Perturbation Tool (V4r4): Installation and Use

Ichiro Fukumori, Ou Wang, and Ian Fenty

DRAFT

August 29, 2022

Introduction

This document describes steps to set up and run programs and scripts to compute the ECCO ocean model's (V4r4) response to perturbations in its control ("Perturbation Tool"); i.e., forward sensitivity. The tool is useful for studying the ocean's response to changes in the controls and to assess the accuracy of adjoint gradients. Available perturbations are, at present, those listed in steps 17) & 18) below.

The ocean model is the flux-forced version of ECCO's Version 4 release 4 (V4r4) ocean state estimate described in Wang et al. [2021]. The output of the Perturbation Tool consists of differences in time-series of model state (sea level, ocean bottom pressure, temperature, salinity, velocity) between model integrations with and without the perturbation, normalized by dividing the results by the amplitude of the perturbation. Perturbation to the control is at a particular model grid point at a particular time defined weekly, starting from 12Z January 01, 1992, which is the starting instant of V4r4. The model time-step is 1-hour and the perturbation is interpolated linearly in time.

A. Steps to set up the tool

User commands/input are given in red. (Steps 1-4 in italic are the same as those for setting up the bulk-formula version of the model described in Wang et al. [2020].) pfe25> denotes Unix prompt.

1) *Create and cd to a work directory*

   pfe25>*mkdir WORKDIR*

   pfe25>*cd WORKDIR*

2) *Download MITgcm "checkpoint 66g"*

   pfe25>*git clone https://github.com/MITgcm/MITgcm.git -b checkpoint66g*

3) *Create and cd to subdirectory*

   pfe25>*cd MITgcm*

   pfe25>*mkdir -p ECCOV4/release4*

pfe25>*cd ECCOV4/release4*

4) *Download V4 configurations*

pfe25>*git clone https://github.com/ECCO-GROUP/ECCO-v4-Configurations*

5) Extract flux-forced configuration of the model

pfe25>mv ECCO-v4-Configurations/ECCOv4\ Release\ 4/flux-forced .

pfe25>rm -rf ECCO-v4-Configurations

pfe25>cd flux-forced

pfe25>set basedir=`pwd`

pfe25>mkdir forcing

6) Download forcing from ECCO Drive. (Substitute username "*fukumori*" below with your own username and use your WebDAV password, <u>**NOT**</u> your Earthdata account password.) The second wget will take a while to complete.

pfe25>wget -P forcing -r --no-parent --user *fukumori* --ask-password -nH --cut-dirs=4 https://ecco.jpl.nasa.gov/drive/files/Version4/Release4/input_init

pfe25>wget -P forcing -r --no-parent --user *fukumori* --ask-password -nH --cut-dirs=4 https://ecco.jpl.nasa.gov/drive/files/Version4/Release4/other/flux-forced

7) Load module for compilation.

pfe25>*module purge*

pfe25>*module load comp-intel/2020.4.304*

pfe25>*module load mpi-hpe/mpt.2.25*

pfe25>*module load hdf4/4.2.12*

pfe25>*module load hdf5/1.8.18_mpt*

pfe25>*module load netcdf/4.4.1.1_mpt*

pfe25>*module list*

8) Compile MITgcm program (generates executable "mitgcmuv")

pfe25>mkdir build

pfe25>cd build

pfe25>../../../../tools/genmake2 -mods=../code
-optfile=../../../../tools/build_options/linux_amd64_ifort+mpi_ice_nas -mpi

pfe25>make depend

pfe25>make all

pfe25>cd ..

9) Download scripts and programs for the Perturbation Tool and compile the tools.

   pfe25>*git clone https://github.com/ECCO-GROUP/ECCO-EIS.git*

   pfe25>mv ECCO-EIS/tool_pert .

   pfe25>rm -rf ECCO-EIS

   pfe25>cd tool_pert

   pfe25>make all

10) Download data files needed by the Perturbation Tool (***pert_xx.grid*** and ***pert_xx.scale***). Substitute username "*fukumori*" below with your own username and use your WebDAV password, **NOT** your Earthdata account password.

   pfe25>wget -r --no-parent --user *fukumori* --ask-password -nH --cut-dirs=7
   https://ecco.jpl.nasa.gov/drive/files/Version4/Release4/other/flux-forced/tool_pert_data

11) Copy files that will be modified, just in case. (optional)

   pfe25>cp -p gen_pert_ref.csh gen_pert_ref.csh_orig

   pfe25>cp -p tool_pert.csh tool_pert.csh_orig

   pfe25>cp -p setup.csh setup.csh_orig

   pfe25>cp -p README_pert README_pert_orig

12) ***Modify*** MITgcm's namelist file "data". (Can be skipped for initial testing.)

In the tool_pert directory in step 9) above, there are three data* files that control the behavior of MITgcm suitable for the Perturbation Tool. In particular, the namelist file "data" is configured for test purposes and should be changed for production runs as below.

Specifically, variable nTimeSteps in file "data" controls the number of time-steps MITgcm will integrate. The tool_pert version in 9) uses

nTimeSteps=13110,

which amounts to an 18-month integration period used for testing purposes. To run the full 26-year period of V4r4, select

nTimeSteps=227903,

On Pleiades, the 18-month and 26-year integrations take 35-min and 10-hours walltime, respectively.

13) ***Modify*** scripts "tool_pert.csh" and "gen_pert_ref.csh" as needed.

Script tool_pert.csh is a script for users to run the perturbation tool. Script gen_pert_ref.csh creates the model's reference without the perturbation. These files downloaded in step 9) are PBS scripts for running the tool in the devel queue on NAS Pleiades; job controls (#PBS …) and modules should be adapted to the computing platform.

In particular, as in namelist file "data" in step 12), the two scripts downloaded are configured for test purposes by limiting the walltime to 2-hours using the "devel" queue;

```
#PBS -l walltime=2:00:00
#PBS -q devel
```

Revise/remove these variables/parameters as needed. For reference, 18-month and 26-year integrations take approximately 35-minutes and 10-hours walltime, respectively, in the Tool's particular configuration on Pleiades.

14) ***Modify*** scripts.

```
pfe25>sed -i -e "s|SETUPDIR|${basedir}|g" gen_pert_ref.csh
```

```
pfe25>sed -i -e "s|SETUPDIR|${basedir}|g" tool_pert.csh
```

15) Run Perturbation Tool without perturbation to obtain reference results.
This job will produce results under a new directory named **run_pert_ref** in basedir (step 5).

```
pfe25>qsub gen_pert_ref.csh
```

16) Copy tools (**tool_csh.csh**, **README_pert**) for user access. Set **FORUSERDIR** below to a full directory path name where you want to install (copy) the tools at.

```
pfe25>set useraccessdir=FORUSERDIR

pfe25>mkdir ${useraccessdir}

pfe25>sed -i -e "s|SETUPDIR|${basedir}|g" setup.csh

pfe25>sed -i -e "s|PUBLICDIR|${useraccessdir}|g" setup.csh

pfe25>sed -i -e "s|PUBLICDIR|${useraccessdir}|g" README_pert

pfe25>cd ${useraccessdir}

pfe25>cp -p ${basedir}/tool_pert/setup.csh .

pfe25>cp -p ${basedir}/tool_pert/README_pert .
```

17) Run Perturbation Tool as a user with user-defined perturbation.

We would like this step to be replaced by a web-based interface, with a feature (e.g., e-mail) to notify users when the computation is complete and where to retrieve the results (step 16).

Here, the perturbation is specified by program ***pert_nml.x*** and the model result is computed by PBS script **tool_pert.csh**.

Program ***pert_nml.x*** requires a user's response, specifying what the user wants to perturb. Example below perturbs "empmr" (evaporation minus precipitation minus river runoff) at model grid (85,601) at week 12 with magnitude -1e-3 (kg/m2/s). Program ***pert_nml.x*** will generate two text files ***pert_xx.nml*** and ***pert_xx.str*** that are used in PBS script **tool_pert.csh**.

File ***pert_xx.nml*** is a namelist file read by program ***pert_xx.x*** invoked by **tool_pert.csh** that perturbs the model's control. File ***pert_xx.str*** contains a character string used by **tool_pert.csh** in naming a directory ("***run_directory***") where the computation's output will be placed under USRDIR below.

The ***run_directory*** for the example here will be **run_pert_1_85_601_12_-1.00E-03**, which bears a description of the perturbation in its name. The results to be provided to the user will be placed in sub-directory **pert_result_output** under this ***run_directory*** (see step 18).

We would like to replace ***pert_nml.x*** with a web interface, where users fill out the queries and submit the job (equivalent to submitting ***tool_pert.csh***) by clicking a button ("submit job").

Create and go to a user directory (**USRDIR** below, change as needed) to run the tool, separate from where the tool was set up above.

```
pfe25>mkdir USRDIR

pfe25>cd USRDIR
```

Run **setup.csh** to set up the perturbation tool in the user directory. (Replace **FOREUSERDIR** below with what it was specified in step 16.)

```
pfe25>source FORUSERDIR/setup.csh

pfe25>pert_nml.x
 Available control variables to perturb ...
    1) empmr
    2) pload
    3) qnet
    4) qsw
    5) saltflux
    6) spflx
    7) tauu
    8) tauv
   Enter control ... (1- 8) ?
1
  ..... perturbing empmr

Choose location for perturbation ...
    Enter 1 to choose native grid location (i,j),
         9 to select by latitude/longitude ... (1 or 9)?
1
    Enter native (i,j) grid to perturb ...
   i ... (1-90) ?
85
   j ... (1-1170) ?
601
  ...... perturbation at (i,j) =          85         601
       which is (long E, lat N) =  -151.8  73.5
       Depth (m) at this location =   3808.8

Enter week to perturb ... (1-1358) ?
12
  ...... perturbing week =           12

 Default perturbation =   -1.00000005E-03
       in unit kg/m2/s (upward freshwater flux)
 Enter 1 to keep, 9 to change ... ?'
```

1
Wrote pert_xx.nml
Wrote pert_xx.str

pfe25>qsub tool_pert.csh

18) Results of the Perturbation Tool will be in sub-directory **pert_result_output** under the ***run_directory*** described in step 17).

Users only need files in this sub-directory, once the computation of step 17) is complete.

pert_xx.nml:
Namelist file with specifics of the perturbation saved for reference. Created by **pert_nml.xx** and used by **pert_xx.x** to perturb controls of MITgcm.

state_2d_set1_day.***TIMESTEP***.data
state_2d_set1_day.***TIMESTEP***.meta

state_2d_set1_mon.***TIMESTEP***.data
state_2d_set1_mon.***TIMESTEP***.meta

state_3d_set1_mon.***TIMESTEP***.data
state_3d_set1_mon.***TIMESTEP***.meta

Result of the perturbation in MITgcm diagnostic output format; "data" are binary, "meta" are text files with "data" file information. The ***TIMESTEP*** in the filenames are model time-steps (center step of average); each file corresponds to a particular instant. The fields are on the model's native grid.

Files "**state_2d_set1_day**" have daily mean dynamic sea level (ssh) and ocean bottom pressure (obp) on the model's 2-dimensional horizontal grid. Files "**state_2d_set1_mon**" have monthly mean dynamic sea level (ssh) and ocean bottom pressure (obp). Units are meters in equivalent sea level for both.

Files "**state_3d_set1_mon**" have monthly mean temperature (theta; deg C), salinity (salt; PSU), i-direction velocity (uvel; m/s), and j-direction velocity (vvel; m/s) on the model's 3-dimensional grid.

Results are normalized as the model's response to a unit perturbation of the control (control's unit noted by **pert_nml.x** in step 17). For reference, the units for the different controls are;

control (1) = 'empmr' 'kg/m2/s (upward freshwater flux)'
control (2) = 'pload' 'N/m2 (downward surface pressure loading)'
control (3) = 'qnet' 'W/m2 (net upward heat flux)'
control (4) = 'qsw' 'W/m2 (net upward shortwave radiation)'

control (5) = 'saltflux''g/m2/s (net upward salt flux)'
control (6) = 'spflx' 'g/m2/s (net downward salt plume flux)'
control (7) = 'tauu' 'N/m2 (westward wind stress)'
control (8) = 'tauv' 'N/m2 (southward wind stress)'

Example code to read temperature, theta (the first record; irec), from file state_3d_set1_mon.0000012396.data as variable "fvar".

FORTRAN

```
      integer nx, ny, nr
      parameter (nx=90, ny=1170, nr=50)
      integer irec
      real*4 fvar(nx,ny,nr)
      character*256 f_in

      f_in = 'state_3d_set1_mon.0000012396.data'
      open(60, file=f_in, access='direct',
     $     recl=nx*ny*nr*4, form='unformatted')

      irec = 1
      read(60,rec=irec) fvar
```

IDL

```
nx = 90
ny = 1170
nr = 50

f_in = 'state_3d_set1_mon.0000012396.data'
close,1 & openr,1,f_in,/swap_if_little_endian
d_file = assoc(1,fltarr(nx,ny,nr))

irec = 0
fvar = d_file(irec)
```

MATLAB

```
nx = 90;
ny = 1170;
nr = 50;

f_in = 'state_3d_set1_mon.0000012396.data';
fid=fopen(f_in,'r','ieee-be');

irec = 1;
```

```
status=fseek(fid,(irec-1)*(nx*ny*nr*4),'bof');
fvar=fread(fid, [nx*ny*nr], 'single');
fvar=reshape(fvar, [nx,ny,nr]);
fclose(fid);
```

PYTHON

```
import numpy as np
nx = 90
ny = 1170
nr = 50

f_in = 'state_3d_set1_mon.0000012396.data'
dt =  np.dtype([ ('fld', '>f4', (nr,ny,nx))])
d_file = np.fromfile(f_in,dtype=dt)

irec = 0
fvar = d_file[‘fld’][irec]
```

B. Perturbation Tool on Pleiades

The perturbation tool has been set up on Pleiades and can be run by the following steps.

1) Create and cd to a work directory. Results and pertinent files will be created under this directory.

   pfe25>*mkdir USRDIR*

   pfe25>*cd USRDIR*

2) Set up perturbation tool.

   pfe25>source /nobackup/ifukumor/ECCO_tools/pert/setup.csh

   For demonstration purpose, and to avoid unintentional long computations, this step sets up a namelist file “**data**” that limits the model’s duration of integration to 18-months (variable “**nTimeSteps**”; see step A-12). Similarly, the PBS script “**tool_pert.csh**” limits the walltime to 2-hours using the “devel” queue;
   #PBS -l walltime=2:00:00
   #PBS -q devel
   Revise/remove these variables/parameters as needed; See step A-13).

3) Specify perturbation (follow the prompt; see step A-17).

   pfe25>pert_nml.x

4) Submit PBS job to run the Perturbation Tool's computation (see step A-17).

   pfe25>qsub tool_pert.csh

5) Results of the computation will be generated in sub-directory **pert_result_output** under the "***run_directory***" (run_pert_*****) that will be created under USRDIR of step B-1). (See steps A-17 & A-18.)

References