SELFE NetCDF output description

# Introduction

The purpose of the work is to output the SELFE simulation results in NetCDF format.

# What the subroutines do?

The NetCDF program does the following things:

1. Output the SELFE results into single NetCDF format based on the NetCDF conversion.(not like the original SELFE output, one processor generate one file)
2. The rank 0 processor will do the following things:
   1. Collect the data from other process
   2. Assemble the collected data with the right order (based on nodes and vertical level)
   3. Output the data to the NetCDF file
3. The other processors send the data to the rank 0 processors, these process will not do output process.

This procedure is proved as the most efficient way to output the results.

# Required package

## Windows:

1. Windows binary version of netcdf version: 3.6.1. (already in the package)

(need to put it in the appropriate location )

1. MPI library, currently MSMPI is used. Mpich2 is also possible
2. Windows version of “METISlib.lib” and “ParMetisLib.li”. (already in the package)

## Linux :

The same as original SELFE except need to use the netcdf library.

The folder\_linux is used for linux operation system.

# Subroutine components:

The package provides the windows version project, which can only run under the release module and the make file for Linux to run:

1. data\_type\_mod.F90: define the data type used.
2. check\_error.F90: define the NetCDF check error part and also the used module.

**NOTE:**

1. **For windows: need to use the module “netcdf90” and “funMaxModule”**
2. **For Linux : comment the “netcdf90” and “funMaxModule” and use “netcdf”**
3. nsf\_container.F90: pack the NetCDF model
4. nsf\_self\_vars.F90: define the variables used for NetCDF output
5. nsf\_selfe\_out\_8.FI: the file included in the SELFE CODE for output

# How to use the NetCDF output subroutines

1. Use the “use nsf\_self\_vars\_mod” in the main program.
2. Include the “nsf\_selfe\_out\_8.FI” at the output part “IF(iwrite == 0 .OR. iwrite ==1 ) THEN ”
3. In param.in if iwrite =1 : original selfe output, if iwrite =2 netcdf output

# How to compile in Linux

The program is tested on CETO, set the variables in CETO and type make.

Don’t forget to change the used models in file “check\_error.F90”

# Details of the subroutines

1. nsf\_selfe\_nectcdf\_out\_init: init the netcdf output by define the variables and so on.
2. nfs\_self\_create\_file: create the netcdf file, define the array and properties.
3. nsf\_selfe\_out\_grid: output the SELFE grid to netCDF file (not time dependent)
4. nfs\_collect\_rdata\_2d: gather the scalar data in netcdf;
5. nfs\_collect\_rdata\_3d: gather the 3D scalar data in netcdf;

# Netcdf output benchmark:

1. # of element 149196 , # of nodes 79786, # of vertical layer: 31. Processors: 32
2. Output variables: elev, vert, temp, salt, hvel;
3. Output data size/step: ~66M
4. Output time : <5s;
5. The time consumption for output is in filer “mirror.out” , for example:

start to write data to netcdf fileoutputs/1.nc

Finished to output netcdf file:

Write data to netcdf file takes: 3.30seconds