Gael Forget
MIT, Jan. 29th 2015

Introduction to ocean data-model analysis

course 5: usage of the MITgcm

1. documentation
2. configuration(s)
3. model output
4. results analysis

5. interactive session : self-guided exercises
6. resources, bibliography

(1) documentation

Chapter 1

Overview of MITgcm

This document provides the reader with the information necessary to carry out numerical experiments using MITgcm. It gives a comprehensive description of the continuous equations on which the model is based, the numerical algorithms the model employs and a description of the associated program code. Along with the hydrodynamical kernel, physical and biogeochemical parameterizations of key atmospheric and oceanic processes are available. A number of examples illustrating the use of the model in both process and general circulation studies of the atmosphere and ocean are also presented.

1.1 Introduction

MITgcm has a number of novel aspects:

- it can be used to study both atmospheric and oceanic phenomena; one hydrodynamical kernel is used to drive forward both atmospheric and oceanic models - see fig 1.1
- it has a non-hydrostatic capability and so can be used to study both small-scale and large scale processes - see fig 1.2
- finite volume techniques are employed yielding an intuitive discretization and support for the treatment of irregular geometries using orthogonal curvilinear grids and shaved cells - see fig 1.3
- tangent linear and adjoint counterparts are automatically maintained along with the forward model, permitting sensitivity and optimization studies.
- the model is developed to perform efficiently on a wide variety of computational platforms.

Key publications reporting on and charting the development of the model are *Hill and Marshall* [1995]; *Marshall et al.* [1997b,a]; *Adcroft et al.* [1997]; *Marshall et al.* [1998]; *Adcroft and Marshall* [1999]; *Chris Hill and Marshall* [1999]; *Marotzke et al.* [1999]; *Adcroft and Campin* [2004]; *Adcroft et al.* [2004a]; *Marshall et al.* [2004] (an overview on the model formulation can also be found in *Adcroft et al.* [2004b]):

http://mitgcm.org/public/r2_manual/latest/online_documents/manual.pdf

(1) documentation

4.1. Build Tools

Many Open Source projects use the "GNU Autotools" to help streamline the build process for various Unix and Unix-like architectures. For a user, the result is the common "configure" (that is, "`./configure && make && make install`") commands. For MITgcm, the process is similar. Typical commands are:

```
$ genmake2 -mods=../code
$ make depend
$ make
```

4.2. The Verification Suite

The MITgcm CVS tree (within the `$ROOTDIR/verification/` directory) includes many (> 90) examples intended for regression testing. Each one of these test-experiment directories contains "known-good" output files along with all the input (including both code and data files) required for their re-calculation. Also included in `$ROOTDIR/verification/` is the shell script `testreport` to perform regression tests.

http://mitgcm.org/public/devel_HOWTO/devel_HOWTO.pdf

(2) configuration(s)

16. **adjustment.128x64x1** - Barotropic adjustment problem on latitude longitude grid with 128x64 grid points (2.8° resolution).

17. **adjustment.cs-32x32x1** - Barotropic adjustment problem on cube sphere grid with 32x32 points per face (roughly 2.8° resolution).
    Also contains a non-linear free-surface adjustment version (*input.nlfs/*).

18. **advect_cs** - Two-dimensional passive advection test on cube sphere grid (32x32 grid points per face, roughly 2.8° resolution)

19. **advect_xy** - Two-dimensional (horizontal plane) passive advection test on Cartesian grid.
    Also contains an additional set-up using Adams-Bashforth 3 (*input.ab3_c4/*).

20. **advect_xz** - Two-dimensional (vertical plane) passive advection test on Cartesian grid.
    Also contains an additional set-up using non-linear free-surface with divergent barotropic flow and implicit vertical advection (*input.nlfs/*).

http://mitgcm.org/public/r2_manual/latest/online_documents/manual.pdf

(2) configuration(s)

1. tutorial_advection_in_gyre - Test of various advection schemes in a single-layer double-gyre experiment. This experiment is described in detail in section 3.11.
2. tutorial_baroclinic_gyre - Four layer, ocean double gyre. This experiment is described in detail in section 3.10.
3. tutorial_barotropic_gyre - Single layer, ocean double gyre (barotropic with free-surface). This experiment is described in detail in section 3.9.
4. tutorial_cfc_offline - Offline form of the MITgcm to study advection of a passive tracer and CFCs. This experiment is described in detail in section 3.20.5.
5. tutorial_deep_convection - Non-uniformly forced ocean convection in a doubly periodic box. This experiment is described in detail in section 3.15.
6. tutorial_dic_adoffline - Offline form of MITgcm dynamics coupled to the dissolved inorganic carbon biogeochemistry model; adjoint set-up.
7. tutorial_global_oce_biogeo - Ocean model coupled to the dissolved inorganic carbon biogeochemistry model. This experiment is described in detail in section 3.17.
8. tutorial_global_oce_in_p - Global ocean simulation in pressure coordinate (non-Boussinesq ocean model). Described in detail in section 3.13.
9. tutorial_global_oce_latlon - 4x4 degree global ocean simulation with steady climatological forcing. This experiment is described in detail in section 3.12.
10. tutorial_global_oce_optim - Global ocean state estimation at 4° resolution. This experiment is described in detail in section 3.18.
11. tutorial_held_suarez_cs - 3D atmosphere dynamics using Held and Suarez (1994) forcing on cubed sphere grid. This experiment is described in detail in section 3.14.
12. tutorial_offline - Offline form of the MITgcm to study advection of a passive tracer. This experiment is described in detail in section 3.20.
13. tutorial_plume_on_slope - Gravity Plume on a continental slope. This experiment is described in detail in section 3.16.

http://mitgcm.org/public/r2_manual/latest/online_documents/manual.pdf

(1) general specifications

Parent Directory

global_oce_cs32/ — Mini setup

global_oce_llc90/ — Primary setup

README — Guidelines

setup_these_exps.csh — Install script

http://mitgcm.org/viewvc/MITgcm/MITgcm_contrib/gael/verification/

(2) configuration(s)

baudelaire	linux_amd64_g77	adjoint-taf	20150128	summary.txt	26:27
baudelaire	linux_amd64_g77	forward	20150128	summary.txt	89:95
baudelaire	linux_amd64_gfortran.dvlp	adjoint-taf	20150128	summary.txt	27:27
baudelaire	linux_amd64_gfortran.dvlp	tanglin-taf	20150125	summary.txt	19:19
baudelaire	linux_amd64_gfortran.dvlp	adjoint-oad	20150128	summary.txt	8:8
baudelaire	linux_amd64_gfortran.dvlp	forward	20150128	summary.txt	95:95
baudelaire	linux_amd64_gfortran.dvlp	restart	20150128	summary.txt	93:95
baudelaire	linux_amd64_gfortran+mpi.dvlp	adjoint-taf	20150128	summary.txt	21:23
baudelaire	linux_amd64_gfortran+mpi.dvlp	forward	20150128	summary.txt	84:88
baudelaire	linux_amd64_gfortran+mpi+mth.dvlp	forward	20150128	summary.txt	77:81
baudelaire	linux_amd64_gfortran+mpi+mth.dvlp	restart	20150128	summary.txt	80:81
baudelaire	linux_amd64_gfortran+mth.dvlp	forward	20150128	summary.txt	83:83
baudelaire	linux_amd64_ifort11.dvlp	forward	20150128	summary.txt	91:95

http://mitgcm.org/public/testing.html

glacier3	linux_amd64_gfortran+mpi	forward	20150128	summary.txt	5:5

ECCO v4 setups (cs32 and llc90)

(3) model output

standard MITgcm text output

```
(PID.TID 0000.0001) // =======================================================
(PID.TID 0000.0001) //                      MITgcm UV
(PID.TID 0000.0001) //                      =========
(PID.TID 0000.0001) // =======================================================
(PID.TID 0000.0001) // execution environment starting up...
(PID.TID 0000.0001)
(PID.TID 0000.0001) // MITgcmUV version:  checkpoint65h
(PID.TID 0000.0001) // Build user:        gforget
(PID.TID 0000.0001) // Build host:        GLACIER2.MIT.EDU
(PID.TID 0000.0001) // Build date:        Tue Jan  6 03:43:49 EST 2015
(PID.TID 0000.0001)
(PID.TID 0000.0001) // =======================================================
(PID.TID 0000.0001) // Execution Environment parameter file "eedata"
(PID.TID 0000.0001) // =======================================================
(PID.TID 0000.0001) ># Example "eedata" file
(PID.TID 0000.0001) ># Lines beginning "#" are comments
(PID.TID 0000.0001) ># nTx - No. threads per process in X
(PID.TID 0000.0001) ># nTy - No. threads per process in Y
(PID.TID 0000.0001) > &EEPARMS
(PID.TID 0000.0001) > useCubedSphereExchange=.TRUE.,
(PID.TID 0000.0001) > nTx=1,
(PID.TID 0000.0001) > nTy=1,
(PID.TID 0000.0001) > /
(PID.TID 0000.0001) ># Note: Some systems use & as the
(PID.TID 0000.0001) ># namelist terminator. Other systems
(PID.TID 0000.0001) ># use a / character (as shown here).
(PID.TID 0000.0001)
(PID.TID 0000.0001) // =======================================================
(PID.TID 0000.0001) // Computational Grid Specification ( see files "SIZE.h" )
(PID.TID 0000.0001) //                                  ( and "eedata"       )
(PID.TID 0000.0001) // =======================================================
(PID.TID 0000.0001)      nPx =   24 ; /* No. processes in X */
(PID.TID 0000.0001)      nPy =    1 ; /* No. processes in Y */
(PID.TID 0000.0001)      nSx =    4 ; /* No. tiles in X per process */
```

a long text file named output.txt or STDOUT.0000

(3) model output

diagnostic MITgcm binary output

Chapter 7

Diagnostics and I/O - Packages II, and Post–Processing Utilities

MITgcm has several packages related to the input and output consumed and produced during a model integration. The packages used are related to the choice of input/output fields and the on-disk format of the model output.

7.1 Diagnostics–A Flexible Infrastructure

http://mitgcm.org/public/r2_manual/latest/online_documents/manual.pdf

(3) model output

diagnostic MITgcm binary output

```
# Diagnostic Package Choices
#-----------------
# for each output-stream:
#   filename(n) : prefix of the output file name (only 8.c long) for outp.stream n
#   frequency(n):< 0 : write snap-shot output every multiple of |frequency| (iter)
#                > 0 : write time-average output every multiple of frequency (iter)
#   levels(:,n) : list of levels to write to file (Notes: declared as REAL)
#                 when this entry is missing, select all common levels of this list
#   fields(:,n) : list of diagnostics fields (8.c) (see "available_diagnostics" file
#                 for the list of all available diag. in this particular config)
#--------------------------------------------------------------------
#
 &diagnostics_list
#
   dumpatlast = .TRUE.,
#---
  frequency(1) = 2635200.0,
   fields(1:25,1) = 'ETAN    ','SIarea  ','SIheff ','SIhsnow ',
#stuff that is not quite state variables (and may not be quite
#synchroneous) but are added here to reduce number of files
                 'DETADT2 ','PHIBOT  ','sIceLoad',
                 'MXLDEPTH','oceSPDep',
                 'SIatmQnt','SIatmFW ','oceQnet ','oceFWflx',
                 'oceTAUX ','oceTAUY ',
                 'ADVxHEFF','ADVyHEFF','DFxEHEFF','DFyEHEFF',
                 'ADVxSNOW','ADVySNOW','DFxESNOW','DFyESNOW',
                 'SIuice  ','SIvice  ',
   filename(1) = 'diags/state_2d_set1',
```

example of user specification (data.diagnostics)

(3) model output

diagnostic MITgcm binary output

```
ocean% du -sh state_2d_set1.0000000732.*ta
11M     state_2d_set1.0000000732.data
4.0K    state_2d_set1.0000000732.meta
ocean%
ocean% cat state_2d_set1.0000000732.meta
 nDims = [   2 ];
 dimList = [
    90,     1,    90,
  1170,     1, 1170
 ];
 dataprec = [ 'float32' ];
 nrecords = [    25 ];
 timeStepNumber = [         732 ];
 timeInterval = [  3.600000000000E+03  2.635200000000E+06 ];
 missingValue = [ -9.99000000000000E+02 ];
 nFlds = [   25 ];
 fldList = {
 'ETAN    ' 'SIarea  ' 'SIheff  ' 'SIhsnow ' 'DETADT2 ' 'PHIBOT  ' 'sIceLoad'
 'MXLDEPTH' 'oceSPDep' 'SIatmQnt' 'SIatmFW ' 'oceQnet ' 'oceFWflx' 'oceTAUX '
 'oceTAUY ' 'ADVxHEFF' 'ADVyHEFF' 'DFxEHEFF' 'DFyEHEFF' 'ADVxSNOW'
 'ADVySNOW' 'DFxESNOW' 'DFyESNOW' 'SIuice  ' 'SIvice  '
 };
ocean% _
```

example of output file (.data, .meta)

(4) results analysis

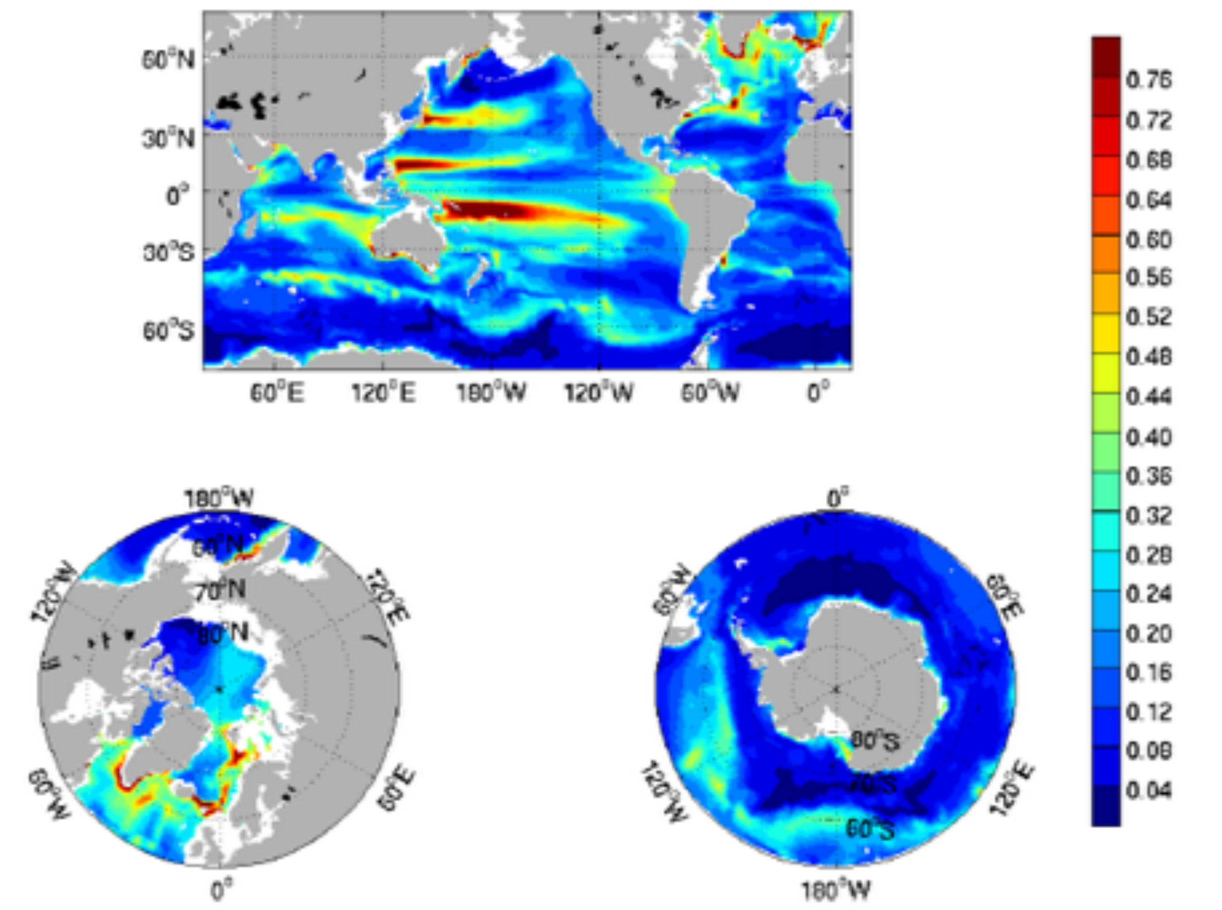


Figure: 1992-2011 standard deviation – temperature (in degC) at 300m

http://mitgcm.org/viewvc/*checkout*/MITgcm/MITgcm_contrib/gael/matlab_class/gcmfaces.pdf

(4) results analysis

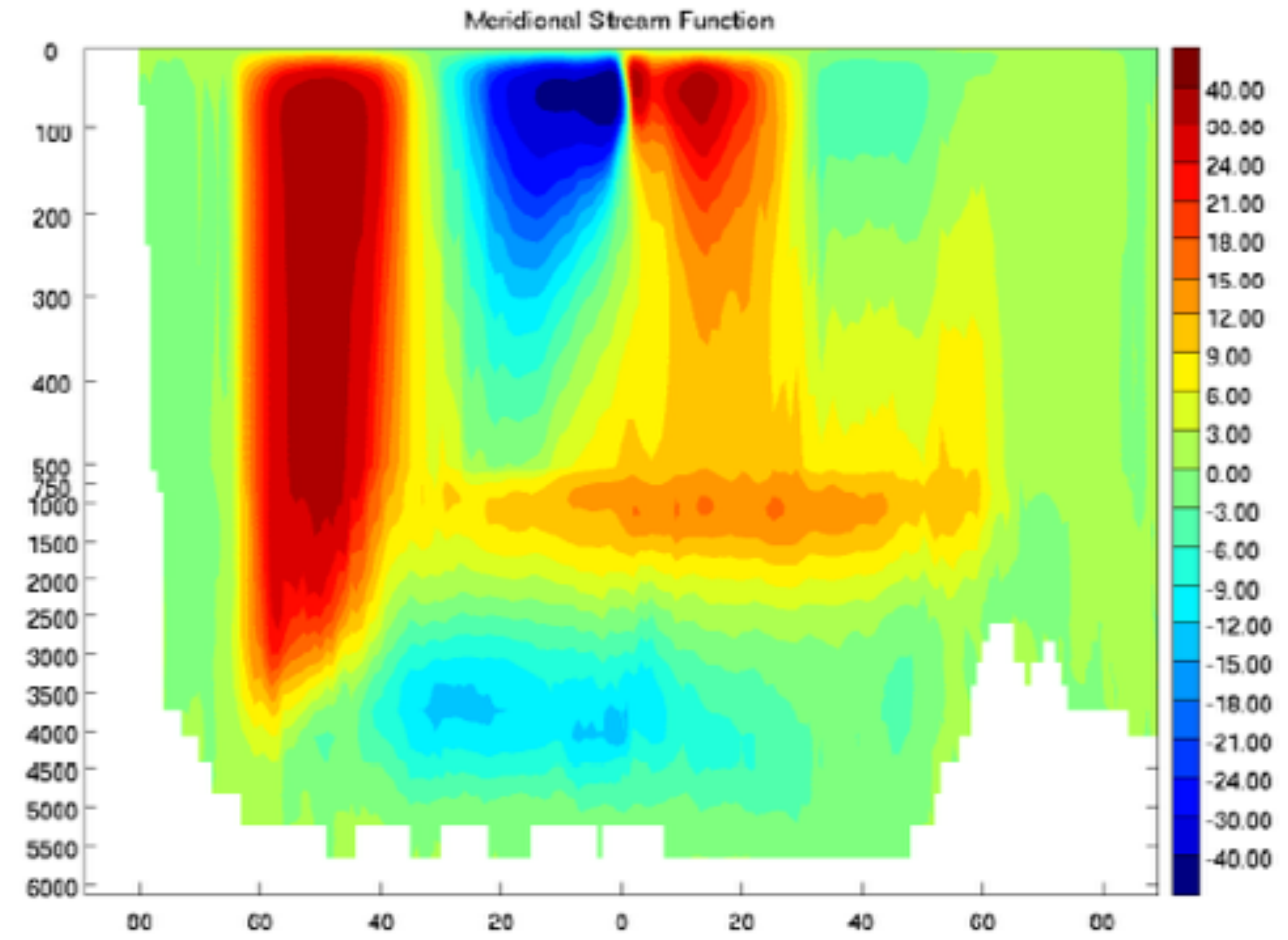
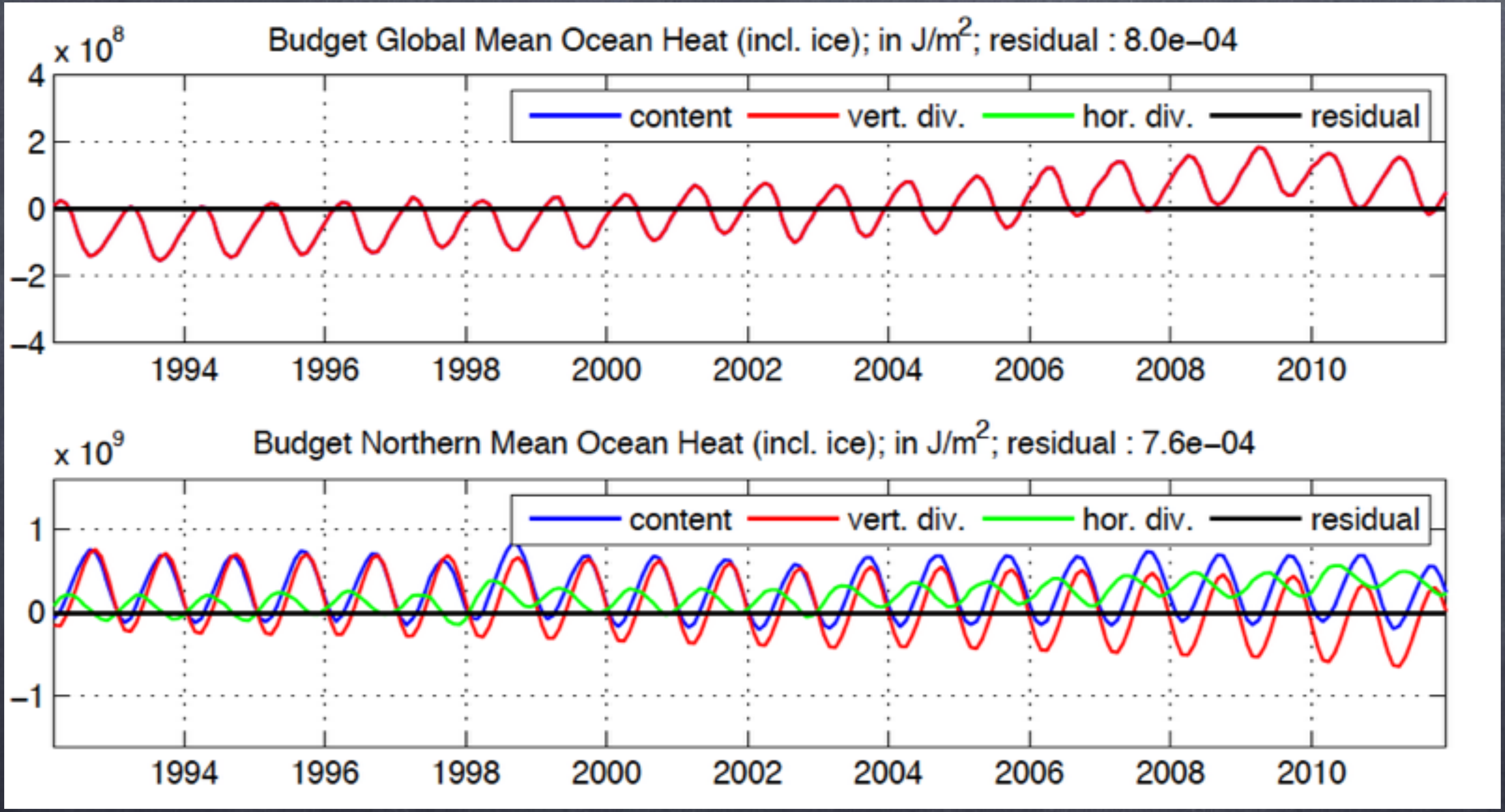


Figure: 1992-2011 mean – overturning streamfunction (Sv)

http://mit.ecco-group.org/opendap/ecco_for_las/version_4/release1/ancillary_data/standardAnalysis.pdf

(4) results analysis



mitgcm output setting : http://mitgcm.org/viewvc/*checkout*/MITgcm/MITgcm_contrib/gael/verification/global_oce_llc90/input/data.diagnostics

matlab analysis code : http://mitgcm.org/viewvc/*checkout*/MITgcm/MITgcm_contrib/gael/matlab_class/gcmfaces_diags/diags_set_E.m

(5) interactive session : self-guided exercises

```
Below is a list of proposed, self guided exercises. I generally tried to order
the exercises by increasing complexity. While none of them is really challenging,
the various exercises aim to give you with first hand experience with the data sets
and tools discussed over the course of the IAP activity.

tips : - look for answers/examples in the programs we ran together in class #1 and #2
       - type 'help read_nctiles' in matlab and similarly for all other functions
       - use the matlab debugger to go through computations step by step
```

The actual listing is at
http://mitgcm.org/viewvc/*checkout*/MITgcm/MITgcm_contrib/gael/comm/course-idma2015/computing/iap-idma-exercises

or run and diagnose MITgcm experiment

Directions were added in
http://mitgcm.org/viewvc/*checkout*/MITgcm/MITgcm_contrib/gael/comm/course-idma2015/computing/iap-idma-readme