Model Engineer's Report to the GCSC

Bob Yantosca

09 Feb 2017

Public release of v11-01

GEOS-Chem v11-01 was officially released on 01 Feb 2017, after a 5-week provisional release period. This version contains many important new features, among which are the FlexChem chemistry solver and the GEOS-Chem species database.

For detailed information about the GEOS-Chem v11-01 release, please see:

- GEOS-Chem v11-01 provisional release newsletter (20 Dec 2017)
- GEOS-Chem Newsletter, Winter 2017 Edition + v11-01 public release (01 Feb 2017)

GEOS-Chem v11-02 development

The following updates are slated to be added to GEOS-Chem v11-02:

v11-02a

In this version we will add several fixes for issues that we were not able to add to the v11-01 public release, as well as updating the chemistry reaction rates to the latest JPL publication.

v11-02a Feature	Type	Submitted by
Updating chemistry rate constants based on *JPL Publication 15-10*	Science	Barron Henderson (US EPA) Mat Evans (U. York) Oxidants and Chemistry WG
Fixes to correct ALK4 lumping issue	Science	Barron Henderson (US EPA)
PAN updates	Science	Emily Fischer (CSU)
Monthly mean NEI2011 emissions	Science	GCST
Implement ISORROPIA v2.0 as a Fortran module (eliminating `COMMON` blocks)	Structural	Seb Eastham (Harvard) GCST
Update to HEMCO v2.0.004	Structural	Christoph Keller (NASA GMAO)
Fix error in production of SO4s and NITs in SEASALT_CHEM routine	Bug fix	Prasad Kasibhatla (Duke)

v11-02a Feature	Type	Submitted by
Fix bug in CHEM_NIT routine	Bug fix	Prasad Kasibhatla (Duke)
Fix for sulfate production in HET_DROP_CHEM	Bug fix	Qianjie Chen (UW)
Save out PM2.5 diagnostic at STP conditions	Bug fix	Aaron van Donkelaar (Dalhousie)

v11-02b

This version will contain a new chemistry mechanism and updated SOA chemistry.

v11-02b Feature	Type	Submitted by
Halogen chemistry updates	Science	Tomás Sherwen (York) Johan Schmidt (Harvard) Oxidants and Chemistry WG
Enhance default GEOS-Chem simple SOA	Science	Aerosols Working Group

v11-02c

This version will contain updates to the isoprene chemistry mechanism.

v11-02c Feature	Type	Submitted by
Updates to isoprene chemistry, includes: • Fast photolysis of carbonyl nitrates • Aerosol uptake of organic nitrates	Science	Jenny Fisher (U. Wollongong) Eloïse Marais (Harvard) Kelvin Bates (Caltech) Katie Travis (Harvard)

v11-02d

This version will contain various chemistry and diagnostic updates.

v11-02d Feature	Type	Submitted by
Update density of BC to 1.8 and add absorption enhancement factor in input.geos	Science	Xuan Wang (MIT)

v11-02d Feature	Type	Submitted by
Add aqueous isoprene uptake to SOA scheme	Science	Eloise Marais (Harvard)
Monthly mean surface methane distributions	Science	Lee Murray (NASA GISS/LDEO)
Radon flux diagnostic	Benchmark	GCST
UCX stratospheric water boundary condition update	Science	Chris Holmes (UC Irvine) Seb Eastham (Harvard)

In the pipeline

For a complete list of the remaining GEOS-Chem v11-02 development items, please see this post on the *GEOS-Chem v11-02* wiki page.

Selecting GEOS-Chem timesteps

The GEOS-Chem v11-01 public release uses the optimal timestep configuration from S. Philip et al [2016]: 20 minute timesteps for chemistry and emissions, and 10 minute timesteps for transport and dynamics. This configuration is also known as C20T10.

It has been noted that using C20T10 can cause simulation run times to increase by almost a factor of two. For this reason we have added cautionary text on the GEOS-Chem wiki and GEOS-Chem v11-01 user's manual advising users to run with coarser timesteps (e.g. C60T30 for 4^0 x 5^0 or C30T15 for 2^0 x 2.5^0) as a way to decrease the run times.

We have posted a number of results from 7-day time tests using with GEOS-Chem v11-01 on the wiki.

In addition, we have also prepared a set of time tests that have not yet been posted on the wiki, which illustrates the speedup that can be obtained by switching between mechanisms and/or timesteps.

For all runs in the table below:

- **Number of CPUs requested:** 16 (in order to reserve an entire node for testing)
- **Number of CPUs used:** 8
- **CPU Type:** GenuineIntel
- **CPU Model:** Intel(R) Xeon(R) CPU E5-2660 0 @ 2.20GHz

Run name	Submitter	Machine or Node and Compiler	CPU time	Wall time	CPU / Wall ratio	% of ideal
v11-01-standard (C20T10)	Bob Yantosca	regal16.rc.fas.harvard.edu ifort 11.1	62554.07 s **17:22:34**	9355.80 s **02:35:59**	6.6861	83.58
v11-01-tropchem (C20T10)	Melissa Sulprizio	regal17.rc.fas.harvard.edu ifort 11.1	26156.34 s **07:15:56**	3881.55 s **01:04:48**	6.7386	84.23
v11-01-tropchem (C60T30)	Melissa Sulprizio	regal18.rc.fas.harvard.edu ifort 11.1	11802.31 s **03:16:42**	1923.35 s **00:32:12**	6.1363	76.70

As you can see from the results above, switching from the v11-01-standard simulation to v11-01-tropchem can result in a factor of 2 speedup. This is because the standard simulation contains the UCX option, which includes a detailed stratospheric chemistry mechanism, which is omitted in the tropchem simulation.

Also note, switching from C20T10 configuration to the coarser C30T60 configuration also introduces another factor of two speedup.

Flexible precision in GEOS-Chem

Prior to the GEOS-Chem v11-01 public release, the GCST performed a 1-month benchmark using the PRECISION=4 option. This forces most floating-point variables in GEOS-Chem to be declared as REAL*4 (i.e. 32 bits) instead of REAL*8 (64 bits). This should reduce the amount of memory required by the GEOS-Chem executable.

With respect to the v11-01-provisional-release (the prior 1-month benchmark):

- Differences in the output were negligible.
- The simulation finished 20 minutes earlier (05:29 vs. 05:49)
- GEOS-Chem used approximately 40% less memory (3.3605 GB vs. 5.3816 GB)

Therefore, we will make PRECISION=4 the default for the public release of GEOS-Chem v11-02 (TBD next year).

Eliminating a computational bottleneck in v11-01-public

After the v11-01 public release was issued, the GCST discovered a computational bottleneck in the convection module. The following DO loop at line 343 of `convection_mod.F` was not parallelized:

```
! Loop over advected species
DO NA = 1, nAdvect

   ! Species ID
   N = State_Chm%Map_Advect(NA)
```

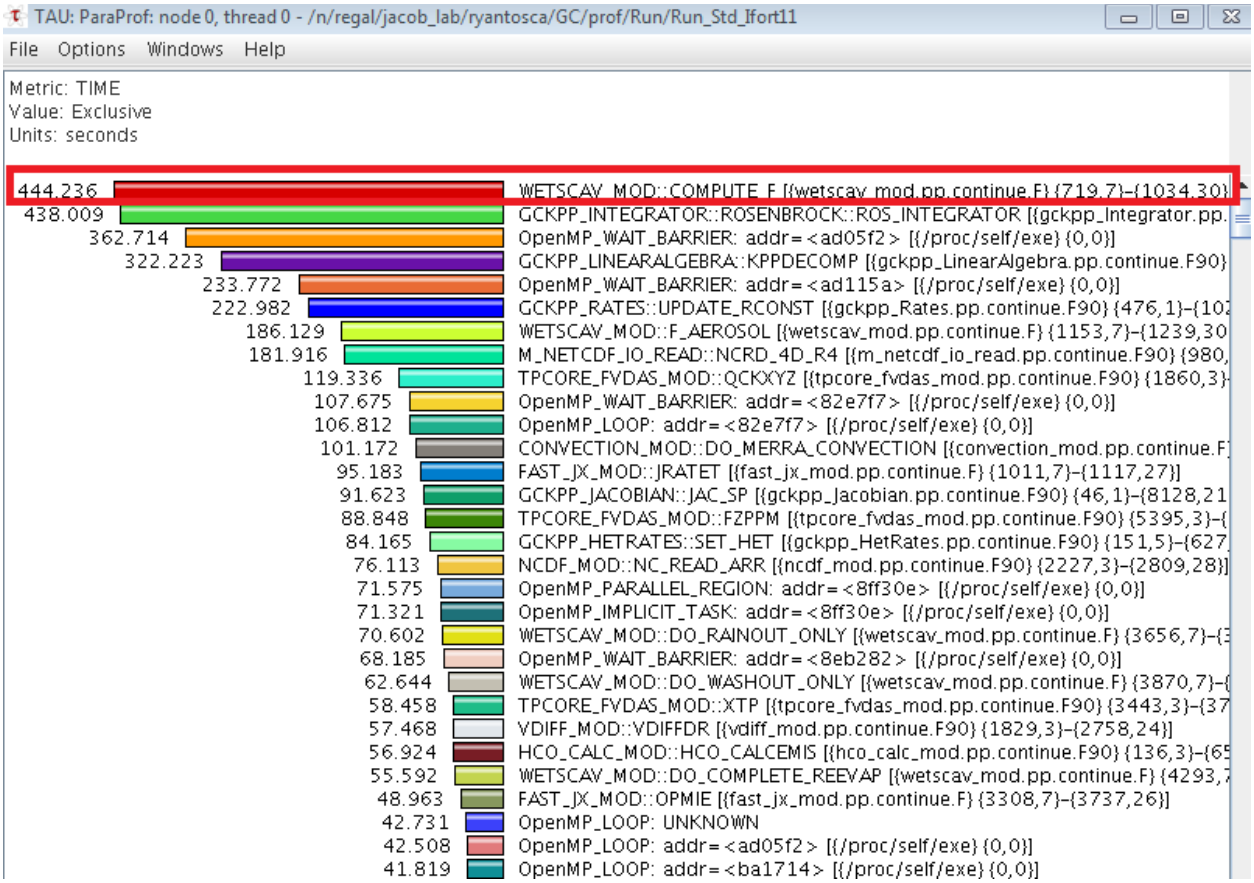
```
        ! Now point to a 3D slice of the FSOL array
        p_FSOL => FSOL(:,:,:,NA)

        ! Fraction of soluble species
        CALL COMPUTE_F( am_I_Root, N, p_FSOL, ISOL(NA),
     &                  Input_Opt, State_Met, State_Chm, RC )

        ! Free pointer memory
        p_FSOL => NULL()

      ENDDO
```

Bob Yantosca profiled the GEOS-Chem v11-01-public code with the TAU performance analyzer, using a 2-model-day simulation with 8 CPUs. The results are shown below:

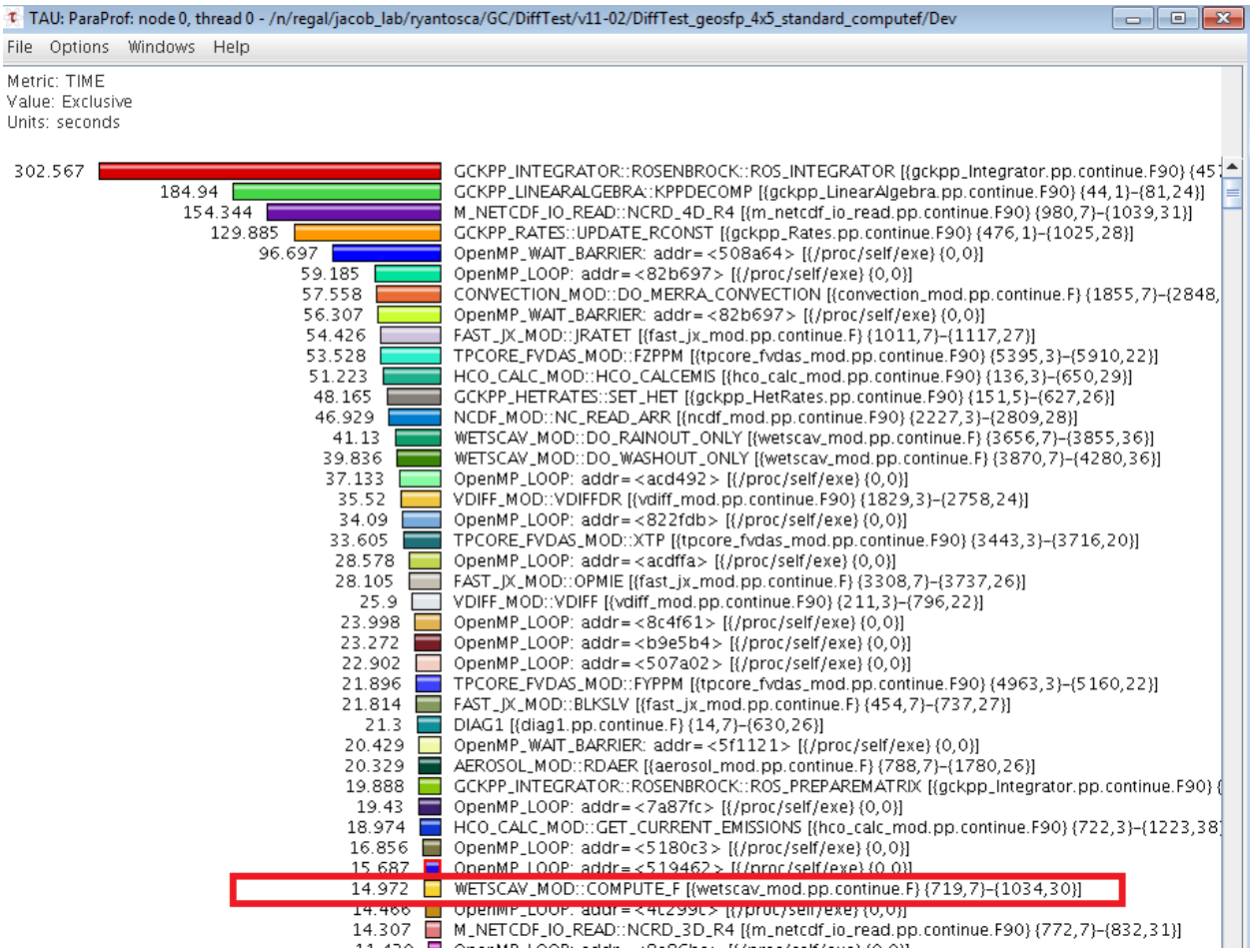


As you can see, because the `COMPUTE_F` routine was not being called from a parallel loop, the subroutine was only being executed on the master CPU. This caused a slowdown because the other CPUs were being forced to wait until `COMPUTE_F` finished before the simulation could proceed.

The solution was to parallelize the DO loop, in order to obtain a better load balancing. We added the code in GREEN:

```
      ! Loop over advected species
!$OMP PARALLEL DO
!$OMP+DEFAULT( SHARED )
!$OMP+PRIVATE( NA, N, p_FSOL, RC )
      DO NA = 1, nAdvect
         ... etc ...
      ENDDO
!$OMP END PARALLEL DO
```

Parallellizing the loop indeed removed the bottleneck:



After parallelizing the loop, the `COMPUTE_F` routine now only spends 14 seconds on the master CPU, instead of ~400 seconds. This means that the work of calling COMPUTE_F is now being efficiently split up among all 8 CPUs. This should result in a significant speedup.

The GCST will add this fix as a patch to the v11-01 public release, as well as into the v11-02a development stream. We will also provide updated timing results shortly.

GCHP validation update

The GCHP development team is working on benchmarking GCHP scientific output. We hope to have this completed by April 1, 2017. Results will be posted on the wiki shortly.

--

Thanks for your continued support of GEOS-Chem!

Bob Y.
with the entire GCST (Melissa, Lizzie, Mike, Junwei, Yanko)