4 mole sulfur, resulting in an increase of alkalinity in water due to phosphate by 1 mole and due to sulfur by 4.8 mole.

14.1.2 Benthic alkalinity

Calcite

It is assumed that a part of species belonging to the functional group "benthic flagellates" and a part of the group "filter feeders" are calcite builders. The calcifcation is calculated similar to 14.1.1. According to the nomenklatur the resulting flux is named fG3BALK

Nitrogen

The change of benthic alkalinity due to changes in nitrogen `fNIALK` is given by the uptake(5) and release(6) of nitrogen (ammonium resp. nitrate) by plankton, the release of nitrogen by bacteria (7), by nitrification and dentrification (8), the inflow of nitrogen from floodgate (9) and atmosphere (10) and the flow of nitrogen into resp. out of the sediment(11). The changes are considered according Tab. 14.1. The changes due to diffusion, advection (incl. North Sea boundary) are directly calculated by transporting alkalinity.

Phosphate

The change of pelagic alkalinity due to changes in phosphate `fN1ALK` is given by the uptake(5) and release(6) of phosphate by plankton, the release of phosphate by bacteria (7), the inflow of phosphate from floodgate (9) and atmosphere (10) and the flow of phosphate into resp. out of the sediment(11). The changes are considered according Tab. 14.1. The changes due to diffusion, advection (incl. North Sea boundary) are directly calculated by transporting alkalinity.

Sulfur

EcoTiM does not treat sulfur explicitly. Therefore the change in TA due to changes in sulfur concentration is derived from the phosphorous to sulfur ratio of marine organic matter (Wolf-Gladrow *et al.* (2007) sec. 5.6)). It is assumed that the S:P is approximatley 2.4:1. This means that f.e. uptake of 1 mole phosphate by algae leads to an uptake of 2.4 mole sulfur, resulting in an increase of alkalinity in water due to phosphate by 1 mole and due to sulfur by 4.8 mole.

Sulphate reduction

Sulphate reduction is not explicitly considered in **EcoTiM**. However, it is assumed that all non-oxic processes in the sediment use nitrate as oxygen source. Nitrate reduction of 1 mole leads to an increase in alkalinity of 1 mole, while sulfate reduction of 1 mole leads to an increase of 2 mole. Thus the increase of alkalinity in non oxic sediments is actually underestimated.

14.2 Carbonate system – derived quantities

Under construction

Files: `carbonate.c`
Parameter file: `carbonate.def`
Switch: `iswCS`

The state variables total alkalinity (TA, pelagic: ALK, benthic BALK) and dissolved inorganic carbon (DIC, pelagic: O3c, benthic G3c) are used to calculate the quantities pH, CO_3^{2-}, HCO_3^- and $CO_{2(aq)}$.
The ode for DIC is given by the original model. The ode for TA and the algorithm is derived from Zeebe & Wolf-Gladrow (2001) and Wolf-Gladrow *et al.* (2007):

15 Model extensions

This chapter describes extensions of the model which exceed the original philosophy of ERSEM. They do not nearly not affect the cycling processes. Due to time step adaptations of the integration the model results may slightly different if extensions are calculated. To keep this problem small, most of the extensions are seperated in an additional integration mode (`multi=4`, `assign=4`). The occuring differences are due to processes which affect all states such as diffusion. If a state variable which is only used in the extensions leads to time step adaption due to such a process, the results will differ very slightly. The advantage to hold extensions in a separate integrations is not only to keep the results as similar as possible but also saves computing time. The calculation of water age, radioactive tracer and salt can be used to get an impression if the transport routine works correctly.

15.1 Pore water advection

File: `ben_advection.c`
Parameter: `ben_advection.def`

Under construction. Advection between the pelagic and benthic is implemented as pore water flows according to Rackebrandt (2012). The processes are only implemented for the Spiekeroog setup (see 5).

15.2 Manganese dynamics

File: `manganese.c`
Switches: `iswEXT`

The manganese cycling is up to now not coupled to the carbon and nutrient cycling processes. The model is preliminary. It is assumed that reduced manganese (Mn^{2+}) only exist in dissolved form while oxidized manganese (Mn^{4+}) only exist in particulate form.

The states are described in A.5.

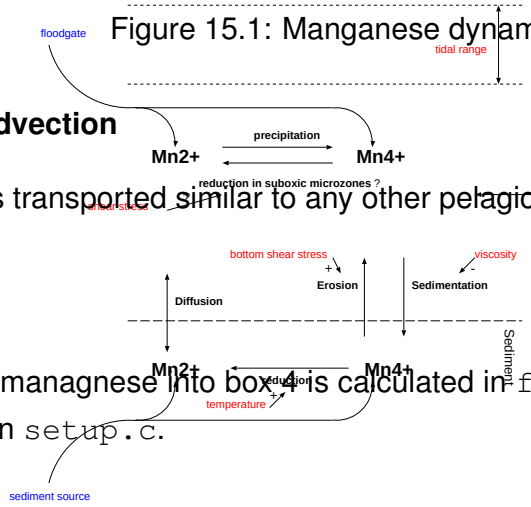


Figure 15.1: Manganese dynamic.

Pelagic diffusion and advection

Manganese is transported similar to any other pelagic quantity.

Input from flood gate

The intake of managnese into box 4 is calculated in `floodgate.c`. The loading file is set in `setup.c`.

Aggregation due to stickiness

It is assumed that aggregation occurs in form of a precipitation process. This process depends on the stickiness (see section 7.5.2) due to the amount of TEP:

$$Mn^{2+} \xrightarrow{r_a \cdot st} Mn^{4+} , \tag{15.1}$$

where r_a is the parametrized aggregation rate and st the stickiness.

Dissagregation due to shear stress

It is assumed that dissagregation occurs due to reduction in suboxic microzones. It depends on the actual shear stress (section 7.5.1).

$$Mn^{4+} \xrightarrow{r_a \cdot \tau} Mn^{2+} \tag{15.2}$$

where r_d is the parametrized disaggregation rate and τ the shear stress in water.

Sedimentation of particulate managnese

The sedimentation of particulate managnese depends on the actual viscosity:

$$Mn^{4+} \xrightarrow{r_{sed}} Mn^{4+}_{ben} , \tag{15.3}$$

with the sedimentation rate

$$r_{sed} = v_{sed} \frac{1}{(\frac{\eta}{\eta_0})^2 \cdot D} , \tag{15.4}$$

where D is the actual water depth , η the actual viscosity, η_0 the reference velocity and v_{sed} the parametrized sinking velocity if $\eta = \eta_0$.

Erosion of particulate manganese

Erosion depends on the local bottom shear stress τ_b of the voxel (section 7.5.1) Higher bottom shear stress leads to higher erosion

$$Mn^{4+}_{ben} \xrightarrow{r_{ero} \cdot \tau_b} Mn^{4+} , \tag{15.5}$$

where r_{ero} is the erosion rate at $\tau_b = 1$.

Outflow of dissoved manganese

The outflow of dissolved manganese out of the sediment is implemented as first order diffusion process.

Bacterial manganese reduction in the sediment

Up to now no influence from the bacterial biomass is assumed. The bacterial manganese reduction is assumed only to be temperature dependent.

$$Mn^{4+}_{ben} \xrightarrow{r_{bac} \cdot e_T} Mn^{2+}_{b} en , \tag{15.6}$$

where r_{bac} is the bacterial reduction rate at 10 °C and e_T is the temeperature factor according to 6.2.

Sediment sources of manganese

For both Mn^{4+} and Mn^{2+} sediment sources are assumed. Such an permanent additional efflux takes place which amount is assumed to be temperature dependent.

15.3 Molybdenum dynamics

File: –
Switch: `iswEXT`

Molybdenum (`U1mo`) is a representant for a conservative quantity within the pelagic system. It is transported (advection and diffusion) and mixed with a fix North Sea concentration (`ns_mo` in `northsea.def`). It is useful to estimate the time until the North Sea concentration is reached within the Wadden Sea.

15.4 Radioactive tracers

File: `radioactive.c`
File: `radioactive.def`
Switch: `iswEXT`

The short living radio isotopes ^{223}Ra (half-life 11.4 days) and ^{224}Ra (half-life 3.6 days) are implemented. Both are products of the uranium-thorium-chain. The activity within the Wadden Sea aera depends on the source within the sediment and on the mixing with the North Sea water which has a realtivley poor activity. The equilibrium activity of the North Sea is assumed to be 0.27 Bq/m^3 for ^{223}Ra and 4.85 Bq/m^3 for ^{224}Ra. Both are mean values from the measurements for coastal areas of **?** and Colbert (2004).
The efflux out of the sediment is given in percent of the mean estuarian volume. It is assumed that the efflux is equal over the whole area and amounts 15 % per day (**?**).

Additional to the natural activity a radioactive event can be simulated. The concerning parameters can be found in `radioactive.def`.

15.5 Water age

File: `calc_age.c`
Switch: `iswEXT`

The age of box 1 can be set in `tidal.def`. Setting the age of box one noticeable greater zero avoids most of the divisions of the time step adaptation routine.

15.6 Dynamic Chl:C ratio

Dir: `chl`
Files: `chl_init.c`, `chl_light.c` `chl_pprod.c`
Parameter file: `chl_pprod.def`
Switch: `iswCHL`

The pelagic primary producers may be simulated with a dynamic chl:C ratio according to Kotzur (2009). The Chl content of the pelagic primary producers are described by the additional state variables `P1chl`, `P2chl`, `P3chl` and `P4chl`.

15.7 Distribution of point tracers

File: `ducks.c`
Parameter file: `ducks.def`
Switch: `iswDUCKS`

It is possible to simulate the distribution of point tracers (f.e. rubber ducks). If `iswDUCKS=1` all tracers are initialized at an arbitrary releasing point (`duck_x`, `duck_y`) at an arbitrary time (`release_time`). Such it is possible to investigate f.e. the distribution of substances which are released from the floodgate in dependency of the time in the tidal cycle. The water age is set to zero at the

releasing time. The simulation should be started with `iswECOL=0` and without correction `iswGC=0` to avoid nonsense results.
The results can be visualized with `ECOVIZ` (see appendix D.4).

16 Inspection and diagnostics of model results

16.1 State variables

The pelagic state variables are indexed by voxel index. Normally, the corresponding `eu`-variables are inspected (see section 16.2). Benthic state variables are indexed by the box index.

16.2 eu-variables of state variables

The variable to be stored are defined in `cemos.par`. Normally, the mean values of all state variables within the geographical boxes are stored. These are the so called `eu`-variables (see section 4.3). These values are of main interest because they correspond to the benthic state values within the boxes. The calculation of these averages is done in `calc_euboxes.c`.

Additionally the mean values within the total back barrier areas are stored:

For the `SPIEKEROOG` setup the mean values of the boxes 3-7 are stored in `euXXX[50]` where XXX is the name of the state variable.

For the `LANGEOOG` setup the mean values of box 3-8 are stored in `euXXX[50]`

For the `BOTH` setup the mean values of box 3-8 are stored in `euXXX[49]`, the mean values of box 10-14 are stored in `euXXX[50]`.

The calculation of these averages is done in `calc_euboxes_mean.c`, the assignment of the indices is done in `topography.def`.

The concentrations in the neighborhood of the Spiekeroog pile station are stored in `euXXX[40]`. All voxels within the rectangular $[205, 49] \times [213, 53]$ are considered.

16.3 eu-variables of other characteristics

For specific quantities of interest, the voxel dependent values are also stored as mean values of the boxes. The calculation is done in `calc_euboxes_fluxes.c` and`calc_euboxes_fluxes_mean.c`.

16.4 Gauge amplitudes

The gauge values for special locations are stored in `pegel`:

`pegel[1]` grid coordinate (218, 50) near Spiekeroog gauge
`pegel[2]` grid coordinate (174, 18) near Bensersiel
`pegel[5]` 211,50 near pile station at 53 °45' N 7 40' E

The coordinates are set in `tide.def`. For calculation of the water level see section **??**.
The stored values do not really correspond to the gauge values from xtide or from the pile station, because the depth and the water movement at the desired grid point is only a rough estimation of the actual depth and the topographical incidents. The variables can give an estimation of the correctness of the flood- and ebb cycle and the spring- and neap-cycle. (Comparison with field data requires data in UTC!).

16.5 Budget calculation

Switch: `iswBUDGET`

Budget computation is useful to check mass conservation. The user must keep in mind that the model is an open model so that mass conservation cannot be expected even in the boxes with index greater 1. But, if all in and outgoing fluxes are inspected the total mass within the system (boxes 3-7 Spiekeroog) can be estimated. Therfore the following variables are calculated in `budget.c` for all voxel:

16.5.1 Pelagic masses in voxels

variable	unit	meaning
`MPTc[vox]`	kg	total mass of carbon in voxel `vox`
`MPTn[vox]`	kMol	total mass of nitrogen in voxel `vox`
`MPTp[vox]`	kMol	total mass of phosphate in voxel `vox`
`MPTs[vox]`	kMol	total mass of silicate in voxel `vox`

16.5.2 Pelagic masses in boxes

For all boxes (index ≥ 3) the total mass of all voxel within the box are added:

variable	unit	meaning
`euMPTc[box]`	kg	pelagic total mass of carbon in box `box`
`euMPTn[box]`	kMol	pelagic total mass of nitrogen in box `box`
`euMPTp[box]`	kMol	pelagic total mass of phosphate in box `box`
`euMPTs[box]`	kMol	pelagic total mass of silicate in box `box`

16.5.3 Benthic masses in boxes

For all boxes (index ≥ 3) the total mass within the sediment is calculated:

variable	unit	meaning
`MBTc[box]`	kg	benthic total mass of carbon in box `box`
`MBTn[box]`	kMol	benthic total mass of nitrogen in box `box`
`MBTp[box]`	kMol	benthic total mass of phosphate in box `box`
`MBTs[box]`	kMol	benthic total mass of silicate in box `box`

16.5.4 Total mass within the back barrier system

Actually this budget computation is only implemented for the `SPIEKEROOG`-setup.

16.6 Mass transport into and out of the tidal inlets

The total mass transport into and out of the tidal inlets for all state variable is calculated in `budget_inlets.c`. The considered boxes which determine the tidal inlet is set in `topography.def`. This is for the `SPIEKEROOG` setup boxes 3-7. Such the transport between these boxes and box 2 is budgeted. The resulting masses are indexed by 3. For the `LANGEOOG` setup boxes 3-8 are considered, the masses are indexed by 3 and for the `BOTH` setup boxes 3-8 for the Langeoog tidal inlet and 10-14 for the Spiekeroog tidal inlet. The indices for the transported masses are 3 and 10. The masses are stored for every state variable in the derived variables `auxI` for masses coming into the tidal inlet and `auxO` for masses leaving the tidal inlet. These variables cannot be stored by default because they are calculated in the first integration and are

zeroized afterwards by the following integrations (see 4.3). For inspection their values are summed up for carbon , nitrogen ,phosphate, and silicate and stored in eTIPTc,eTOPTc (name: transport in pelagic total carbon), eTIPTn,eTOPTn (name: transport in/out pelagic total nitrogen), eTIPTp,eTOPTp (name: transport in/out pelagic total phosphate), and eTIPTs,eTOPTs (name: transport in/out pelagic total silicate).

16.7 Additional output files

The activation of additional output file does not influence the model results! To get the output at the right time, the following values should be set in the file c:emos.par

```
multi=4
%integration_par4
maxdelt=0.01;
method=99;
```

16.7.1 Voxel coordinates

Switch:	iswCOOR
Parameter:	release_time in entchen.def
Files:	store_coor.c
Output files:	<resultname>.coo, <resultname>_mdepth.dat, <resultname>_box

If iswCOOR=1 the coordinates of all voxel are written to the file <resultname>.coo at every time step of the integration four (recommended: every 0.01 day) starting at day release_time (set in entchen.def). This options slows down the calculation. It is recommended to store the coordinates only for a few days. The file <resultname>_mdepth.dat is written within the first step and contains the mean depth of every grid point. The file <resultname>_boxdat is written within the first step and contains the number of the box of every grid point.

These informations can be visualized f.e. with the tool **EcoViz** (see section **??**).

16.7.2 Frequency of occurrence

Switch: `iswSTAT`
File:
Output file: `<resultname>_stat.dat`

The frequency of occurrence is stored in the file `<resultname>_.dat`. The calculation begins if the simulation time reaches the half of endtime. For every grid point the number of occurrences of voxel divided by the maximum possible number (time steps) is stored. The data structure is 324 $\times$ 88. This information can be visualized f.e. with the tool **EcoViz** (see section **??**).

16.7.3 Velocity field

Switch: `iswVELO`
File: `store_velo.c`
Output file: `velo.dat, velo_u.dat, velo_v.dat ,tide.dat`

The file `velo.dat` contains the absolute velocities, the files `velo_u.dat` and `velo_v.dat` the velocities in u- resp- v-direction.at and `tide.dat` the tidal range for every grid point at every time.

The output files are not named after the result files to save storage capacity. The files sizes amount approximately 300 KB per time step! This option slows down the calculation. It is recommended to store the information only for a few days.

This information can be visualized f.e. with the tool **EcoViz** (see section **??**).

16.8 Statistics

Switch: `iswSTD`
File: `stdev.c`

For further inspection of the model's behavior the following statistical quantities of all voxels within a box can be calculated. The calculation is done in `stdev.c`

Standard deviation

The Standard deviation of all voxel within a box is stored in `sdXXX[box]` where `XXX` is the desired state variable. This value is only reasonable if the number of voxels within a box is large enough. The user should check this with the variable `voxel` which contains the number of voxel for every box.

The Standard deviation of a box as variation of a day is stored in`dsdXXX[box]`. The considered values depend on the maxdelt of the integration.

Minimum and maximum

The minimum and maximum value of a state variable of all voxel within a box are stored in`minXXX[box]` and `maxXXX[box]`.

The minimum and maximum values of a state variable within one day are stored in `dminXXX[box]` and `dmaxXXX[box]`. The considered values depend on the maxdelt of the integration.

Mean deviation within a box

The mean deviation, defined as the mean deviation from the mean value is stored in `mdXXX[box]`.

Variation coefficient

The variation coefficient V is defined as the percentual fraction of the standard deviation s from the mean value $\overline{x}$,

$$V = \frac{s}{\overline{x}} \cdot 100 \ ,$$

it is stored in `vkXXX[box]`.

The variation coefficient with one day is stored in `dvkXXX[box]`.The considered values depend on the maxdelt of the integration.

17 Auxiliary programs

17.1 Creating North Sea Boundary file

The boundary conditions at the North Sea boundary are normally read from a file. This file must have the structure of a **CEMoS** `_pk.outc`-file (spectral values). Such the easiest thing is to run a **CEMoS** model for the boundary and convert it to spectral values by **MoViE**. The file structure and the proceeding to get spectral values is described in Kohlmeier & Hamberg (2004).

The file must contain a store command for every variable which shall act as boundary condition for exact one box. The name of the variables must be equal to the names in **EcoTiM** (without leading `eu`).

The actual default file is given by simulation results from the COCOA-model, the Continental Coastal Application of ERSEM (Lenhart *et al.* , 1997) for the year 1995. A special application of this is given as **EcoBound** which is adapted to the needs of **EcoTiM**. **EcoBound** provides the possibility to overrule the model simulations with measured data.

`iswBAH`=1
The North Sea values for the dissolved nutrients phosphate (`N1p`), ammonium (`N4n`) and silicate (`N5s`) are overruled by data from BAH (Biologische Anstalt Helgoland) from Helgoland Reede for the year 1995 (Wahl, 1997). The file containing these values is a **CEMoS** `_dpk.outc`-file (spectral values from data).

`iswBSH`=1
The North Sea values for the dissolved nutrients phosphate (`N1p`), nitrate/nitrite (`N3n`) and silicate (`N5s`) and salinity (`X1x`) are overruled by data from BSH (Bundesamt für Seeschifffahrt und Hydrografie) from the station 'Deutsche Bucht' in 2002 (www.bsh.de). The files containg these values are `.csv`-files (comma seperated values). The first line cotianing the header is omitted, in the following lines time stamp and value must be seperated by a comma. The time stamp must be a decimal value (2. Januar 3:00 am = 2.125). The calculation of the time stamps can be done by Tcl/TK (see `data/BSH/nut2csv`).

In **EcoBound** the files for overruling can be set in `ndz/overruling/setup.c`. Such `iswBAH`=1 reads one `_dpk.outc`-file . `iswBSH`=1 reads csv-files and

interpolates inbetween.

To get the conditions for building spectral values right, the following settings in `cemos.par` are necessary for the default file (overruling with BAH data):

```
%simulation_parameters
startim=0.0;
endtime=365.0;
storestart=1;

outdelt=1.0;
year=1995;
cycle=366.0;
model_dir ndz
multi=3;
recalc_globals=1;

%integration_par1
maxdelt=1.0;
method=99;

%integration_par2
mindelt=1.e-7;
maxdelt=1.0;
accuracy=0.1;
method=2;

%integration_par3
maxdelt=1.0;
method=99;

%change
iswBAH=1;
iswBSH=0;
iswRES=1;
```

The `.cin`-files in `ECOBOUND/runs` (see Hamberg & Kohlmeier, 2004) provide these settings.

The switch `iswRES=1` takes care that the simulation runs independently of the overuling data.
If `iswRES=0` the model calculation is done on the base of the read data. This means that the model is started with new initial conditions within every time step. Such a model run cannot be mass conserving and represents transient effects.

17.2 Files from xtide

By default, the tide length, the tide amplitude, sunrise and daylen are extracted from the program xtide (Flater, 2005). If these files shall be modified or extracted for future years, xtide must be installed and the harmonic data file must be available for the considered region.
The data files are extracted with some auxiliary Tcl-shell scripts in the directory `xtide`. This extraction must be done manually before the simulation and the created files must be placed manually into the correct directories.

17.2.1 Extracting data from xtide

To get the data for a specific year the command
`make_xtide_files.tcl <year>`
must be given. This will create the directory `xtidedata` and the files
`xtidedata/xtider_pegel<year>.dat`
`xtidedata/xtider_info<year>.dat`
for the location Spiekeroog.

Optional the command `make_xtide_files.tcl <year> <location>` may be given where location must be a location which is available by the xtide harmonic data file.

17.2.2 Gauge data

The file `xtidedata/xtide_pegel<year>.dat` contains half-hourly gauge values for the desired location and year. The values are given in UTC and start at the last day the year before. The script `make_pegel.tcl` calculates the day and time from the date informations as decimal value and the adjacent gauge value. The gauge values are normalized to values between -1 and 1. The data are written in `pegel_amp_<year>.csv` and `pegel_amp_<year>.symb` .

For the model simulation only the file `pegel_amp_<year>.csv` is needed. This file should be placed in `main/tide/data`.

17.2.3 Information about day and tide

The file `xtidedata/xtide_info<year>.dat` contains information about sunrise, sunset, high tide, low tide etc. The values are given in UTC and start at the last day the year before. The script `make_daylen.tcl` calculates the day length from this file and writes the time of the sunrise and the actual daylen into the file `daylen<year>.csv`. This file must be placed in `main/forcing/data`.

The script `make_tidelen.tcl` calculates the tide length from this info-file and writes the time stamp of high water and the actual tide length into the file `tidelen<year>.csv`. This file must be placed in `main/tide/data`. The information of the tide length is needed for the Fourier synthesis of the tracer movement. The time of high water is only useful for inspection.

The script `make_lowtide.tcl` writes the time stamps of low water into the file `lowtide<year>.csv`. This file must be placed in `main/tide/data`. The time of low water is only useful for inspection.

The script `make_gatetime.tcl` writes the time of low water at daytime into the file `gatetime<year>.csv`. This file must be placed in `main/tide/data`. This information is an estimation for the time of flooding. At some flood gates the water is released only once a day nearly at low tide. Due to the work schedule this usually occurs at daytime. The file should be replaced by real data if existing. In the currenct setup the time stamps are read from `lowtide<year>.csv` because the gate at Neuharlingen works automatically at every low tide. The recalclulation from daily to tidal loadings is done externally (see 17.3).

The script `make_springneap.tcl` writes the time stamps of spring time and neap time into file. This is only needed for inspection.

The script `make_all` executes all scripts for a given year (and optional for a location). The default location is Spiekeroog.

17.3 Flood gate data

The time stamps for releasing fresh water from the flood gate are read from `lowtide<year>.csv`. This has the advantage that for years without data the default loadings and concentration can be used and the water is released at the right time. To recalculate daily loadings to tidal loadings the script `daily2tide` can be used. The daily loadings and the script are placed in `floodgate/data/onceaday`. The script creates a directory `xxx` containing the tidal loadings. which must be copied to `floodgate/data/` manually.

A State variables, parameters and fluxes

The names of states, fluxes and parameters follow in general the ERSEM-naming-convention (Blackford & Radford, 1995). State variables are also called prognostic variables in some communities. These variables will be integrated. For every state variable an initial value is needed.

A.1 State variables of the pelagic model

name	unit	meaning
N1p	mmol/m^3	phosphate
N3n	mmol/m^3	nitrate
N4n	mmol/m^3	ammonium
N5s	mmol/m^3	silicate
O2o	mmol/m^3	oxygen
O3c	mmol/m^3	dissolved inorganic carbon
P1c	mg C/m^3	diatoms (C-part)
P1n,P1p,P1s	mmol/m^3	(N,P,Si-parts)
P2c	mg C/m^3	flagellats (C-part)
P2n,P2p	mmol/m^3	(N, P-parts)
P3c	mg C/m^3	pikophytoplankton (C-part)
P3n,P3p	mmol/m^3	(N, P-parts)
P4c	mg C/m^3	dinoflagellats (C-part)
P4n,P4p	mmol/m^3	(N, P-parts)
Z3c	mg C/m^3	carn. mesozooplankton
Z4c	mg C/m^3	omni. mesozooplankton
Z5c	mg C/m^3	mikrozooplankton (C-part)
Z5n,Z5p	mmol/m^3	(N, P-parts)
Z6c	mg C/m^3	hetero. nanoflagellats (C-part)
Z6n,Z6p	mmol/m^3	(N, P-parts)
B1c	mg C/m^3	bacteria (C-part)
B1n,B1p	mmol/m^3	(N, P-parts)
R1c	mg C/m^3	diss. org. material (C-part)
R1n,R1p	mmol/m^3	(N, P-parts)
R6c	mg C/m^3	Detritus (C-part)
R6n,R6p,R6s	mmol/m^3	(N, P, Si-parts)
PIi	W/m^2	optimal light
X1x	–	salinity (calculated by the model)

Table A.1: Pelagic state variables

Pelagic State variables of model extensions			
name	**unit**	**meaning**	**section**
Age	mmol/m^3	age of water	15.5
U2mn	mmol/m^3	manganese	15.5
U4mn	mmol/m^3	manganese	15.5
Ume	mmol/m^3	methane	15.5
U1mo	mmol/m^3	molydene	15.5
U223ra	mmol/m^3	radium	15.5
U224ra	mmol/m^3	radium	15.5
Ura	mmol/m^3	urane	15.5
P1chl	mmol/m^3	urane	15.5
P2chl	mmol/m^3	urane	15.5
P3chl	mmol/m^3	urane	15.5
P4chl	mmol/m^3	urane	15.5
ALK	mmol/m^3	alkalinity	15.5

Table A.2: Pelagic state variables of modelextensions

State variables beloging to model extensions are decribed in A.5.

A.2 Parameters of the pelagic model

Common parameters of pelagic phytoplankton			
name	**value**	**unit**	**meaning**
q10P	2.0	–	Q10 value for primary production
et1	20.0	$^{\circ}$C	temperature where inhibition starts
ets	21.0	$^{\circ}$C	temperature where eT reaches its maximum
qnRPc	0.0126	mmol/mg C	Redfield ratio N:C
qpRPc	0.7862e-3	mmol/mg C	Redfield ratio P:C
xnlP	0.5454	–	multiple of Redfield N:C as lowest ratio
xplP	0.5454	–	multiple of Redfield P:C as lowest ratio
xnhP	2.0	–	multiple of Redfield N:C as highest ratio
xphP	2.0	–	multiple of Redfield P:C as highest ratio
xqcPp	1.0	–	multiple of Redfield N:C below growth limitation occurs
xqcPn	1.0	–	multiple of Redfield P:C below growth limitation occurs
quPn3	0.0025	m^3/(d.mg C)	nitrate uptake rate
quPn4	0.0025	m^3/(d.mg C)	ammonium uptake rate
qurPp	0.0025	m^3/(d.mg C)	phosphate uptake rate
clPli	4.0	W/m^2	minimum value of optimal light
ruPli	0.25	1/d	adaptation rate for optimal light
pEIR	0.50	–	P-synthetically available radiance
ad_dep	5.0	m	adaptation depth for optimal light

Table A.3: Common parameters of pelagic phytoplankton, equal for all functional groups

Specific parameters of phytoplankton						
Identifier	P1	P2	P3	P4	Unit	Meaning
etlim	0	1	0	0	–	temperature limitation of growth above et1, 0: no limitation, 1: limitation
sum_P	3.70	2.60	3.80	2.10	1/d	maximum productivity at 10 $^{\circ}$
srs_P	0.15	0.10	0.10	0.10	1/d	respiration rate at 10 $^{\circ}$
pu_eaP	0.05	0.20	0.20	0.05	–	fraction of primary prod. excreted as POM
pu_raP	0.20	0.25	0.25	0.25	1/d	activity respiration
chPs	0.30	0.00	0.00	0.00	mmol/m^3	half-value of silicate limitation
qs_Pc	0.03	0.0	0.0	0.0	mmol/mg C	maximum ratio of silicate
esNIP	0.7	0.75	0.75	0.75	–	nutrient limitation value below which sedimentation occurs
resPm	5.0	0.0	0.0	5.0	m/d	sinking velocity under total nutrient limitation
seo_P	0.0	0.0	0.0	0.01	1/d	mortality rate at 100 100 mg C/m^3
sdo_P	0.05	0.05	0.05	0.05	1/d	lysis rate
uhPlc	25.0	50.0	50.0	50.0	mg C/mg Chl	conversion chlorophyll to carbon

Table A.4: Specific parameters of pelagic phytoplankton, individually set for the specific functional group

Parameters of Microzooplankton			
name	**value**	**unit**	**meaning**
q10Z5	2.0	–	Q10 value
sumZ5	1.2	1/d	maximum uptake rate at 10 $^\circ$
chuZ5c	200.0	mg C/m^3	half saturation value for uptake
suP3_Z5	0.0	–	preference value for P3
suP2_Z5	1.0	–	preference value for P2
suP1_Z5	0.75	–	preference value for P1
suZ5_Z5	1.0	–	preference value for Z5
suZ6_Z5	1.0	–	preference value for Z6
suB1_Z5	0.0	–	preference value for B1
puZ5	0.5	–	assimilation efficiency
pe_R1Z5	0.5	–	fraction of excretion to R1
srsZ5	0.02	1/d	respiration rate at 10 $^\circ$
chrZ5o	0.3	–	half saturation value for oxygen limitation expressed in oxgen saturation
pu_eaZ5	0.5	–	excreted fraction of uptake (activity excretion)
sdZ5o	0.25	1/d	maximum mortality rate due to oxygen limitation
sdZ5	0.05	1/d	mortality rate
qn_Z5c	0.0167	mmol/mg C	maximum quota N:C
qp_Z5c	0.001	mmol/mg C	maximum quota P:C
minfoodZ5	30.0	mg C/m^3	half value for food uptake
stempZ5n	0.5	1/d	releasing rate for N
stempZ5p	0.5	1/d	releasing rate for P

Table A.5: Parameters of Microzooplankton

Parameters of heterotrophic nanoflagellates			
Identifier	Value	Unit	Meaning
q10Z6	2.0	–	Q10 value
sumZ6	5.0	1/d	maximum uptake rate at 10 $^{\circ}$
chuZ5c	250.0	mg C/m^3	half saturation value for uptake
suP2_Z6	0.0		preference value for P2
suP3_Z6	1.0		preference value for P3
suZ6_Z6	0.2		preference value for Z6
suB1_Z6	1.0		preference value for B1
puZ6	0.4	–	assimilation efficiency
pe_R1Z6	0.5	–	fraction of excretion to R1
srsZ6	0.02	1/d	respiration rate at 10 $^{\circ}$
chrZ6o	0.3	–	half saturation value for oxygen limitation expressed in oxgen saturation
pu_eaZ6	0.5	–	excreted fraction of uptake (activity excretion)
sdZ6o	0.25	1/d	maximum mortality rate due to oxygen limitation
sdZ6	0.05	1/d	mortality rate
qn_Z6c	0.0167	mmol/mg C	maximum quota N:C
qp_Z6c	0.001	mmol/mg C	maximum quota P:C
minfoodZ6	100.0	mg C/m^3	half value for food uptake
stempZ6n	0.5	1/d	releasing rate for N
stempZ6p	0.5	1/d	releasing rate for P

Table A.6: Parameters of heterotrophic nanoflagellates

Common parameters of mesozooplankton			
name	**value**	**unit**	**meaning**
qnZlc	0.011	mmol/mg C	fixed N:C ratio in mesozooplankton
qpZlc	0.001	mmol/mg C	fixed P:C ratio in mesozooplankton

Table A.7: Allgemeine Parameter des Mezozooplanktons

Parameters of omnivorous zooplankton			
Identifier	Value	Unit	Meaning
q10Z4	2.1	–	Q10 value
sumZ4	0.7	1/d	maximum uptake rate
vumZ4	0.008	m^3/(d.mg C)	search volume
rvP1Z4	0.75	–	rel. search volume for P1
rvP2Z4	0.6	–	rel. search volume for P2
rvZ5Z4	0.25	–	rel. search volume for Z5
rvZ4Z4	0.5	–	rel. search volume for Z4
srsZ4	0.01, 0.01, 0.02, 0.0	1/d	rest respiration rates for C,N,P,Si
sraZ4	0.07, 0.07, 0.07, 0.0	–	part of gross uptake respired (C,N,P,Si)
PeI_Z4R6	0.2, 0.2, 0.2, 1.0	–	part of ingested material to faeces production for C,N,P,Si
PeI_Z4	0.05, 0.03, 0.03, 0.0	–	excreted part of of ingested material (C,N,P,Si)
rdZ4	0.04	1/d	mortality rate

Table A.8: Parameters of omnivorous Zooplankton

Parameters of carnivorous zooplankton			
Identifier	Value	Unit	Meaning
q10Z3	2.1	–	Q10 value
sumZ3	0.7	1/d	maximum uptake rate
vumZ3	0.02	m^3/d/mg C	search volume
rvZ3Z3	1.0	–	rel. search volume for Z3
rvZ5Z3	0. 5	–	rel. search volume for Z5
rvZ4Z3	0.5	–	rel. search volume for Z4
srsZ3	0.03, 0.06 ,0.02, 0.0	1/d	rest respiration rates for C,N,P,Si
sraZ3	0.20, 0.34, 0.32, 0.0	–	part of gross uptake respired (C,N,P,Si)
PeI_Z3R6	0.20, 0.03, 0.03, 1.0	–	part of ingested material to faeces production for C,N,P,Si
PeI_Z3	0.05, 0.03, 0.03, 0.0	–	excreted part of of ingested material (C,N,P,Si)
rdZ3	0.02	1/d	mortality rate

Table A.9: Parameters of carnivorous zooplankton

Pelagic bacteria and regeneration parameters			
name	**value**	**unit**	**meaning**
sumB1	8.38	1/day	maximum uptake rate at 10 $^{\circ}$
chdB1o	0.3125	–	half saturation value for oxygen limitation expressed in oxygen saturation
puR6_B1	0.01	–	fraction of R6-pool available for B1
puB1	0.6	–	bacterial efficiency
puB1o	0.2	–	bacterial efficiency at low oxygen concentrations
q10B1	2.95	–	Q10 value for metabolic processes
srsB1	0.01	1/d	rest respiration rate at 10 $^{\circ}$
sdB1	0.0	1/d	mortality rate
qn_B1c	2.084e-2	mmol/mg C	maximium N:C quota
qp_B1c	2.083e-3	mmol/mg C	maximum P:C quota
chB1p	0.5	mmol/m^3	Michaelis constant for P uptake
chB1n	1.0	mmol/m^3	Michaelis constant for N uptake
sN4N3	0.1	1/d	nitrification rate at 10 $^{\circ}$
rR6N5s	0.00	1/d	silicate regeneration rate
pR6cR6n	6.625	mmolC/mmol N	C:N reference ratio for R6 uptake
uB1c_O2	0.1115625	mmol/mg C	conversion of C (produced) into oxygen
urB1_O2	0.1040625	mmol/mg C	conversion of C (respired) into oxygen
REACON	0.96	1/m	relative rate of reaeration
ct0	0.023	1/ $^{\circ}$	temperature dependence of reaeration

Table A.10: Parameter of pelagic bacteria

	Food source								
Consumer	Diatoms	Flagellats	Pikophytoplankton	Dinoflagellats	Carn. Mesozooplankton	Omn. Mesozooplankton	Mikrozooplankton	Hetero. Nanoflagellats	Bacteria
Carn. Mesozooplankton	0	0	0	0	1.0	0.5	0.5	0	0
Omn. Mesozooplankton	0.75	0.6	0	0	0	0.5	0.25	0	0
Mikrozooplankton	0.75	1.0	0	0	0	0	1.0	1.0	0
Hetero. Nanoflagellats	0	0	1.0	0	0	0	0	0.2	1.0

Table A.11: Preference factors of pelagic consumers for the specific food components.

A.3 State variables of the benthic model

name	unit	meaning
K1p	mmol/m^2	phosphate
K3n	mmol/m^2	nitrate
K4n	mmol/m^2	ammonium
K5s	mmol/m^2	silicate
G2o	mmol/m^2	oxygen
G3c	mmol/m^2	dissolved inorganic carbon
A1c	mg C/m^2	diatoms (C-part)
A1n,A1p,A1s	mmol/m^2	(N, P, Si-parts)
A2c	mg C/m^2	non-diatoms (C-part)
A2n,A2p	mmol/m^2	(N, P-parts)
Y1c	mg C/m^2	epibenthic predators
Y2c	mg C/m^2	deposit feeders
Y3c	mg C/m^2	suspension feeders
Y4c	mg C/m^2	meiobenthos
Y5c	mg C/m^2	predators
H1c	mg C/m^2	aerobic bakteria
H2c	mg C/m^2	anerobic bacteria
Q1c	mg C/m^2	diss. org. material (C-part)
Q1n,Q1p	mmol/m^2	(N, P-parts)
Q6c	mg C/m^2	part. detritus (C-part)
Q6p,Q6n,Q6s	mmol/m^2	(N, P, Si-parts)
Q7c	mg C/m^2	refractory detritus (C-part)
Q7nQ7p	mmol/m^2	(N, P-parts)
Ali	W/m^2	optimal light
D1m	m	penetration depth of oxygen
D2m	m	penetration depth of nitrate
D3m,D4m,D5m	m	mean penetration depth of refractory detritus (Q7) for C, N, P
D6m,D7m,D8m,D9m	m	mean penetration depth of detritus (Q6) for C, N, P, Si

Table A.12: Benthic state variables

State variables beloging to model extensions are decribed in A.5.

A.4 Parameters of the benthic model

Common parameters of benthic phytoplankton			
name	**value**	**unit**	**meaning**
q10A	2.0	–	Q10 value for primary production
et1_A	20.0	°C	temperature where inhibition starts
ets_A	21.0	°C	temperature where temperature factor reaches its maximum
qnRPc	0.0126	mmol/mg C	Redfield ratio N:C
qpRPc	0.7862e-3	mmol/mg C	Redfield ratio P:C
xnlP	0.5454	–	multiple of Redfield N:C as lowest ratio
xplP	0.5454	–	multiple of Redfield P:C as lowest ratio
xnhP	2.0	–	multiple of Redfield N:C as highest ratio
xphP	2.0	–	multiple of Redfield P:C as highest ratio
xqcPp	1.0	–	multiple of Redfield N:C below growth limitation occurs
xqcPn	1.0	–	multiple of Redfield P:C below growth limitation occurs
quAn3	0.025	m^2/(d.mg C)	nitrate uptake rate
quAn4	0.025	m^2/(d.mg C)	ammonium uptake rate
qurAp	0.025	m^2/(d.mg C)	phosphate uptake rate
clAli	2.0	W/m^2	minimum value of optimal light
ruAli	0.25	1/d	adaptation rate for optimal light
pEIR_A	0.50	–	P-synthetically available radiance
ad_dep_A	2.0	m	min. water depth for adaptation
ad_dep_S	1.0	mm	adaptation depth in the sediment
xeps_S	6.0	1/mm	extinction coefficient in the sediment

Table A.13: Common parameters of benthic phytoplankton, equal for all functional groups.

Specific parameters of benthic phytoplankton				
Identifier	A1	A2	Unit	Meaning
etlim_A	1	0	–	temperature limitation of growth above et1, 0: no limitation, 1: limitation
sum_A	2.50	2.00	1/d	max. productivity at 10 $^{\circ}$
srs_A	0.10	0.10	1/d	respiration rate at 10 $^{\circ}$
pu_eaA	0.05	0.05	–	fraction of primary prod. excreted as POM
pu_raA	0.10	0.10	1/d	activity respiration
chPs	0.03	0.00	mmol/m^2	half-value of silicate limitation
qs_Ac	0.03	0.0	mmol/mg C	maximum ratio of silicate
kap_A	1500	1500	mg C/m^2	capacity at which mortality reaches seo_A
seo_A	0.0005	0.0005	1/d	mortality rate at kap_A
sdo_A	0.05	0.05	1/d	lysis rate
uhAIc	25.0	50.0	mg C/mg Chl	conversion chlorophyll to carbon

Table A.14: Specific parameters of benthic phytoplankton, individually set for the specific functional group

Common parameters of zoobenthic groups			
name	**value**	**unit**	**meaning**
qnYIc	0.0119	mmol/mg C	fixed N:C ratio
qpYIc	0.000792	mmol/mg C	fixed P:C ratio

Table A.15: Common parameters of zoobenthic groups

Epibenthic predators			
Identifier	Value	Unit	Meaning
q10Y1	2.0	–	Q10 value
suY1	0.03	1/d	uptake rate
huY1	5000.0	mg C/m^2	half saturation uptake
luY1	200.0	mg C/m^2	lower threshold uptake
pueY1	0.30	–	excreted fraction of uptake
purY1	0.25	–	respired fraction of assimililation
pudilY1	0.8	–	nutrient dilution factor in fecal pellets
srY1	0.0027	1/d	rest respiration rate
sdY1	0.002	1/d	mortality rate
rlO2Y1	0.0	mmol/m^2	O2 at which survival becomes impossible
hO2Y1	0.0	mmol/m^2	O2 at which half limitation occurs

Table A.16: Parameters of epibenthic predators

Deposit Feeders			
Name	Value	Unit	Meaning
q10Y2	2.0	–	Q10 value
suY2	0.11	1/d	uptake rate
huY2	3000.0	mg C/m^2	half saturation uptake
luY2	250.0	mg C/m^2	lower threshold uptake
pueY2	0.35	–	excreted fraction of uptake
pueQ6Y2	0.8	–	excreted fraction of detritus uptake
purY2	0.35	–	respired fraction of assimililation
pudilY2	0.8	–	nutrient dilution factor in fecal pellets
srY2	0.0027	1/d	rest respiration rate
sdY2	0.001	1/d	mortality rate
dQ6Y2	0.3	m	layer available
rlO2Y2	0.0	mmol/m^2	O2 at which survival becomes impossible
hO2Y2	0.0	mmol/m^2	O2 at which half limitation occurs

Table A.17: Parameters of deposit feeders

Suspension Feeders			
Name	Value	Unit	Meaning
q10Y3	2.0	–	Q10 value
dwatY3	1.0	m	food layer in the water
suY3	0.09	1/d	uptake rate
huY3	300.0	mg C/m^2	Half saturation uptake
luY3	10.0	mg C/m^2	Lower threshold uptake
pueY3	0.35	–	excreted fraction of uptake
pueR6Y3	0.85	–	excreted fraction of detritus uptake
purY3	0.4	–	respired fraction of assimililation
pudilY3	0.8	–	nutrient dilution factor infecal pellets
srY3	0.0027	1/d	rest respiration rate
sdY3	0.001	1/d	mortality rate
dQ6Y3	0.0025	m	layer available
rlO2Y3	0.0	mmol/m^2	O2 at which survival becomes impossible
hO2Y3	0.0	mmol/m^2	O2 at which half limitation occurs
xclY3	2500.0	mmol/m^2	minimum density for shadingInterference to uptake due to shading
xcsY3	3000.0	mmol/m^2	half saturation value of effective biomass
xchY3	5000.0	mmol/m^2	half saturation value for shading

Table A.18: Parameters of suspension feeders

Meiobenthos			
Name	Value	Unit	Meaning
q10Y4	2.0	–	Q10 value
suY4	0.40	1/d	uptake rate
huY4	1000.0	mg C/m^2	half saturation uptake
luY4	50.0	mg C/m^2	lower threshold uptake
pueY4	0.25	–	excreted fraction of uptake
pueQ6Y4	0.4	–	excreted fraction of detritus uptake
purY4	0.45	–	respired fraction of assimilation
pudilY4	0.8	–	nutrient dilution factor in fecal pellets
srY4	0.01	1/d	rest respiration rate
sdY4	0.01	1/d	mortality rate
dQ6Y4	0.03	m	layer available.
rlO2Y4	0.0	mmol/m^2	O2 at which survival becomes impossible
hO2Y4	0.0	mmol/m^2	O2 at which half limitation occurs

Table A.19: Parameters des Meiobenthos

Infaunal Predators			
Name	Value	Unit	Meaning
q10Y5	2.0	–	Q10 value
suY5	0.08	1/d	uptake rate
huY5	5000.0	mg C/m^2	half saturation uptake
luY5	200.0	mg C/m^2	lower threshold uptake
pueY5	0.30	–	excreted fraction of uptake
purY5	0.3	–	respired fraction of assimililation
pudilY5	0.8	–	nutrient dilution factor in fecal pellets
srY5	0.0027	1/d	rest respiration rate
sdY5	0.003	1/d	mortality rate
rlO2Y5	0.0	mmol/m^2	O2 at which survival becomes impossible
hO2Y5	0.0	mmol/m^2	O2 at which half limitation occurs

Table A.20: Parameters of infaunal predators

Common parameters of benthic bacteria			
name	**value**	**unit**	**meaning**
qnHlc	0.0167	mmol/mg C	fixed N:C ratio
qpHlc	0.00125	mmol/mg C	fixed P:C ratio

Table A.21: Allgemeine Parameter der benthischen Bakterien

Aerobic Bacteria			
Identifier	Value	Unit	Meaning
q10H1	2.0	–	Q10 value
suQ6fH1	0.0002	1/d	degradation rate for good detritus
suQ6sH1	0.00002	1/d	degradation rate for bad detritus
suQ7H1	0.000002	1/d	degradation rate for ugly detritus
suQ1H1	0.0005	1/d	DOC decomposition rate
purH1	0.3	–	fraction respired
srH1	0.02	1/d	rest respiration rate
ddH1	0.001	m	half mortality layer
sdH1	0.05	1/d	maximum mortality rate
chH1n	20.0	$mmol/m^3$	Michaelis constant for NH4 uptake
chH1p	5.0	$mmol/m^3$	Michaelis constant for PO4 uptake
puincH1	2.0	–	preference of nutrient content
pdH1Q1	0.1	–	fraction to Q1 of mortality
pue6H1Q1	0.1	–	fraction to Q1 of Q6 uptake
pue7H1Q1	0.1	–	fraction to Q1 of Q7 uptake

Table A.22: Parameters of benthic aerobic bacteria

Anaerobic Bacteria			
Identifier	Value	Unit	Meaning
q10H2	2.0	–	Q10 value
suQ6fH2	0.0002	1/d	degradation rate for good detritus
suQ6sH2	0.00002	1/d	degradation rate for bad detritus
suQ7H2	0.000002	1/d	degradation rate for ugly detritus
purH2	0.3	–	fraction respired
srH2	0.02	1/d	rest respiration rate
ddH2	0.01	m	half mortality layer
sdH2	0.05	1/d	maximum mortality rate
chH2n	20.0	$mmol/m^3$	Michaelis constant for NH4 uptake
chH2p	5.0	$mmol/m^3$	Michaelis constant for PO4 uptake
puincH2	2.0	–	preference of nutrient content

Table A.23: Parameters of benthic anaerobic bacteria

	Food sources														
Consumer	Ben. Diatoms	Ben. Non-Diatoms	Megabenthos	Deposit Feeders	Susp. Feeders	Meiobenthos	Predators	Aerobic Bacteria	Anaerobic Bacteria	Part. Detritus	Pel. Diatoms	Pel. Flagellats	Pel. Pikophytoplankton	Pel. Bacteria	Pel. part. Detritus
Megabenthos	0	0	5	0.7	1	0	0.5	0	0	0	0	0	0	0	0
Deposit feeders	1	0.25	0	0	0	1	0	1	1	0.1	0	0	0	0	0
Susp. feeders	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
Meiobenthos	1	1	0	0	0	1	1	1	1	0.3	0	0	0	0	0
Predators	0	0	0	1	0	0.5	3	0	0	0	0	0	0	0	0

Table A.24: Preference factors of benthic consumers for the specific food components

Bioturbation and Bioirrigation			
name	**value**	**unit**	**meaning**
Etur	0.000002	m^2/d	turbation value without organisms
htur	116.	mg C/(d.m^2)	half saturation value for turbation enhancement
pturY1	0.5	–	relative Y1 contribution to bioturbation
pturY2	1.0	–	relative Y2 contribution to bioturbation
pturY5	0.5	–	relative Y5 contribution to bioturbation
q10tur	2.0	–	temperature effect on turbation
dtur	0.02	m	turbation depth
mtur	10.0	–	maximum turbation enhancement
hirr	101.	mg C/(d.m^2)	half saturation value for irrigation
pirrY2	1.0	–	relative Y2 contribution to bioirrigation
pirrY4	0.2	–	relative Y4 contribution to bioirrigation
pirrY5	1.0	–	relative Y5 contribution to bioirrigation
irr_min	2.0	–	minimum diffusion enhancement
mirr	10.0	–	maximum irrigation enhancement

Table A.25: Parameters of bioturbation and bioirrigation

Parameters of detritus dynamic			
name	**value**	**unit**	**meaning**
pe_R1P1	0.20	–	part of sinking diatoms to dissolved organic matter
pe_R1P4	0.50	–	part of sinking dinoflagellates to dissolved organic matter
pe_R7P1	0.005	–	part of sinking diatoms to refractory organic matter
pe_R7P4	0.05	–	part of sinking dinoflagellates to refractory organic matter
pe_R7R6	0.10	–	part of sinking particulate organic matter to refractory organic matter
xR1p	1.2	–	relative phosphate content in the matter transformed to dissolved organic matter
xR1n	1.0	–	relative nitrogen content in the matter transformed to dissolved organic matter
xR7p	0.6	–	relative phosphate content in the matter transformed to refractory organic matter
xR7n	1.0	–	relative nitrogen content in the matter transformed to refractory organic matter

Table A.26: Parameters of detritus dynamic

Parameter of benthic nutrient model			
name	**value**	**unit**	**meaning**
poro	0.5,0.6	–	volumetric porosity of boxes 2-4 resp. boxes 5-7
pM1	350,450	–	adsorbed:dissolved phosphate in oxic layer boxes 2-4 resp. boxes 5-7
M11ads	2.0	–	adsorbed:dissolved phosphate in anoxic layer (all boxes)
M4ads	3.0	–	adsorption between adsorbed and dissolved ammonium
relax_o	5.0	d	relaxation time for oxygen layer adaptation (water-water)
relax_dry	5.0	d	relaxation time for oxygen layer adaptation (water-air)
relax_m	5.0	d	relaxation time for nitrate layer
EDZ_1	0.00005	m^2/day	diffusion-constant surface
EDZ_2	0.00005	m^2/day	diffusion-constant D1-D2
EDZ_3	0.00005	m^2/day	diffusion-constant D2-D3
EDZ_mix	2.0	m/d	epibenthic mixing constant
q10nit	2.0	–	Q10 value for nitrification
sM4M3	5.0	1/d	nitrification rate
hM4M3	10.0	mmol/m^3	suppression of nitrification
hM3G4	1.0	mmol/m^3	nitrate limitation of denitrification
pammon	0.5	–	fraction of H_2-oxygen-consumtion taken from NO_3
pdenit	0.0	–	fraction of pammon denitrified in N_2
sQ6M5	0.0035	1/d	silicate regeneration rate
g2osat	275.0	mmol/m^3	saturation value for O_2 in pore water

Table A.27: Parameters of benthic nutrient model

A.5 Additional state variables

Names of additional states which do not belong to the original ERSEM.

identifier	unit	meaning	section
Age	days	water age in back barrier area	15.5
U223ra	mmol/m^3	pelagic radium Ra-223	15.4
U224ra	mmol/m^3	pelagic radium Ra-224	15.4
Ura	mmol/m^3	pelagic radioactive tracer	15.4
U2mn	mmol/m^3	Mn^{2+} , pelagic dissolved manganese (reduced)	15.2
U4mn	mmol/m^3	Mn^{4+} , pelagic particulate manganese (oxidized)	15.2
V2mn	mmol/m^2	Mn^{2+} , benthic dissolved manganese (reduced)	15.2
V4mn	mmol/m^2	Mn^{4+} , benthic particulate manganese (oxidized)	15.2
P1chl	mmol/m^3	Chl diatoms	15.6
P2chl	mmol/m^3	Chl flagellates	15.6
P3chl	mmol/m^3	Chl picophytoplankton	15.6
P4chl	mmol/m^3	Chl dinoflagellates	15.6
U1mo	—mol/m^3	pelagic molybdenum	-
ALK	mmol/m^3	pelagic total alkalinity	14
BALK	mmol/m^2	benthic total alkalinity	14

Table A.28: Additional state variables for model extensions

A.6 Fluxes

The names of fluxes follow in general the ERSEM-naming-convention (Blackford & Radford, 1995). Most of the fluxes are define as gloabls in model.def (possibly via include file) to allow storing and inspection. A flux from `XXx` to `YYx` is named `fXXYYx`, where the capitals denote the state variabel and the small x denotes the substance class. Example: the uptake flux of phosphate (`N1p`) by diatoms (`P1p`) in voxel 5 is named `fN1P1p[5]`. The units of these fluxes are in general given by by the unit of the first variable per day.

Some fluxes are also available for the boxe, these are denoted by the prefix eu.

A.6.1 Derived fluxes

Derived fluxes are defined in `model.def` as `%real_derived_from_states`. They are used to change all S-terms simultaneously in `trsptorates.c`. The abbreviation `XXX` has to be replaced by the state variable names. The units depend on the variable.

name	unit	meaning
`wDI+XXX`	mmol/m^3/d	pelagic, diffusion into voxel
`wDO+XXX`	mmol/m^3/d	pelagic, diffusion into voxel
`bPTI+XXX`	mgC/m^2/d, mmol/m^2/d	benthic, particle transport into box
`bPTO+XXX`	mgC/m^2/d, mmol/m^2/d	benthic particle transport out of box
`bDI+XXX`	mgC/m^3/d, mmol/m^3/d	benthic, benthic-pelagic diffusion into box
`bDO+XXX`	mgC/m^3/d, mmol/m^3/d	benthic, benthic-pelagic diffusion out of box
`bTI+XXX`	mgC/m^2/d, mmol/m^2/d	benthic, transport into box
	mgC/m^3/d, mmol/m^3/d	pelagic, benthic-pelagic advection
`bTO+XXX`	mgC/m^2/d, mmol/m^2/d	benthic, transport out of box
	mgC/m^3/d, mmol/m^3/d	pelagic, benthic-pelagic advection
`eubTI+XXX`	mgC/m^3/d, mmol/m^3/d	mean over box of pelagic bTI's
`eubTI+XXX`	mgC/m^3/d, mmol/m^3/d	mean over box of pelagic bTO's

B Tables of switches

Most of the switches can have the values 0 or 1, where 0 means that the concerning activity is disabled. If there are more possible values or if the values have other meanings it is described. Variable names related to the switches are given. The allocation of filenames to variable names is done in `setup.c`

B.1 Switches for additional output files and diagnostics

Switch	Default	Definition
`iswSTAT`	0	Writing statistic file `statistic.dat`
`iswCOOR`	0	Writing coordinate file `coor.dat`
`iswVELO`	0	Writing velocity files `velo.dat` and `tide.dat`
`iswBUDGET`	0	budget computation
`iswEXT`	1	calculation of extensions:radioactive tracers,water age, methane, manganese (section 15)
`iswCS`	1	calculation of carbonate system (section 14)
`iswSTD`	0	calculation of standard deviations

These diagnostic switches have in principal no influence on the ecological model results. Due to time step adaptation the results may differ slightly.

B.2 Switches for alternative model formulations

Switch	Default	Definition
iswXXX	1	0: state variableXXX set to zero at start 1: state variableXXX is active XXX has to be replaced by the desired variable name
iswTRACER	1	0 reading tracer positions according to trajec_result_file 1: reading tracer velocities from EOF data file according to tracer_data_file
iswNORTHSEA	2	0: fix values for box 1 1:calculating slave model in parallel 2-: reading result file for box 1 according to northsea_xxx_file in setup.c
iswNSXXX	0	0: no overruling of North Sea value for variable XXX 1:overruling of North Sea value for variable XXX by data from a file according to northsea_result_file in setup.c
iswGC	1	grid correction by shifting voxels enabled
iswECOL	1	calculating the ecology model
iswBEN	1	0: no benthos calculation 1: benthos 2: benthic shortcut
iswMIX	1	diffusion between all voxels
iswPTP	1	particulate inward transport of benthic detritus
iswGATE	1	input of freshwater by floodgates according to gate_load_file_nhs, gate_load_file_ben and gate_load_file_dor at time according to gate_time_file
iswAQUI	0	input of freshwater by aquifer according to aqui_load_file_nhs, aqui_load_file_ben and aqui_load_file_dor

Switch	Default	Definition
iswPRE	1	precipitation according to pre_file
iswEVA	1	loss of freshwater according to evaporation
iswETW	1	0: calculation of temperature according to calc_euetw 1: field data according to etw_file
iswETB	0	0: cosine temperature according to calc_euetw 1: field data according to etb_file
iswEPAR	2	0: cosine irradiance parametrisation 1: reading of photosynthetic available irradiance PAR from data files 2: reading mean irradiance from boundary model
iswESS	1	0: cosine suspended matter parametrisation 1: field data 2: any other function or data set (e.g. Fourier series)
iswESAL	1	0: values from model X1x 1: field data
iswTIDE	1	1: reading of gauge level according to tidelen_data_file and pegel_data_file 2: calculation of gauge level

C The mystic recalc_globals statement

The setting `recalc_globals=1` forces a recalculation of the model without integration to get `global` and `global_derived_from_states` variables recalculated before storing simulation results. This is needed if the integration method calls the model at intermediate interpolation points. Normally the values of global variables are calculated at the last interpolation point and therefore this value is stored. This is a well known problem with accurate integration methods. Even if the differences between the values at the end of the step and the values at some intermediate points is not serious, the results might be misinterpreted (f.e in budget computation where total mass conservation is expected). The differences increase if the system is non autonomous (directly dependent from the actual time, f.e. in the case of a forcing function). **CEMoS** provides the possibility to recalculate the values at the end of the step with the statement `recalc_globals=1;` in the file `cemos.par` (the default setting is `recalc_globals=0`).
In this case the model is called once again to calculate the global variables at the sampling point (with the actual simulation time) but without changing state variable values.
This is also very helpful, if the model is run with different integration methods on different time steps (operator splitting), and not all derived variables are affected by all integration methods, but shall be stored for diagnostic purposes.

Remark: No differences in the values of state variables shall occur with or without setting `recalc_globals=1` because the integration is not affected by this!!

Under some practical circumstances differences might occur:

- A state variable is directly set within the model to a new value (this is not allowed in the context of differential equation but may occur if state variables a misused, f.e. for diagnostic purposes.). During the recalculation such a state variable gets a new value which may force the integration routine to a slightly different behavior. If f.e. the state is set to a total different value the integration adapts the time step and this may lead to differences in all state variables.

- A global variable which determines the rate of a state variable is calculated at the wrong position in the model code. Globals are initialized with zero by **CEMoS** . Such no warning is given if a global is used before setting it to its right value. Because **CEMoS** passes through the model once before starting the simulation, normally no problems occur. But if the global itself is determined by the value of another state variables things go wrong. The following model will show the effect:

```
#include "struct.h"
void model(void)
{
SX[1]=a[1]*X[1];
SY[1]=Y[1];
a[1]=Y[1];
}
```

The state variables `X[1],X[2],Y[1],Y[2]` get all the initial value 1. The global variable `a[1]` is initialized by zero (**CEMoS** does it). The time step is fixed to 1. The results are taken from a simulation with a second order Runge-Kutta integration which has an intermediate calculation point:

	recalc=0	recalc=1
Time	x(1)	x(1)
0.00000	1.00000	1.00000
1.00000	2.50000	2.50000
2.00000	18.12500	16.56250

D Visualization of additional output files

The additional output files of **EcoTiM** can be visualized with **EcoViz**.
EcoViz has three main functionalities

- visualizing pictures
  - depth information: `<resultname>_mdepth.dat`,
  - box information: `<resultname>_box.dat`
  - statistic: `<resultname>_stat.dat`
- creating films
  - velocities: `velo.dat velo_u.dat velo_v.dat`
  - tide level: `tide.dat`
- showing tracer movement
  - tracer: `<resultname>.coo`

EcoViz is only a working tool for internal use. Such it is not as comfortable as **MoViE** and has not been tested with Windows. **EcoViz** is written in **Tcl/TK**. The image processing to ppmfiles is written in C.To start **EcoViz** it is necessary to go to the directory `ECOVIZ` and give the command `./ecoviz`.
By default the background file `ECOVIZ/data/mdepth.dat` is loaded and the region of the `SPIEKEROOG` setup is shown. The depth information and the grid coordinates are shown by moving the mousewithin the graphic window.

D.1 The input file format

The input files are ASCII files. Normally the input files are written with **EcoTiM**. The procedure to create these files is described in 16.7

D.1.1 Picture files

The input file format for picture files is a matrix of 324x88 grid points describing the total Lower Saxony Wadden Sea. The data structure is realized by the following loops

```
for (j=88;j>=1;j--)
        for (i=1;i<=324;i++)
            fprintf(store_file,"%f\n",(float) value);..
```

where the inner loop runs from west to east and the outer loop runs from north to south (this is convenient because typical graphic coordinates start in the upper left corner). The file must contain one number per per row.

D.1.2 Film files

The input file format for film files is the same as for picture files but repeated for every time step.

D.1.3 Tracer files

The coo file contains a row for every time step. The first column contains the time as floating value. The following columns contain integer values for x- and y- coordinates of the tracer (x and y alternating).

D.2 Area and canvas size

By default not all grid points are shown (very slow). The values for the different setups are defined in `ECOVIZ/custom/set_area.tcl` and can be selected in the **[configure]** tool.
The canvas size can be adapted to different monitor resolutions.

D.3 Visualizing pictures

The picture file can be selected by **[Graphic File]** . The background file is read from **EcoViz** itself. Only the islands are shown. Select File opens a file browser in the directory `ECORES` where the files of **EcoTiM** are normally stored.
[Print to PS] writes the actual canvas content into a PostScript file in `ECOVIZ/plots`.
Pictures can be also stored as gif files as films with one time step (see D.4.1)

D.4 Showing tracer movement

By default the file `ECORES/result.coo` is selected. **[Tracer File]** opens a file browser in the directory `ECORES` where the `coo`-files of **EcoTiM** are normally stored. **[Start tracer]** initializes the tracer movement and shows the tracer at its initial position. **[Step/Stop tracer]** makes one single step resp. stops the animation, **[Animate Tracer]** shows the movement. If only specific tracers should be shown, the numbers can be set in the entry **[Tracer]** . The numbers must be seperated by space. If the entry is empty all tracers will be shown. If **[show trace]** is activated the trace of the tracers is shown.

It is possible to show the tracer with a picture in background.

D.4.1 Making a tracer film

If **[prepare film]** is set and the animation is started, every pictures is stored as PostScript in `ECOVIZ/tracertmp`. **[Make Tracer Film]** creates an animated `gif` file from all PostSript Files and stores it in `ECOVIZ/films` by default.

The suggested file name contains date and time information. Depending on the number of files, this conversion may take several minutes up to hours.

Animated gif files can be played by XAnim (Freeware multimedia player for Linux). If XAnim is installed it can be started from **EcoViz**. In this case the **EcoViz** window is hidden until XAnim is exited.

D.5 Creating films

The files which can be directly selected are located in `ECORES`. Alternatively a file selector box can be opened. Depending on the number of time steps and the selected area the processing takes up to hours. **[Make Gif Film]** creates an animated gif file from the selected film file and writes it to `ECORES/films`. The suggested file name contains date and time information. Additionally all gif frames are stored temporarily in filmtmp and can be directly viewed by **EcoViz**. After processing a film control window opens which allows to step throug the set of frames. This frames are available until a new film is processed. The stored animated gif file still exist.

References

Index