CP2K 9.1

ウェブページ

https://www.cp2k.org/

バージョン

9.1 (9.1.0)

ビルド環境

- Intel Parallel Studio 2020 Update 2 (MPI のみ)
- GCC 9.3.1 (devtoolset-9)
- cmake 3.16.3

ビルドに必要なファイル

- cp2k-9.1.0.tar.gz
- tc-install-intelmpi.sh.diff (強制的に mpicc, mpicxx, mpif90 を使うため)

```
--- scripts/stage1/install_intelmpi.sh.org      2022-01-26 09:13:02.023215170 +0900
+++ scripts/stage1/install_intelmpi.sh  2022-01-26 09:14:03.485735051 +0900
@@ -32,15 +32,9 @@
   __SYSTEM__)
     echo "==================== Finding Intel MPI from system paths ===================="
     check_command mpirun "intelmpi" && MPIRUN="$(command -v mpirun)" || exit
-    check_command mpiicc "intelmpi" && MPICC="$(command -v mpiicc)" || exit
-    check_command mpiifort "intelmpi" && MPIFC="$(command -v mpiifort)" || exit
-    if [ $(command -v mpiicpc >&- 2>&-) ]; then
-      check_command mpiicpc "intelmpi" && MPICXX="$(command -v mpiicpc)"
-    elif [ $(command -v mpic++ >&- 2>&-) ]; then
-      check_command mpic++ "intelmpi" && MPICXX="$(command -v mpic++)"
-    else
-      check_command mpicxx "intelmpi" && MPICXX="$(command -v mpicxx)" || exit
-    fi
+    check_command mpicc "intelmpi" && MPICC="$(command -v mpicc)" || exit
+    check_command mpif90 "intelmpi" && MPIFC="$(command -v mpif90)" || exit
+    check_command mpicxx "intelmpi" && MPICXX="$(command -v mpicxx)" || exit
     add_include_from_paths INTELMPI_CFLAGS "mpi.h" $INCLUDE_PATHS
     add_lib_from_paths INTELMPI_LDFLAGS "libmpi.*" $LIB_PATHS
     check_lib -lmpi "intelmpi"
```

- tc-install-libint.sh.diff (エラー回避; 以前よりも少し真っ当な方法に変更)

```
--- scripts/stage3/install_libint.sh.org        2022-01-26 09:34:28.922143775 +0900
+++ scripts/stage3/install_libint.sh    2022-01-26 09:41:44.440708192 +0900
@@ -90,6 +90,7 @@
       # Fix bug in makefile for Fortran module
       sed -i "s/\$(CXX) \$(CXXFLAGS)/\$(FC) \$(FCFLAGS)/g" fortran/Makefile
     fi
+    sed -i 's/FCLIBS)/FCLIBS) -lstdc++/' fortran/Makefile

     make -j $(get_nprocs) > make.log 2>&1
     make install > install.log 2>&1
```

ビルド手順

```
#!/bin/sh

INSTDIR=/local/apl/lx/cp2k910

GITHUB_VERSION=9.1.0
VERSION=9.1.0
```

```
DBCSR_VERSION=v2.2.0

SOURCE_ROOT=/home/users/${USER}/Software/CP2K/${GITHUB_VERSION}

TARBALL=${SOURCE_ROOT}/cp2k-${VERSION}.tar.gz

TC_PATCH_1_1=${SOURCE_ROOT}/tc-install-intelmpi.sh.diff
TC_PATCH_3_1=${SOURCE_ROOT}/tc-install-libint.sh.diff

PARALLEL=12

#-------------------------------------------------------------------------------
umask 0022
export LANG=C
export LC_ALL=C
ulimit -s unlimited

module purge
module load scl/devtoolset-9
module load mpi/intelmpi/2019.8.254
module load cmake/3.16.3

cd $INSTDIR
if [ -d cp2k-${VERSION} ]; then
  mv cp2k-${VERSION} cp2k-erase
  rm -rf cp2k-erase &
fi
tar zxf ${TARBALL}
sleep 5
mv cp2k-${VERSION}/* .
sleep 5
rm -rf cp2k-${VERSION}/{.dockerignore,.github}
rmdir cp2k-${VERSION}

cd ${INSTDIR}/tools/toolchain

# apply patches
patch -p0 < ${TC_PATCH_1_1}
patch -p0 < ${TC_PATCH_3_1}

export CC=gcc
export CXX=g++
export FC=gfortran
export MPICC=mpicc
export MPICXX=mpicxx
export MPIFC=mpif90

./install_cp2k_toolchain.sh --mpi-mode=intelmpi \
                            --math-mode=openblas \
                            --with-gcc=system \
                            --with-cmake=system \
                            --with-openmpi=no \
                            --with-mpich=no \
                            --with-intelmpi=system \
                            --with-libxc=install \
                            --with-libint=install \
                            --with-fftw=install \
                            --with-acml=no \
                            --with-mkl=system \
                            --with-openblas=install \
                            --with-scalapack=install \
                            --with-libsmm=no \
                            --with-libxsmm=install \
                            --with-elpa=install \
                            --with-ptscotch=install \
```

```
                    --with-superlu=install \
                    --with-pexsi=install \
                    --with-quip=install \
                    --with-plumed=install \
                    --with-sirius=install \
                    --with-gsl=install \
                    --with-libvdwxc=install \
                    --with-spglib=install \
                    --with-hdf5=install \
                    --with-spfft=install \
                    --with-spla=install \
                    --with-cosma=no \
                    --with-libvori=install \
                    -j ${PARALLEL}

## -Werror is no longer a problem!
#sed -e "s/-Werror / /g" install/arch/local.psmp > ../../arch/rccs.psmp
cp install/arch/local.psmp ../../arch/rccs.psmp

cd ${INSTDIR}

# prep dbcsr
cd ${INSTDIR}/exts
rmdir dbcsr
git clone https://github.com/cp2k/dbcsr.git
cd dbcsr
git checkout refs/tags/${DBCSR_VERSION}
git submodule update --init --recursive
cd ${INSTDIR}

make -j ${PARALLEL} ARCH=rccs VERSION=psmp
make -j ${PARALLEL} ARCH=rccs VERSION=psmp libcp2k
```

(--with-mkl=no とすべきだったが忘れていた…使われているわけではないので、影響はないはず)

テスト

```
#!/bin/sh
#PBS -l select=1:ncpus=16:mpiprocs=16:ompthreads=1:jobtype=core
#PBS -l walltime=12:00:00

export LC_ALL=C
export LANG=""
export OMP_STACKSIZE=64M

module purge
module load scl/devtoolset-9
module load mpi/intelmpi/2019.8.254
module load cmake/3.16.3

CP2K=/local/apl/lx/cp2k910

CP2K_ARCH=rccs
CP2K_VER=psmp
TIMEOUT=600
PARALLEL=16

ulimit -s unlimited
cd ${CP2K}/regtesting/${CP2K_ARCH}/${CP2K_VER}
rm -rf LAST-${CP2K_ARCH}-${CP2K_VER}

# serial test
../../../tools/regtesting/do_regtest \
      -nobuild \
      -arch ${CP2K_ARCH} \
```

```
        -version ${CP2K_VER} \
        -mpiranks 1 \
        -ompthreads 1 \
        -jobmaxtime ${TIMEOUT} \
        -cp2kdir ../../../ \
        -maxtasks ${PARALLEL} >& regtest_mpi1_omp1.log
rm -rf LAST-${CP2K_ARCH}-${CP2K_VER}

# omp test
../../../tools/regtesting/do_regtest \
        -nobuild \
        -arch ${CP2K_ARCH} \
        -version ${CP2K_VER} \
        -mpiranks 1 \
        -ompthreads 2 \
        -jobmaxtime ${TIMEOUT} \
        -cp2kdir ../../../ \
        -maxtasks ${PARALLEL} >& regtest_mpi1_omp2.log
rm -rf LAST-${CP2K_ARCH}-${CP2K_VER}

# mpi test
../../../tools/regtesting/do_regtest \
        -nobuild \
        -arch ${CP2K_ARCH} \
        -version ${CP2K_VER} \
        -mpiranks 2 \
        -ompthreads 1 \
        -jobmaxtime ${TIMEOUT} \
        -cp2kdir ../../../ \
        -maxtasks ${PARALLEL} >& regtest_mpi2_omp1.log
rm -rf LAST-${CP2K_ARCH}-${CP2K_VER}

# mpi/openmp test
../../../tools/regtesting/do_regtest \
        -nobuild \
        -arch ${CP2K_ARCH} \
        -version ${CP2K_VER} \
        -mpiranks 2 \
        -ompthreads 2 \
        -jobmaxtime ${TIMEOUT} \
        -cp2kdir ../../../ \
        -maxtasks ${PARALLEL} >& regtest_mpi2_omp2.log
rm -rf LAST-${CP2K_ARCH}-${CP2K_VER}

# yet another mpi test
../../../tools/regtesting/do_regtest \
        -nobuild \
        -arch ${CP2K_ARCH} \
        -version ${CP2K_VER} \
        -mpiranks 8 \
        -ompthreads 1 \
        -jobmaxtime ${TIMEOUT} \
        -cp2kdir ../../../ \
        -maxtasks ${PARALLEL} >& regtest_mpi8_omp1.log
rm -rf LAST-${CP2K_ARCH}-${CP2K_VER}

# yet another mpi/openmp test
../../../tools/regtesting/do_regtest \
        -nobuild \
        -arch ${CP2K_ARCH} \
        -version ${CP2K_VER} \
        -mpiranks 8 \
        -ompthreads 2 \
        -jobmaxtime ${TIMEOUT} \
        -cp2kdir ../../../ \
```

```
      -maxtasks ${PARALLEL} >& regtest_mpi8_omp2.log
rm -rf LAST-${CP2K_ARCH}-${CP2K_VER}
```

テスト結果

```
[qf7@ccfep8 9.1.0]$ cd /local/apl/lx/cp2k910/regtesting/rccs/psmp/
[qf7@ccfep8 psmp]$ grep "GREPME" *.log
regtest_mpi1_omp1.log:GREPME 0 0 3596 0 3596 X
regtest_mpi1_omp2.log:GREPME 0 0 3596 0 3596 X
regtest_mpi2_omp1.log:GREPME 0 0 3652 0 3652 X
regtest_mpi2_omp2.log:GREPME 0 0 3652 0 3652 X
regtest_mpi8_omp1.log:GREPME 3 20 3643 0 3666 X
regtest_mpi8_omp2.log:GREPME 3 20 3643 0 3666 X
```

8 MPI 時のみでいくつかのテストでエラー、数値誤差が発生

- 8 MPI 1 OMP
  - h2o_f35.inp: RUNTIME FAIL
    - Can not add blocks to matrix with no rows. (dbcsr_block_access.F:555)
  - HeH-mixed-cdft-5.inp & HeH-mixed-cdft-7.inp: RUNTIME FAIL
    - Load balancing error: too much data to redistribute. (mixed_cdft_methods.F:3041)
    - may be fixed by changing LOAD_SCALE?
  - H2O-32-dftb-ls-2_mult.inp: WRONG 11 (ENERGY| Total FORCE_EVAL error: 3.15639644e-04 tolerance: 1e-12)
  - H2O-32-dftb-ls-2.inp: WRONG 11 (ENERGY| Total FORCE_EVAL error: 3.15639644e-04 tolerance: 1e-12)
  - H2O-OT-ASPC-1.inp: WRONG 1 (Total energy error 2.09579836e-11 tolerance 4e-14)
  - H2O_grad_gpw.inp: WRONG 11 (ENERGY| Total FORCE_EVAL error 1.31898310e-09 tolerance: 7e-11)
  - H2O-6.inp: WRONG 1 (Total energy error: 3.34421827e-11 tolerance: 2e-14)
  - h2o_v01.inp: WRONG 8 (VIB|Frequency error: 2.36308063e-02 tolerance: 1.0E-07)
  - H2O-debug-5.inp: WRONG 86 (DIPOLE : CheckSum error: 2.20748333e-07 tolerance: 4e-10)
  - H2O-debug-6.inp: WRONG 86 (DIPOLE : CheckSum error: 2.20751799e-07 tolerance: 4e-10)
  - H2O_grad_mme.inp: WRONG 11 (ENERGY| Total FORCE_EVAL error: 6.67224387e-09 tolerance: 6e-09)
  - H2O_grad_ri-hfx.inp: WRONG 11 (ENERGY| Total FORCE_EVAL error: 6.40217238e-09 tolerance: 6e-09)
  - ethene-vib-mode-sel-int.inp: WRONG 21 (VIB| error: 5.18090909e+02 tolerance: 1.0E-14)
  - N3dye_vib_bfgs2.inp: WRONG 18 (MS| TRACKED FREQUENCY error: 2.61646572e+00 tolerance: 2e-08)
  - N3dye_vib_inv_atoms.inp: WRONG 18 (MS| TRACKED FREQUENCY error: 2.11358036e+00 tolerance: 1.0E-14)
  - N3dye_vib_restart_vec3.inp: WRONG 18 (MS| TRACKED FREQUENCY error: 1.42820600e-03 tolerance: 1.0E-14)
  - N3dye_vib_restart_vec2.inp: WRONG 18 (MS| TRACKED FREQUENCY error: 2.04088385e+00 tolerance: 1.0E-14)
  - N3dye_vib_restart_vec.inp: WRONG 18 (MS| TRACKED FREQUENCY error: 1.42831089e-03 tolerance: 1.0E-14)
  - N3dye_vib_restart_vec4.inp: WRONG 18 (MS| TRACKED FREQUENCY error: 3.26939470e+00 tolerance: 1.0E-14)
  - H2O-VIB-MS-INT-1.inp: WRONG 21 (VIB| error: 6.54403214e-02 tolerance: 1.0E-14)
  - H2O-VIB-MS-INT-2.inp: WRONG 21 (VIB| error: 6.54403214e-02 tolerance: 1.0E-14)
  - vib-mixed.inp: WRONG 8 (VIB|Frequency error: 1.57503770e-02 tolerance: 1e-14)
- 8 MPI 2 OMP
  - 細かすぎる数字を除けば 1 OMP 時とすべて同じエラー

ベンチマーク

8.2 と同じように H2O-64.inp を利用。(時間は grep "CP2K " *.log で表示される値から)
20 回実行し、最初の 1 回を除いた平均値。(初回は速度が安定しないため)

jobtype	総コア数 (ノード数)	MPI	OMP	GPU	elapse(sec)
core	18 (1)	18	1	-	59.63
small	40 (1)	40	1	-	45.74
small	80 (2)	80	1	-	29.37
small	160 (4)	160	1	-	22.4

- core についてはノードを全て確保した上で 18 コアで実行。(実行ノードに他のジョブがある場合は速度がこの数字よりも落ちます。)
- 若干の差はあるものの、8.2 の時と大きくは変わらない傾向
  - OpenMP についても 8.2 と同様にこの規模では有効では無い模様

メモ

- OpenMPI 3.x を使った場合、追加でいくつかのテストに失敗するため、Intel MPI を利用
  - OpenMPI 4.x を使った場合は状況がさらに悪化するため、OpenMPI を使う場合には 3.x が推奨
    - https://github.com/cp2k/cp2k/issues/1274#issuecomment-753291518
- Parallel Studio 2020 update 2 の Intel MPI では GCC 10 以降は利用できず(対応する mpi.mod が存在しない様子)
  - MVAPICH + GCC10 のような環境については未検証
- COSMA は念のため無効化
  - https://github.com/cp2k/cp2k/issues/1826
  - https://github.com/cp2k/cp2k/issues/1861
- インテルコンパイラについては未検証
- MKL を有効にした場合若干速度が落ちているように見えるため、OpenBLAS+scalapack を利用
- 今回は libcp2k.a も作成(Gromacs との連携を想定)
- 同一インプットでも異常に遅くなるケースが稀に発生する？
  - 一度発生したものの再現取れず