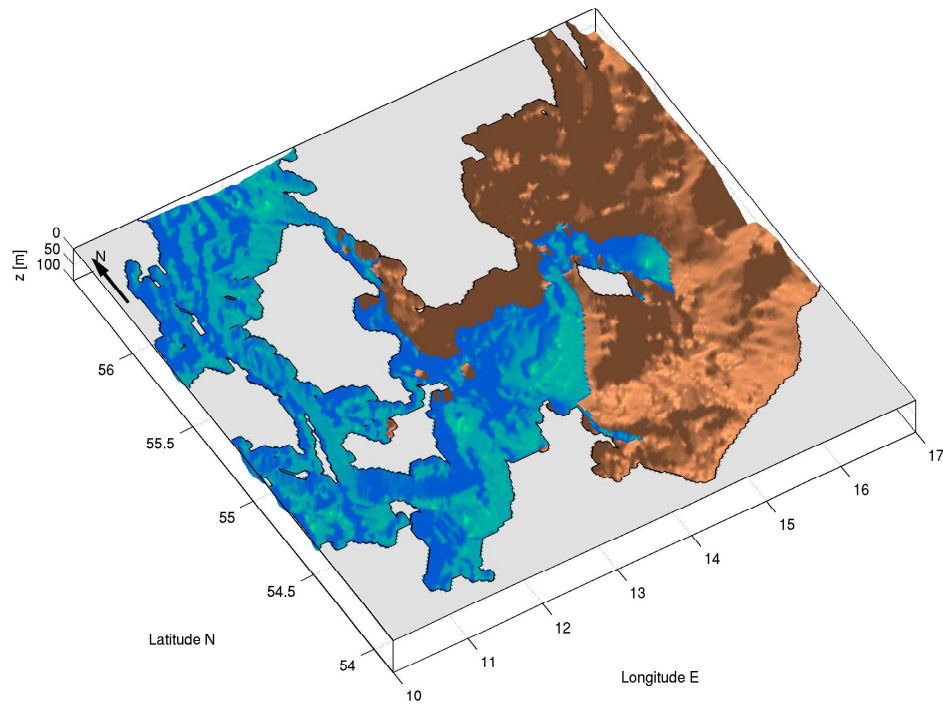


A Baltic Sea model test case for MOM

Martin Schmidt, Hagen Radtke, Torsten Seifert
Leibniz-Institute of Baltic Sea Research Warnemünde,
Seestraße 15, 18119 Rostock, Germany

May 11, 2012



Contents

1	Introduction	3
1.1	Contents of the test case	3
1.2	Summary of hydrographic conditions in the Baltic Sea	4
1.3	The Ecological Regional Ocean Model ERGOM	5
2	Model implementation	7
2.1	Preprocessing steps and model boundary data	7
2.1.1	Grid and topography	7
2.1.2	A note on open boundary conditions (OBC)	7
2.1.3	A note on vertical mixing	9
2.1.4	A note on shortwave radiation	10
2.1.5	Atmospheric forcing and river runoff	11
2.1.6	Masking land-only ocean domains	13
2.2	Defining passive tracers	13
2.3	A simple wave model	14
2.4	The sea-ice model	16
2.5	Activating the ecosystem model ERGOM	16
2.5.1	Namelist and tables	16
2.5.2	Gas dynamics and fluxes	17
2.5.3	Dry and wet deposition, river runoff	17
A	Suspended matter and sediment in ERGOM	19
A.1	Data types and storage of prognostic variables	20
A.1.1	The array <i>spm(i)</i> , <i>sed(i)</i> and the types <i>spm_type</i> and <i>sed_type</i>	20
A.1.2	The variable <i>sed_defs</i> and the type <i>sed_defs_type</i>	22
A.2	The subroutine <i>sedimentation_and_resuspension</i>	23
A.2.1	Sedimentation	24
A.2.2	Resuspension because of currents and waves	24
A.2.3	Resuspension by benthic animals	25
A.2.4	Sediment layer propagation	25
A.3	Implementation of 2d-tracers	26
A.3.1	Adding 2d-tracers	26
A.3.2	Storing data in 2d-tracers	27
A.3.3	Providing initial conditions for 2d-tracers	27
A.4	Detritus and mineralisation	28
A.4.1	Representation of detritus in the water column	28
A.4.2	Representation of detritus in the sediment	29
B	The ERGOM namelist - a summary	30

C	Bottom boundary layer model of combined current and wave action on bottom drag and skin friction	35
D	Basic relations	35
E	Three-layer bottom boundary model	36
F	Solution limits	38

1 Introduction

1.1 Contents of the test case

This document describes the Baltic Sea test case as an example for a regional application of MOM. Model grid and topography, forcing and model parameters look realistic and reasonable, but the results should be considered with caution, because a 9 nm resolution is not eddy resolving for this highly variable system. Similar model configurations but with better horizontal and vertical resolution are used for several research projects at the Leibniz-Institute for Baltic Sea Research Warnemünde (IOW). It is interesting and important to note, that an ocean model system originally developed for global ocean simulations, is also suitable for a regional application. On the other hand, methods and tools developed for research in marginal seas could be expanded potentially to global and climate modelling. This example demonstrates that model development for global and regional models does not necessarily result in diverging model codes. A consistent concept of keeping the numerical tools opens several valuable new pathways for fruitful developments in both the directions of computational ocean sciences. MOM is one outstanding example for this approach and succeeded to gain synergy from results of numerous modelling efforts worldwide. This achievement is a result of the clear supporting GFDL-policy regarding modelling, the general acceptance and support of modellers' ideas.

Stephen M. Griffies and his co-workers deserve credit for carrying the MOM flag with deep knowledge, patience, encouragement and energy over more than the last decade.

Several features implemented in MOM will be used and the configuration of MOM is:

- the Ecological Regional Ocean Model (ERGOM),
- a simple wind wave model to consider the resuspension under the combined influence of currents and waves,
- a special setup with open boundary conditions (OBC),
- examples for passive tracers.

1.2 Summary of hydrographic conditions in the Baltic Sea

The test case considers the ecosystem of the Baltic Sea, a humid and semi-enclosed sea in the north-east of the Atlantic. It is a chain of basins separated by more or less shallow sills. The exchange between the Baltic Sea and the North Sea is strongly limited by the narrow Danish straits. Episodic inflow events of saline water in combination with a seasonal fresh water surplus from rivers and precipitation maintain a permanent salinity gradient between the Kattegat in the west and the Bothnian Bay in the north-east, that exhibits only a small salinity. For a description and discussion of the hydrographic conditions in the Baltic Sea see for example Feistel and Wasmund [2008].

Inflowing saline water has a higher density and enters the Baltic Sea as bottom water. In the long time average (30 years) the inflow of salt is balanced by the vertical salt flux from the deep waters to the less saline surface waters and finally by the outflow of salt with the surplus of fresh water to the North Sea. Vertical fluxes of dissolved substances are due to upwelling at the edges of the subbasins of the Baltic Sea and by turbulent mixing. Hence, the Baltic Sea is a dilution area where a small amount of saline water comes in from the North Sea and a larger amount of less saline water is flowing towards the North Sea leaving the total amount of salt in the Baltic Sea nearly constant.

This balance applies for all conservative substances like nutrients or dissolved gasses, modified by river discharge, atmospheric inputs and atmosphere-ocean gas fluxes. However, for the dynamics of oxygen an additional flux has to be considered, which is much faster than the main circulation, namely the flux of dead organic matter (detritus) from surface waters to the sea floor. This flux may take place either vertically as sinking or horizontally as a transport of suspended fluffy material near the sea floor towards the deeper basins. It implies a large consumption of oxygen in the deep layers from mineralisation of detritus which disturbs the balanced oxygen flux with the mean circulation. As a result the Bornholm Basin and the Gotland Basin are always oxygen depleted and often even hydrogen sulphide is found in the water column.

Only parts of the detritus are mineralised in the water column, the major part forms a sediment where also mineralisation takes place. Two aspects are important here:

1. in the sediments denitrification takes place, i.e. nitrate is transformed into molecular nitrogen released to the water column. This shifts the ratio of the two limiting nutrients, nitrogen and phosphorus towards the phosphorus. This means that blooms of primary producers are usually limited by nitrogen availability. In conjunction with high summer temperature in the surface waters this turns the Baltic Sea into a preferable habitat for diazotrophs (cyanobacteria). These organisms take advantage from their ability to use molecular nitrogen for production and come into play when nitrate and ammonium are exhausted in the euphotic zone but phosphorus is still available.
2. If the sediment thickness is growing, the diffusion of oxidising agents (oxygen, nitrate, sulphate) into the deeper sediment layers is slow and organic matter can be considered as “buried”. This mechanism provides a substantial nutrient loss from the biological cycles.

In this frame the Baltic Sea ecosystem develops a typical seasonal cycle:

- Nutrients are accumulated in the surface water during winter time, because the mixing layer depth is large and the low amount of available light is not sufficient for a substantial phytoplankton growth.
- During spring time, heating of the surface water generates a seasonal thermocline with enough average light in the warm surface layer to support a phytoplankton spring bloom. Sinking of diatoms and detritus but also zooplankton grazing removes the organic matter from the surface waters.
- When ammonium and nitrate are exhausted in the surface waters but phosphorus is still available, diazotrophs may develop if the surface temperature is sufficiently high.
- When autumnal surface cooling with convection starts, additional nutrients from the subthermocline layer become available. This may support a secondary phytoplankton bloom that is, however, limited by the reduced available light.
- Detritus is mineralised in the water column and in the sediment. Upwelling at the edges of the basins and turbulent diffusion return the nutrients to the surface.

This simplified description comprises the minimum complexity of an ecosystem model of the Baltic Sea.

1.3 The Ecological Regional Ocean Model ERGOM

ERGOM is an acronym for the *Ecological Regional Ocean Model*, Neumann et al. [2002], Neumann and Schernewski [2008]. Developed originally by Thomas Neumann and Wolfgang Fennel for the Baltic Sea ecosystem, Fennel and Neumann [2004], it has gained some level of generality from its application to other areas of the ocean like the North Sea or the upwelling area on the Namibian shelf. The version implemented in FMS is a result of merging code from the original model branch and methods added for applications in the African eastern boundary currents.

ERGOM is a nutrient cycle model, but some aspects of the phosphorus cycle, the sulfur cycle and gas dynamics are included.

Fig. 1 comprises the general outline of the model. Round objects are state variables,

- **phytoplankton**. Up to three groups are considered: **diatoms** comprise groups with large cells, growing fast, requiring high nutrient supply and sinking; **flagellates** are small, grow slowly and are satisfied with less nutrients; **cyanobactus** stand for diazotrophs growing slowly and only in warm waters using molecular nitrogen (fixation).
- **zooplankton**. An arbitrary number of different zooplankton groups can be specified. They differ in grazing preferences, growth rate, vertical migration and mortality.
- **detritus**. Sinking dead organic matter.

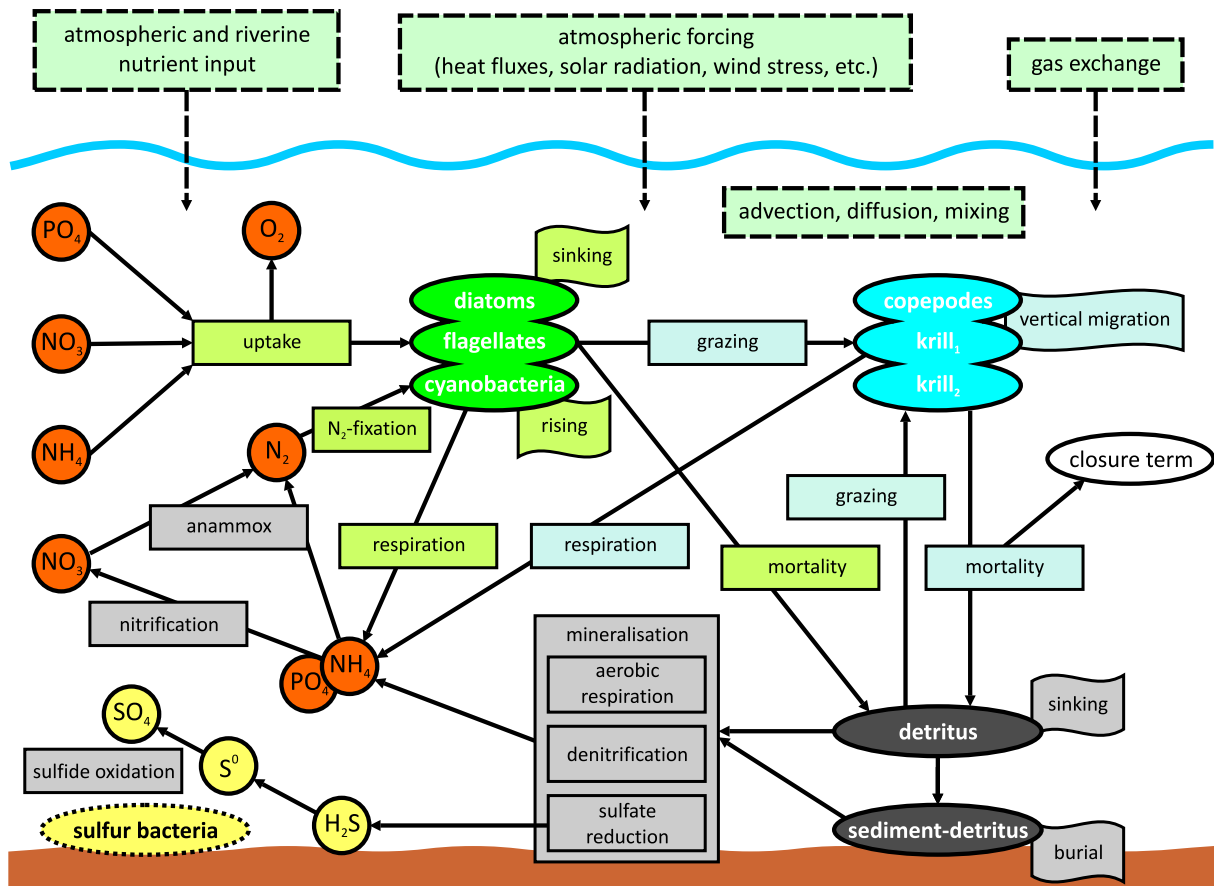


Figure 1: Nutrient flux diagram of ERGOM. (Graphics by A. Eggert, IOW.)

- nitrate and ammonium,
- phosphate,
- iron phosphate,
- dissolved oxygen, dissolved molecular nitrogen
- sulfur, hydrogen sulphide
- organic sediment
- iron phosphate in sediments

All organic matter means “nitrogen in ...”, i.e., “detritus” means “nitrogen in detritus”. The content of carbon and phosphorus in organic matter is assumed to be in Redfield ratio $C/N/P = 106/16/1$ and is taken into account in the biochemical reactions.

The cornered objects stand for processes. The “reaction speed” depends mostly on temperature (q-10 rules), the availability of source material like nutrients, oxygen concentration. Mineralisation processes are mostly due to bacteria. In many cases little is known on the nature of the microbial agents. Hence, these are not considered explicitly but are assumed to be omnipresent and become active in dependence on the environmental conditions. Still missing is the carbon cycle and iron limitation. A description of the model equations and more details on the implemented schemes can be found in Schmidt et al., 2012

2 Model implementation

2.1 Preprocessing steps and model boundary data

2.1.1 Grid and topography

The model area is covered with grid consisting of 76×82 horizontal grid cells and 60 vertical levels. This horizontal resolution of about 9 n.m. does not resolve the first baroclinic Rossby radius, Fennel et al. [1991]. Hence, coastal processes and baroclinic waves are not well represented. The results of this model look realistic but should not be used for scientific purpose. The topography is derived from the data base compiled by Seifert and Kayser [1995], Seifert et al. [2001]. The data base is available at <http://www.io-warnemuende.de/topography-of-the-baltic-sea.html>. Bottom cells are partial cells with a minimum thickness of 1 m or 20% of the full cell thickness, non-advective cells are closed.

2.1.2 A note on open boundary conditions (OBC)

The general scheme of OBC in MOM is described in Herzfeld et al. [2011]. For this test case a radiation condition applies in combination with relaxation toward boundary data. This is done for all tracers and for the sea level as well. It should be noted that not all prognostic tracers are relaxed toward boundary values. It is not straightforward to see the names of all prognostic tracers and their number in the tracer list. It is recommended to start a short model run and to analyse the file generated as *stdout*. One may find the tracer list below the lines

```
==>Note from ocean_tracer_mod(ocean_prog_tracer_init):
Reading prognostic tracer initial conditions or restarts
```

All options for OBC in *ocean_obc_nml* must refer to these tracer numbers. For example in this test case phosphate is specified like:

```
Initializing tracer number          10
at time level tau. This tracer is called po4
```

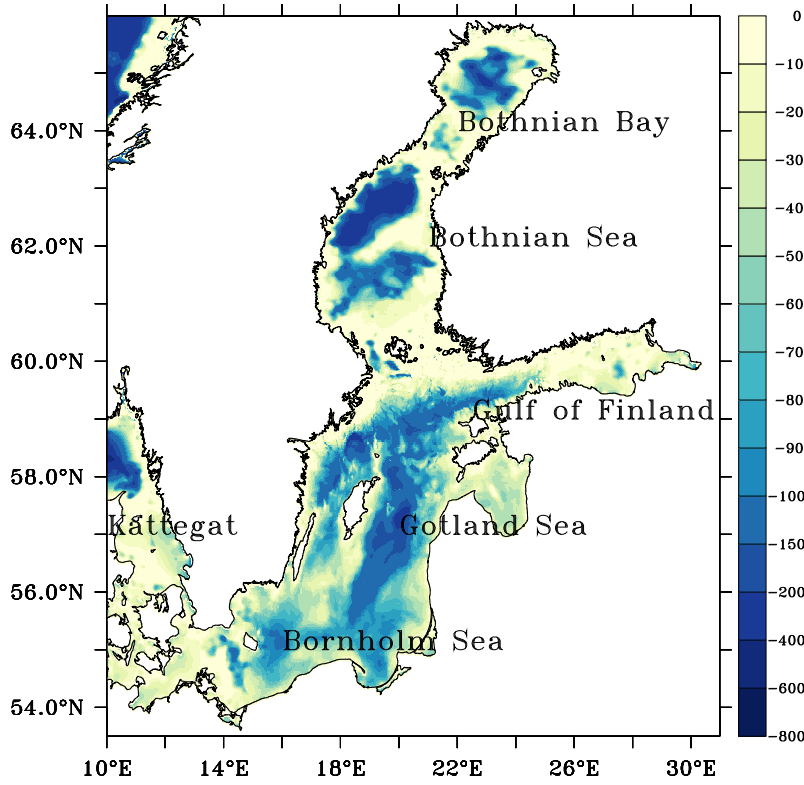


Figure 2: Sketch of the Baltic Sea topography.

The corresponding definition in *ocean_obc_nml* are

```
obc_tra(1,10)      = 'UPSTRM|FILEIN|IOW'
obc_relax_tracer(1,10) = .true.
filename_tracer(1,10) = 'INPUT/obc_clin_west.dta.nc'
fieldname_tracer(1,10) = 'po4'
rel_coef_tracer_in(1,10) = 2.78e-4
rel_coef_tracer_out(1,10) = 2.78e-4
rel_clin_pnts(1,10) = 1
```

which means that the tracer *po4* is using the methods *UPSTRM|FILEIN|IOW*, relaxation toward boundary data applies, the input file is *INPUT/obc_clin_west.dta.nc* and the corresponding file variable is named also *po4*. This must be specified for all tracers, there are no reasonable defaults. It is possible to use boundary data in a file variable for more than one tracer variable. For example all tracer variables relaxed toward zero could use a common file variable named *zero*.

For the relaxation of the sea level the option *MEANIN|IOW* is specified. Using *MEANIN* is suitable for the case that the sea level near the open boundary is known only at a single point.

Specifying the sea level profile along the boundary defines the geostrophic flow through the boundary. Hence, any error in the sea level profile may result in numeric artifacts near the open boundary. Specifying MEANIN relaxation is applied only to the sea level averaged along the boundary leaving the sea level profile undisturbed.

2.1.3 A note on vertical mixing

Vertical mixing is critical for the nutrient supply of the euphotic zone, F. Janssen, T. Neumann, M. Schmidt [2004]. Stratification in the Baltic Sea is characterised by two pycnoclines of different nature. The first one is a more or less permanent halocline established during strong (convective) mixing events going along with strong cooling and stormy conditions. The second one is a seasonal thermocline developing during summer. Below the thermocline but above the halocline very cold “winter water” is found that stays there enclosed during summer until deep convection resolves the seasonal thermocline. If the halocline is too shallow, available nutrient concentrations are underestimated.

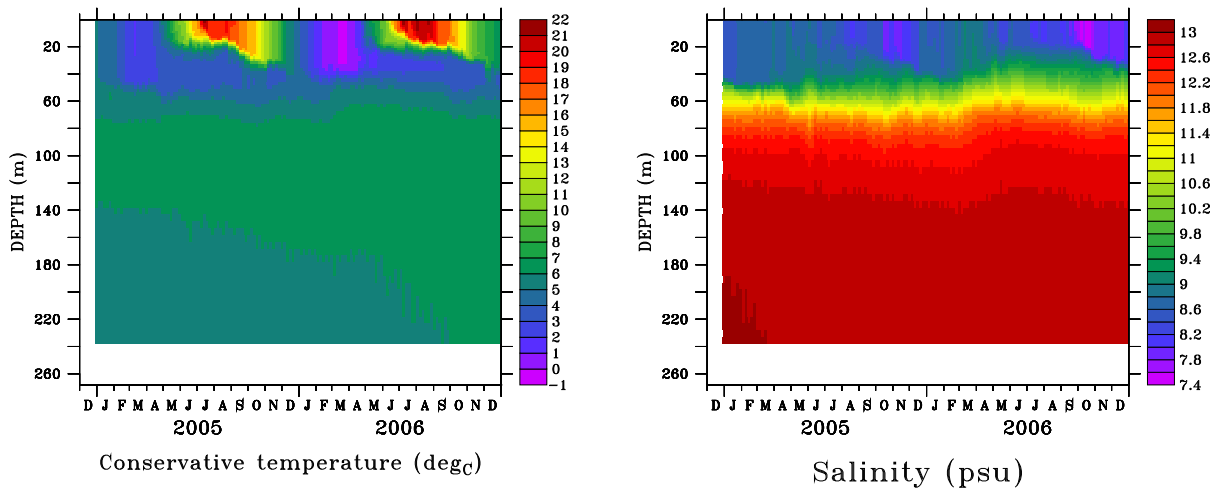


Figure 3: The seasonal cycle of the Baltic Sea stratification in the Gotland Basin. Simulations are carried out with a 1 n.m. model version. See <http://phy-51.io-warnemuende.de/thredds>

The simulated halocline depth depends on the numerical properties of the vertical mixing scheme. In this example the non-local kpp-scheme of Large et al. [1994], Large et al. [1997] is used. Some namelist parameters need attention:

```
diff_cbt_iw = 1.34e-6
visc_cbu_iw = 1.34e-5
Ricr=0.3
concv=4.25
limit_with_hekman=.false.
limit_ghats=.true.
```

```

hbl_with_rit=.true.
kl_min=2
radiation_iow=.true.
use_max_shear=.true.
variable_vtc=.true.
bvf_from_below=.true.
smooth_blmc=.false.
kbl_standard_method=.false.

```

The kpp-scheme has the tendency to underestimate the halocline depth. To improve the results the parameter `concv` is enhanced, which controls mixing strength at the pycnocline. For the same reason `use_max_shear=.true.` and `bvf_from_below=.true.` are used. `hbl_with_rit=.true.` avoids errors in the determination of the mixed layer depth which would generate numerical artifacts. The parameter `limit_ghats=.true.` assures that the vertical divergence of the non-local tracer fluxes does not exceed the total tracer flux. The option `radiation_iow=.true.` corrects an error in the original scheme, that is observed only in fresh water influenced model areas. The non-local vertical mixing redistributes the total surface heat flux over the mixing layer. This includes also the short wave radiation. However, radiation is treated separately in MOM. It is not absorbed in the surface layer only, but is distributed through the water column. Distributing the radiative surface flux twice may generate surface cooling from radiation as a numerical artifact. As long as solar radiation, hence warming, enhances surface layer stability, the numerical effect will be a rare event, since the non-local vertical mixing requires an unstable mixed layer. However, surface salinity in the Baltic sea is small enough that warming cold surface waters may lead to convection. If this happens, the original kpp-scheme warms deep layers and cools the surface layer down below the freezing temperature. The `radiation_iow=.true.` excludes radiation partially from the non-local mixing and corrects this shortcoming.

2.1.4 A note on shortwave radiation

Shortwave radiation spreads downward through the water column until it is totally absorbed. The absorption depends on the amount of chlorophyll in the water. In the Baltic Sea many other dissolved substances modify the optical water properties. This motivates to use “Jerlov-3” water properties, Jerlov [1968]:

```

&ocean_shortwave_nml
  use_shortwave_jerlov=.true.

&ocean_shortwave_jerlov_nml
  jerlov_3=.true.
  zmax_pen=300.

```

Alternatively

```
baltic_optics=.true.
```

can be used. ERGOM allows to pass chlorophyll concentration to the shortwave-module. Hence

```
use_shortwave_gfdl=.true.
```

and

```
&ocean_shortwave_gfdl_nml  
  read_chl=.false.  
  zmax_pen=1800.  
  optics_for_uniform_chl=.false.  
  optics_morel_antoine=.false.  
  optics_manizza=.true.  
  zmax_pen=300.
```

is also a reasonable choice.

The parameter `zmax_pen=300.` enables light penetration down to the sea floor. This is needed since zooplankton may detect light of tiny intensity that would be neglectable otherwise.

2.1.5 Atmospheric forcing and river runoff

The test case is configured like a fully coupled atmosphere-ocean-ice model. However, the atmosphere model is never called and file data are used instead for the calculation of the surface fluxes. The atmosphere variables are 10 m winds, 2 m air temperature, 2 m specific humidity, sea level pressure and precipitation. The downward shortwave radiation is calculated in a preprocessing step from a formula after Bodin, Bodin [1979], cloud coverage is taken from ERA40 data, Uppala et al. [2006]. Down-welling long-wave radiation is calculated from “grey body”- formulas following either Berliand , Berliand and Berliand [1952], or Bunker, see Kondratyev [1969].

The method to calculate the albedo is specified in namelist

```
&ocean_albedo_nml  
  ...  
  ocean_albedo_option=3
```

Using this method, the sun angle must be known, which in turn is calculated by the atmosphere model. However, if the atmosphere model is never called, the sun angle stays always 0. To avoid this error, the sun angle calculation is initiated from the ice model before

the albedo is calculated. This requires to define

```
&ice_model_nml
...
do_sun_angle_for_alb=.true.
```

This motivates the following entries in the *data_table*:

```
"ATM", "slp"      , "PAIR"   , "./INPUT/pair.mom.dta.nc" ,"default", 1.0e2
"ATM", "p_surf"   , "PAIR"   , "./INPUT/pair.mom.dta.nc" ,"default", 1.0e2
"ATM", "p_bot"    , "PAIR"   , "./INPUT/pair.mom.dta.nc" ,"default", 1.0e2
"ATM", "t_bot"    , "TAIRK"  , "./INPUT/tairK.mom.dta.nc","default", 1.0
"ATM", "sphum_bot", "shumi"   , "./INPUT/shum.mom.dta.nc" ,"default", 1.0
"ATM", "u_bot"    , "WINDX"  , "./INPUT/windx.mom.dta.nc","bicubic", 1.0
"ATM", "v_bot"    , "WINDY"  , "./INPUT/windy.mom.dta.nc","bicubic", 1.0
```

Data on radiation and fresh water fluxes are sent to the ice model:

```
"ICE", "lw_flux_dn", "lwdn_bun", "./INPUT/lwdn.mom.dta.nc", "default", 1.0
"ICE", "sw_flux_vis_dir_dn", "swdn"   ,
"                ./INPUT/swdn.mom.dta.nc" ,"default", 1.0
"ICE", "lprec", "rain"   , "./INPUT/rain.mom.dta.nc", "default", 1.157e-5
"ICE", "fprec", "snow"   , "./INPUT/snow.mom.dta.nc", "default", 1.157e-5
"ICE", "runoff", "runoff" , "./INPUT/runoff.nc",      "none",    1.0
```

The variable *snow* has been calculated in a preprocessing step from the precipitation. It is assumed, that all precipitation at air temperature below 0°C is snow.

Please note, the style of the *data_table* differs from the usual style. This allows to specify interpolation method “bicubic” for the wind field components, which results in a smoother representation of the wind stress curl compared with the default interpolation method “bilinear”. Only time interpolation is required for the runoff data, which are prescribed as monthly means at the model cells corresponding to river mouths. Together with a positive advection scheme no special measures like enhanced mixing are needed to prevent salinity from dropping below zero in river mouths.

Finally the specific humidity must be defined as an atmosphere surface (bottom) flux. This is the purpose of the following lines in *field_table*

```
"TRACER", "atmos_mod", "sphum"
      "longname",      "specific humidity"
      "units",         "kg/kg"
      "profile_type",  "fixed", "surface_value=3.e-6" /
```

```
# sphum must be present on land as well
```

```

"TRACER", "land_mod",      "sphum"
          "longname",      "specific humidity"
          "units",         "kg/kg" /

```

2.1.6 Masking land-only ocean domains

Data on ocean domains with land points only are never used. Hence, it is possible to mask these domains to avoid the attachment of compute nodes to such domains which would stand fully idle in this case. Here the following definitions are used:

```

&coupler_nml
...
n_mask = 26
layout_mask = 6, 11
mask_list = 5,1,6,1,5,2,6,2,6,3,6,4,2,5,2,6,1,7,2,7,
            1,8,2,8,5,8,6,8,1,9,2,9,5,9,6,9,1,10,2,10,
            3,10,6,10,1,11,2,11,3,11,6,11

```

Hence, from a layout with $6 \times 11 = 66$ tasks 26 tasks are masked out and the job is started with $6 \times 11 - 26 = 40$ cores.

No domain layout must be specified in `&ice_model_nml`. It is impossible to find these definitions by hand but there is an appropriate preprocessing tool in the ocean grid generator. Defining in `ocean_grid_generator.csh`

```

&topog_nml
...
check_mask=.true.

```

offers a list of options for masking for a given domain layout in `ocean_grid.fms.out` suitable to be placed in `&coupler_nml` simply by cut&paste.

2.2 Defining passive tracers

Passive tracers provide additional information on water mass spreading. The Baltic Sea test case provides three tracers, *passive_batre*, *passive_z20* and *passive_z50*. The *field_table* is modified like follows:

```

"tracer_packages","ocean_mod","ocean_passive"
names = batre, z20, z50
horizontal-advection-scheme = mdfl_sweby
vertical-advection-scheme = mdfl_sweby
restart_file = ocean_passive.res.nc

```

```

min_tracer_limit=0.0
/

"prog_tracers","ocean_mod","passive_batre"
const_init_tracer = f
/
"prog_tracers","ocean_mod","passive_z20"
const_init_tracer = f
/
"prog_tracers","ocean_mod","passive_z50"
const_init_tracer = f
/
"namelists","ocean_mod","ocean_passive/batre"
restore = t
/
"namelists","ocean_mod","ocean_passive/z20"
restore = f
/
"namelists","ocean_mod","ocean_passive/z50"
restore = f
/

```

The first tracer “batre” (named after a tracer release experiment BATRE) is defined with `restore = t`. This means that all grid cells are marked, where initial values different from zero are found. In these cells the tracer concentration is restored to 1 every time step. The other passive tracers are defined with `restore = f`. Hence, they are initialised once and spread passively and conservative with the ocean currents and the turbulent mixing. The initial fields are found in *ocean_passive.res.nc*, *passive_batre* is released in the deepest area of the Gotland Basin at about 20°E and 57°N, *passive_z20* and *passive_z50* are released at depths below 20 m and 50 m respectively.

2.3 A simple wave model

The ocean model MOM is now supplemented with a simple wave model. The original code is described in Schwab et al. [1984], D.J. Schwab and Donelan [1984] Liu and Bennett [1984]. The wave model is of spectral type and uses the JONSWAP spectrum. It considers only one peak frequency and is not able to handle swell. However, for the Baltic Sea the wind driven surface waves play a major role. The prognostic quantity in the wave model is the wave momentum, peak frequency, significant wave height and wave number are diagnosed. The interaction of waves with the sea floor is calculated in the TMA-approximation, Hughes [1984], Bouws et al. [1985], Bouws et al. [1987], which considers only a modification of shape and maximum value of the spectrum.

The aim of the wave model is to provide a measure for the wave action at the sea

floor. This is needed to estimate resuspension of sedimentary material. It is also possible to estimate the modification of the vertical momentum flux into the bottom due to the combined action of waves and momentum. Details of the Grant and Madsen scheme implemented for this purpose are given in Appendix C.

Generally, handling of the time steps for waves, currents and tracers is very simple. The wave model is called as part of the ocean model. Like the ocean barotropic mode it uses some sub-steps to avoid CFL-violations. Because the relation between wave momentum, peak frequency and significant wave height is non-linear, instead of some time average the result of the last wave model time step is used to analyse bottom stress for velocity and resuspension of tracers neglecting the time staggering of velocities and tracers.

To switch on the wave model the wave module must be initialised:

```
&ocean_wave_nml
  use_this_module = .true.
```

Here the wind action on the waves is calculated in the wave model and not in the coupler. To provide the information on the 10 m winds and on the ice mass for the ocean waves sub-model a modified data structure of type `ice_ocean_boundary_type` is used. For data overriding the *data-table* needs the lines:

```
"OCN", "u_bot", "WINDX" , "./INPUT/windx.mom.dta.nc", "bicubic", 1.0
"OCN", "v_bot", "WINDY" , "./INPUT/windy.mom.dta.nc", "bicubic", 1.0
```

It would be also possible to use the 10 m winds calculated by an atmosphere model. This requires a few changes in the module *flux_exchange*. However, the applicability of the wave model is kept restricted until the full atmosphere-ice-wave-ocean coupling is implemented eventually for a more realistic wave model.

In order to use the wave information for the bottom stress calculation, `&ocean_bbc_nml` must be changed like follows:

```
&ocean_bbc_nml
  cdbot=2.e-5
  cdbot_wave=.true.
```

This enables the calculation of the vertical momentum flux with a 3-layer wave-current boundary layer model, see Sec. C. A modified drag coefficient is provided for the bottom stress acting on the ocean currents. The combined wave-current bottom stress is calculated to estimate resuspension in the ecosystem model.

To see details enable the output of surface wave quantities in the *diag-table*

```
"ocean_model","windx_wave","windx_wave","ocean_day2d" ,"all",.true., "none",2
"ocean_model","windy_wave","windy_wave","ocean_day2d" ,"all",.true., "none",2
"ocean_model","xmom"          ,"xmom"          ,"ocean_day2d" ,"all",.true., "none",2
```

```

"ocean_model","ymom"      ,"ymom"      ,"ocean_day2d" ,"all",.true., "none",2
"ocean_model","height"    ,"height"    ,"ocean_day2d" ,"all",.true., "none",2
"ocean_model","wave_p"    ,"wave_p"    ,"ocean_day2d" ,"all",.true., "none",2
"ocean_model","wave_k"    ,"wave_k"    ,"ocean_day2d" ,"all",.true., "none",2
"ocean_model","wave_s"    ,"wave_s"    ,"ocean_day2d" ,"all",.true., "none",2
"ocean_model","wave_u"    ,"wave_u"    ,"ocean_day2d" ,"all",.true., "none",2
"ocean_model","iter"      ,"iter"      ,"ocean_day2d" ,"all",.true., "none",2
"ocean_model","current_wave_stress","current_wave_stress",
"ocean_day2d" ,"all",.true., "none",2

```

2.4 The sea-ice model

The Baltic Sea develops an seasonal ice cover, which melts totally away during summer. Ice thickness of about 1 m is common, but it may occasionally grow up to 3 m. In correspondence with the high sea level variability due to the wind one may encounter empty model surface cells, if the z-coordinate option is chosen for the vertical grid. This does never happen with z^* -coordinates even in the case of thick ice and strong wind driven sea level variations.

Considering horizontal grid cell size of 1 n.m. or more, the ice model develops reasonable ice thickness distribution comparable with field data. Hence, the ice model parameters are left unchanged to their defaults.

2.5 Activating the ecosystem model ERGOM

2.5.1 Namelists and tables

The numerical scheme of the ecosystem model ERGOM is similar to TOPAZ. Main differences are the treatment of hypoxia, vertical migration of zooplankton and a low order sediment model. Still missing is the carbon cycle in ERGOM.

To use ERGOM the generic tracer package must be activated and ERGOM must be chosen,

```

&generic_tracer_nml
  do_generic_ERGOM=.true.

```

In the *field_table* the ERGOM variables must be added to the tracer packages,

```

"tracer_packages","ocean_mod","generic_ergom"
names = _
horizontal-advection-scheme = mdf1_sweby
vertical-advection-scheme = mdf1_sweby
min_tracer_limit=0.0

```


It is possible to change all parameters of ERGOM via the *field_table*. However, this is not tested. Instead a namelist `&ergom_nml` may be used to configure the ecosystem model.

2.5.2 Gas dynamics and fluxes

ERGOM maintains the dynamics of three gasses, oxygen, hydrogen sulphide and nitrogen. The explicit account for oxygen and hydrogen sulphide permits studies of processes in hypoxic and sulphidic water like in the deep basins of the Baltic Sea or the oxygen minimum zones in the eastern boundary current systems. Molecular nitrogen concentration can be measured with special sensors and provides additional information on denitrification processes. This motivates the explicit inclusion of molecular nitrogen.

To find the flux of dissolved gasses through the ocean surface several ingredients are needed:

- the saturation concentration of the gasses as function of temperature and salinity. For oxygen the formula of Garcia and Gordon [1992] is used, for nitrogen that of, Hamme and Emerson [2004].
- the Schmidt numbers to calculate the “piston velocity” K_w . Here the formula of Keeling, R. F., Stephens, B. B., Najjar, R. G., Doney, S. C., Archer, D. and M. Heimann [1998] is used.
- the surface concentrations and the concentrations in the atmosphere.

For details of the gas flux calculation see

<http://www.ipsl.jussieu.fr/OCMIP/phase2/simulations/CFC/HOWTO-CFC.html>.

In the *data_table* the partial atmosphere pressure of the gas must be specified. The file `gasses.bc.nc` contains only one field `nitrogen` with a constant value of 1 everywhere.

```
"ATM","nitrogen_flux_pcair_atm","nitrogen","INPUT/gasses.bc.nc","default",1.0
"ATM","o2_flux_pcair_atm","nitrogen","INPUT/gasses.bc.nc","default",1.0
```

2.5.3 Dry and wet deposition, river runoff

Dry and wet deposition of nutrients are one important source of nutrients. The fluxes are prescribed from file. The corresponding lines in the *data_table* are:

```
"OCN", "wet_dep_no3_flux_ice_ocn", "NO3_WET_DEP",
      "./INPUT/depflux_total.mean.1990.nc", "default" , -1.0
"OCN", "dry_dep_no3_flux_ice_ocn", "NO3_DRY_DEP",
      "./INPUT/depflux_total.mean.1990.nc", "default" , -1.0
"OCN", "wet_dep_nh4_flux_ice_ocn", "NH4_WET_DEP",
      "./INPUT/depflux_total.mean.1990.nc", "default" , -1.0
"OCN", "dry_dep_nh4_flux_ice_ocn", "NH4_DRY_DEP",
```

```

        "./INPUT/depflux_total.mean.1990.nc", "default" , -1.0
"OCN", "wet_dep_po4_flux_ice_ocn", "NO3_WET_DEP",
        "./INPUT/depflux_total.mean.1990.nc", "default" , -0.0625
"OCN", "dry_dep_po4_flux_ice_ocn", "NO3_DRY_DEP",
        "./INPUT/depflux_total.mean.1990.nc", "default" , -0.0625
"OCN", "runoff_no3_flux_ice_ocn", "NO3",
        "./INPUT/runoff.nc" , "default", 1.0
"OCN", "runoff_nh4_flux_ice_ocn", "NH4",
        "./INPUT/runoff.nc" , "default", 1.0
"OCN", "runoff_po4_flux_ice_ocn", "PO4",
        "./INPUT/runoff.nc" , "default", 1.0

```

A Suspended matter and sediment in ERGOM

contributed by Hagen Radtke, Leibniz-Institute of Baltic Sea Research

Two distinct aspects of the dynamics of suspended particulate matter and sediment must be considered:

- part of the suspended particulate matter is of organic nature and serves as food for zooplankton or is mineralised by bacteria. This holds in the water column and in the sediment as well.
- Sediment is a mixture of organic matter and inorganic components of different grain size and density and is formed in a complex diagenetic process. For the ecosystem resuspension or bedload transport of the organic component is most important which is strongly influenced by the other sediment compartments. However, ERGOM considers the organic components only and some simplification is needed for the influence of inorganic sediment components.

Hence, in future more advanced sediment transport model should replace this part of ERGOM code.

Suspended matter is represented as a usual tracer but with an additional sinking speed. Sediment is organised in several layers. The layers do not have a fixed size but account for different sediment decomposition and redox state. To be more specific consider the required oxygen supply for mineralisation in sediments. Surface sediments undergo fast exchange with bottom water but with increasing depth sediments become more and more isolated because pore water flow and diffusion are slow. At some depth organic matter can be considered as "buried" since mineralisation is slow. A first order measure if some organic matter is buried or not is the amount of matter above. This simple model implies the following:

- Each type of sediment matter has its own layer system.
- Sedimentation means the transfer of matter from the lowest ocean cell to the surface sediment layer.
- Resuspension transfers matter from the surface sediment layer to the lowest ocean cell.
- If the amount of matter within a layer exceeds some threshold material is transferred to the next sediment layer.

The model distinguishes different types of sedimentary material. For the Baltic Sea model these are detritus and iron phosphate. Each type has a specific critical shear stress for resuspension, molar volume, sinking velocity and settling velocity. For each type and layer the concentration is given in mol m^{-2} .

For the current experiments only two layers are used for each sediment type.

A.1 Data types and storage of prognostic variables

Particulate matter is represented by three data structures:

- The array `spm(i)` of type `spm_type` contains all data that belong to a specific type i of suspended material.
- The array `sed(i)` of type `sed_type` contains all data that belong to a specific type i of settled material.
- The variable `sed_defs` of type `sed_defs_type` contains all definitions concerning the sediment model in general.
- The variable `sed_bio` of type `sed_bio_type` contains all data concerning biological processes in the “detritus” type of the sediment, like mineralisation or denitrification.

The first two of them make up what may be called a physical sediment model and may be moved to a “sediment module” some day. The last one is an integral part of the biological model.

A.1.1 The array `spm(i)`, `sed(i)` and the types `spm_type` and `sed_type`

The allocatable arrays `spm(i)` and `sed(i)` are the main data structure in the sediment model. Its dimensions are the namelist parameters `NUM_SPM` and `NUM_SED` defining the number of types of suspended and settled particulate matter. Although `NUM_SPM` and `NUM_SED` may be the same in many cases, it is an option to consider of different configuration. For example several types of detritus from different sources with different character (sinking velocity) may be collected in a single sediment variable. Hence, the correspondence between the variables may be non-trivial and is established by pointers `suspend_to` and `sediment_to`. The 3d-fields for the suspended matter are pointers to usual tracers maintained together with the generic tracer manager. Default is a correspondence between variables for suspended and settled matter in the order of definition. However, if `NUM_SED` and `NUM_SPM` are different, an explicit definition is required.

The main data structures for the prognostic variables of type `spm_type` and `sed_type` are:

<code>p_wat</code>	<code>p_wat(i, j, k)</code> → Pointer to the 3d-field for concentration in water column [mol/kg], stored by the tracer manager
<code>f_sed</code>	<code>f_sed(i, j, layer)</code> → 2d-arrays for sediment concentration [mol/m ²] in the different layers.

The parameters controlling the behaviour of the sediment or the suspended particles can be modified via the sediment model namelist, see Tab. 1 (here part of `ergom_nml`).

Examples for such configurations are:

1. 2-layer sediment and detritus: We define in the namelist

Table 1: **Namelist parameters controlling the sediment model**

Name in namelist	Name in code	Description
Unit in namelist	Unit in code	
wsink0_spm m/d	wsink0 m/s	sinking velocity (< 0 for sinking)
wsed_spm m/d	wsed m/s	sedimentation rate
erosion_rate_sed 1/d	erosion_rate 1/s	erosion rate of a sediment type by waves and currents
bioerosion_rate_sed 1/d	bioerosion_rate 1/s	erosion rate of a sediment type by benthic animals
molar_volume_sed m^{-3}/mol	molar_volume m^{-3}/mol	molar volume of this sediment type
critical_stress_sed Nm^{-2}	critical_stress Nm^{-2}	critical shear stress for erosion
name_spm	name	name of 3d-tracer, 'none' means undefined
name_sed	name	name of 2d-tracer, 'none' means undefined
longname_spm	longname	long name for description in output
longname_sed	longname	long name for description in output
sediment_to	sediment_to	name of 2d-tracer to which sedimentation takes place
suspsed_to	suspsed_to	name of 3d-tracer to which resuspension takes place

```

NUM_SPM = 1
NUM_SED = 1
NUM_SED_LAYERS = 2
name_spm(1) = 'det'
name_sed(1) = 'sed'
longname_spm(1) = 'detritus'
longname_sed(1) = 'detritus'

```

This means we will get three output variables:

```

det (detritus concentration in water),
sed_1 (detritus concentration in sediment layer 1)
sed_2 (detritus concentration in sediment layer 2).

```

2. one 2-layer sediment variable and two detritus variables. One is sinking fast the other one is sinking slowly. We define

```
NUM_SPM = 2
NUM_SED = 1
NUM_SED\_LAYERS = 2
name_spm(1) = 'fast'
name_spm(2) = 'slow'
longname_spm(1) = 'fast detritus'
longname_spm(2) = 'slow detritus'
sediment_to(1) = 'sed'
sediment_to(2) = 'sed'
name_sed(1) = 'sed'
longname_sed(1) = 'detritus'
suspend_to(1) = 'fast'
```

There is only one type of sediment. Both, the fast and the slow sinking detritus are transferred to `sed` when settling. If `sed` is resuspended, it becomes fast sinking detritus, justified by the assumption that small particles stick to larger particles. We get four output variables:

```
fast (fast detritus concentration in water),
slow (slow detritus concentration in water),
sed_1 (detritus concentration in sediment layer 1)
sed_2 (detritus concentration in sediment layer 2).
```

A.1.2 The variable `sed_defs` and the type `sed_defs_type`

The variable `sed_defs` contains parameters defining physical sediment processes. The parameters are:

Name in namelist	Name in code	Description
NUM_SED_LAYERS	NUM_LAYERS	number of vertical layers
sed_layer_propagation	layer_propagation	sediment layer propagation mode 1=SLP_DOWNWARD 2=SLP_FULL_BOX 3=SLP_OLD_ERGOM
sed_erosion_mode	erosion_mode	sediment erosion mode 1=SEM_INDEPENDENT 2=SEM_MAXSTRESS 3=SEM_ORGANIC
sed_layer_height(n)	layer_height(n)	maximum height of each sediment layer; < 0 means the layer may get infinitely thick

The parameter **NUM_SED_LAYERS** provided in the namelist is the number of vertical model sediment layers. The maximum thickness of each layer is defined by the parameter **sed_layer_height(n)**, where n is the index of the layer, counted from $n=1$ at the top to $n=\text{NUM_SEDIMENT_LAYERS}$ for the deepest layer. The maximum thickness is a threshold for the onset of downward propagation of sediment through all layers. The deepest layer should be allowed to become infinitely thick. This is enabled by a negative value of **sed_layer_height**. Otherwise, the material is lost and the model is not conserving. (This is a numerical and not a physical process, not the material itself moves downward, but the vertical counting of positions of the layers is changed.)

A.2 The subroutine *sedimentation_and_resuspension*

This subroutine is called from the main ecosystem model subroutine **generic_ERGOM_update_from_source**. It may be replaced by a separate sediment module later.

The subroutine **sedimentation_and_resuspension** needs a number of parameters:

- The timestep τ , for the Euler-forward integration.
- The grid cell height array dzt , to convert a settling velocity (m/s) to a relative fraction of the concentration that settles per time step (1/s).
- The density times grid cell height array ρdzt , to convert between the units mol/kg (for 3d-tracers) and mol/m² (for 2d-tracers).
- The array **suspsed** containing the prognostic variables and its dimension **NUM_SUSPSED**.
- The variable **sed_defs**
- The 2d-array **current_wave_stress** that contains the combined shear stress of currents and waves at the bottom.

- The 2d-array `bioerosion` that indicates where benthic organisms live that may suspend particles from the sediment. This field is delivered from the ecosystem model.

Within this subroutine four processes take place in the following order:

- Sedimentation
- Resuspension because of current and wave action
- Resuspension by benthic animals
- Sediment layer propagation.

The details how this propagation takes place are determined by the choice of the parameter `sed_layer_propagation`.

The parameter `sediment_erosion_mode` determines how erosion takes place when a mixture of sediment types with different critical shear stress is present at the bottom.

A.2.1 Sedimentation

Sedimentation of particles in the water column takes place from the lowest T-cell with a specific settling velocity `wsed`. The namelist parameter `sediment_to(i)` specifies the corresponding 2d-tracer in the sediment. Usually the settling velocity, `wsed`, should be less than sinking velocity, `wsink0`. This leads to accumulation of suspended particles in the lowest T-cell like in a “fluffy layer”.

A.2.2 Resuspension because of currents and waves

If the current-wave stress, `current_wave_stress`, exceeds the critical shear stress, `suspsed(i)%critical_stress` resuspension of sediment takes place.

There are three numerical schemes implemented for erosion, selected by `sed_defs%erosion_mode`.

For the choice `SEM_INDEPENDENT=1`, the sediment types are eroded independently. Each sediment type is suspended when its specific critical shear stress is exceeded. The shear stress acts only as a switch between the states “erosion” and “sedimentation”. If a sediment type is suspended, a constant percentage of its volume within the uppermost sediment layer is transferred to the water column. This percentage is determined by the namelist parameter `erosion_rate_suspsed(i)`.

If the choice is `SEM_MAXSTRESS=2` (not implemented yet), the maximum critical shear stress of all present sediment classes needs to be exceeded to allow sediment resuspension.

If the choice is `SEM_ORGANIC=3`, only organic matter is suspended when its specific critical shear stress is exceeded. The erosion rate is proportionally to the difference between the

wave-current stress and the critical stress. It depends also on the amount of other sediment types like fine sand, which, however, is unknown. Hence, simple assumptions are made to account for the ratio of organic sediment to the total amount of sediment.

By default, suspending sediment is transferred to the corresponding 3d-tracer. This can be modified with the namelist variable `suspend_to(i)`.

A.2.3 Resuspension by benthic animals

In the case of resuspension by benthic animals all sediment types are resuspended, no matter what the choice of `sed_defs%erosion_mode` is and independent off the shear stress. The array `bioerosion` provided by the ecosystem model contains information (0 to 1) if benthic animals eroding the sediment. The erosion takes place at a rate `bioerosion_rate_suspsed(i)` specified in the namelist.

A.2.4 Sediment layer propagation

The way sediments propagate through the layers depends on the choice of `sed_defs%layer_propagation`.

- `sed_layer_propagation=SLP_DOWNWARD` is a very simple scheme that allows downward propagation only. If the surface box is filled, the excess volume propagates downward to the next layer. If this is also filled completely, material propagates to the next layer and so on. The ratio between the sediment types in the source box determines the ratio of the sediment types in the downward flux. If the lowest box is limited in height, the material that leaves it downwards is lost. This scheme does not allow for upward flux, if the surface box is empty, no more erosion can take place.
- `sed_layer_propagation=SLP_FULL_BOX` (not implemented yet) is a scheme that allows both upward and downward propagation. To use this scheme, all boxes must have the same maximum height h , except for the surface box which has the maximum height $2h$. If the actual height of the surface box exceeds $2h$, sedimentary material of the height h is moved downwards, that is, the content of each box is stored in the box below. If the actual height of the surface box drops below h and the box underneath is not empty, material of the height h is transported upwards again, that is, the content of each box is transferred to the box above. This scheme prevents numerical mixing of different sediment layers by the layer propagation.
- `sed_layer_propagation=SLP_OLD_ERGOM` is a very specific scheme that realizes sediment burial of detritus and iron phosphate just as it was done in the MOM-31 version of ERGOM. There are two boxes, the lower one is unlimited in height and collects all material that is buried forever. There are two types of sediment, a Redfield-ratio detritus and iron phosphate.

Redfield-ratio detritus is buried if its height exceeds the height of the uppermost box.

Iron phosphate is buried by immobilization when iron phosphate is transferred downwards by pore-water diffusion. If the amount of iron phosphate in the top layer does not exceed 0.1 mol m^{-2} , a certain ratio r of the existing iron phosphate is buried that depends on the sediment thickness,

$$r = \frac{h}{h_{max}} \cdot 0.007 \text{ d}^{-1},$$

where h is the actual and h_{max} the maximum thickness of the surface layer. This thickness is only calculated taking the Redfield-ratio detritus, not the iron phosphate, into account. According to this formula the more iron phosphate is immobilized the thicker the sediment layer is. If the amount of iron phosphate in the active sediment, I , exceeds the threshold of 0.1 mol m^{-2} , iron phosphate is buried faster:

$$r = \left(1 + \frac{I - 0.1 \text{ mol m}^{-2}}{0.0045 \text{ mol m}^{-2}}\right) \frac{h}{h_{max}} \cdot 0.007 \text{ d}^{-1}.$$

A.3 Implementation of 2d-tracers

For the sediment module, 2-dimensional tracers are needed which are not provided by MOM tracer manager. They need the same methods like 3d-diagnostic tracers, no advection or diffusion but restarts and output. Hence, all 2d-tracers are collected in 3d-tracer fields that can be handled by the tracer manager.

A.3.1 Adding 2d-tracers

2d-tracers are stored in 3d-tracers serving as a storage container in the MOM tracer package. The list of 2d-tracers is initialised by a call of subroutine `user_init_2d_tracer_list`. Then 2d-tracers can be added to the 2d-tracer list with subroutine `user_add_2d_tracer`.

The subroutine `user_add_2d_tracer` needs the following parameters:

- **tracer_list**, the list of tracers in which the 3d-tracer(s) containing the 2d-tracers shall be added.
- **name**, the variable name of the tracer used for output
- **longname**, the variable description in the output file(s)
- **units**, the unit of the tracer, for example ' mol m^{-2} '.

The subroutine does the following

- If needed, the subroutine generates a new 3d-diagnostic tracer to store the 2d-tracers. The number of 2d-tracers that can reside in a 3d-tracer is determined by the namelist parameter `vert_levels`. This parameter is needed, because `kmt` is still unknown when 3d-tracers are registered. It never must exceed the parameter `kmt`. The first 3d-tracer will be called `tracer_2d_1`, the second `tracer_2d_2`, and so on.

- The information `name`, `longname` and `units` is kept together with the information on the 3d-tracer and the level where the 2d-tracer is stored. (They are stored in the following order. `tracer_2d_1` level 1, ..., `tracer_2d_1` level `vert_levels`, `tracer_2d_2` level 1, `tracer_2d_2` level 2, ...)

The subroutine `user_register_2d_tracers` registers output fields and restarts. It will call the subroutine `register_diag_field` for each of the 2d-tracers.

A.3.2 Storing data in 2d-tracers

2d-tracers do not contain a two-dimensional array on their own. Originally, their data is only stored in the 3d-diagnostic tracer array. To access the data, you need to allocate a 2d-array on your own. Then assign it to the 2d-tracer by calling the subroutine `a_two_dimensional_array` to be assigned to. You need to allocate a two-dimensional array of the correct size on your own. Then, call the subroutine `user_2d_tracer_assign_array`. It needs the name of the 2d-tracer and the array as a parameter. This subroutine then stores a pointer to the 2d-array in the 2d-tracer list.

To get the 2d-tracer fields, call the subroutine `user_get_2d_tracer_values`. For performance reasons, this subroutine will get the values of all 2d-tracers from the 3d-diagnostic tracer field, and store them in the 2d-arrays that have been assigned to them before. Then you may modify the data in the 2d-tracer fields.

The subroutine serves to store `user_set_2d_tracer_values`. It does two things:

- store the data of all 2d-tracers in the corresponding 3d-tracer array
- send the data to the registered output field.

A.3.3 Providing initial conditions for 2d-tracers

To get your model running, you will need to provide initial conditions for the 2d-tracers. They have to be stored in the corresponding 3d-tracer arrays. Provide initial conditions for `tracer_2d_1`, `tracer_2d_2` and so on.

For example, assume you have specified `NUM_SEDIMENT_LAYERS=7`, and you have 3 particulate matter variables. Suppose the names of their 2d-tracers are:

```
name_2d_suspsed(1) = 'mud'
name_2d_suspsed(2) = 'sand'
name_2d_suspsed(3) = 'gravel'
```

Suppose `vert_levels=20` to be specified. Then initial conditions for two 3d tracers called `tracer_2d_1` and `tracer_2d_2` are needed. The layers of the sedimentary tracers are stored

there as follows:

```

mud_1  →  tracer_2d_1(:, :, 1)
mud_2  →  tracer_2d_1(:, :, 2)
...
mud_7  →  tracer_2d_1(:, :, 7)
sand_1  →  tracer_2d_1(:, :, 8)
sand_2  →  tracer_2d_1(:, :, 9)
...
sand_7  →  tracer_2d_1(:, :, 14)
gravel_1 → tracer_2d_1(:, :, 15)
gravel_2 → tracer_2d_1(:, :, 16)
...
gravel_6 → tracer_2d_1(:, :, 20)
gravel_7 → tracer_2d_2(:, :, 1)

```

A.4 Detritus and mineralisation

A.4.1 Representation of detritus in the water column

Several types of detritus can be included in the model. They may differ in sinking velocity, sedimentation rate, and the speed of recycling.

In the model there are two structures to represent a functional group of detritus. In `suspsed(:)` of type `spm_type` an array is needed to store concentrations in the water column. The 3d-state variable is handled by the tracer manager with help of a pointer to `suspsed(n)%p_wat(i,j,k)` in `spm_type`. This type also contains parameters that control the physical properties of detritus as sinking and sedimentation. All these parameters can be modified via a namelist. For example, define the name of the 3d-tracer specifying the parameter `name_3d_suspsed` in the namelist.

There are not only physical processes modifying the detritus state variable. Biological processes create detritus or destroy it by mineralization. This is a difference to other suspended matter, for example fine sand. Hence, a second structure exists in the model for representing biological processes with detritus. The array `det(:)` of type `detritus` represents several functional groups of detritus. However, it does not contain an own 3d-state variable, this is just a references to the appropriate data in the `suspsed(:)` array. This implies that detritus concentration is stored as `spm_type` 3d-variable, but the `detritus` parameters in the array `det(:)` represent the biological processes with detritus.

We consider the example of only one detritus state variable, when the array `det(:)` has dimension 1. Assume three `spm_type` 3d-variables to exist, these could be fine sand, coarse sand and detritus. Then the parameter `det(1)%index_suspsed` is set to 3 pointing to the third suspended matter variable. The biological model part gets access to the state variable

via

```
suspsed(det(1)%index_suspsed)%p_wat(i,j,k)
```

while the physical processes (sinking and sedimentation) will act directly on

```
suspsed(3)%p_wat(i,j,k).
```

The idea of using this rather complicated structure is that the three processes sinking, sedimentation and erosion may one day be calculated by a separate sediment module that operates independently from the biological module.

The index `det(1)%index_suspsed` is not set manually, instead, the namelist parameters `name_3d_suspsed(n)` and `name_det(m)` must be the same.

The number of functional groups of detritus is given by the namelist parameter `NUM_DET`. The parameters stored in the `det(:)` array are listed in the table below.

Name in namelist	Name in code	Description
Unit in namelist	Unit in code	
name_det	name_det	variable name of the detritus tracer a variable with the same name must be defined also in name_3d_suspsed .
dn 1/d	dn 1/s	detritus recycling rate at 0°C
q10_rec 1/°C	q10_rec 1/°C	q-10 rule factor for detritus recycling

A.4.2 Representation of detritus in the sediment

Detritus in the sediment is represented just in the same two-fold way as detritus in the water column is. A state variable for sedimentary detritus is contained in the array `suspsed(:)`, in this case, as 2d-variable. Time tendency representing physical processes (erosion and burial) apply to this the state variable.

To allow biological processes to access the sedimentary detritus state variables, a variable `sed` of type `sediment` is defined. Time tendency representing biological processes in the sedimentary detritus apply to `suspsed(sed%index_sed)%f_sed(i,j,1)` The index 1 in the last parentheses indicates that biological processes only influence the uppermost sediment layer. The variable `sed` contains all parameters controlling the biological processes in the sediment.

The parameter `sed%index_sed` must not be defined manually, instead, the namelist parameter `name_sed` shall have the same value as a parameter `name_2d_suspsed(n)`. This establishes the correspondence between `sed` and the sedimentary detritus state variable found under index n in the array `suspsed(:)`.

A state variable for sedimentary iron phosphate is also required for ERGOM.

The parameters which must be defined and can be modified via namelist are summarized in Tab. 2.

Table 2: **Namelist parameters controlling mineralisation in the sediment**

Name in namelist	Name in code	Description
Unit in namelist	Unit in code	
name_sed	name_sed	Variable name of the sedimentary detritus tracer. The same name needs to occur in the array name_2d_suspsed(:)\verb .
name_ips	name_ips	Variable name of the sedimentary iron phosphate tracer. The same name needs to occur in the array name_2d_suspsed(:)\verb .
dn_sed\verb 1/d	dn\verb 1/s	sediment detritus recycling rate at 0^\circC
q10_rec_sed	frac_dn_anoxic	ratio between recycling rate under oxic or anoxic conditions
q10_rec_sed\verb 1/°C	q10_rec\verb 1/°C	q10 parameter for recycling
thio_bact_min\verb mol/m ²	thio_bact_min\verb mol/m ²	minimum amount of nitrogen (organic matter) in active sediment for thiomargarita colonies
np_sed/nn_sed	pnr	P/N ratio
nc_sed/nn_se	cnr	C/N ratio
po4_lib_rate\verb 1/d	po4_lib_rate\verb 1/s	liberation rate for iron phosphate
po4_retention	po4_retention	fraction of phosphorous retained in the sediment while recycled
po4_ret_plus_BB	po4_ret_plus_BB	value added to po4_retention north of 60.75°N (special treatment for the Bothnian Bay to suppress cyanobacterial blooms)
o2_bioerosion\verb mol/kg	o2_bioerosion\verb mol/kg	oxygen threshold to enable bio-erosion
den_rate	den_rate	proportion of denitrification at the sediment redox-cline

B The ERGOM namelist - a summary

Here all model parameters are summarized that can be modified via the ERGOM namelist. Not all parameters should be changed, since not all changes will meet code ready for these changes. ERGOM is ready to use the `fied_table` for parameter changes too. But this is not tested yet.

```
namelist /ergom_nml/
```

```
dimensions of tracer arrays
```

NUM_PHYTO	Number of phytoplankton groups. Default is 3 for diatoms, flagellates and diazotrophs. Adding more requires code changes.
NUM_ZOO	Number of zooplankton groups. Unlimited. Differ in migration, growth rate and grazing preferences.
NUM_DET	Number of detritus variables. Differ in sinking and mineralisation rates.
NUM_SPM	Number of suspended particulate tracer variables undergoing sinking and settling. Note, phytoplankton and zooplankton is not in this number.
NUM_SED	Number of sediment tracers.
NUM_SEDIMENT_LAYERS	Number of sediment layers. Two is recommended.

phytoplankton parameters

name_phyt		name of phytoplankton variable
imin	W/m ²	minimum light saturation for phytoplankton growth
tmin	°C	minimum growth temperature for phytoplankton
smin		minimum growth salinity for phytoplankton
smax		maximum growth salinity for phytoplankton
alpha	mol/kg	half-saturation constants of nutrient uptake by phytoplankton
talpha	°C	Michaelis Menton-like temperature
rp0	1/d	growth rates of phytoplankton
p0	mol/kg]	background concentration for initial phytoplankton growth
np		number of P atoms in uptake, Redfield ratio
nn		number of N atoms in uptake, Redfield ratio
nc		number of C atoms in uptake, Redfield ratio
lpd	1/d	phytoplankton loss rates to detritus
lpr	1/d	phytoplankton respiration rates to nh4
sinkp	m/d	phytoplankton sinking velocities, usually for diatoms. Diazotrophs may raise.

h1>zooplankton parameters

name_zoo		name of zooplankton variable
sinkz	m/d	zooplankton sinking velocities
risez	m/d	zooplankton rise velocities
vdiff_max	m ² /s	zooplankton maximum enhanced diffusion
wfood		weight for food gradients in zooplankton rise velocity
o2min	mol/kg	minimum oxygen concentration where migration stops
h2smax	mol/kg	maximum h2s concentration where migration stops
wtemp		weight number for temperature sensitivity
wo2		weight number for o2 sensitivity
wh2s		weight number for h2s sensitivity
t_opt_zoo	°C	temperature optimum of zooplankton for grazing
t_max_zoo	°C	temperature maximum of zooplankton for grazing
beta_zoo		parameter for temperature dependence of grazing
oxy_sub_zoo	mol/kg	oxygen level below which reduced respiration starts
oxy_min_zoo	mol/kg	oxygen level below which no respiration takes place
resp_red_zoo		reduction factor for respiration under suboxic conditions
sigma_b	1/d	zooplankton loss rate to detritus
nue	1/d	zooplankton respiration rate to nh4
food_to_nh4		fraction of eaten food that is directly lost to respiration
food_to_det		fraction of eaten food that is directly lost to detritus
food_to_nh4_2		fraction of food eaten potentially, directly lost to respiration
food_to_det_2		fraction of food eaten potentially, directly lost to detritus
iv	kg/mol	Ivlev parameter for zooplankton grazing
zcl1	kg/mol	closure parameter for zooplankton
graz	1/d	zooplankton maximum grazing rate
z0	mol/kg	background concentration for initial zooplankton growth
Imax	W/m ²	maximum light for zooplankton activity
vertical_migration		logical to enable zooplankton vertical migration
blanchard_temperature		.false.: old ERGOM temperature dependence, .true.: Blanchard 1996 formula
pref_phy	0-1	preferences of zooplankton for phytoplankton
pref_zoo	0-1	preferences of zooplankton for zooplankton
pref_det	0-1	preferences of zooplankton for detritus
graz_pref		flag to select grazing preference method

detritus parameters

name_det		name of detritus, must be the same like one suspended particulate matter (spm) variable name
dn	1/d	recycling rate
q10_rec	°C	q10 parameter for recycling of detritus

generic_ERGOM_type parameters

nf	1/d	nitrification rate
q10_nit	°C	q10 parameter for nitrification
alpha_nit	mol/kg	half-saturation constant for nitrification
q10_h2s	°C	q10 parameter for chemolithotrophs (h2s oxidation)
k_h2s_o2	kg/mol/d	reaction constant h2s oxidation with o2
k_h2s_no3	kg/mol/d	reaction constant h2s oxidation with no3
k_sul_o2	kg/mol/d	reaction constant sulfur oxidation with o2
k_sul_no3	kg/mol/d	reaction constant sulfur oxidation with no3
k_an0	1/d	maximum anammox rate
k_DN		detritus recycling fraction (denitrification) in the water column
k_DS		detritus recycling fraction with sulphate reduction in the water column

biological sediment parameters

name_redfield_sed		ERGOM requires one sediment variable with Redfield C/N/P
name_iron_phosphate		ERGOM requires iron phosphate
dn_sed	1/d	recycling rate of detritus in the sediment
frac_dn_anoxic		fraction of recycling rate in shallow sediments for anoxic bottom water
thio_bact_min	mol/m ² 2	minimum nitrogen content of active sediment for thiomargarita
q10_rec_sed	°C	q10 paramter for recycling of detritus in the sediment
den_rate		proportion of denitrification at the sediment redoxcline
np_sed		number of P atoms in the sediment, Redfield ratio
nn_sed		number of N atoms in the sediment, Redfield ratio
nc_sed		number of C atoms in the sediment, Redfield ratio
po4_lib_rate	1/d	liberation rate of iron phosphate in the sediment
po4_retention		fraction of phosphorous retained in the sediment while recycled
po4_ret_plus_BB		value added to po4_retention north of 60.75°N (special treatment for the Bothnian Bay to suppress cyanobacterial blooms)
o2_bioerosion	mol/kg	oxygen threshold to enable bio-erosion

suspended particulate matter parameters

name_spm		name of spm tracer
longname_spm		long name for output
wsink0_spm	m/d	sinking velocity ($\neq 0$ for sinking)
wsed_spm	m/s	sedimentation rate
sediment_to		name of 2d-tracer to which sedimentation takes place

settled matter parameters

name_sed		name of sed tracer
longname_sed		long name for output
molar_volume_sed	m ³ /mol	specific volume of this sediment type
critical_stress_sed	N/m ²	critical shear stress when erosion starts
erosion_rate_sed	m/d	erosion rate
bioerosion_rate_sed	m/d	erosion rate by benthic animals
suspend_to		name of 3d-tracer to which resuspension takes place

sediment parameters

sed_layer_height		maximum height of vertical layers [m]. $\neq 0$ means the layer may become infinitely thick.
sed_layer_propagation		Sediment layer propagation mode, SLP_DOWNWARD=1, SLP_FULL_BOX=2, SLP_OLD_ERGOM=3
sed_erosion_mode		Sediment erosion mode, INDEPENDENT=1, MAXSTRESS=2, ORGANIC=3

other parameters

vlev_sed		number of 2d-tracers that may be stored in one diagnostic 3d-tracer
----------	--	---

C Bottom boundary layer model of combined current and wave action on bottom drag and skin friction

contributed by Torsten Seifert, Leibniz-Institute for Baltic Sea Research

D Basic relations

Friction at the rough sea bottom decelerates the magnitude of currents by transferring momentum from the moving water body to the resting ground. The corresponding momentum flux is usually described by a quadratic approach, relating the bottom shear stress $\vec{\tau}_b$ to the current velocity \vec{u} via a drag coefficient C_D

$$\vec{\tau}_b = \rho_w C_D |u| \vec{u}, \quad (1)$$

where ρ_w is the density of water. Assuming a constant flux of momentum, deceleration establishes a logarithmic boundary layer within which the current of the water body drops down from the free flow velocity u_c to zero according to the law-of-the-wall

$$u_c(z) = \frac{u_{*c}}{\kappa} \ln \left(\frac{z}{z_o} \right). \quad (2)$$

Here, z describes the height above the sea bottom and $\kappa = 0.4$ is von Karman's constant. The shear stress of Eq. (1) is expressed by the so-called friction velocity u_* according to

$$|\tau_b| = \rho_w u_*^2 \quad (3)$$

The roughness length z_o is a length scale describing the impact of the rough sea bed. z_o includes all decelerating effects at the bottom, i. e. adhesive forces as well as momentum transfer connected with small-scale turbulence induced by roughness elements at the sea bottom as single grains or structures like ripples. Therefore, z_o might be related to geometric characteristics like the median of the grain size distribution d_{50} and the height η and distance λ of ripples. The sum of grain and ripple roughness is the “natural choice” for z_o , however, it has to be specified for a certain bottom boundary layer model which roughness effects are included, see the three-layer approach below.

A relation between the bottom drag coefficient C_D and the roughness length z_o follows from Eqs. (1-3)

$$C_D(z, z_o) = \left(\frac{u_*}{u_c(z)} \right)^2 = \left(\frac{\kappa}{\ln(z/z_o)} \right)^2. \quad (4)$$

Obviously, the drag coefficient is not a constant like z_o but changes with the height z above ground. It has to be noted, that table values of C_D refer to observations scaled to a reference height of $z_{ref} = 1 \text{ m}$ usually. These data yield equivalent roughness lengths of

$$z_o = z_{ref} \exp \left(-\kappa / \sqrt{C_D(z_{ref})} \right). \quad (5)$$

This relation reveals that the “standard value” for the drag coefficient $C_D = 0.0025$ corresponds to a roughness length of $z_o = \exp(-8)m$ which characterizes fine sand with $d_{50} \approx 120\mu m$ if $z_o = 2.75d_{50}$ is assumed including grain roughness of $d_{50}/12$ and form drag by ripples of $(8/3)d_{50}$, see section F.

E Three-layer bottom boundary model

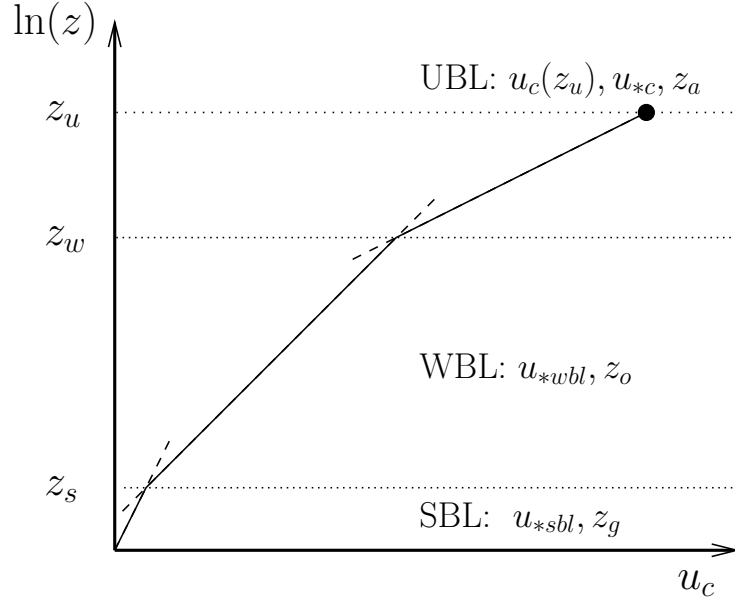


Figure 4: Sketch of the three-layer bottom boundary. In the upper boundary layer (UBL) the current velocity u_c is known at a height z_u above bottom and the logarithmic velocity profile is characterized by the friction velocity u_{*c} and the apparent roughness length z_a . The wave boundary layer (WBL), starting at depth z_w , is determined by a combined current-wave friction velocity u_{*wbl} and the bottom roughness z_o , which includes form drag induced by ripples at the sea bottom. The skin friction layer of thickness z_s is characterized by u_{*sbl} and grain roughness z_g .

In order to take into account the combined impact of currents and surface waves on the momentum flux to the sea bed and to find the shear stress which might cause movement and resuspension of sediment particles the bottom boundary is subdivided into three layers, see Fig. (4)

$$\begin{aligned}
 UBL \quad u_c(z) &= (u_{*c}/\kappa) \ln(z/z_a), & z_w \leq z \\
 WBL \quad u_c(z) &= (u_{*wbl}/\kappa) \ln(z/z_o), & z_s \leq z \leq z_w \\
 SBL \quad u_c(z) &= (u_{*sbl}/\kappa) \ln(z/z_g), & z \leq z_s
 \end{aligned} \tag{6}$$

According to this approach the logarithmic velocity profile is determined by an apparent bottom roughness z_a within the upper boundary layer (UBL). The corresponding momentum

flux, which is withdrawn from the current field at a height $z \geq z_w$, is expressed by the friction velocity u_{*c} or alternatively by an effective bottom drag coefficient $C_D(z, z_a)$, see Eq. (4) and Eq. (1).

The apparent roughness is caused by the underlying wave boundary layer (WBL) where the oscillations of the water body which are induced by surface waves enhance the level of small-scale turbulence. The turbulent structures react as additional roughness elements to the water layers above. The combined action of currents and waves is approximated after Grant and Madsen [1979] by

$$u_{*wbl} = u_{*c}^2 / u_{*cw} \quad \text{with} \quad u_{*cw} = \sqrt{u_{*c}^2 + u_{*w}^2}. \quad (7)$$

The wave induced contribution to the friction velocity is be calculated using a parameterization of Nielsen [1992] by

$$u_{*w}^2 = f_w U_w^2 / 2 \quad \text{with} \quad f_w = \min \left\{ \exp \left[5.5 \left(\frac{30z_o}{A_w} \right) - 6.3 \right], 0.3 \right\}, \quad (8)$$

where $A_w = (H/2) / \sinh(kh)$ denotes the damped amplitude of a surface wave of height H and wave number k in water depth h . The maximum velocity of wave motion in the same depth is $U_w = A_w \omega$. The wave angular frequency is determined by the dispersion relation $\omega = \sqrt{gk \tanh(kh)}$ with g the gravity of earth. It is assumed that wave characteristics are known by the significant wave height H_{sig} and the corresponding peak frequency ω_p and wave number k_p of the wave spectrum.

The shear stress acting on particles at the surface of the sea bed is derived within a thin skin friction sublayer (SBL) located at $z \leq z_s$ where the logarithmic velocity profile is determined by the grain roughness z_g only. There $u_{*sbl} = u_{*sc}$ is considered as the current induced friction velocity. The contribution of waves u_{*sw} is calculated according to Eq. (8) using the grain roughness z_g instead of z_o thus leading to a total skin friction of

$$u_{*s} = \sqrt{u_{*sc}^2 + u_{*sw}^2}. \quad (9)$$

The three-layer bottom boundary model leads to a unique solution for the three unknowns z_a , u_{*c} , and u_{*sc} , if the current velocity u_c is known at $z_u > z_w$ in the upper boundary layer. Rearranging the first equation of Eq. (6) yields

$$u_{*c} = \kappa u_c(z_u) / \ln(z_u / z_a). \quad (10)$$

Applying the parameterization of Grant and Madsen [1979] for the thickness of the wave boundary layer

$$z_w = 2\kappa u_{*cw} / \omega, \quad (11)$$

matching the logarithmic profiles at this interface leads to

$$z_a = z_o (z_w / z_o)^\beta = z_o (2\kappa u_{*cw} / \omega z_o)^\beta \quad \text{with} \quad \beta = 1 - u_{*c} / u_{*cw}. \quad (12)$$

Finally, matching to the skin boundary layer at z_s results in

$$u_{*sc} = \frac{u_{*c}^2 \ln(z_s/z_o)}{u_{*cw} \ln(z_s/z_g)}. \quad (13)$$

According Smith and McLean [1977] the thickness of the skin layer is determined by the grain roughness z_g and the distance λ between ripples at the sea bottom

$$z_s = 0.09 z_g (\lambda/z_g)^{0.8}. \quad (14)$$

A magnitude-of-order estimate shows that both sublayers are well below $1m$ as long as $u_{*cw}/\omega \ll 1m$ and $\lambda/z_g \ll 10^5$ which is fulfilled for typical values of $u_{*cw} \sim 0.1m/s$, $\omega \gtrsim 1/s$, $\lambda \sim 0.1m$ and $z_g \sim 10^{-5}m$. This justifies the application of the present approach to numerical models with layer thickness $\Delta z \geq 1m$.

In order to find a consistent solution, an iteration has to be applied to solve Eq. (10) and Eq. (12) which are interdependent by z_a and u_{*c} . This is easily done for the effective bottom drag coefficient $C_D(z_u, z_a)$ according to Eq. (4) in the following way: Starting from $z_a = z_o$ as initial guess the equations for u_{*c} , u_{*cw} , z_a and C_D are repeatedly solved until a prescribed accuracy is reached. For the sake of clearness the calculation is indicated in the following using the subscript n for the iterative level and resulting in the effective bottom drag according to Eq. (4)

$$\begin{aligned} C_{D,0} &= [\kappa / \ln(z_u/z_o)]^2, \\ u_{*c,n} &= u_c(z_u) \sqrt{C_{D,n-1}}, \\ u_{*cw,n} &= \sqrt{u_{*c,n}^2 + u_{*w}^2}, \\ z_{a,n} &= z_o [2\kappa u_{*cw,n} / \omega z_o]^{(1-u_{*c,n}/u_{*cw,n})}, \\ C_{D,n} &= [\kappa / \ln(z_u/z_{a,n})]^2, \\ \text{until} &: |(C_{D,n} - C_{D,n-1})/C_{D,n}| < \varepsilon. \end{aligned} \quad (15)$$

The iteration above converges quickly for comparable contributions from current and waves. A relative accuracy of $\varepsilon = 10^{-3}$ is achieved after 2 – 7 repetitions.

F Solution limits

It is obvious from Eq. (12) that the apparent roughness z_a tends to z_o in case of vanishing wave action because of $u_{*cw} \rightarrow u_{*c}$ and $\beta \rightarrow 0$. This confirms that $z_a = z_o$ is the natural choice for the initial guess of the iteration, see Eq. (15). Consequently, also the effective bottom drag approaches the value defined by the prescribed bottom roughness z_o , but scaled to the actual height above bottom z_u , compare Eqs. (4) and (15)

$$C_{D,nowave} = (\kappa / \ln(z_u/z_o))^2. \quad (16)$$

The wave boundary layer thickness z_w of Eq. (11) is diminishing for weak currents and weak waves, which imply $H_{sig} \rightarrow 0$, $\omega \sim 1/\sqrt{(H_{sig})}$, and $u_{*cw} \rightarrow 0$. In the strict sense, the wave boundary layer disappears if $z_w \rightarrow z_s$, however, this case needs not to be considered explicitly, because the solution for the drag coefficient is practically the above mentioned limit.

The resulting skin friction in the no-wave limit is easily, see Eqs. (9), (10), and (13)

$$u_{*s,nowave} = u_{*sc} = \frac{\kappa u_c(z_u)}{\ln(z_u/z_o)} \frac{\ln(z_s/z_o)}{\ln(z_s/z_g)}. \quad (17)$$

The first term is equal to the total momentum flux corresponding to $u_{*c,nowave}$ which is transferred from the current to the sea bottom. The second term defines that part which acts as a shear stress on grains at the bottom. It is determined by the relation between the bottom roughness z_o , the grain roughness z_g and the distance of ripples λ , which cause form drag. The second term reduces to a constant if these parameters are fixed, assuming an invariable type of sedimentary material, see below.

Finally, some remarks to the relationships between ripples, roughness, and grain size. The letter is usually specified by the median value of the grain size distribution d_{50} . As already mentioned at the end of section D the grain roughness is generally set to $z_g = d_{50}/12$. After Xu and Wright [1995] the bottom roughness may be approximated by three parts $z_o \approx z_g + z_f + z_t$, where z_f denotes form drag induced by structures like ripples, and z_t stands for the contribution by sediment transport, which is ignored here. It would play a role if very large amounts of sedimentary material are suspended in the water column above bottom. The form drag roughness might be estimated by a parametrization of Nielsen [1983] $z_f = (8/30)\eta(\eta/\lambda)$, where η and λ denote the mean height of ripple crests and the distance in-between. Following Yalin [1977] the ripple characteristics again may be related to the grain size by $\lambda \simeq 1000d_{50}$. Assuming a ripple steepness of $\eta/\lambda \simeq 0.1$ leads to $z_f = (8/3)d_{50}$, $z_o = z_g + z_f = 33z_g$ and $z_s \simeq 165z_g = 5z_o$, see Eq. (14). Consequently, the logarithmic term in Eqs. (13) and (17) reduces to a constant of $\ln(5)/\ln(165) \simeq 0.3152$ defining the relative part of momentum flux which comes into effect for sediment transport.

However, the roughness and ripple parameters have to be explicitly time-stepped if mixtures of different grain size classes are to be considered taking into account changing ripples

Acknowledgment

The authors wish to acknowledge use of the Ferret program for analysis and graphics in this paper. Ferret is a product of NOAA's Pacific Marine Environmental Laboratory. (Information is available at www.ferret.noaa.gov)

The title figure shows saline water spreading through the basins of the Baltic Sea during a major inflow event. The figure was produced by J. König (IOW) using model results from a 1 n.m. Baltic Sea model based on MOM.

References

- G. Nausch Feistel, R. and N. Wasmund, editors. State and evolution of the Baltic Sea, 1952-2005. A detailed 50-year survey of meteorology and climate, physics, chemistry, biology, and marine environment. Hoboken: Wiley-Interscience, 2008.
- T. Neumann, W. Fennel, and Ch. Kremp. Experimental simulations with an ecosystem model of the Baltic Sea: A nutrient load reduction experiment. Global Biogeochemical Cycles, 16:7–1 – 7–19, 2002.
- T. Neumann and G. Schernewski. Eutrophication in the baltic sea and shifts in nitrogen fixation analyzed with a 3d ecosystem model. Journal of Marine Systems, 74:592–602, 2008. doi: 0.1016/j.jmarsys.2008.05.003.
- W. Fennel and Th. Neumann. Introduction to the Modelling of Marine Ecosystems. Elsevier, Amsterdam, 2004.
- W. Fennel, T. Seifert, and B. Kayser. Rossby radii and phase speeds in the baltic sea. Continental Shelf Research, 11:23–36, 1991.
- T. Seifert and B. Kayser. A High Resolution Spherical Grid Topography of the Baltic Sea, 1995.
- T. Seifert, F. Tauber, and B. Kayser. A high resolution spherical grid topography of the baltic sea - revised edition. http://www.io-warnemuende.de/admin/de_index.html, 2001.
- M. Herzfeld, M. Schmidt, S. M. Griffies, and Z. Liang. Realistic test cases for limited area ocean modelling. Ocean Modelling, January 2011. ISSN 14635003. doi: 10.1016/j.ocemod.2010.12.008. URL <http://dx.doi.org/10.1016/j.ocemod.2010.12.008>.
- F. Janssen, T. Neumann, M. Schmidt. Inter-annual variability in cyanobacteria blooms in the baltic sea controlled by wintertime hydrographic conditions. Marine ecology. Progress series, 275:59–68, 2004.
- W. G. Large, J. C. McWilliams, and S. C. Doney. Oceanic vertical mixing: A review and a model with a nonlocal boundary layer parameterization. Review of Geophysics, 32:363 – 403, 1994.
- W. G. Large, G. Danabasoglu, S. C. Doney, and J. C. McWilliams. Sensitivity to surface forcing and boundary layer mixing in a global ocean model: annual-mean climatology. JPO, 27:2418–2447, 1997.

- N. G. Jerlov. Optical Oceanography. Elsevier, 1968.
- S. Bodin. A predictive numerical model of the atmospheric boundary layer based on the turbulent energy equation. Technical report, Swedish Meteorological and Hydrological Institute, SE-60176, Norrköping, Sweden, 1979.
- S.M. Uppala, P.W. Kållberg, A.J. Simmons, U. Andrae, V. da Costa Bechtold, M. Fiorino, J.K. Gibson, J. Haseler, A. Hernandez, G.A. Kelly, X. Li, K. Onogi, S. Saarinen, N. Sokka, R.P. Allan, E. Andersson, K. Arpe, M.A. Balmaseda, A.C.M. Beljaars, L. van de Berg, J. Bidlot, N. Bormann, S. Caires, F. Chevallier, A. Dethof, M. Dragosavac, M. Fisher, M. Fuentes, S. Hagemann, E. Hólm, B.J. Hoskins, L. Isaksen, P.A.E.M. Janssen, R. Jenne, A.P. McNally, J.-F. Mahfouf, J.-J. Morcrette, N.A. Rayner, R.W. Saunders, P. Simon, A. Sterl, K.E. Trenberth, A. Untch, D. Vasiljevic, P. Viterbo, and J. Woollen. The ERA-40 re-analysis. Quart. Journal Royal Meteorological Society, 131:2961–3012, 2006. 2961-3012.doi:10.1256/qj.04.176.
- M. E. Berliand and T. G. Berliand. Determining the net long-wave radiation of the Earth with consideration of the effect of cloudiness. Number 1 in Ser. Geofis. Akad. Nauk. SSSR, 1952.
- K. Ya. Kondratyev. Radiation in the Atmosphere. Academic Press, 1969.
- D. J. Schwab, J. R. Bennett, P. C. Liu, and M. A. Donelan. Application of a Simple Numerical Wave Prediction Model to Lake Erie. Journal of Geophysical Research, 89 (C3):3586–3592, 1984.
- P.C. Liu D.J. Schwab, J.R. Bennett and M.A. Donelan. Application of a simple numerical wave prediction model to lake erie. Journal of Geophysical Research, 89 (C3):3586, 1984.
- D. J. Schwab Liu, P. C. and J. R. Bennett. Comparison of a two-dimensional wave prediction model with synoptic measurements in lake michigan. Journal of Physical Oceanography, 14:1514, 1984.
- S. A. Hughes. TMA shallow-water spectrum: Description and application. Technical Report CERC-84-7, US Army Engineer Waterways Experiment Station, Vicksburg, Miss., 1984.
- E. Bouws, H. Guenther, W. Rosenthal, and C. L. Vincent. Similarity of the wind wave spectrum in finite depth water: 1. Spectral form. Journal of Geophysical Research, 90 (C1):975–986, 1985. TMA-Spektrum.
- E. Bouws, H. Guenther, W. Rosenthal, and C.L. Vincent. Similarity of the wind wave spectrum in finite depth water: 2. Statistical relationships between shape and growth stage parameters. Deutsche Hydrographische Zeitschrift, 40:1–24, 1987. TMA-Spektrum.
- Hernan E. Garcia and Louis I. Gordon. Oxygen solubility in seawater: Better fitting equations. Limnology and Oceanography, 37:1307–1312, 1992.

- Roberta C Hamme and Steven R. Emerson. The solubility of neon, nitrogen and argon in distilled water and seawater. Deep-Sea Research Part I-Oceanographic Research Papers, 51(11):1517 – 1528, 2004. doi: 10.1016/j.dsr.2004.06.009.
- Keeling, R. F., Stephens, B. B., Najjar, R. G., Doney, S. C., Archer, D. and M. Heimann. Seasonal variations in the atmospheric o_2/n_2 ratio in relation to the kinetics of air-sea gas exchange. Global Biogeochemical Cycles, 12:141–163, 1998. doi: 10.1029/97GB02339.
- W. D. Grant and O. S. Madsen. Combined Wave and Current Interaction With a Rough Bottom. Journal of Geophysical Research, 84 (C4):1797–1808, 1979.
- P. Nielsen. Coastal Bottom Boundary Layer and Sediment Transport. Advanced Series on Ocean Engineering, World Scientific Edition, 4, 1992.
- J. D. Smith and S. R. McLean. Spatially Averaged Flow Over a Wavy Surface. Journal of Geophysical Research, 82 (12):1735–1746, 1977.
- J. P. Xu and L. D. Wright. Tests of Bed Roughness Models Using Field Data from the Middle Atlantic Bight. Continental Shelf Research, 15 (11/12):1409–1434, 1995.
- P. Nielsen. Analytical Determination of Nearshore Wave Height Variation due to Refraction Shoaling and Friction. Coastal Engineering, 7:233–251, 1983.
- M. S. Yalin. Mechanics of Sediment Transport. Pergamon Press, New York:298 pp, 1977.