How to set up an MPP configuration with NEMO OPA9 ?

J.M. Molines [*]

update for netcdf files: May 26, 2005

This document quickly describes a set of tools that were developped to facilitate the setting up of MPP configuration. It is supposed that all the programs described here are installed and compiled. (Ask Jean-Marc.Molines@hmg.inpg.fr for the sources).

Process the bathymetric file

- The starting points for this process is the bathymetry file, netcdf format, either the level one or the meter one, as soon as they have the same land/sea mask. For visualization purposes it is probably more interesting to work with the 'meter' file. It is important to consider the very same file than for the model run. The programs used hereafter use a 'namelist' file to know various parameter. The namelist file is described in Appendix, but note that the bathymetric file name is one of the item of the namelist. It is associated with a logical flag (ln_zps) to known if the bathymetry is for model level (ln_zps=.false.) of in meters (ln_zps=.true.)

- The `mpp_optimiz_zoom_nc.f90` program was formely written by Maurice Imbard at Lodyc, during the early CLIPPER years. It was slightly modified to take its input from a namelist. It basically try almost all the possible processor layout within a given range and output a log file (`processor.layout`). Namelist file is decribed in appendix A.

- Choose in the log file the domain decomposition (in term on jpni, jpnj) that you want to keep, and put those values in the KEEP section of the namelist. In the same block give a root-name to the output file of visualization program. The program gives its 'best' choice at the end of the file. It is not necessarily the one you want to keep, but you must

---

[*]Laboratoire des Ecoulements Géophysiques et Industriels, CNRS UMR 5519, Grenoble, France

consider the real number of points used in the computation. For the same amount of processors, the best choice corresponds to the lower number.

- Run the program `mppopt_showproc.f90` that will create an overlay file (very simple ASCII format), ready to be used with a chart script, or another adecuate graphic package.
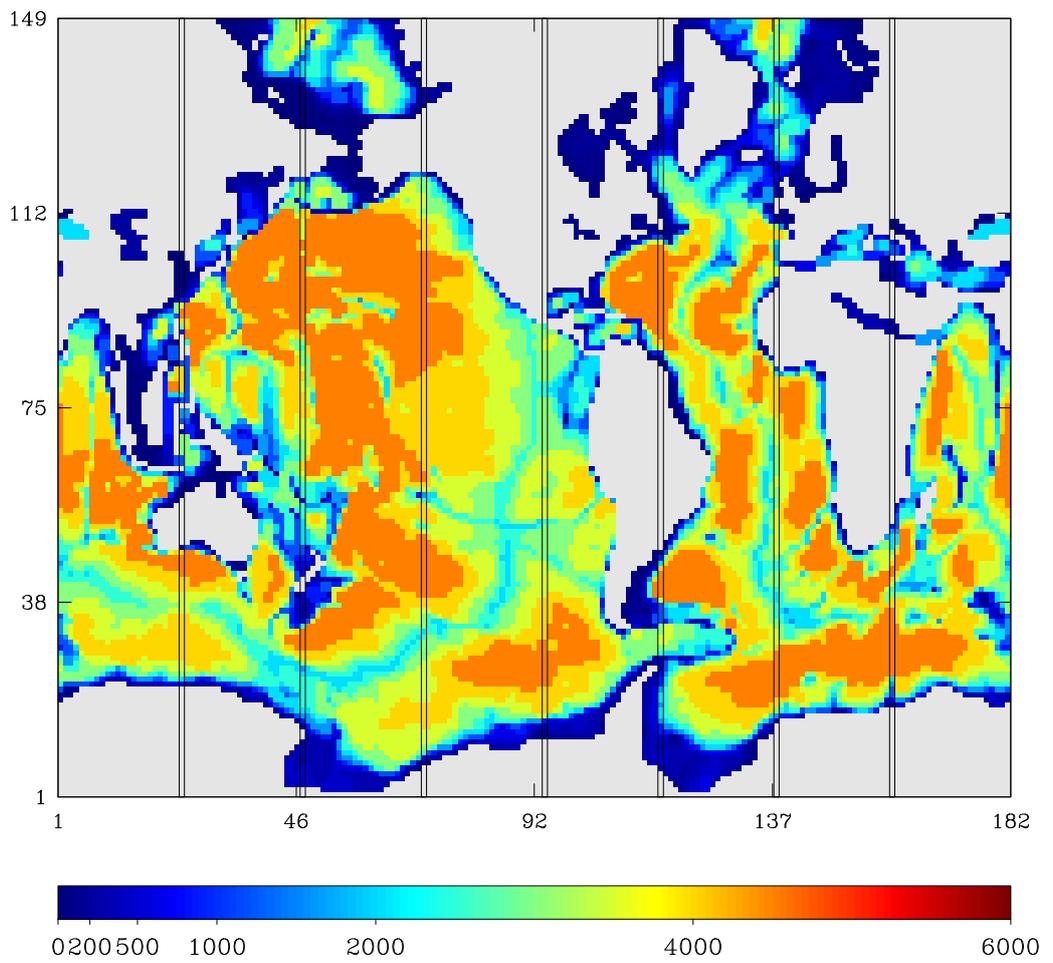
Look at the domain decomposition (chart users).

A script file is proposed (`plover_nc.csh`) which takes the name of the over-data file as input. It produces a cgm file with the same name and the cgm extension. This script needs the namelist file (to retrieve the name of the bathy file, converted in bimg) and a color mark file (`bathy.clrmark`). The results look like figure 1 or 2.

Examples

This set of preprocessing program is 'distributed' as a tar file for the MPP_PREP-1.0 directory. In this directory, there are the source codes (SRC), this file in (DOC) and some examples built on ORCA2 grid (EXAMPLES). The example have been run for 2 domain decomposition

- **jpni= 4, jpnj = 8, jpnij = 8** , with all processors as oceanic processors. The results and plot are under the 004x002_008 directoy

- **jpni= 7, jpnj = 11, jpnij = 69** , with 8 land processors. Again results and plots are under the 007x011_069 directory.
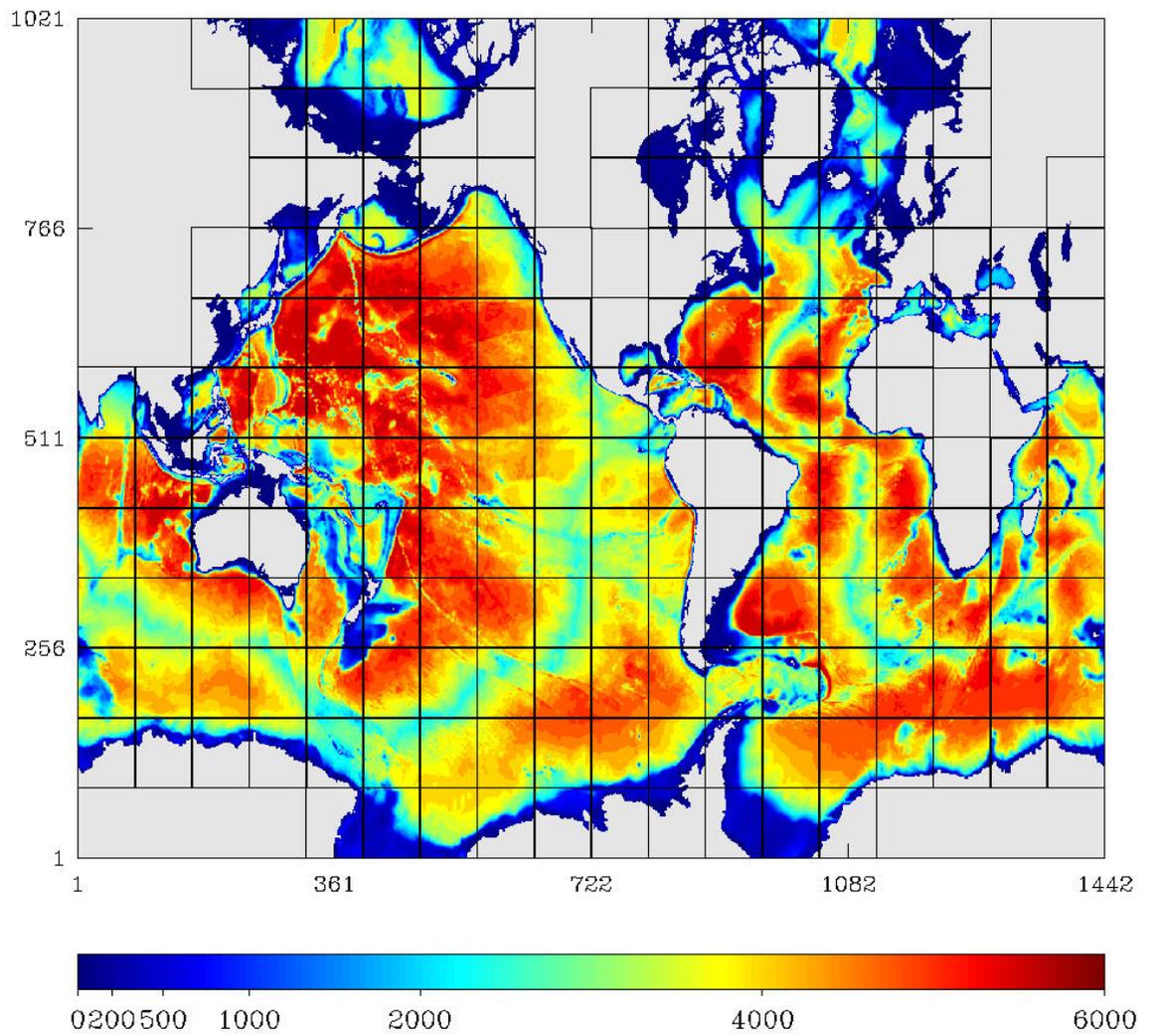
Figure 1: Example of domain decomposition for ORCA2 model

Figure 2: Example of domain decomposition for ORCA025 model. Some processors with only land points are eliminated.

A   Namelist

```
!>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
!  MPP_OPTIMIZ_ZOOM namelist
! -------------------------
!>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
!
!,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
!      namspace  spatial indexes
!,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
! NAMELIST /namspace/ jpiglo,jpjglo,jpidta,jpjdta,nizoom,njzoom
! jpiglo = overall size of the zoomed region (i-direction)
! jpjglo = overall size of the zoomed region (j-direction)
! jpidta = overall size of the domain (i-direction)
! jpjdta = overall size of the domain (j-direction)
! nizoom = i -index of point (1,1) of the zoomed region/ jpidta
! njzoom = j -index of point (1,1) of the zoomed region/ jpjdta
!
&NAMSPACE
    jpk=31
    jpiglo =  182
    jpjglo =  149
    jpidta =  182
    jpjdta =  149
    nizoom = 1
    njzoom = 1
/
!,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
!     namproc
!,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
!   jprocx = maximum number of proc
!   jpmem = 0 : dont care about the use of memory
!           1 : try to optimize the use of memory
!
&NAMPROC
     jprocx= 8
     jpmem = 0
/
!,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
!      namparam
!,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
!  ppmcal = memoire en octet d'un processeur
!  ppmin = ??
!  ppmax = ??
```

```
!
&NAMPARAM
   ppmcal = 225000000.
   ppmin = 0.4
   ppmax = 0.9
/
!
!,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
!      namfile   of filename
!,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
! NAMELIST /namfile/ cbathy
! cbathy = name of the ASCII bathymetric file
!
&NAMFILE
     cbathy='bathymetry.orca2.nc'
     ln_zps=.true.
/
!
!,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
!      namkeep   jpni jpnj   kept
!,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
! NAMELIST /namkeep/ jpni,jpnj,covdta
!   jpni = number of procs. in the i-direction
!   jpnj = number of procs. in the j-direction
!   covdta = Root for the overdata file name .
!          Complete name will be {covdta}.{jpni}x{jpnj}_{jpnij}
&NAMKEEP
   jpni = 1
   jpnj = 8
   covdta = 'ORCA2-zahir'
/
```