

# **User guide for the CSIRO Environmental Modelling Suite (EMS): Optical, biogeochemical, and carbon chemistry models.**

**Mark Baird, Mathieu Mongin, Karen Wild-Allen, Barbara Robson, Jennifer Skerratt, Farhan Rizwi, Monika Soja-Wozniak, Anna Maggiorano, Lachlan Phillips**

Public GitHub Version 1.3.2

Local Github Version (13<sup>th</sup> August 2025)

16<sup>th</sup> August 2025

### **Original citation**

Baird ME, Mongin M Wild-Allen KA Robson B Skerratt J. Rizwi F. 2018. User guide for the CSIRO Environmental Modelling Suite (EMS): Optical, biogeochemical, and carbon chemistry models. CSIRO Hobart.

### **Copyright**

© Commonwealth Scientific and Industrial Research Organisation 2018. To the extent permitted by law, all rights are reserved and no part of this publication covered by copyright may be reproduced or copied in any form or by any means except with the written permission of CSIRO.

### **Further information, acknowledgments and code access**

The code is available from: <https://github.com/csiro-coasts/EMS>. All conditions for the use of the code can be found on the GitHub archive, as well as acknowledgement of funding sources and authors of the CSIRO EMS code. Thank you

# Contents

1	Introduction .....	1
2	Code availability.....	3
3	Model configuration files.....	4
	3.1 Master configuration file .....	4
	3.2 The process list .....	7
	3.3 The parameter file .....	10
4	Temperature dependence .....	13
5	Sediment – water column exchange rates .....	14
6	Optical model configuration .....	15
	6.1 Original spectrally-resolved model.....	15
	6.2 Specification of CDOM absorption .....	18
	6.3 Column-based light calculations .....	18
	6.4 Outputs of optical variables .....	27
	6.5 Moonlight .....	28
	6.6 Colour visualisation. ....	29
7	Carbon chemistry model .....	30
8	Ecological integration options .....	31
9	Pre-processing tools for BGC .....	33
10	Running simulations .....	34
	10.1 Fully-coupled mode .....	34
	10.2 Off-line transport mode .....	34
11	Computational diagnostics .....	36
	11.1 Mass balance .....	36
12	RECOM / test case .....	38

# Figures

Figure 1 The EMS model suite used for the eReefs project..... 2

Figure 2 Schematic illustrating the vertical grid of the model and the locations on this grid that the optical properties have been calculated. .... 28

# Tables

Table 1 Standard process lists..... 10

Table 1 Standard set of parameter values ..... 12

Table 2 Simulated observations..... 17

# 1 Introduction

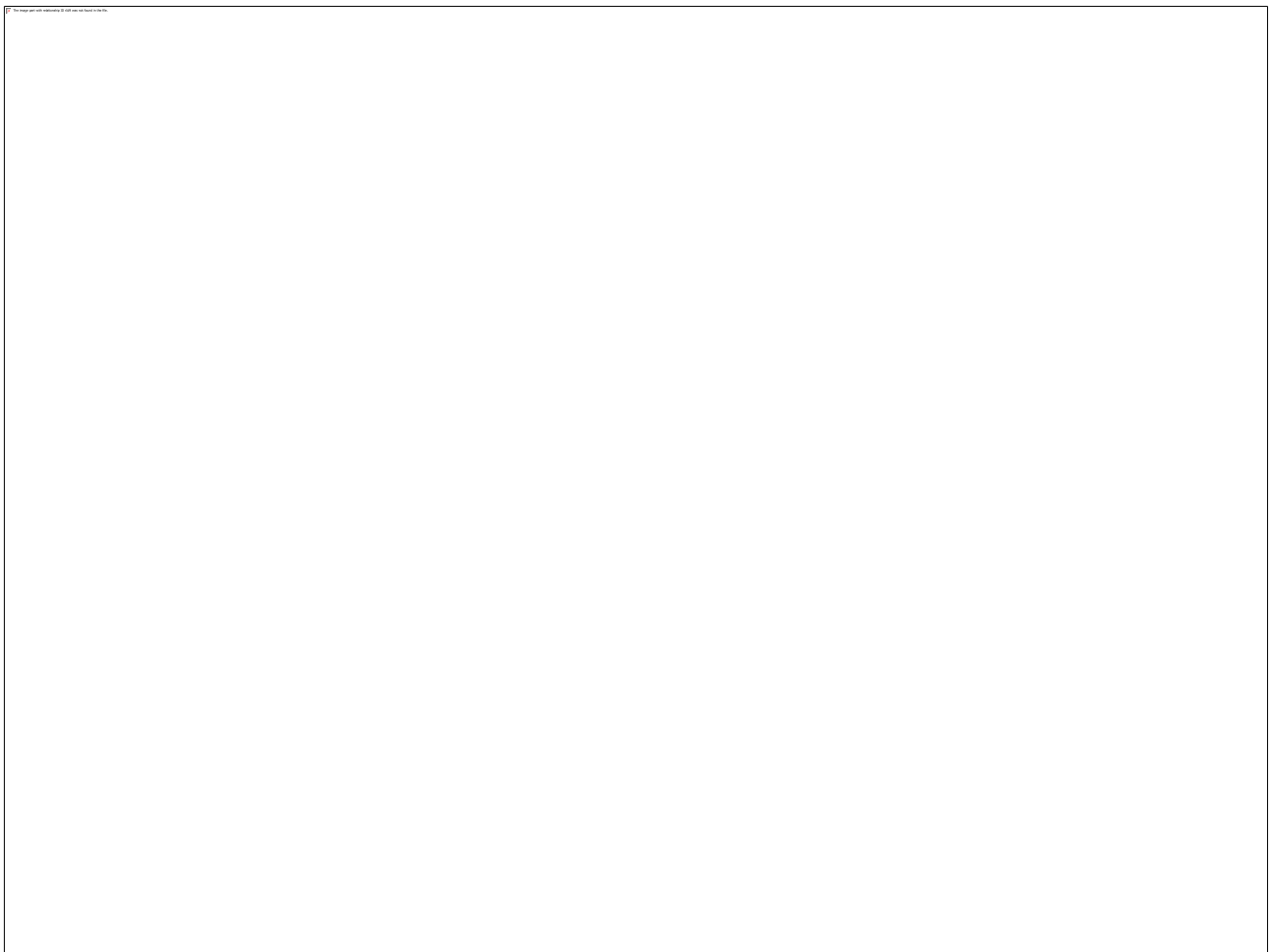
The CSIRO Environmental Modelling System (EMS) is a modelling tool that allows for the coupling of hydrodynamic / sediment / optical / biogeochemical (BGC) processes. It has been designed to represent shallow water systems, where processes at the sediment interface and below impact the water column, and vice versa, but also works well in deep waters.

One feature of EMS is the ability to include / exclude biogeochemical processes using the same model executable (or version), but different input configuration files. For example, the same executable can run a model of a tropical region with coral reefs, as well as a temperate region with multiple seagrass species or to just model the optical properties of the water column. The inclusion of model processes is therefore an important component of the model configuration, and makes up a significant portion of this User Guide.

This documentation is designed to be used only for user options that are not detailed in the much more extensive SHOC / COMPAS user guide (<https://research.csiro.au/cem/software/ems/ems-documentation/>). In particular, details regarding model running, model grids, advection and diffusion schemes, boundary conditions, as well as input / output formats are, are all documented in the SHOC / COMPAS user guide.

For developers considering adding new ecological processes, the original description of the ecological module (The “new” ecological model: user guide by Pavel Sakov) is available on the documentation page.

For more information on the model equations, please see the [Biogeochemical Scientific Description for version B3p0](#).



**Figure 1** The EMS model suite used for the eReefs project.

## 2 Code availability

The source code, and conditions of use, is available from a GitHub archive:

<https://github.com/csiro-coasts/EMS>.

Model documentation (including the latest version of this document) are available at:

<https://research.csiro.au/cem/software/ems/ems-documentation/>

## 3 Model configuration files

EMS is run from a compiled executable (see Section 10) that points to a number of configuration files, all of which are text files. The “Master” configuration is called in the execution line, and contains the names of any other files that are called.

For convenience, instead of using the filename suffix `.txt`, we use the following (noting they are still text files):

- `.prm` files are text files containing the model parameters.
- `.tran` files are text files used to run the off line transport model.
- `.pss` files are text files that list the location of the tracer point source.
- `.ts` files are text which contains time series of data of model input or outputs.
- `.mnc` files are text file listing source of multiple netcdf files.

Within these files any line starting with `#` is a comment line.

### 3.1 Master configuration file

This is the main file to run the model it contains:

- model run time
- model output and restart files
- hydrodynamic flags and parameters
- grid & bathymetry definition
- model tracer list definition
- model forcing
- model time series outputs
- model boundaries
- vector of bathymetry
- x grid
- y grid

See Section 8 for detailed ecology options.

#### 3.1.1 Tracer list

The tracer list contains all the prognostic and diagnostic tracers that are available for output from the model. Most of the flags are self-explanatory, but the “diagn” is subtle.

A prognostic tracer has a tracer flag `TRACER.diagn` of 0, while a diagnostic tracer can take a value of 1 or 2. Diagnostic variables not altered by advection, diffusion or sediment processes.

In the `ECOLOGY` model, if `diagn = 1` (flux diagnostic), then the diagnostic is the mean flux for the ecological time period (obtained by taking the change in the tracer from the initial value to the end and dividing by the timestep.). If `diagn = 2` (mass diagnostic), then the diagnostic is the value of the tracer at the end of the time period.



In the rest of the code (hydro, sediments) diagn 1 and 2 simply the value at the output time, but are not advected or diffused. Therefore, tracers which are never given an increment by the ecology, such as TSS, the use of diagn = 1 by the sediment model is not a problem.

#### Example prognostic hydrodynamic tracer:

```
TRACER0.name          salt      # SHOC variable name.
TRACER0.long_name     Salinity  # long name in netcdf
TRACER0.units         PSU
TRACER0.type          WATER SEDIMENT
TRACER0.fill_value    35.0 # default value initial value.
TRACER0.valid_range   0.0 40.0 # gives warning if outside range
TRACER0.advect        1        # this tracer is advected
TRACER0.diffuse        1        # this tracer is diffused
TRACER0.diagn         0        # this tracer is not a diagnostic
tracer
TRACER0.data          data/temp.nc      # if present, overwrite the
                                     fill_value, the data in temp.nc are interpolated onto
                                     the model grid (see SHOC user guide for specification
                                     of temp.nc)
```

#### Example prognostic dissolved BGC tracer:

```
TRACER23.name DIP
TRACER23.long_name Dissolved Inorganic Phosphorus
TRACER23.units mg P m-3
TRACER23.valid_range_wc 0 1e+35
TRACER23.valid_range_sed 0 1e+35
TRACER23.diagn 0
TRACER23.type          WATER SEDIMENT
TRACER23.fill_value_wc 0.5
TRACER23.fill_value_sed 100 # fill value in the sediment, if present,
                             overwrite the value_wc in the sediment
TRACER23.inwc 1
TRACER23.insed 1
TRACER23.dissol 1 # dissolved tracer
TRACER23.partic 0
TRACER23.advect 1
TRACER23.diffuse 1
TRACER23.decay 0
TRACER23.svel 0 # sinking rate
TRACER23.resusp_rate 0
```

#### Example prognostic particulate BGC tracer:

```
TRACER33.name DetR_P
TRACER33.long_name Refractory Detrital Phosphorus
TRACER33.units mg P m-3
TRACER33.valid_range_wc 0 1e+35
TRACER33.valid_range_sed 0 1e+35
```

```

TRACER33.diagn 0
TRACER33.type          WATER SEDIMENT
TRACER33.fill_value_wc 0.01
TRACER33.fill_value_sed 1.1e+03
TRACER33.inwc 1
TRACER33.insed 1
TRACER33.dissol 0
TRACER33.partic 1  # particulate tracer altered by sediment code
TRACER33.advect 1
TRACER33.diffuse 1
TRACER33.decay 0

```

# the following parameter are used in the sediment code to do sinking and resuspension

```

TRACER33.psize 1e-05  # size
TRACER33.b_dens 1e+09  # density
TRACER33.i_conc 2e+08
TRACER33.f_conc 2e+08
TRACER33.svel -2.31e-05
TRACER33.crit_stress 0.2
TRACER33.resusp_rate 0

```

TRACER0.TYPE #which type of variable:

```

WATER      # 3D water column only
SEDIMENT   # 3D sediment only
WATER SEDIMENT# 3D water column and SEDIMENT
WC2D# 2D water column
BENTHIC# 2D BENTHIC LAYER (sees both the last layer of the water column and
top layer of the sediment)

```

An option available is the automatic generation of the ecological tracers in the tracer list based on the choice of processes. This development is the key to ecological implementations of RECOM, as the user only specifies the process list and parameter values, not the tracers. For this option, see standard process list (Section 3.2).

NOTE: A number of tracer name formats have special uses in EMS:

- TRACERNAME\_WC, TRACERNAME\_EPI, TRACERNAME\_SED should not be used.
- R\_XXX signifies reflectance at wavelength XXX. No tracer should have 'R\_' beginning unless it is specifying a reflectance.
- Ed\_XXX signifies downwelling irradiance at wavelength XXX. No tracer should have 'Ed\_' at the beginning unless it is specifying downwelling irradiance at a particular wavelength.

### 3.1.2 Tracer statistics

To calculate a diagnostic tracer from other tracer output, or obtain a time-mean of the tracer of the, the `tracerstat` option can be used:

```
TRACER50.name Chl_a_sum
TRACER50.long_name Chlorophyll_sum
TRACER50.units mg Chla m-3
TRACER50.type WATER SEDIMENT
TRACER50.valid_range_wc 0 1e+35
TRACER50.valid_range_sed 0 1e+35
TRACER50.diagn 2
TRACER50.tracerstat sum(PhyL_Ch1,PhyS_Ch1,MPB_Ch1,Tricho_Ch1)
TRACER50.fill_value_wc 0
TRACER50.fill_value_sed 0
```

The time-means are especially useful when calculating a daily mean of a variable that changes cyclically through the day, such as chlorophyll concentration.

## 3.2 The process list

EMS contains many processes (i.e. phytoplankton growth, gas exchange etc.) that have been developed for projects around Australia and in Chile. To understand how these processes work the reader is encouraged to look at the top of the \*.c process file. This section discusses how to implement a sub-set of the EMS processes in a particular model configuration.

The process list file contains the listing of all the BGC processes that are to be run in the model, broken into the 3 vertical zones: water column, epibenthic and sediment. Each of these processes has 2 associated files (a .h and .c file), located in the source code folder at:

```
https://github.com/csiro-coasts/EMS/tree/master/model/lib/ecology/process\_library
```

The process list use is invoked in the configuration file with the statement:

```
PROCESSFNAME      process_list.prm
```

All processes in the source codes (listed in `allprocessess.c`) are included in the model executable when it is compiled. But processes are only implemented in the model run if they are in the process list. This allows the same executable to be used for multiple applications. Some processes have arguments. In this case, the comments at the top of the \*.c file explain the options.

In general, processes with a `_wc`, `_sed`, or `_epi` suffix can only be applied in the water column, sediment or epibenthic respectively.

The processes `values_common` and `values_common_epi` are used to define variables either in pre-calc, such that they can be used in the other processes, or in post-calc, so they can be outputted. Examples of variables defined include DIN.

Occasionally conflicts can occur with combinations of processes, and the user needs to be careful. For example, if a process is listed twice, the code will attempt to execute the rate process twice. Also, where two alternate versions of a process exists, such as `trichodesmium_grow_wc` and `trichodesmium_spectral_grow_wc`, applying both will have unintended consequences.

Furthermore, a peculiarity of the code is that it checks for the mass balance of model through adding state variable to the sums TN, TP, TC and TO. It does these at the beginning and end of the process. It uses the fact that it is in a particular process to know to add a variable to this sum. So for example, if it is in the process `trichodesmium_spectral_grow_wc`, it adds the nitrogen associated with the

trichodesmium structural material and the reserves of nitrogen to TN. However, if `trichodesmium_spectral_grow_wc` is not specified, but there is trichodesmium in the tracer list, and `zooplankton_large_spectral_grow_wc` is called, then the mass balance will fail because TN does not include structural material of trichodesmium, and it is altered by grazing. Usually these conflicts are due to genuine incompatibility of processes. But occasionally it may be necessary to turn off the mass balance (see Section 11.1).

The configuration used in this example contains the following processes:

- **in the water column:**

```
tfactor          # Q10 temperature dependence of some parameters
viscosity        # calculate the viscosity from T and S
moldiff          # calculate the molecular diffusivity from T and S
remineralization # contain all the remineralization processes.
microphytobenthos_spectral_grow_wc
phytoplankton_spectral_grow_wc(small)
phytoplankton_spectral_grow_wc(large)
trichodesmium_mortality_wc
trichodesmium_grow_wc
phytoplankton_spectral_mortality_wc(small)
phytoplankton_spectral_mortality_wc(large)
zooplankton_mortality_wc(small)
zooplankton_mortality_wc(large)
zooplankton_large_spectral_grow_wc
zooplankton_small_spectral_grow_wc
nitrification_wc
p_adsorption_wc
carbon_chemistry_wc # OCMIP routines
gas_exchange_wc(oxygen,carbon) # if just one, (gas,dummy)
carbon_chemistry_wc
values_common      # calculate value across the different processes
massbalance_wc
light_spectral_wc
age_wc # perform age tracer experiment from a source
```

Notes:

1. `phytoplankton_spectral_grow_wc`, `phytoplankton_spectral_mortality_wc` and `zooplankton_mortality_wc` contain an argument of either (small) or (large). For these three processes the code is identical, but the parameters vary between small and large.
2. `gas_exchange_wc(oxygen,carbon)` – Gas exchange is calculated once, and applied to the gases in the argument. If carbon specified, `carbon_chemistry` must be called above.
3. `Trichodesmium` processes can only be used with the spectrally-resolved processes.
4. Mass balance processes– this is a test within a cell and an ecological time step that the model conserves mass. It's most appropriate use is to ensure the code is consistent. It does not assess whether the model losses mass across boundaries or during advection or diffusion.

- **in the epibenthic layer (~ benthic layer):**

```
tfactor_epi
macroalgae_spectral_grow_epi
seagrass_spectral_grow_epi(Zostera)
coral_spectral_grow_epi
macroalgae_mortality_epi
seagrass_spectral_mortality_epi(Zostera)
values_common_epi
massbalance_epi
light_spectral_epi
diffusion_epi
```

#### Notes.

1. `diffusion_epi` calculates a diffusive exchange between the sediment and water column based on a constant diffusion coefficient specified by parameters `EpiDiffCoeff` and `EpiDiffDz`. It is in addition to the diffusive flux specified in the sediment model by

- **in the sediment layers:**

```
tfactor
viscosity
moldiff
remineralization
light_sed
microphytobenthos_spectral_grow_sed
microphytobenthos_spectral_mortality_sed
phytoplankton_spectral_mortality_sed(small)
phytoplankton_spectral_mortality_sed(large)
trichodesmium_mortality_sed
nitrification_denitrification_sed
p_adsorption_sed
values_common
massbalance_sed
```

- **for the entire column:**

From SHOC version 6773 onwards, a fourth process type became available, “column”

```
column {
    light_spectral_col()
}
```

This has been developed for the purposes of radiative transfer solution for the light field, but will no doubt be used in more applications in the future.

### 3.2.1 Arguments to processes and multiple process calling.

Some of the processes contain arguments. These are to allow the user to pursue different options (such as calculating moonlight in `light_spectral_wc`) or to invoke a processes using a different set of state variables and parameter values (such as large and small phytoplankton). In the second case, processes can be called more than once to allow for both varieties. So, for example, `seagrass_spectral_grow_epi` can be called with “Zostera” or “Halophila”. The

equations are the same, but “Zostera” uses SG and SGROOT tracers and parameters starting with SG, while “Halophila” uses SGH and SGHROOT tracers and SGH parameters.

### 3.2.2 Standard process lists

Instead of specifying a list of processes from a file as described above, a number of key words are reserved for standard process lists. For example,

```
PROCESSFNAME      BCG2p0
```

invokes a standard list of processes. When a standard process list is used, the model also generates the tracers that is required by the processes in this list. The list it uses is also written into the execution folder with a filename `processes_*.prm`, where `*` is the key word. A list of the most commonly used lists is given below (Table 1).

**Table 1 Standard process lists**

Keyword	Description of process list
<b>BGC2p0</b>	Biogeochemical model used in the eReefs implementation B2p0 with 4 microalgae, 2 zooplankton, 3 seagrass, 1 macroalgae and a simple coral host symbiont model.
<b>BGC3p1</b>	Biogeochemical model used in the eReefs implementation B3p1 with 4 microalgae, 2 zooplankton, 3 seagrass, 1 macroalgae and the coral host symbiont model with coral bleaching.
<b>gas_only</b>	Air-sea gas exchange of carbon and oxygen only.
<b>optics_only</b>	The spectrally-resolved optical model calculations only.
<b>dead_only</b>	All processes in BGC3p1, except those involving living processes, and also including anammox. This includes the spectrally-resolved optical model.
<b>porewater_age</b>	Aging process in sediment porewaters only.
<b>TASSE1p0</b>	Processes for eTas model.

Most of the standard process lists are available through the RECOM interface.

## 3.3 The parameter file

This file contains a list of the BGC model parameters and their metadata.

```
NPARAMETERS      141
```

```
PARAMETER0.name   Tref
PARAMETER0.desc   Reference temperature
PARAMETER0.units  Deg C
PARAMETER0.value  20.000000
PARAMETER0.stderr  0.000000
PARAMETER0.ref    Set by parameter library options.
```

Notes:

1. EMS does read the “units”, and if a time unit other than “seconds” is specified, it will convert to seconds for use in the code.
2. As a matter of principle, all biological parameters are read into the code during the initialisation phase.
3. The top of the `bio_spectral.prm` file contains a line: “NPARAMETERS 98” which must be manually changed to the number of parameters (remember the count starts at 0).
4. Just because a variable appears in the `bio.prm` files does not imply it is used in the code. If it is never called by any processes, or a typo exists in the “name”, the code will not use it. Often this results in a default value in the code being used. To ensure the value of the parameter that is implemented is known, **ecology\_setup.txt** prints out each time the code uses a parameter value. The self-explanatory statements, are one of the following three:

```
Ecol. parameter forced read, <NAME> = XXX
Ecol. parameter tried to read <NAME> and found XXX
Ecol. parameter tried to read <NAME>, but not in parameter file.
Code default of <NAME>, XXX
```

5. The variables that are read from the parameter file and are used in an ecological process are output to a netcdf file named **ecology\_parameter\_setup.nc** which has a format like:

```
netcdf ecology_parameter_setup {
variables:
    double Q10 ;
    Q10:name = "Q10" ;
    Q10:description = "Temperature coefficient for rate
parameters" ;
    Q10:units = "none" ;
    Q10:function_call = "try_parameter" ;

    // global attributes:
        :title = "CSIRO Environmental Modelling Suite (EMS) biological
parameter setup." ;
        :description = "All variables specified in the biological
parameter file" ;
        :date_created = "Mon Apr 1 09:35:09 2024" ;
    data:
        Q10 = 2 ;
}
```

Apart from providing a providence of parameter values, this file can be used to automate recalculation of model rates (e.g.  $u_{max} = u_{max\_t0} * Q_{10}^{((T-T_{ref})/10)}$ , where  $T_{ref}$ ,  $Q_{10}$  and  $u_{max\_t0}$  are read (in matlab, python etc.) from the `ecology_parameter_setup.nc` and  $T$  comes from an output file). The attribute function call informs whether the parameter needed to be specified in the `bio.prm` file (`"try_parameter"`), or whether it is optional (`"get_parameter"`) in the first instance that it is read.

### 3.3.1 Standard parameter file

Instead of specifying parameter values from a file as described above, a number of key words are reserved for a set of standard parameter values. For example,

```
BIOFNAME    BCG2p0
```

invokes a standard set of parameter values used in the B2p0 configuration. The list used is also written into the execution folder with a filename `bio_*.prm`, where `*` is the key word. A list of the most commonly used lists is given below (Table 1).

**Table 2 Standard set of parameter values**

Keyword	Description of set of parameter values
<b>BGC2p0</b>	Biogeochemical model used in the eReefs implementation B2p0 with 4 microalgae, 2 zooplankton, 3 seagrass, 1 macroalgae and a simple coral host symbiont model.
<b>BGC3p1</b>	Biogeochemical model used in the eReefs implementation B3p1 with 4 microalgae, 2 zooplankton, 3 seagrass, 1 macroalgae and the coral host symbiont model with coral bleaching.
<b>porewater</b>	Just parameter values for porewater aging.
<b>TASSE1p0</b>	Biogeochemical model used in the eTas.



## 4 Temperature dependence

The functional form of the temperature-dependence of the vast majority of ecological rates is set by choosing a process, either `tfactor` or `tfactor_clim`. In both cases a power law relationship is applied, of the form  $k_T = k_{ref} Q_{10}^{(T-T_{ref})/10}$ , where  $k_T$  is the rate at temperature  $T$ ,  $k_{ref}$  is the rate at  $T_{ref}$ , and  $Q_{10}$  is a unitless coefficient, often 2.0 to represent a doubling in the rate over 10 degrees.

If `tfactor` is called,  $T_{ref}$  is calculated based  $T_{ref}$  from the biological parameter file.

If `tfactor_clim` is called,  $T_{ref}$  is spatially- and temporally changing, and is specified by:

```
TRACER142.name          temp_clim
TRACER142.long_name     Climatological temperature
TRACER142.units         deg C
TRACER142.valid_range   0.00e+00 1.00e+35
TRACER142.fill_value    26.0
TRACER142.advect        0
TRACER142.diffuse        0
TRACER142.diagn          0
TRACER142.partic         0
TRACER142.dissol         0
TRACER142.type           WC3D  SEDIMENT ECOLOGY  DIAGNOSTIC
TRACER142.eco_flag       NORESET
STRACER142.reset_file    forcing_bgc/TS_CARSa.nc4(temp_clim=temp_CARS)
TRACER142.reset_dt      1 day
```

where the value of  $T_{ref}$  is updated from file `reset_dt` every days.

## 5 Sediment – water column exchange rates

The exchange of dissolved tracers between the water column and sediment occurs because of:

1. Deposition of sediments, entraining the porewater from the bottom layer of the water column (a flux into the sediment).
2. Erosion of sediments, releasing porewater into the water column (a flux out of the sediment).
3. Diffusion in the sediment model (which is slow).
4. Diffusion in the ecology model (implemented in the process `diffusion_epi`).

The sum of processes 1-4 is difficult to calculate from the state variables in the model output, so inline diagnostic calculations are available.

The sediment model calculates the sum of the instantaneous rates of 1-3 of variable `XXX` if the tracer `XXX_fluxsedi_inst` is specified :

<code>TRACER196.name</code>	<code>DIC_fluxsedi_inst</code>
<code>TRACER196.long_name</code>	<code>Sed-water sed. DIC flux</code>
<code>TRACER196.units</code>	<code>mg C m-2 s-1</code>
<code>TRACER196.fill_value</code>	<code>0.00e+00</code>
<code>TRACER196.valid_range</code>	<code>-1.00e+35 1.00e+35</code>
<code>TRACER196.diagn</code>	<code>0</code>
<code>TRACER196.advect</code>	<code>0</code>
<code>TRACER196.diffuse</code>	<code>0</code>
<code>TRACER196.type</code>	<code>WC2D DIAGNOSTIC</code>

The ecological model calculates diffusion in the ecological model if the process `diffusion_epi` is evoked, and the diagnostic tracer `XXX_sedflux` is specified.

<code>TRACER195.name</code>	<code>DIC_sedflux</code>
<code>TRACER195.long_name</code>	<code>Sed-water ecol. DIC flux</code>
<code>TRACER195.units</code>	<code>mg C m-2 s-1</code>
<code>TRACER195.fill_value</code>	<code>0.00e+00</code>
<code>TRACER195.valid_range</code>	<code>0.00e+00 1.00e+35</code>
<code>TRACER195.diagn</code>	<code>2</code>
<code>TRACER195.type</code>	<code>WC2D ECOLOGY DIAGNOSTIC</code>

At present the only variables available for `XXX_sedflux` are `DIC`, `DIP`, `NO3`, `NH4`, `alk`, and `Oxygen`. Unless a large resuspension or deposition event has occurred, `XXX_sedflux` will generally be an order of magnitude more than `XXX_fluxsedi_inst`. The total instantaneous sediment-water exchange (positive in the sediment to water direction) is the sum of the `XXX_sedflux` and `XXX_fluxsedi_inst`.

## 6 Optical model configuration

### 6.1 Original spectrally-resolved model

From 2010-2021 (up until B3p3), the spectrally-resolved optical was implemented using the processes `light_spectral_wc`, `light_spectral_uq_epi` and `light_spectral_sed`. This is being replaced with a column-based calculation (`light_spectral_col`) described in Sec. 6.2.

The spectrally-resolved optical model calculates the inherent optical properties (IOPs) from the optically-active state variables, and then uses the IOPs to calculate the apparent optical properties (AOPs) such as the light field and remote-sensing reflectance.

The optical model is configured using parameter variables specified in **bio.prm**, through option calls to light processes in the process list models, and through tracer list specification. The relationships used to relate state variables to IOPs (such as the pigment-specific absorption coefficient of each of the pigments) are specified in the file `bio_opt.c`:

[https://github.com/csiro-coasts/EMS/blob/master/model/lib/ecology/bio\\_opt.c](https://github.com/csiro-coasts/EMS/blob/master/model/lib/ecology/bio_opt.c).

The wavelengths (centre and bandwidths) used are specified in the parameter file using:

```
PARAMETER4.name      Light_lambda
PARAMETER4.desc       Wavelengths of light
PARAMETER4.units      nm
PARAMETER4.value      290.000000 310.000000 330.000000 350.000000 370.000000
390.000000 410.000000 430.000000 440.000000 450.000000 470.000000 490.000000
510.000000 530.000000 550.000000 570.000000 590.000000 610.000000 630.000000
650.000000 670.000000 690.000000 710.000000 800.000000
PARAMETER4.stderr     0.000000
PARAMETER4.ref         Approx. 20 nm resolution with 10 nm about 440 nm. PAR
(400-700) is integral of bands 2-22.
```

The ‘hard’ coded shortest wavelength edge is 140 nm, and the longest edge at 8000 nm. The spectral ‘grid’ that results from this specification is output into **ecology\_setup.txt**:

```
Centre of waveband (edge):
290.00 (140.00 - 300.00)
310.00 (300.00 - 320.00)
330.00 (320.00 - 340.00)
350.00 (340.00 - 360.00)
370.00 (360.00 - 380.00)
390.00 (380.00 - 400.00)
410.00 (400.00 - 420.00)
430.00 (420.00 - 435.00)
440.00 (435.00 - 445.00)
450.00 (445.00 - 460.00)
470.00 (460.00 - 480.00)
490.00 (480.00 - 500.00)
510.00 (500.00 - 520.00)
530.00 (520.00 - 540.00)
550.00 (540.00 - 560.00)
570.00 (560.00 - 580.00)
590.00 (580.00 - 600.00)
```

```

610.00 (600.00 - 620.00)
630.00 (620.00 - 640.00)
650.00 (640.00 - 660.00)
670.00 (660.00 - 680.00)
690.00 (680.00 - 700.00)
710.00 (700.00 - 755.00)
800.00 (755.00 - 8000.00)

```

For this particular optical grid, and the clear sky spectrally-resolved solar radiation, approximately 40 % of the energy of the sun is between 400 and 700 nm. Note that in this configuration we have used a 20 nm resolution, but added an extra wavelength at 440 nm to better resolve absorption by photosynthetic pigments.

In the tracer list, the inclusion of tracers with tracer names of R\_\*\*\* results in the optical model calculating remote-sensing reflectance,  $R_{rs}$ , at wavelength \*\*\* nm:

```

TRACER93.name          R_640
TRACER93.long_name     Remote-sensing reflectance @ 640 nm
TRACER93.units         sr-1
TRACER93.fill_value    0.00e+00
TRACER93.valid_range   0.00e+00 1.00e+35
TRACER93.diagn         2
TRACER93.type          WC2D  ECOLOGY  DIAGNOSTIC

```

**Warning: Do not include a tracer with R\_ as its first two characters unless the next characters are specifying a wavelength.**

In the tracer list, the inclusion of tracers with tracer names of Ed\_\*\*\* results in the optical model calculating downwelling irradiance,  $E_d$ , at wavelength \*\*\* nm:

```

TRACER224.name          Ed_490
TRACER224.long_name     Downwelling light at 490 nm
TRACER224.units         W m-2 nm-1
TRACER224.fill_value    0.00e+00
TRACER224.valid_range   0.00e+00 1.00e+35
TRACER224.fill_value_sed 0.00e+00
TRACER224.valid_range_sed 0.00e+00 1.00e+35
TRACER224.advect        0
TRACER224.diffuse        0
TRACER224.diagn         2
TRACER224.partic        0
TRACER224.dissol         0
TRACER224.type          WC3D  SED  ECOLOGY  DIAGNOSTIC
TRACER224.psize          0.00e+00
TRACER224.b_dens         0.00e+00
TRACER224.i_conc         0.00e+00
TRACER224.svel           0.0
TRACER224.cohesive       1
TRACER224.calcvol        0
TRACER224.floc           0
TRACER224.resuspend      1
TRACER224.deposit        1
TRACER224.eco_flag       0

```

**Warning: Do not include a tracer with Ed\_ as its first three characters unless the next characters are specifying a wavelength.**

In the case that the wavelength specified in R\_\*\*\* or Ed\_\*\*\* is not one of the centre wavelengths the output will be a linear interpolation of centre wavelength.

Further, a number of simulated satellite products can be outputted if the required remote-sensing reflectance at the required wavelengths, and the key tracer name are specified (with similar options to the Remote-sensing reflectance above) (Table 3). If using the `light_spectral_epi` process, the code requires remote-sensing wavelengths (e.g. OC3M would require R\_443, R\_488 and R\_547). If using the `light_spectral_col` process that uses a spectral response curve, the code requires bands (e.g. OC3M\_SR would require MODIS\_B2, MODIS\_B4, MODIS\_B6).

**Table 3 Simulated observations.** [variables in brackets refer to spectral response curves used in `light_spectral_col`].

Tracer name	Description of simulated variable	wavelength (nm)	Source
<b>OC3M</b> [OC3M_SR]	Three-band chlorophyll algorithm calibrated for the MODIS sensor.	443, 488, 547; [MODIS_B2, MODIS_B4, MODIS_B6]	
<b>OC3V</b> [OC3V_SR]	Three-band chlorophyll algorithm calibrated for the VIIRS sensor.	443, 486, 551; [VIIRS_B2, VIIRS_B3, VIIRS_B4]	
<b>OC4Me</b> [OC4Me_SR]	Four-band chlorophyll algorithm calibrated for the MERIS and Sentinel3A sensors	443, 488, 510, 560; [Sentinel_B3, Sentinel_B4, Sentinel_B5, Sentinel_B6]	
<b>Secchi</b>	Secchi depth calculated from vertical integral of attenuation of light at 488 nm.	488	Lee et. al (2016).
<b>TSSM</b> [TSSM_SR]	Turbidity based on red reflectance – local GBR algorithm.	645 [MODIS_B9]	Petus
<b>KD490M</b> [KD490M_SR]	Two band vertical attenuation at 490 nm.	488, 547; [MODIS_B4, MODIS_B6]	
<b>nFLH</b>	Normalized Fluorescence Line Height based on MODIS wavelengths.	667, 678, 748 [MODIS_B9, MODIS_B10, MODIS_B11]	
<b>Turbidity</b>	Turbidity calculated from total scattering at 590 nm.	590	
<b>Fluorescence</b>	Chlorophyll concentration ‘scaled’ by processes that reduce fluorescence such as self-shading and photochemical quenching.	Integral of photosynthetic wavelengths.	

Finally, the default calculation of the zenith angle is on the time zone (in below example +10) specified at the top of the master configuration file:

```
TIMEUNIT          seconds since 1990-01-01 00:00:00 +10
```

However, if the option:

```
PROJECTION      geographic
```

is invoked, the zenith angle is calculated based on the longitude (as well as the hour, day, and latitude).

## 6.2 Specification of CDOM absorption

Colour dissolved organic matter absorption can be added as a mixture of absorbing CDOM tracers (CDOM\_pale, CDOM\_amber, CDOM\_dark), each with an absorption at 440 nm and exponential coefficient specified in the biological parameter file (as a440cdom\_pale, Scdom\_pale etc.) with an ocean value as well (a440cdom\_ocean, Scdom\_ocean).

Additionally, a bespoke relation for GBR has been developed using the tracer 'cdom\_gbr'. When using cdom\_gbr, an additional flexibility is introduced of the biological parameter 'CDOMgbrscale' which allows all absorption to be scale, the default being 1.

Finally, in the column-based calculations an absorbing due can always be added, but it will require a spectrally-resolved absorption coefficient to be added through a netcdf library.

## 6.3 Column-based light calculations

A major upgrade to the optical model occurred through 2021/22, with improved configuration and calculation of the light field.

The optical model contains two classes of spectrally-resolved parameters:

1. Mass-specific coefficients, such as the mass-specific absorption rate of carbonate sand. As these are applied only to specific tracers they are specified as tracer attributes in the tracer list (see 6.2.1).
2. Generic optical coefficients, such as the absorption rate of sea water. As generic optical coefficients are not linked to any particular tracer, they are specified in the .prm or .tran file.

The generic optical coefficients that can be specified in the configuration files (ideally placed in section including DO ECOLOGY) are:

```
optical_library_filename xx.nc
```

- this option determines the netcdf file that contains constants  $g_0$ ,  $g_1$ ,  $g_i$ ,  $g_{ii}$ ,  $a_w$ ,  $b_w$  as well as the reflectance of corals, luminescence and fluorescence emission, the normalised scattering of phytoplankton cells and the microalgae pigments. The default filename is "csiro\_optical\_parameters\_library.nc".

```
optical_library_variable_name_aw aw_HL
```

```
optical_library_variable_name_bw btw_HL
```

- for two variables  $a_w$ ,  $b_w$ , a further option is provided to choose the variable name within the specified netcdf file. The default names are 'aw' and 'btw'. The example above with variable name "aw\_HL" is given for the clear water data used in HydroLight.

### 6.3.1 Tracer attribute specification of spectrally-resolved mass-specific optical parameters.

In `light_spectral_col`, the bulk of the spectrally-resolved optical parameters (e.g. mass-specific absorption by inorganic particles) are provided to model in a netcdf file. In order to assign a mass-specific optical property to a tracer, the file name and variable name are given in the tracer list (here for a particulate substance).

TRACER4.name	Mud-mineral
TRACER4.units	kg m <sup>-3</sup>
TRACER4.optical_file	csiro_siop_library.nc
TRACER4.type	WC3D SED SEDIMENT PROGNOSTIC OPTICAL
TRACER4.absorp_name	ap_terr
TRACER4.scatter_name	btp_terr
TRACER4.backscat_name	B_terr
TRACER4.benreflt_name	rho_mud

On initialisation, the process `light_spectral_col` finds all tracers that are optically-active (as specified by type `OPTICAL`). It then reads their mass-specific optical properties from the file specified. The values are then linearly interpolated onto the optical grid. Not all the attributes for each tracer must come from the same file.

During the integration the component of absorption due to the tracer *X* is calculated from  $ap\_X \cdot X$ . Similarly scattering, the backscattering ratio and benthic reflectance can be specified.

For benthic variables, the optical properties are the absorbance, reflectance and transmission of the surface:

TRACER59.name	MA_N
TRACER59.long_name	Macroalgae N
TRACER59.units	g N m <sup>-2</sup>
TRACER59.fill_value	0.2
TRACER59.valid_range	0.00e+00 1.00e+35
TRACER59.diagn	0
TRACER59.type	WC2D ECOLOGY PROGNOSTIC OPTICAL
TRACER59.optical_file	project_siop_library.nc
TRACER59.abtance_name	Ahalophila
TRACER59.refltce_name	rho_macroalgae
TRACER59.trnmiss_name	Thalophila

For suspended macroalgae (implemented in `macroalgae_spectral_grow_wc`), which use the 2D benthic optical properties for a 3D tracer, the two options available are:

```
TRACER59.abtance3d_name      Ahalophila
TRACER59.refltce3d_name      rho_macroalgae
```

At initialisation the model produces a file `optical_setup.nc` that contains the spectrally-resolved optical parameters and the optical grid (wave band specification etc.), as well as useful optical attributes of the tracer. An example is absorption of FineSed:

```
double ap_FineSed(wave_centre) ;
    ap_FineSed:description = "Mass-specific absorption by non-CaCO3 particles" ;
    ap_FineSed:codename = "ap_FineSed" ;
    ap_FineSed:sourcefile = "project_siop_library.nc" ;
    ap_FineSed:sourcename = "ap_terr" ;
    ap_FineSed:symbol = "a_{non-CaCO3}" ;
    ap_FineSed:units = "m-1 (mg m^{-3})-1" ;
    ap_FineSed:interpolation_percent_error_in_data_range = 7.86551155453676 ;
    ap_FineSed:source_data_range_in_nm = 401.6, 750.3 ;
    ap_FineSed:model_range_used_for_interpolation_error_calculation = 410., 690. ;
    ap_FineSed:data_range_used_for_interpolation_error_calculation = 410., 686.3 ;
    ap_FineSed:rgb_colour_space = 164.85628351858, 151.793433426242, 116.99795659274 ;
    ap_FineSed:mass_concentration_used_for_rgb_colour_space_in_clear_water = 1. ;
```

Attributes include the percent area generated by interpolating onto the optical grid and the MODIS true colour produced (given as RGB) of suspending the particles in clear water.

### 6.3.2 Tracer attribute specification of spectral response curves.

Tracer attributes can also be used to specify the spectral response of a sensor. For the tracer attribute of `specresp2d`, the response is interpreted as a sensor to remote-sensing reflectance and is ideally used for an airborne or satellite based sensor with measurements of both downwelling irradiance and upwelling radiance (i.e.  $R_{rs} = E_d / L_u$ ). For example:

```
TRACER218.name                Himawari8_Band1_SR
TRACER218.long_name           Col:Himawari8 Band 1
TRACER218.units                -
TRACER218.fill_value          0.00e+00
TRACER218.valid_range          -1.00e+35 1.00e+35
TRACER218.diagn                2
TRACER218.type                 WC2D ECOLOGY  DIAGNOSTIC OPTICAL
TRACER218.optical_file         csiro_siop_library.nc
TRACER218.specresp2d_name      Himawari8_Band1_SR
```



The tracer name does not need to be the same as the `specresp2d_name`.

Alternatively, the tracer attribute of `specresp3d` is used for a downwelling radiometer:

```
TRACER224.name           Seabird_SR_490
TRACER224.long_name      Col: Downwelling light at 490 nm
TRACER224.units          W m-2 nm-1
TRACER224.fill_value     0.00e+00
TRACER224.valid_range    0.00e+00 1.00e+35
TRACER224.fill_value_sed 0.00e+00
TRACER224.valid_range_sed 0.00e+00 1.00e+35
TRACER224.advect         0
TRACER224.diffuse        0
TRACER224.diagn          2
TRACER224.partic         0
TRACER224.dissol         0
TRACER224.type           WC3D  SED  ECOLOGY  DIAGNOSTIC  OPTICAL
TRACER224.psize          0.00e+00
TRACER224.b_dens         0.00e+00
TRACER224.i_conc         0.00e+00
TRACER224.svel           0.0
TRACER224.cohesive       1
TRACER224.calcvol        0
TRACER224.floc           0
TRACER224.resuspend      1
TRACER224.deposit        1
TRACER224.eco_flag       0
TRACER224.optical_file   seabird.nc
TRACER224.specresp3d_name Seabird_SR_490
```

In this case the spectral response curve is multiplied by the downwelling irradiance, and then normalised by the area under the spectral response curve, in order to give a measure of the downwelling irradiance that would be measured by that sensor by nm.

Spectral response curves are often quite narrow (say 10 nm for Sentinel ocean colour bands), so using the optical model grid for the response curves is unreliable. As a result, the remote-sensing reflectance is interpolated onto a 1 nm grid that is then used against the spectral response curve.

The file `optical_file` can be a netcdf file of any name. Eight libraries that can be requested from CSIRO are:

- Spectral response curves - `csiro_spectral_response_library.nc`
- General optical properties - `csiro_optical_properties_library.nc`
- Benthic plant reflectance - `csiro_benthic_plant_reflectance_library.nc`
- Mass-specific inherent optical properties - `csiro_mass_specific_iops_library.nc`
- Generic coloured particles and tracers - `csiro_coloured_dyes_particles_library.nc`

A sixth library, `csiro_iops_standards_library.nc`, contain spectrally-constant values of 0, 0.001, 0.01, 0.1, 1, 10, 100 and 1000 that are useful for testing the code and model configuration.

A seventh library, `csiro_photoperception_library.nc`, contains absorption of photoperception molecules. An eighth library is from the AquaWatch project and contains spectral response curves of satellite sensors.

### 6.3.3 Column-based light calculations.

`Light_spectral_col` is a column-based process meaning it has access to state variables in the entire water column. This allows the calculation to be solved via the ray tracing technique used in the earlier optical model, or alternatively, using radiative transfer models. The code provides two schemes do the conversion from IOPs to AOPs, a geometric optics scheme (default and the only scheme available until August 2024), and a three directional radiative transfer model (RT3) based on Gregg and Carder (1990).

To undertake the RT3 calculations, a diagnostic tracer for the PAR-integrated direct downwelling light, `PAR_d_3RT`, must be specified in the tracer list. The remaining outputs of light field can then be specified such as the diffuse downwelling (`PAR_s_3RT`), scalar light (`PAR_sc_3RT`), and diffuse upwelling (`PAR_u_3RT`).

The calculated light field is used to drive photosynthesis and to calculate remote-sensing reflectance. To choose which light model (geometric optics or three directional radiative transfer model), an argument is given in the process list.

Process specification: `light_spectral_col (G|R|P)`

G - Geometric or ray tracing used to calculate AOPs (including `Rrs`) and photosynthesis (default).

R - Three directional radiative transfer model used to calculate AOPs including remote-sensing reflectance. Photosynthesis calculated using the Geometric solution.

P - Three directional radiative transfer model used to calculate AOPs and photosynthesis.

The three directional radiative transfer model (RT3) requires a diffuse and direct component of the surface radiative forcing. If the 2D tracers `SWR_diffuse` and `SWR_direct` appear in the tracer list containing the diffuse and direct shortwave radiation (< 6000 nm) above the sea surface, then these will be used in the RT3 calculation. Further, with `SWR_diffuse` and `SWR_direct` specified in the tracer list, the surface albedo will be calculated separately for the direct and diffuse streams using the OASIM model formulation.

If the tracers `SWR_diffuse` and `SWR_direct` are not in the tracer list or the sun is below the horizon, then an approximate calculation of diffuse versus direct is used based on the zenith of the dominant light source (sun, moon or galactic centre) and fractional total cloud cover (`tcld`):

$$\text{frac\_diffuse} = (5.6023\text{e}9/(\lambda^4 + 0.0267)) * (\text{tcld} + (1-\text{tcld})*(1-\cos(\text{zenith})))$$

`frac_direct = 1 - frac_diffuse`

where  $\lambda$  is the wavelength in nm, tcld is total cloud from all levels.

### 6.3.4 Radiative transfer calculation (HydroLight library).

**HydroLight** is a radiative transfer numerical model computes radiance distributions and related quantities (irradiance, reflectance, diffuse attenuation functions, etc.) in any water body. After Curtis Mobley, the developer of HydroLight, retired from Sequoia in December 2016, HydroLight became a product of Numerical Optics, Ltd. and is managed by Dr. John Hedley.

EMS contains an interface to hydrolight that allows calling to their routines. The HydroLight Code itself is not part of the EMS distribution, and must be obtained from Numerical Optics, Ltd. For users without an EMS licence, `light_spectral_col` needs to comment out the following line:

```
// #define INCLUDE_HYDROLIGHT 1 // Compile with Hydrolight calls.
```

`light_spectral_col` is archived in SVN and GitHub with this line commented out, so no action is required by users not intending to use the HydroLight interface.

### 6.3.5 Generation of a Matlab / python script to plot optical setup.

During model initialisation, EMS generates a Matlab script (`optical_setup_matlab_check.m`) for plotting all the optical properties that have been interpolated onto the optical grid. For each optically property of each optically-active variable, a segment of code is produced:

```
Figure;set(gca,'FontSize',14);
var = ncread(file,'ap_FineSed');
sourcefile = 'project_siop_library.nc';
var1 = ncread(sourcefile, 'ap_terr');
rgb = ncreadatt(file, 'ap_FineSed','rgb_colour_space');
pp = plot(wave,var,'+k');set(pp,'LineWidth',2);hold on;
pp = plot(var1(:,1),var1(:,2),'-k');set(pp,'LineWidth',4.4);hold on;
pp = plot(var1(:,1),var1(:,2),'-k');set(pp,'LineWidth',4,'Color',rgb/256);hold on;
xlabel('wavelength [nm]');
ylabel('ap_FineSed [m-1 (mg m^{-3})-1]','Interpreter','none');
title('Interpolation error: 7.865512e+00');
set(gca,'xlim',[200 1000]);
```

The Matlab script creates a text file (`references.txt`) that lists all the spectrally-resolved optical properties and the original source of the data, which must be an attribute in the source netcdf file listed in the tracer list.

EMS also generates a similar python script (`optical_setup_matlab_check.py`), although it doesn't create the reference list.

### 6.3.6 Output of optical properties in a column

The optical properties of single columns are outputted into a separate netcdf files named `optical_column_out_0.nc` at the `ECOLOGY_DT` timestep. The columns used are specified in the biological parameter file, using the format:

```
PARAMETER144.name      opt_output_i_index
PARAMETER144.desc      Output of optical column at i_index
PARAMETER144.value     266 266 266 266 266 266

PARAMETER145.name      opt_output_j_index
PARAMETER145.desc      Output of optical column at j_index
PARAMETER145.value     23 24 25 26 27 28
```

The first column also becomes the column on which `HYDROLIGHT` calculations, if that option is invoked, are undertaken. The variables are re-grid with depth relative to the surface.

```
double depth(record, k_grid) ;
    depth:description = "metres from sea surface of k_grid" ;
    depth:units = "m" ;
    depth:air_value = 1.e+35 ;
    depth:seafloor_value = -9999. ;
```

### 6.3.7 Other aspects of `light_spectral_col`

Points to note:

- When using `light_spectral_col` a water column, epibenthic and sediment process must be implemented or otherwise the variable list for the column is not generate properly.
- As the light field is calculated at the output time (or postcalc), the rate of light absorption and the incremental benefit of adding chlorophyll must be stored as tracers

- For each microalgae added, two new tracers:

```
TRACER195.name          PhyL_kI
TRACER195.long_name     Col: Large Phytoplankton absorption
TRACER195.units         mol cell-1
TRACER195.fill_value    0.00e+00
TRACER195.valid_range   0.00e+00 1.00e+35
TRACER195.fill_value_sed 0.00e+00
TRACER195.valid_range_sed 0.00e+00 1.00e+35
TRACER195.advect        0
TRACER195.diffuse        0
TRACER195.diagn          0
TRACER195.partic         0
```

TRACER195.dissol	0
TRACER195.type	WC3D SED ECOLOGY DIAGNOSTIC
TRACER195.psize	0.00e+00
TRACER195.b_dens	0.00e+00
TRACER195.i_conc	0.00e+00
TRACER195.css_erosion	-1.00e+00
TRACER195.css_deposition	-1.00e+00
TRACER195.svel	0.000000
TRACER195.cohesive	1
TRACER195.calcvol	0
TRACER195.floc	0
TRACER195.resuspend	1
TRACER195.deposit	1
TRACER195.adsorb	0
TRACER195.adsorbkd	0.00e+00
TRACER195.adsorbrate	0.00e+00
TRACER195.eco_flag	NORESET
TRACER196.name	PhyL_yCfac
TRACER196.long_name	Col: Large Phytoplankton opaqueness
TRACER196.units	-
TRACER196.fill_value	0.00e+00
TRACER196.valid_range	0.00e+00 1.00e+35
TRACER196.fill_value_sed	0.00e+00
TRACER196.valid_range_sed	0.00e+00 1.00e+35
TRACER196.advect	0
TRACER196.diffuse	0
TRACER196.diagn	0
TRACER196.partic	0
TRACER196.dissol	0
TRACER196.type	WC3D SED ECOLOGY DIAGNOSTIC
TRACER196.psize	0.00e+00
TRACER196.b_dens	0.00e+00
TRACER196.i_conc	0.00e+00
TRACER196.css_erosion	-1.00e+00
TRACER196.css_deposition	-1.00e+00
TRACER196.svel	0.0
TRACER196.cohesive	1
TRACER196.calcvol	0
TRACER196.floc	0

TRACER196.resuspend	1
TRACER196.deposit	1
TRACER196.adsorb	0
TRACER196.adsorbkd	0.00e+00
TRACER196.adsorbrate	0.00e+00
TRACER196.eco_flag	NORESET

○ **For each benthic plant added, one new tracer:**

TRACER201.name	SGH_kI
TRACER201.long_name	Col: SGH absorption
TRACER201.units	mol m-2
TRACER201.fill_value	0.00e+00
TRACER201.valid_range	0.00e+00 1.00e+35
TRACER201.diagn	0
TRACER201.type	WC2D ECOLOGY PROGNOSTIC

○ **For suspended macroalgae, one new tracer:**

TRACER211.name	SMA_kI
TRACER211.long_name	Floating MA absorption
TRACER211.units	mol m-2
TRACER211.fill_value	0.00e+00
TRACER211.valid_range	0.00e+00 1.00e+35
TRACER211.fill_value_sed	0.00e+00
TRACER211.valid_range_sed	0.00e+00 1.00e+35
TRACER211.advect	0
TRACER211.diffuse	0
TRACER211.diagn	0
TRACER211.partic	0
TRACER211.dissol	0
TRACER211.type	WC3D SED ECOLOGY DIAGNOSTIC
TRACER211.psize	0.00e+00
TRACER211.b_dens	0.00e+00
TRACER211.i_conc	0.00e+00
TRACER211.css_erosion	-1.00e+00
TRACER211.css_deposition	-1.00e+00
TRACER211.svel	0.000000
TRACER211.cohesive	1
TRACER211.calcvol	0
TRACER211.floc	0

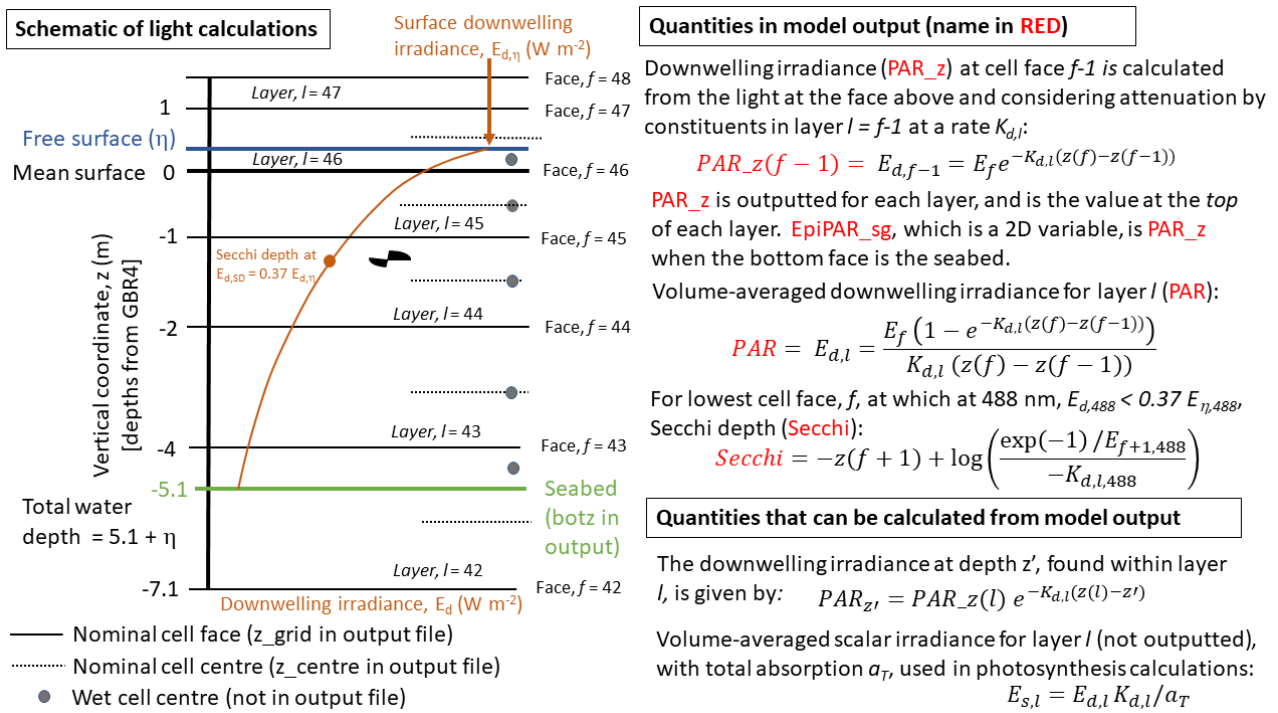
TRACER211.resuspend	1
TRACER211.deposit	1
TRACER211.adsorb	0
TRACER211.adsorbkd	0.00e+00
TRACER211.adsorbrate	0.00e+00
TRACER211.eco_flag	NORESET

The most important improvements are:

- Specification of reflectance, absorptance (photosynthetic and non-photosynthetic) and transmission for benthic plants.
- CDOM absorption only uses CDOM tracers.
- Reflectance calculated for satellite sensor bands.
- Spectrally-resolved bioluminescence emission spectra.
- Wavelength dependence of air-sea water interface refraction.
- The fraction of light into the far-red band is the remainder of the light not allocated into other bands.

## 6.4 Outputs of optical variables

The output of optical variables is complicated by the calculation of downwelling irradiance (PAR<sub>z</sub>) at the cell faces, but PAR at the cell centres. Further, cell centres of the partially-filled surface and bottom layers deviate from the nominal cell centre (z<sub>centre</sub> in output files). The schematic below (Figure 2) illustrates these complications and gives equations that can be calculated from the output. None of these complications affect the output of 2D variables (such as remote-sensing reflectance) which are calculated on the moving free surface at the output time.



**Figure 2 Schematic illustrating the vertical grid of the model and the locations on this grid that the optical properties have been calculated.**

## 6.5 Moonlight

The optical model includes the calculation of moonlight parameters using orbital calculations in the hydrodynamic model, and the USGS ROLO spectral irradiance model (<https://astrogeology.usgs.gov/moon-cal/modeling/irradiance.php>, Kieffer, H. H and T. C. Stone (2005) The spectral irradiance of the moon. The Ast. J. 129: 2887-2901).

In order to invoke moonlight calculations with `light_spectral_wc`, the third argument to the process must be 'M'. For `light_spectral_col`, moonlight is always calculated.

In order for moonlight calculations to be undertaken, `PROJECTION` must be `geographic`.

The output variables for moonlight are:

Moonlight	Moonlight (downwelling PAR-integrated irradiance)
Lunar_zenith	Moon zenith (0 overhead; $\pi/2$ at horizon)
Lunar_phase	Moon phase (0 new moon; $\pi/4$ first quarter; $\pi/2$ full moon; $3\pi/4$ last quarter; 0 new moon)
Moon_fulldisk	Moonlight brightness (PAR-integrated) if looked at directly.
Lunar_azimuth	Moon azimuth angle, following a north clockwise-wise convention (0 north; $\pi/4$ west; $\pi/2$ south; $3\pi/4$ east)

The full disk reflectance (i.e. looking straight at the moon without including the reduction to the zenith angle,  $1/\cos \theta$ ) includes the effect of moon phase on irradiance.



The USGS ROLO model include the effect of Earth-Sun distance, Earth Moon distance, spatially varying albedo of the moon (i.e. the craters are darker than the plains), the point of the Moon facing the Earth, phase of the moon, time of day, point on the Earth (latitude).

The Moonlight calculations in `light_spectral_col` include the same atmospheric attenuation calculations as used for Sunlight in the EMS hydrodynamic model (although using the lunar instead of solar zenith). For these to be undertaken the `.tran` file must include inputs of CLOUD, PRESSURE and HUMIDITY:

```
CLOUD /home/eReefs/forcing/access/gbr_2016_03_access.nc
CLOUD_INPUT_DT 1 hour
PRESSURE /home/eReefs/forcing/access/gbr_2016_03_access.nc (pressure=mslp)
PRESSURE_INPUT_DT 1 hour
HUMIDITY /home/eReefs/forcing/access/gbr_2016_03_access.nc (hr=hr2m)
HUMIDITY_INPUT_DT 1 hour
```

In addition to Moonlight, a small starlight term is given (~1000 x less than the full moon overhead) that allows calculations to be undertaken with a non-zero light field. For simplicity, the galactic centre of the Milky Way is assumed to be at a zero zenith.

## 6.6 Colour visualisation.

For production of true colour a red, green and blue wavelength of remote-sensing reflectance needs to be specified. In the CSIRO visualisation software DIVE, `MODIS_true_colour` uses tracers `R_470`, `R_555` and `R_667`, and `MODIS_SR_true_colour` uses tracers `MODIS_B9`, `MODIS_B6` and `MODIS_B3`. If these tracers aren't available in the output files then no option to visual true colour is given by DIVE.

## 7 Carbon chemistry model

To implement the carbon chemistry model requires that the tracers temperature (`temp`), salinity (`salt`), total alkalinity (`alk`), dissolved inorganic carbon (`DIC`) and pH (`PH`) be specified. Additional diagnostic variables that can be produced by the carbon chemistry model are calcite saturation, aragonite saturation, surface partial pressure of carbon dioxide, bicarbonate ion concentration, carbonate ion concentration and atmospheric CO<sub>2</sub> concentration. The atmospheric CO<sub>2</sub> concentration can either be set to a constant using the parameter file:

```
PARAMETER3.name      xco2_in_air
PARAMETER3.desc       Atmospheric CO2
PARAMETER3.units      ppmv
PARAMETER3.value      396.480000
PARAMETER3.stderr     0.000000
PARAMETER3.ref        Mean '13, Mauna Loa: http://co2now.org/current-co2/co2-now/
```

Or by using a resetting spatially-resolved tracer:

```
TRACER77.name         xco2_in_air
TRACER77.long_name     Atmospheric CO2
TRACER77.units         ppmv
#TRACER77.fill_value   400.0
TRACER77.valid_range   0.00e+00 1.00e+35
TRACER77.advect        0
TRACER77.diffuse        0
TRACER77.diagn         0
TRACER77.partic         0
TRACER77.dissol         0
TRACER77.type          WC3D    SEDIMENT ECOLOGY  PROGNOSTIC
TRACER77.eco_flag       0
TRACER77.reset_file     forcing_bgc/global_co2_merged_2016.ts
TRACER77.reset_dt       30 days
```

The Ocean-Carbon Cycle Model Intercomparison Project (OCMIP) numerical methods are used to quantify air-sea carbon fluxes and the carbon dioxide system equilibria in seawater. The OCMIP procedures quantify the state of the CO<sub>2</sub> system using two prognostic variables, the concentration of dissolved inorganic carbon,  $C_T$ , and total alkalinity,  $A_T$ . The value of these prognostic variables, along with salinity and temperature, are used to calculate the partial pressure of carbon dioxide,  $p\text{CO}_2$ , in the surface waters using a set of governing chemical equations that are solved using a Newton-Raphson method with literature-derived constants. One alteration from the typical implementation of the OCMIP algorithm is that we increased the search space for the iterative scheme from  $\pm 0.5$  pH units (appropriate for global models) to  $\pm 2.5$ . The air-sea flux of carbon dioxide is calculated based on wind speed data (requiring process `gas_exchange_wc(carbon)`), and the air-sea gradient in  $p\text{CO}_2$ , using an empirical net flux relationship.

## 8 Ecological integration options

In the model configuration file .prm or .tran file: ECOLOGY options are:

```
DO_ECOLOGY 1      ## DO ECOLOGY FLAG SET TO 1 TO run ecology
ECOLOGY_DT 1200   # ECOLOGY time step
```

Note: make sure that the ecology timestep is the same length or less than the shortest timestep in the physics or forcing data. If this is not true, the model will change `ECOLOGY_DT` from what is specified in the file, which will be noted in the **runlog**.

### Specify the other configuration files:

```
biofname          in/bio_spectral.prm # file that contain model parameters
processfname      in/processes_spectral.prm # file that contains the list of
which processes.
```

### Specify integration checks:

```
check_nans 0      # test to check is there is any NaN in the state variable
array , computationally expensive so only used during debugging.
check_negs 0 # test to check is there is any <0 in the model , expensive so
only used during debugging.
```

Both `check_nans` and `check_negs` cause the model to stop, and output diagnostic information to the **runlog**. `check_negs` is will often fail with dissolved oxygen, that is allowed to go slightly negative.

### Miscellaneous options:

```
internal_tracers 1 # legacy
mandatory_water  0 # legacy
mandatory_sediment 0 # legacy
```

### Light forcings:

The short wave solar radiation (i.e up to about 5000 nm, or which approximately 40% is PAR, see Section 4) can be specified from a file:

```
LIGHT              in/light.ts      # get the light from a .ts text file
LIGHT_INPUT_DT     1 days
ALBEDO_LIGHT       0.2# legacy
```

Or from an internal tracer:

```
LIGHT              swr      # get the light from the internal swr hydro tracer
(form meteorological forcing)
```

If it is coming from an internal tracer, the tracer can either be from the fully-coupled model, or in the transport model. For convenience, if the following diagnostic tracer is specified:

```
TRACER120.name      PAR_z
TRACER120.long_name  Downwelling PAR at top of layer
TRACER120.units      mol photon m-2 s-1
```

```

TRACER120.fill_value      0.00e+00
TRACER120.valid_range     0.00e+00 1.00e+35
TRACER120.fill_value_sed  0.00e+00
TRACER120.valid_range_sed 0.00e+00 1.00e+35
TRACER120.advect          0
TRACER120.diffuse         0
TRACER120.diagn           2
TRACER120.partic          0
TRACER120.dissol          0
TRACER120.type            WC3D  SED  ECOLOGY  DIAGNOSTIC
TRACER120.psize           0.00e+00
TRACER120.b_dens          0.00e+00
TRACER120.i_conc          0.00e+00
TRACER120.svel           0.0
TRACER120.cohesive        1
TRACER120.calcvol         0
TRACER120.floc            0
TRACER120.resuspend       1
TRACER120.deposit         1
TRACER120.eco_flag        0

```

then the surface value of PAR<sub>z</sub> will be the surface PAR radiation (see Section 4).

### Numerical scheme for ecology integration

The model uses an adaptive time step to solve the ecological ordinary differential equations (ODEs). The model configuration file allows specification of the adaptive integrator used, and the precision to which it is applied.

```

integrator dopri5          # BGC integrator scheme
integration_precision 1e-5 # BGC integrator precision

```

where the above options are the default. Integrator schemes include (dopri8, dopri5, euler1). The adaptive schemes use an approximation of the solution through the time-step (for dopri5, four and five time-steps), and use the difference in solution at the end to determine if the precision requirement has been met. Only the solution at the end of the time step is kept.

### By-passing ecological integration

The ecological integration is not undertaken if:

- The cell is not wet. Note whether a cell is 'wet' or 'dry' can change in time due to surface elevation changes.
- If the adaptive integrator fails to converge after NMAX = 2000 steps (this is hardcoded), the ODEs for that timestep are set to zero, the tracer ECOLOGY\_ERROR is incremented. In this case the updating excludes the whole column (wc, epibenthic and sed).
- If the master configuration file lists the cell below "EXCLUDE\_BGCSED". For example:

```

EXCLUDE_BGCSED 2
(169,15) - (170,15)
7 11

```

## 9 Pre-processing tools for BGC

[TO BE COMPLETED]

## 10 Running simulations

Simulations can be run either fully coupled to one of the two hydrodynamic models available in the Environmental Modelling Suite (SHOC – Sparse Hydrodynamic Ocean Code; COMPAS – Coastal Ocean Marine Prediction Across Scales), or in an off-line mode where the current, temperature and salinity fields from an earlier hydrodynamic run are used to force the BGC model.

### 10.1 Fully-coupled mode

By convention, model configuration files that are run in the fully-coupled mode have endings `.prm`.

To execute the model in the fully coupled SHOC model:

```
./shoc -p test_estuary_spectral.prm
```

To execute the model in the fully coupled COMPAS model:

```
./compas -p test_estuary_spectral.prm
```

### 10.2 Off-line transport mode

There are two transport models available, one in SHOC and one in COMPAS.

This is just a quick description of the off-line transport model, more details are available in the SHOC user guide.

The following files has been saved when running the hydrodynamic model:

<code>file8.name</code>	<code>out/test_estuary_tran.nc</code>
<code>file8.filetype</code>	<code>sparse</code>
<code>file8.tinc</code>	<code>1 hour</code>
<code>file8.bytespervalue</code>	<code>4</code>
<code>file8.vars</code>	<code>eta salt temp ulmean u2mean wmean Kzmean swr</code>

The transport model then reads `test_estuary_tran.nc` and runs the BGC and sediment in an off line version. This file need to be copied across to the trans folder (so it does not get deleted).

Instead of using a `test_estuary_spectral.prm` as a main parameter file, we use a stripped version called `test_estuary_spectral.tran` (grid and model forcing are removed only replaced by transport model options, all the ecology tracer list sediment ocean boundary stay the same, expect the flag advection and diffusion for temp and salt need to be set to 0 as they are read from files).

If the model source code has been configured with the `--enable-omp` option during configuration before compiling (see SHOC user guide):

```
./conf/configure --enable-omp
```

Then the BGC model can be run using more than 1 CPU using the following option specified in the `test_estuary_spectral.tran`

```
eco_omp_num_threads 12 # 12 cpus in this case
```

To run the transport model in the two models:

```
./shoc -t trans/test_estuary_spectral.tran  
./compas -t trans/test_estuary_spectral.tran
```

# 11 Computational diagnostics

The model outputs a few files during execution that are informative. Files written during the initialization are:

1. **setup.txt** – describes the model configuration, including tracers, boundary conditions etc. This is from the transport model.
2. **sedlog.txt** – outputs at the beginning the sediment configuration, plus any errors in the sediment model (requires **VERBOSE\_SED** to be activated in the in.tran or in.prm file).
3. **ecology\_setup.txt** – describes the optical model characteristics, and lists the biological parameters read and used, as well as the processes. Arguments given in the process list, such as gases calculated in gas\_exchange\_wc etc. are also listed.
4. **optical\_setup.nc** – netcdf file containing the spectrally-resolved parameters on the optical grid.
5. **optical\_setup\_matlab\_check.m** – a matlab script for plotting the optical configuration. In particular, because this file is generated during model setup it only displays optical properties that have been implemented in the simulation.
6. **optical\_setup\_python\_check.py** – a python script for plotting the optical configuration (see above).
7. **ecology\_parameter\_setup.nc** – a netcdf file containing all of the biological parameters that have been read from the biological parameter file. Any variables in the biological parameter file, but not in this file, have not used by the model run. The file does not contain biological variables that use the default value from the code.

Files written during the initialization and model integration:

8. **diag.txt** – this file is written to each ecological time step, and updates computing performance and expected finish time.
9. **runlog** – outputs during runtime warnings and errors.

The files **ecology\_setup.txt**, **setup.txt**, **ecology\_parameter\_setup.nc**, **optical\_setup.nc**, **optical\_setup\_python\_check.py** and **optical\_setup\_python\_check.m** are automatically copied to the OUTPUT folder at the completion of the model initialization.

## 11.1 Mass balance

Like all biogeochemical models, maintaining mass balance is an important requirement of the biogeochemical model. The **Biogeochemical Science Guide** details the sink / source equations that are conserved. The model code contains checks for the conservation of carbon, nitrogen, phosphorus and oxygen. This check involves calculating the mass before and after each ecological step. The difference between the two is calculated. If the difference exceeds a relative tolerance (which is printed in **ecology\_setup.txt**, and is calculated as the mass difference over the original value), then the model prints an error to the **runlog**. The hardcoded value for the relative tolerance is  $10^{-9}$  for the water column and sediment, and  $10^{-6}$  for the epibenthic. The exception to



the above calculation is oxygen, for which the mass difference is divided by the minimum of the original mass and  $8000 \text{ mg m}^{-3}$ .

In order to implement the mass balance the process list must contain `massbalance_wc`, `massbalance_epi` and `massbalance_sed` in the process list.

To determine mass budgets, including external inputs, circulation etc. see the SHOC manual region files.

## 12 RECOM / test case

[TO BE COMPLETED]

age.ts : time series file for tracer age concentration at the river boundary (set to 0 in this example)

dicriver.ts : time series file for tracer DIC and alkalinity concentration at the river boundary

gbr\_gravel\_600x180.txt : text file with gravel data to initialize the model

input.nc : initial condition file created using `./shoc -g prm/bgc/prm data/input.nc`

met.ts : time series file for meteorological forcing (time varying but spatially constant across the model domain)

passive0.ts : point source data file that contain NO<sub>3</sub> and passive tracer fluxes need to be in

data.txt : text file with geographic short wave radiation bottom absorption ~ spatially variable parameter.

flow.ts : time series file with river flow temperature and salinity condition at the river boundary.

gbr\_mud\_600x180.txt : text file with mud data to initialize the model

nutrient\_loads.ts : time series file with nutrient and sediment concentration at the river boundary (flow is also present but not used in fully coupled, only when used in transport mode).

dicocean.ts : time series file for tracer DIC and alk concentration at the river boundary

forcing.ts : time series file for meteorological forcing (time varying but spatially constant across the model domain)

gbr\_sand\_600x180.txt : text file with sand data to initialize the model

met2.ts : time series file for meteorological forcing (time varying but spatially constant across the model domain)