Getting started with ETRAC: A model-independent Element TRACing software for biogeochemical models

Disclaimer: This document describes an ETRAC test case for the northern Gulf of Mexico (NGoM) based on the Regional Ocean Modeling System (ROMS; Haidvogel et al., 2008) on the ComputeCanada cluster Cedar. If you need to set up ETRAC for a different case, please get in touch with Fabian Große (grosse@bafg.de).

1 Compile ROMS for ETRAC

- Log onto the Graham cluster and change to the directory where you want to place your copy of ROMS for ETRAC
- Get your version of ROMS 3.7 with an implementation of output for ETRAC from https://github.com/FabianGrosse/ROMS_3.7_for_ETRAC.git using:

  git clone https://github.com/FabianGrosse/ROMS_3.7_for_ETRAC.git
- Move the directory **test_case_NGoM** to another location of your choice. This folder contains the setup and input files for the northern Gulf of Mexico test case
- Enable the following CPP flags in **mch_bio_etrac.h** located in **test_case_NGoM**

  (mandatory)
  - SOLVE3D => 3-D model
  - BIO_FENNEL => Fennel biology
  - BIO_SEDIMENT => immediate benthic remineralization
  - DENITRIFICATION => benthic denitrification
  - TBNT_OUT => output needed for ETRAC
  - PSRC_UV_BUGFIX => inhibits mass fluxes from water on land (see: https://www.myroms.org/forum/viewtopic.php?f=19&t=4643)

  (optional, but recommended)
  - TBNT_NFDOUBLE => write double-precision output for ETRAC allowing for more accurate mass balances
  - PERFECT_RESTART => ensure accurate mass balance in case of a model restart (e.g. during a multi-year simulation on a cluster with job time limits)
- In **ROMS_3.7_for_ETRAC**, execute **chmod 700 ROMS/Bin/***
- Change directory to **test_case_NGoM**
- Execute **module purge**
- Execute **module load yaxt udunits gsl hdf5 netcdf netcdf-fortran**

- In **build.sh**, check that the following statement is included:

  setenv ROMS_APPLICATION MCH_BIO_ETRAC

- In **build.sh**, check that **MY_ROOT_DIR** points to the directory with your copy of **ROMS_3.7_for_ETRAC**
- Execute **chmod 700 build.sh**, then execute **./build.sh** to compile ROMS. This may take a few minutes.

2 Set up and run ROMS for ETRAC

- For the NGoM test case, make sure you have the files listed below available and located in one folder. These files are too big to upload to github. For access, please get in touch with Fabian Große (grosse@bafg.de)
    - mch_atmo_frc_1980_2016.nc
    - mch_bry_carbonHZ_PO4_WOA_TAmean_1984_2020.nc
    - mch_grd.nc
    - mch_ini20000101_049.nc
    - mch_river_1982_2016_Rdet_prov.nc
    - mch_wind_narr_3h_2000_2016.nc
- In **run_ROMS_CHAIN.sh**, set variable **inputPath** so that it points to the folder containing the files listed above. The file names are defined in **ocean.in** (see variable **oceanBase**). Define your simulation setup etc. at the beginning of **run_ROMS_CHAIN.sh** according to the instructions/comments in the file.
- In **bio_Fennel.in** file (see variable **bioFile** in **run_ROMS_CHAIN.sh**), make sure that **TBNTout(NITROGEN) == T** is set after the **Dout** list (diagnostics output). This switch activates the output needed for ETRAC.
- In **ocean_CHAIN-JOB.in,** set the name of the ROMS output files used for ETRAC as you like by adapting the value of variable **TBNTNAME**
- Execute **module purge**
- Execute **module load yaxt udunits gsl hdf5 netcdf netcdf-fortran**
- Execute **chmod 700 run_ROMS_CHAIN.sh**
- When running only a short test with manual job submission (to be defined in run_ROMS_CHAIN.sh), execute **./run_ROMS_CHAIN.sh**
- When running a series of jobs, execute **nohup ./run_ROMS_CHAIN.sh &**
- If you did not apply further changes to **run_ROMS_CHAIN.sh**, this will launch a 5-day job starting on January 1, 2000. Note that if you start from initial conditions and not from a restart file you will need to run for at least about two months, so

that tracers, which are not included in the initial conditions but enter the domain, e.g. through river input (like river detritus), can be distributed in the domain. Otherwise ETRAC is likely to fail.

3 Compiling ETRAC

- Change into the directory where you want to create your copy of ETRAC
- Execute **git clone https://github.com/FabianGrosse/ETRAC.git**
- Change into directory **ETRAC/software**
- Execute **chmod 700 compile.sh**
- Execute **./compile.sh**
- The compiled code and executable are now in the new **Build** folder
- **NOTE #1**: *make* configuration files for different architectures are located in **/software/src/make-config**. If there is no file available for your architecture and compiler, you need to create one.
- **NOTE #2**: The **makefile** in **ETRAC/software/src** provides a few CPP compiler flags. The current definitions are the ones required for ROMS applications (*TBNTconvert_3Dto1D* and *TBNTmass_fluxes*). The function of all CPP flags is described in the comments in the makefile.

4 Set up and run ETRAC (general description)

- In general, you need to modify the following files according to your setup (see comments in files) before running ETRAC.
    - In the directory **setup_files/model_setup**:
        - **model_dummy_vars.txt** => list of variables that are used internally by the model and can be calculated from other available variables by simple multiplication with a constant factor, but are not stored in the model output
        - **model_fluxes.txt** => a list of all model fluxes and the variables they link to each other
        - **model_grid.txt** => the model grid described in terms of 1-D vectors of neighboring cells for each spatial dimension, and surface and bottom cell indices
        - **model_iDep.txt** => an ASCII "map" of the model grid with the number of wet cells per horizontal grid cell
        - **model_rivers.txt** => a list of all river input locations in the 1-D indexing scheme; river names used in this file are used to for definition of river source groups

All these files are mandatory for running ETRAC. The files **model_grid.txt**, **model_idep.txt** and **model_rivers.txt** can be generated with the tools provided in **tools/make_ETRACcontrol4ROMS**.

- In the directory **setup_files/etrac_setup**
  - **etrac_openb_source.txt** => a list of open boundary input sources defined as grid cell indices of source and target cells using the 1-D indexing scheme
  - **etrac_atmos_source.txt** => a list of open boundary input sources defined as grid cell indices using the 1-D indexing scheme
  - **linked_fluxes.txt** => a list of fluxes for which relative contributions can be calculated based on selected state variables
  - **target_areas.txt** => list of target/integration regions defined as 1-D index vectors; simple .csv output will be created for these regions
  - **target_variables.txt** => list of target variables to be stored in .csv files for target areas

  Only **etrac_openb_source.txt** is mandatory for running ETRAC. **etrac_atmos_source.txt** is only needed if the tracing includes air-sea fluxes. Both files can be generated with the tools provided in **tools/make_ETRACcontrol4ROMS**. The other files are only needed, if the user wants to analyze linked fluxes or target variables/areas.

- In the **ETRAC** main directory
  - **etrac_set_BASE.nml** => file in which your ETRAC setup is defined. It points to the setup files listed above and to the files containing the bulk variables and fluxes and the initialization (if applicable). It uses placeholder strings, which are replaced during the execution of **run_ETRAC_CHAIN.sh** (see below).
  - **ETRAC_SINGLE_BASE.slurm** => the batch job script used to run ETRAC with SLURM. It uses placeholder strings, which are replaced during the execution of **run_ETRAC_CHAIN.sh** (see below).
  - **run_ETRAC_CHAIN.sh** => the main script to run ETRAC; it replaces the placeholders in the two previous files and deals with running a sequence of jobs etc.

- After having prepared all your files, execute **chmod 700 run_ETRAC_CHAIN.sh**
- Then execute: **nohup ./run_ETRAC_CHAIN.sh &**
- **NOTE**: Currently, **run_ETRAC_CHAIN.sh** is designed to run (a series of) complete years using model output containing information for complete years, or individual shorter test jobs. However, ETRAC itself is capable of running shorter,

yet not longer jobs. That means **run_ETRAC_CHAIN.sh** could be adapted to allow running series of short jobs.

5 Set up and run ETRAC for the Northern Gulf of Mexico

5.1 Preparing setup files for ETRAC

- to prepare your setup files for ETRAC (e.g. location of Rivers, open boundaries etc.) use the tools provided in **ETRAC/tools/make_ETRACcontrol4ROMS/**
- change directory to **ETRAC/tools/make_ETRACcontrol4ROMS/**
- execute **chmod 770 compile.sh**
- execute **./compile.sh make_ETRACcontrol4ROMS**
- execute **cp setup_NGoMex.nml setup.nml** (The Fortran tool reads the setup.nml, which is domain specific; see another example for the BYECS domain)
- in **setup.nml**, make sure you provide the correct paths to the model grid netCDF file ("gridFile"), a model HIS output file ("hisFile"), and the river forcing netCDF file ("riverFile") (catz users can find appropriate files in the paths provided in **setup_NGoMex.nml**)
- in the folder **Input/nice-maps/NGoMex/etrac-maps**, make sure you provide ASCII maps of the model domain ("nice-maps"), which define the locations of open boundaries, grid cells to be excluded from the ETRAC calculation and, if applicable, the location of atmospheric sources (e.g. atmospheric deposition region; not needed for N tracing with ROMS); the following files are needed:
    - **etrac_openb_source_map_from_NGoMex.txt** => nice-map with origin cells of open boundary sources
    - **etrac_openb_source_map_to_NGoMex.txt** => nice-map with target cells of open boundary sources
    - **etrac_exclude_map_NGoMex.txt** => nice-map with grid cells to be excluded from ETRAC calculation (i.e. all cells outside the open boundaries incl. the open boundary origin grid cells)
- **NOTE:** The "domain" variable in **setup.nml** must match the subfolder name in **Input/nice-maps/[domain]/etrac-maps** and the filename strings in the nice-map files. An empty nice-map that only contains the land-sea mask can be created using the **make_nicemaps4ROMS** tool by providing a model grid netCDF file (see **grid_NGoMex.nml**). The empty nice-map can be used to create a different setup of open boundaries. An example of an empty nice-map is located in: **Input/nice-maps/NGoMex/etrac-maps/ROMS_NGoMex_nice-map.txt**
- (optional) in the folder **Input/box-defs**, make sure you provide ASCII files describing your selected "target areas" (only needed if you want to use them

during ETRAC, see short description in Section 3 and comments in **setup_NGoMex.nml**, and example files in **Input/box-defs**)

- Execute **./make_ETRACcontrol4ROMS**
    - This may take a few moments
    - Output is stored in the subfolder **Output**
    - Output files starting with "model_" need to be copied to **ETRAC/setup_files/model_setup** ; all other output files need to be copied to **ETRAC/setup_files/etrac_setup**

5.2 Preparing ROMS output for ETRAC

- Currently, ETRAC is designed to either run (series of) complete years or an individual shorter job (e.g. 10 days for testing). However, ROMS output is not stored in yearly files but in multiple files with a certain output interval (e.g. 50 days for the NGoM). To create yearly netCDF files with all (nitrogen) model tracers and fluxes used as input for ETRAC, Climate Data Operators (CDO; https://code.mpimet.mpg.de/projects/cdo/embedded/cdo.pdf) can be used, the relevant commands are:
    - **cdo seltimestep** => to cut the relevant time indices for the first and last file of the year to be prepared (note that the data for the very last day of the previous year needs to be included in the first file of the year (i.e. December 31, 24:00/January 1, 00:00)
    - **cdo mergetime** => to merge the individuals files with data for the year into a single file with data for the entire year
    - **NOTE:** for the BYECS domain (East China Sea), you can not use ETRAC on the whole domain due to memory limitations (it's a sequential program), hence you need to cut the domain using, e.g. **cdo selindexbox**. However, this also implies that you need to provide the corresponding setup files (see Sections 4 and 5.1) for this subdomain.
- Place the file in the folder you want as input folder for ETRAC

5.3 Run ETRAC for the NGoM

- After having prepared all your input files and placed them in the right folders, change directory to: **ETRAC**
- **NOTE:** an initialization file and a bulk variable/flux to run the year 2001 can be found on **Cedar** in **/project/def-kfennel/grosse/ETRAC_data**
- Update the header of the main control script **run_ETRAC_CHAIN.sh** according to your job requirements (see comments in file)
- Update **etrac_set_BASE.nml**

    - File names in **&model_nml**
    - Other changes can be made as required (see comments in file)
- Execute **nohup ./run_ETRAC_CHAIN.sh &**
- Wait until the job is complete … this may take quite some time depending on your setup: a single year with Fabian's setup (Mississippi, Atchafalaya and one open-boundary source) takes about 1.5 days; the test is setup for 10 days (January 1-10, 2001)
- At the end of the log file (only if job succeeds within given job time), ETRAC prompts the mass balance for all bulk state variables computed from the initial mass and the cumulated fluxes in comparison to those read from the bulk file at the end of the calculation. The relative errors need to be very small ($<10^{-10}$) or something is wrong with your setup.