CMAQ: compile v5.0.1

1. Content

2. Introduction

We would like to compile CMAQ then the CMAQ wiki may be a great help: http://www.airqualitymodeling.org/cmaqwiki/index.php?title=CMAQ_version_5.0_%28February_2010_release%29_OGD

The I/O API (IOAPI) version 3.1 has to be available on the system. The procedure of compiling IOAPI is also described in this Wiki.

A prepared set of IOAPI and CMAQ code is available in the Download section together with a README. Some steps described below can be skipped when these prepared versions are used.

3. Downloads

3.1. config.cmaq

config.cmaq.original unchanged config file

config.cmaq.modified (use this version but change M3HOME; compiler flags are modified)

3.2. Prepared CMAQ and IOAPI Files

CMAQv5.0.1.hzg.tar.gz

IOAPIv3.1.hzg.tar.gz

README_fastInstall_CMAQ_neumannd_v03

NOTE: There are minor differences compared to the guide below.

- IOAPI: No modifications of Makefiles except of the BASEDIR variable in the top-level Makefile are necessary if compiler and architecture are not changed. Thus, just call 'make configure' and 'make all' and run the tests.
- CMAQ: (a) Open MPI should be linked into lib/x86_64/pgf/openmpi instead of into lib/x86_64/pgf/mpich (the pathes in the build scripts are corrected accordingly). (b) Modifications are only necessary in the scripts/config.cmaq file and in no other compilation scripts. Only the variables for different gas phase and aerosol chemistry mechanisms need to be adjusted in icon, bcon and cctm if wished. The standard mechanism is cb05tucl_ae6.
- CMAQ Benchmark data (mentioned in the README): It nearly equals the standard CMAQ Benchmark data at the homepage. Difference: Benchmark initial conditions were created and added to the tar.gz archive. Running the icon program without any modifications reproduces these initial conditions.

See the Readme for details.

4. Preparation

4.1. Preconditions

Make sure that **MPI**, **netCDF** and **IOAPI** are installed. MPI and netCDF are already installed on Ocean by Jens and Hartmut. IOAPI needs to be installed by ourselves. See CMAQ: Needed Libraries for details.

4.2. Download and unpack CMAQ code

Download your appropriate CMAQ version from https://www.cmascenter.org/ . For CMAQ version 5.0.1 these are

- CMAQv5.0.1.tar.gz -> source code
- DATA.CMAQv5.0.1.tar.gz -> data for the benchmark case
- ~~SCRIPTS.CMAQv5.tar.gz -> scripts to build CMAQ~~ (note: deprecated; now contained in CMAQv5.0.1.tar)
- ~~DATA_REF.CMAQv5.0.1.tar.gz -> data for the benchmark case~~ (note: contained in DATA.CMAQv5.0.1.tar.gz)

Choose a directory in which CMAQ in installed later on and create it (here: /data/M3HOME/CMAQ/myCMAQ).

```
mkdir /data/M3HOME/CMAQ/myCMAQ
```

The CMAQv5.0.1.tar.gz needs to be extracted into the directory /data/M3HOME/CMAQ/myCMAQ. The DATA.CMAQv5.0.1.tar.gz file also needs to be extracted if Benchmark runs should be performed.

```
tar xzfv CMAQv5.0.1.tar.gz
# x = extract
# z = gzip compression
# f = use archive file (here: CMAQv5.0.1.tar.gz)
# v = verbose mode (see which files are extracted to which place)

tar xzfv CMAQv5.0.1.tar.gz -C /data/M3HOME/CMAQ/myCMAQ
# Extract the archive
# into /data/M3HOME/CMAQ/myCMAQ

tar xfv CMAQv5.0.1.tar
CMAQv5.0.1/models/CCTM/src/emis/emis/SSEMIS.F,v -C
/data/M3HOME/CMAQ/myCMAQ
# Extract the file SSEMIS.F,v (+directory tree)
# to /data/M3HOME/CMAQ/myCMAQ/CMAQv5.0.1/models/CCTM/src/emis/emis/
```

After the un-tar process finished we may go into /data/M3HOME/CMAQ/myCMAQ/CMAQv5.0.1. There we find the directories models and scripts.

- models: source code (in CVS format => each text file has an extra header and footer + '@' is given as '@@')
- scripts: scripts for compiling and running CMAQ

4.3. Links to external libraries

In order to simplify the compilation process we will create links to some libraries in a newly created folder structure in '/data/M3HOME/CMAQ/myCMAQ/CMAQv5.0.1'.

```
mkdir lib lib/x86_64 lib/x86_64/pgf
cd lib/x86_64/pgf
mkdir netcdf ioapi_3.1 mpich
```

We create links to needed external libraries with 'ln -s [TARGET] [LINKNAME]'

```
# create links to netcdf
cd netcdf
ln -s /opt/netcdf/4.2.1.1/pgi/openmpi/ib/bin bin
ln -s /opt/netcdf/4.2.1.1/pgi/openmpi/ib/lib lib
ln -s /opt/netcdf/4.2.1.1/pgi/openmpi/ib/include include
cd ..

# create links to MPI
cd mpich
ln -s /opt/openmpi/pgi/ib/bin bin
ln -s /opt/openmpi/pgi/ib/lib lib
ln -s /opt/openmpi/pgi/ib/include include
cd ..

# create links to IOAPI
cd ioapi_3.1
ln -s /data/M3HOME/CMAQ/IOAPI3.1_pgi14.1/ioapi ioapi
ln -s /data/M3HOME/CMAQ/IOAPI3.1_pgi14.1/Linux2_x86_64pg_pgcc_nomp
Linux2_x86_64pgf
cd ..
```

The IOAPI directories may differ depending on your compiler version! See CMAQ: Needed Libraries for possible directories. You may also have to compile it by yourself.

NOTE: It would be nicer to create a directory 'openmpi' instead of 'mpich' and link the Open MPI libraries the 'openmpi' directory. However, one needs to modify the include and library pathes in the bldit.se, bldit.pario and bldit.cctm in the latter case. Hence, it is more convenient to link the Open MPI into the mpich folder.

5. Compilation

(The page http://www.airqualitymodeling.org/cmaqwiki/index.php?title=CMAQ_version_5.0_(February_2010_release)_OGD#Compiling_ CMAQ_for_the_Benchmark_Test_Case_Simulation may help.)

We change into /data/M3HOME/CMAQ/myCMAQ/CMAQv5.0.1/scripts for our further work.

5.1. Preparation

Different programmes need to be compiled

- build: create bldmake; needed for all other compilation processes
- stenex: stencil exchange (STENEX) libraries for serial and parallel processing
- pario: parallel CCTM operation
- jproc (optional): create photolysis tables
- icon (optional): create initial conditions
- bcon (optional): create boundary conditions
- cctm: chemistry transport model

Each compilation task is performed with a script: [TASK]/bldit.[TASK] (e.g. icon/bldit.icon). All of these scripts source a configure script 'config.cmaq' which is located in '.'. In this script, compilers, compiler options, directories and environment variables are set.

5.2. config.cmaq

Due to problems with the optimisation options of the 2014's version of the pgif90 compiler (see TERMINATED by signal 11 / ERROR 127 in CMAQ: compiling FAQ for details), we need to modify the compiler/linker calls and the flags parsed to the compiler/linker. It is recommended not to use any optimization flags (such as -O3). The following setup works.

```
setenv compiler pgf
  setenv LM_LICENSE_FILE /opt/pgi/license.dat

  # NEUMANND: If mpif90 is called several parameters are attached.
Especially
  #          the flag '-fast' caused problems during CMAQ
compilation.
  #          Therefore, we call pgf90 directly and attach the
necessary MPI
  #          flags below.
  setenv myFC pgf90
  setenv myCC pgcc
  setenv myLINK_FLAG "-lmpi_f90 -lmpi_f77 -lmpi -lnetcdf -lnetcdff"
  setenv myFFLAGS "-Mfixed -Mextend -Mflushz -Mdaz -Kieee
-Mlarge_arrays"
  setenv myFRFLAGS "-Mfree"
  setenv myCFLAGS ""
```

NOTE: The -lnetcdff, -lnetcdff, -lmpi_f90, -lmpi_f77 and -lmpi flags should be set in the individual build scripts (bldit.*). However, it is fast to do it this way.

Additionally one has to adjust the M3HOME and M3DATA directories in the beginning to the file.

```
setenv M3HOME   /data/M3HOME/CMAQ/myCMAQ/CMAQv5.0.1
 setenv M3MODEL $M3HOME/models
 setenv M3DATA   /data/M3HOME/data
```

and one library variable in the beginning of the file

```
#setenv extra_lib "-lrdmacm -libumad -lopa -lmpl -lrt -lpthread
-libverbs -ldl"
 setenv extra_lib "-libumad -lrt -lpthread -libverbs -ldl -lnsl
-lutil -lm"
```

5.3. Compiling

5.3.1. buildmake, stenex and pario

Compile buildmake.

```
cd build
./bldit.bldmake
cd ..
```

Compile stenex (*_noop = no parallelisation).

```
cd stenex
./bldit.se
./bldit.se_noop
cd ..
```

Compile pario.

```
cd pario
./bldit.pario
cd ..
```

5.3.2. icon and bcon

Now we come to ICON, BCON and CCTM. We need to modify the scripts according to our future simulations setup. CMAQ offers different gas phase ('saprc99' and 'cb05' with subversions such as 'cl' and 'tucl') and aerosol chemistry mechanisms (aero5 and aero6) and some more configuration options. Let us assume we want to use cb05tucl and aero5.

We open icon/bldit.icon and bcon/bldit.bcon and

- set the variable 'Mechanism' to 'cb05tucl_ae5_aq' <- *effects the output of the icon programme*
- set the variable 'APPL' to 'cb05tucl_ae5_aq' <- *effects the name of the icon programme and the name of the output filebut not the content*
- check the variable 'ModMech' (should be 'cb05') <- *use cb05*
- check the variable 'ModType' (should be 'profile') <- *take a standard atmospheric profile; one could also use an own profile*

Now compile icon and bcon:

```
cd icon
./bldit.icon
mv BLD_cb05tucl_ae5_aq/ICON_cb05tucl_ae5_aq_Linux2_x86_
ICON_cb05tucl_ae5_aq_Linux2_x86_64pgf
# File name in the build directory is cut because it it too long.
# Therefore we have to move it manually
cd ..


cd bcon
./bldit.bcon
mv BLD_cb05tucl_ae5_aq/BCON_cb05tucl_ae5_aq_Linux2_x86_
BCON_cb05tucl_ae5_aq_Linux2_x86_64pgf
cd ..
```

5.3.3. cctm

Open cctm/bldit.cctm and (uncomment the appropriate option!)

- set the variable 'Mechanism' to 'cb05tucl_ae5_aq' <- *effects the output of the icon programme*
- set the variable 'APPL' to 'cb05tucl_ae5_aq' <- *effects the name of the icon programme and the name of the output filebut not the content*
- set 'ModCloud' to 'module cloud_acm_ae5'
- set 'ModAero' to 'module aero5'
- set 'ModChem' to 'module ebi_cb05tucl' <- *here we choose the numerical solver*
- set 'ModVadv' to 'module yamo'
- set 'ModDriver' to 'module ctm_wrf'
- replace '-lmpich' by '-lmpi' (see 'set LIB4', about in line 224)
- add '-lnetcdff' to 'set NETCDF = "${M3LIB}/netcdf/lib -lnetcdf"' (about line 174)

If you want to run CMAQ on a single core you have to comment out the line 'set ParOpt'.

Finally we call

```
cd cctm
./bldit.cctm
mv BLD_cb05tucl_ae5_aq/CCTM_cb05tucl_ae5_aq_Linux2_x86_
CCTM_cb05tucl_ae5_aq_Linux2_x86_64pgf
cd ..
```