EC-Earth/COSP performance analysis results

EC-Earth Branch

```
URL: https://svn.ec-earth.org/ecearth3/branches/development/2016/r3473-cosp2
Repository Root: https://svn.ec-earth.org
Repository UUID: 7d5749e4-8d9b-47cb-b16d-f28b5b69cda6
Revision: 4040
Node Kind: directory
Schedule: normal
Last Changed Author: klausred
Last Changed Rev: 4037
Last Changed Date: 2017-03-03 11:09:40 +0100 (Fri, 03 Mar 2017)
```

Performance analysis completed on Triolith using standard compilation information in the `sources/config-build.xml` file.

Single executable

To allow for the use of the Allinea MAP profiler (https://www.allinea.com/products/map) and to simplify the analysis, the IFS code was modified to revert to reading ICMSEA files as suggested by Klaus. Files `sources/ifs-36r4/src/ifs/climate/updclie.F90`, `sources/ifs-36r4/src/ifs/setup/sumcc.F90` modified (uncommenting relevant ICMSEA sections) and `sources/ifs-36r4/src/ifs/cplng/cplng_data_mod.F90` modified to include a trap for `CPLNG_ACTIVE=.FALSE.` These changes allow the IFS model to be launched as a single binary without the need to couple to the `ampi-forcing.exe` model.

Simulations

Standard 24 hour, atmosphere only tests were used for the analysis (standard setup from the `runtime/classic/config-run.xml` file). The Allinea MAP profiler uses sampling (default 1000 points) and so it was recommended to run short simulations (limited number of time steps) to ensure the sampling is representative.

IFS timestep: 2700 sections
Initial data path: /nobackup/rossby20/rossby/joint_exp/ecearth/inidata
Resolution: T255L91
No restarts.
CMIP6: TRUE
CMIP5: TRUE
RCP: 0
CMIP5_FIXYEAR: 0
nproc: 48

Results

Reference simulation

Results from the reference simulation (ISSCP and CALIPSO on, no code or compiler flag changes) clearly show the impact of the COSP simulators on the runtime performance of the model. For example the ifs.stat file shows that for this configuration, the timesteps where COSP is being called (timesteps 9, 17, 25 below) take ~14.1 seconds compared with either 2.05/4.3 seconds for standard IFS timesteps.

```
09:40:19 000000000 CNT3      -999      0.64      0.64      0.96       0:00        0:00 0.00000000000000E+00
09:40:28 AAAB00AAA STEPO        0      3.66      3.66     10.24       0:03        0:10 0.95562419379363E-05
09:40:52 0AAA00AAA STEPO        1     24.04     24.04     24.34       0:27        0:34 0.95562419379363E-05
09:40:54 0AAA00AAA STEPO        2      2.02      2.02      2.04       0:29        0:36 0.95562419379363E-05
09:40:57 0AAA00AAA STEPO        3      2.08      2.08      2.10       0:31        0:38 0.95562419379363E-05
09:40:59 0AAA00AAA STEPO        4      2.05      2.05      2.06       0:33        0:41 0.95562419379363E-05
09:41:03 0AAA00AAA STEPO        5      4.31      4.31      4.65       0:38        0:45 0.95562419379363E-05
09:41:05 0AAA00AAA STEPO        6      2.03      2.03      2.04       0:40        0:47 0.95562419379363E-05
09:41:07 0AAA00AAA STEPO        7      2.07      2.07      2.08       0:42        0:49 0.95562419379363E-05
09:41:10 AAAB00AAA STEPO        8      2.22      2.22      2.23       0:44        0:52 0.95562419379363E-05
09:41:24 0AAA00AAA STEPO        9     14.17     14.17     14.28       0:58        1:06 0.95562419379363E-05
09:41:26 0AAA00AAA STEPO       10      2.00      2.00      2.02       1:00        1:08 0.95562419379363E-05
09:41:28 0AAA00AAA STEPO       11      2.06      2.06      2.07       1:02        1:10 0.95562419379363E-05
09:41:30 0AAA00AAA STEPO       12      2.01      2.01      2.03       1:04        1:12 0.95562419379363E-05
09:41:34 0AAA00AAA STEPO       13      4.37      4.37      4.40       1:09        1:16 0.95562419379363E-05
09:41:36 0AAA00AAA STEPO       14      2.00      2.00      2.02       1:11        1:18 0.95562419379363E-05
09:41:39 0AAA00AAA STEPO       15      2.06      2.06      2.07       1:13        1:20 0.95562419379363E-05
09:41:41 AAAB00AAA STEPO       16      2.04      2.04      2.10       1:15        1:23 0.95562419379363E-05
09:41:55 0AAA00AAA STEPO       17     14.07     14.07     14.16       1:29        1:37 0.95562419379363E-05
09:41:57 0AAA00AAA STEPO       18      1.99      1.99      2.01       1:31        1:39 0.95562419379363E-05
09:41:59 0AAA00AAA STEPO       19      2.04      2.04      2.06       1:33        1:41 0.95562419379363E-05
09:42:01 0AAA00AAA STEPO       20      2.14      2.14      2.17       1:35        1:43 0.95562419379363E-05
09:42:05 0AAA00AAA STEPO       21      4.29      4.29      4.38       1:39        1:47 0.95562419379363E-05
09:42:07 0AAA00AAA STEPO       22      2.01      2.01      2.02       1:41        1:49 0.95562419379363E-05
09:42:10 0AAA00AAA STEPO       23      2.20      2.20      2.21       1:44        1:52 0.95562419379363E-05
09:42:12 AAAB00AAA STEPO       24      2.05      2.05      2.09       1:46        1:54 0.95562419379363E-05
09:42:26 0AAA00AAA STEPO       25     14.17     14.17     14.26       2:00        2:08 0.95562419379363E-05
09:42:28 0AAA00AAA STEPO       26      1.98      1.98      2.01       2:02        2:10 0.95562419379363E-05
09:42:30 0AAA00AAA STEPO       27      2.02      2.02      2.05       2:04        2:12 0.95562419379363E-05
09:42:32 0AAA00AAA STEPO       28      2.00      2.00      2.02       2:06        2:14 0.95562419379363E-05
09:42:36 0AAA00AAA STEPO       29      4.37      4.37      4.38       2:10        2:18 0.95562419379363E-05
09:42:38 0AAA00AAA STEPO       30      2.00      2.00      2.01       2:12        2:20 0.95562419379363E-05
09:42:41 0AAA00AAA STEPO       31      2.03      2.03      2.05       2:14        2:22 0.95562419379363E-05
09:42:43 FAAB00000 STEPO       32      2.05      2.05      2.06       2:16        2:24 0.10992460223854E-04
09:42:59 000000000 CNT0        32     14.84     14.84     16.08       2:31        2:41 0.10992460223854E-04
```

Initial Allinea Reports and Allinea MAP results demonstrate that the COSP performance is CPU bound (not MPI or IO) as expected. Allinea MAP profiling indicates that there is no one single, critical performance bottleneck in the COSP implementation, nevertheless some performance optimization targets were identified.

sources/cosp-v2.0/src/simulator/actsim/lidar_simulator.F90

Two subroutines (`cmp_backsignal`, line 378 and `cmp_beta`, line 410) contain code blocks with EXP calculations of vector arguments, where the EXP calculations are repeated. e.g. `cmp_backsignal`, lines 393-407 (repeated EXP highlighted)

```
! Other layers
  do k=2,nlev
     tautot_lay(:) = tau(:,k)-tau(:,k-1)
     WHERE ( EXP(-2._wp*tau(:,k-1)) .gt. 0. )
        WHERE (tautot_lay(:) .gt. 0.)
           pnorm(:,k) = beta(:,k)*EXP(-2._wp*tau(:,k-1)) /&
                (2._wp*tautot_lay(:))*(1._wp-EXP(-2._wp*tautot_lay(:)))
```

```
         ELSEWHERE
            ! This must never happen, but just in case, to avoid div. by 0
            pnorm(:,k) = beta(:,k) * EXP(-2._wp*tau(:,k-1))
         END WHERE
      ELSEWHERE
         pnorm(:,k) = 0._wp!beta(:,k)
      END WHERE
   END DO
```

Furthermore the green highlighted `WHERE` block is technically redundant unless the argument(s) to the EXP function are -inf. Modifying the code to store the `EXP(-2._wp*tau(:,k-1))` vector and removing the outer `WHERE` block:

```
   do k=2,nlev
      tautot_lay(:) = -2._wp*(tau(:,k)-tau(:,k-1))
      tau2 = -2._wp*tau(:,k-1)
      exptau = EXP(tau2(:))
!       pnorm(:,k) = 0._wp!beta(:,k)
!       WHERE (exptau(:) .gt. 0. )
         WHERE (tautot_lay(:) .lt. 0.)
            pnorm(:,k) = beta(:,k)*exptau(:) /&
                 (-1._wp*tautot_lay(:))*(1._wp-EXP(tautot_lay(:)))
         ELSEWHERE
            ! This must never happen, but just in case, to avoid div. by 0
            pnorm(:,k) = beta(:,k) * exptau(:)
         END WHERE
!       END WHERE
   END DO
```

reduces the COSP timesteps from ~14.1 to ~13.5 seconds. No tests of whether the inner `WHERE` block is necessary were made.

sources/cosp-v2.0/src/cosp_stats.F90

Allinea MAP identified two lines in the `sources/cosp-v2.0/src/cosp_stats.F90` file, subroutine `COSP_CHANGE_VERTICAL_GRID` (lines 239,240, highlighted below).

```
      ! Do the weighted mean
      do j=1,Ncolumns
         do k=1,M
            ws = sum(w(:,k))
            if (ws > 0.0) r(i,j,k) = sum(w(:,k)*yp(j,:))/ws
         enddo
      enddo
```

The first line (sum of weights) can be calculated earlier in the subroutine. This gives a small improvement in the overall timing (reduction from ~14.1 to 13.7 sec). With further code changes, it may be possible to precalculate the second sum (`sum(w(:,k)*yp(j,:))`) but this was not tested.

It was noted that the `sources/cosp-v2.0/src/cosp_stats.F90` file contains a complete alternative `COSP_CHANGE_VERTICAL_GRID` subroutine that is commented out. This alternative routine appears to have the same function as the default routine (regrid vertical coordinates). It takes the same arguments and compiles successfully. The model also runs without crashing if this alternative regridding subroutine is used, and runs significantly faster (~12.8 compared to 14.1 sec for the default), however the resulting output (lidar) does not match the reference simulation. This result at least suggests that the `COSP_CHANGE_VERTICAL_GRID` is a good target for further performance optimization and the alternative subroutine in the `sources/cosp-v2.0/src/cosp_stats.F90` file may be of some use as a starting point, although more work is required to verify this.

Compiler options

Based on the assumption that the COSP calculations are purely diagnostic and therefore do not affect the IFS state variables, relaxing some of the compiler options for the COSP library (mostly related to floating point precision) was tested.

Default compiler options (F90FLAGS):
`F90FLAGS=-fp-model precise -traceback -r8 -O3 -xHost -g`

Modified flags:
`-traceback -O3 -g -xhost -mkl=sequential`

This change results in a COSP timestep decrease from ~14.1 to ~12.2 seconds.

Overall improvement in timing

The timing improvement when applying all three changes described can be seen in the ifs.stat file listed below (timesteps 9, 17 and 25, compared to the ifs.stat file listed above).

```
16:12:17 000000000 CNT3      -999      0.72     0.72     1.17       0:00       0:00 0.00000000000000E+00
16:12:26 AAAB00AAA STEPO        0      3.87     3.87    10.53       0:03       0:10 0.95562419379363E-05
16:12:49 0AAA00AAA STEPO        1     22.72    22.72    22.95       0:26       0:33 0.95562419379363E-05
16:12:51 0AAA00AAA STEPO        2      2.00     2.00     2.03       0:28       0:35 0.95562419379363E-05
16:12:53 0AAA00AAA STEPO        3      2.08     2.08     2.09       0:30       0:37 0.95562419379363E-05
16:12:55 0AAA00AAA STEPO        4      2.05     2.05     2.06       0:32       0:39 0.95562419379363E-05
16:12:59 0AAA00AAA STEPO        5      4.43     4.43     4.45       0:37       0:44 0.95562419379363E-05
16:13:02 0AAA00AAA STEPO        6      2.05     2.05     2.20       0:39       0:46 0.95562419379363E-05
16:13:04 0AAA00AAA STEPO        7      2.07     2.07     2.08       0:41       0:48 0.95562419379363E-05
16:13:06 AAAB00AAA STEPO        8      2.16     2.16     2.19       0:43       0:50 0.95562419379363E-05
16:13:18 0AAA00AAA STEPO        9     11.84    11.84    11.93       0:55       1:02 0.95562419379363E-05
16:13:20 0AAA00AAA STEPO       10      1.97     1.97     2.01       0:57       1:04 0.95562419379363E-05
16:13:22 0AAA00AAA STEPO       11      2.05     2.05     2.07       0:59       1:06 0.95562419379363E-05
16:13:24 0AAA00AAA STEPO       12      2.02     2.02     2.04       1:01       1:08 0.95562419379363E-05
16:13:28 0AAA00AAA STEPO       13      4.39     4.39     4.40       1:05       1:13 0.95562419379363E-05
16:13:30 0AAA00AAA STEPO       14      2.00     2.00     2.03       1:07       1:15 0.95562419379363E-05
16:13:33 0AAA00AAA STEPO       15      2.07     2.07     2.08       1:09       1:17 0.95562419379363E-05
16:13:35 AAAB00AAA STEPO       16      2.06     2.06     2.16       1:11       1:19 0.95562419379363E-05
16:13:47 0AAA00AAA STEPO       17     12.04    12.04    12.13       1:23       1:31 0.95562419379363E-05
16:13:49 0AAA00AAA STEPO       18      2.00     2.00     2.01       1:25       1:33 0.95562419379363E-05
16:13:51 0AAA00AAA STEPO       19      2.02     2.02     2.06       1:28       1:35 0.95562419379363E-05
16:13:53 0AAA00AAA STEPO       20      2.01     2.01     2.03       1:30       1:37 0.95562419379363E-05
16:13:57 0AAA00AAA STEPO       21      4.36     4.36     4.38       1:34       1:42 0.95562419379363E-05
16:13:59 0AAA00AAA STEPO       22      2.01     2.01     2.02       1:36       1:44 0.95562419379363E-05
16:14:02 0AAA00AAA STEPO       23      2.09     2.09     2.26       1:38       1:46 0.95562419379363E-05
16:14:04 AAAB00AAA STEPO       24      2.04     2.04     2.09       1:40       1:48 0.95562419379363E-05
16:14:16 0AAA00AAA STEPO       25     11.84    11.84    11.93       1:52       2:00 0.95562419379363E-05
16:14:18 0AAA00AAA STEPO       26      1.99     1.99     2.00       1:54       2:02 0.95562419379363E-05
16:14:20 0AAA00AAA STEPO       27      2.03     2.03     2.06       1:56       2:04 0.95562419379363E-05
16:14:22 0AAA00AAA STEPO       28      2.00     2.00     2.02       1:58       2:06 0.95562419379363E-05
16:14:26 0AAA00AAA STEPO       29      4.38     4.38     4.40       2:02       2:10 0.95562419379363E-05
16:14:28 0AAA00AAA STEPO       30      1.99     1.99     2.00       2:04       2:12 0.95562419379363E-05
16:14:30 0AAA00AAA STEPO       31      2.03     2.03     2.05       2:06       2:15 0.95562419379363E-05
16:14:32 FAAB00000 STEPO       32      2.05     2.05     2.06       2:08       2:17 0.10992460223854E-04
16:14:46 000000000 CNT0        32     12.58    12.58    13.37       2:21       2:30 0.10992460223854E-04
```

Changes in output fields

For all the changes described above, all the output fields from the COSP simulators (grib codes 90-96) were compared with the default reference fields. The only non-zero changes that were detected were found in grib code 91 (associated with the ISSCP simulator) when the F90FLAGS compiler options were changed. The relative changes are <0.1% and so not considered significant.

The relevant files from the tests described here can be found on Triolith:
`/proj/nsc/users/struthers/Public/cosp-analysis`

Key:
ref = reference simulation
exp = simulation where changes in `sources/cosp-v2.0/src/simulator/actsim/lidar_simulator.F90` applied
vlev = simulation where changes in `sources/cosp-v2.0/src/cosp_stats.F90` applied
all = simulation where all changes were applied

the *.map files are the Allinea MAP sampling files. They can be viewed by e.g.

```
module load allinea-MAP/6.1
map ref.map
```

Other comments

Profiling results indicate that other than the changes suggested above, significant time is being spent in memory access. Based on this and after some inspection of the COSP code and interface, it is recommended that the amount/number of allocations, deallocations and copies of variables between the IFS and COSP data structures is carefully managed to ensure there is no (or little) redundant or unnecessary data movement. It appears, based on emails and from the EC-Earth development portal, that work is already being done on this point.