

NWChem-7.2.2

Webpage

<https://nwchemgit.github.io/>

Version

7.2.2

Build Environment

- GCC 12.1.1 (gcc-toolset-12)
- Intel MKL 2023.2.0
- Open MPI 4.1.6

Files Required

- nwchem-7.2.2-release.revision-74936fb9-src.2023-11-03.tar.bz2
- runtest.md.mpi

```
#!/bin/sh
./runtests.mpi.unix procs 48 \
na_k/nak \
na_k/nak_md \
crown/crown_md \
ethanol/ethanol_md \
ethanol/ethanol_ti \
had/had_em \
had/had_md \
prep/a3n \
prep/aal \
prep/fsc \
water/water_md
```

Build Procedure

```
#!/bin/sh

VERSION=7.2.2
INSTALL_PREFIX=/apl/nwchem/7.2.2

BASEDIR=/home/users/${USER}/Software/NWChem/${VERSION}
TARBALL=${BASEDIR}/nwchem-7.2.2-release.revision-74936fb9-src.2023-11-03.tar.bz2

WORKDIR=/gwork/users/${USER}

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

module -s purge
module -s load gcc-toolset/12
module -s load mkl/2023.2.0
module -s load openmpi/4.1.6/gcc12

cd ${WORKDIR}
if [ -d nwchem-${VERSION} ]; then
  mv nwchem-${VERSION} nwchem-erase
  rm -rf nwchem-erase &
fi

tar jxf ${TARBALL}
```

```

export NWCHEM_TOP=${WORKDIR}/nwchem-${VERSION}
export NWCHEM_MODULES="all python"
export NWCHEM_TARGET=LINUX64
export ARMCI_NETWORK=MPI-PR

export USE_OPENMP=y
export USE_MPI=y
export USE_MPIF=y
export USE_MPIF4=y

export USE_NOFSCHECK=TRUE
export USE_NOIO=TRUE
export MRCC_METHODS=TRUE
export CCSDTQ=TRUE
export LIB_DEFINES=-DDFLT_TOT_MEM=180000000 # 1.8 GB/process

export PYTHONVERSION=3.6

export BLAS_SIZE=8
export BLASOPT="-L${MKLRROOT}/lib/intel64 -lmkl_intel_ilp64 -lmkl_gnu_thread -lmkl_core -liomp5 -lpthread -lm -ldl"
export LAPACK_SIZE=8
export LAPACK_LIB="${BLASOPT}"
export USE_SCALAPACK=y
export SCALAPACK_SIZE=8
export SCALAPACK="-L${MKLRROOT}/lib/intel64 -lmkl_scalapack_ilp64 -lmkl_blacs_openmpi_ilp64 -lmkl_intel_ilp64 -lmkl_gnu_thread -lmkl_core -liomp5 -lpthread -lm -ldl"

cd ${NWCHEM_TOP}/src

# cheat
mkdir ../bin
ln -s /usr/libexec/platform-python3.6-config ../bin/python3.6-config
export PATH="${PATH}:${NWCHEM_TOP}/bin:${NWCHEM_TOP}/QA"

make nwchem_config
make

# involve also version info
cd ${NWCHEM_TOP}/src/util
make version
make
cd ${NWCHEM_TOP}/src
make link
cd ${NWCHEM_TOP}

# installation; assume INSTALL_PREFIX directory already exists
cp -fr LICENSE.TXT README.md release.notes.* examples ${INSTALL_PREFIX}

mkdir ${INSTALL_PREFIX}/bin
mkdir ${INSTALL_PREFIX}/data

cp -f ${NWCHEM_TOP}/bin/${NWCHEM_TARGET}/nwchem ${INSTALL_PREFIX}/bin
chmod 755 ${INSTALL_PREFIX}/bin/nwchem

cp -fr ${NWCHEM_TOP}/src/data          ${INSTALL_PREFIX}
cp -fr ${NWCHEM_TOP}/src/basis/libraries  ${INSTALL_PREFIX}/data
cp -fr ${NWCHEM_TOP}/src/basis/libraries.bse ${INSTALL_PREFIX}/data
cp -fr ${NWCHEM_TOP}/src/nwppw/libraryyps  ${INSTALL_PREFIX}/data

# create default.nwchemrc
cat << EOS > ${INSTALL_PREFIX}/default.nwchemrc
nwchem_basis_library ${INSTALL_PREFIX}/data/libraries/
nwchem_nwppw_library ${INSTALL_PREFIX}/data/libraryyps/
ffield amber

```

```

amber_1 ${INSTALL_PREFIX}/data/amber_s/
amber_2 ${INSTALL_PREFIX}/data/amber_x/
#amber_3 ${INSTALL_PREFIX}/data/amber_q/
#amber_4 ${INSTALL_PREFIX}/data/amber_u/
spce  ${INSTALL_PREFIX}/data/solvents/spce.rst
charmm_s ${INSTALL_PREFIX}/data/charmm_s/
charmm_x ${INSTALL_PREFIX}/data/charmm_x/
EOS

# some tests need this...
cp -f ${INSTALL_PREFIX}/default.nwchemrc ~/.nwchemrc

# run test
export NWCHEM_EXECUTABLE=${INSTALL_PREFIX}/bin/nwchem
export OMP_NUM_THREADS=1

cd ${NWCHEM_TOP}/QA
./doqmtests.mpi 2 fast >& doqmtests.mpi.fast.log
#./doqmtests.mpi 48 >& doqmtests.mpi.log

mv testoutputs testoutputs-serial

mkdir -p ${INSTALL_PREFIX}/testlog
cp -fr doqmtests.mpi.fast.log \
    testoutputs-serial \
    ${INSTALL_PREFIX}/testlog

```

Test

In addition to the serial test above, we performed a large test using the job script below.

```

#!/bin/sh
#PBS -l select=1:ncpus=48:mpiprocs=48:ompthreads=1
#PBS -l walltime=24:00:00

VERSION=7.2.2
INSTALL_PREFIX=/apl/nwchem/7.2.2

BASEDIR=/home/users/${USER}/Software/NWChem/${VERSION}
TARBALL=${BASEDIR}/nwchem-7.2.2-release.revision-74936fb9-src.2023-11-03.tar.bz2

WORKDIR=/gwork/users/${USER}

RUNTESTMD=runtest.md.mpi
RUNTESTMD_PATH=${BASEDIR}/${RUNTESTMD}

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

module -s purge
module -s load gcc-toolset/12
module -s load mkl/2023.2.0
module -s load openmpi/4.1.6/gcc12

export NWCHEM_TOP=${WORKDIR}/nwchem-${VERSION}
# run test
export NWCHEM_EXECUTABLE=${INSTALL_PREFIX}/bin/nwchem
export PATH="$PATH:$NWCHEM_TOP/QA"

cd ${NWCHEM_TOP}/QA

./doqmtests.mpi 48 >& doqmtests.mpi.log
cp -f ${RUNTESTMD_PATH} .

```

```
sh ${RUNTESTMD} >& runtest.md.mpi.log

mv testoutputs testoutputs-mpi

mkdir -p ${INSTALL_PREFIX}/testlog
cp -fr doqmtests.mpi.log \
    runtest.md.mpi.log \
    testoutputs-mpi \
    ${INSTALL_PREFIX}/testlog
```

Test Results

Test results are copied to `/apl/nwchem/7.2.2/testlog`.

Failed tests for serial version.

- tests/h2o-b3lyp-disp/h2o-b3lyp-disp: insufficient memory
- (tests/oh2/oh2: to be failed)
- tests/dft_siosi3/dft_siosi3: insufficient memory
- tests/pspw_md/pspw_md: minor numerical error (-14.38970 vs -14.38971)
- tests/tce_cr_eom_t_ozone/tce_cr_eom_t_ozone: createfile: failed ga_create size/nproc bytes:
- tests/tce_mrcc_bwcc/tce_mrcc_bwcc: minor numerical error (-76.0643135404 vs -76.0643135403)
- tests/tce_mrcc_mkcc/tce_mrcc_mkcc: minor numerical error (-76.0630229504 vs -76.0630229503, -76.0702306002 vs -76.0702306001)
- tests/tce_mrcc_bwcc_subgroups/tce_mrcc_bwcc_subgroups: Floating point exception: integer divide by zero
- tests/qmmm_grad0/qmmm_grad0: insufficient memory
- tests/lys_qmmm/lys_qmmm: insufficient memory
- tests/ethane_qmmm/ethane_qmmm: insufficient memory
- tests/qmmm_freq/qmmm_freq: insufficient memory
- tests/h2o-b3lyp-disp/h2o-b3lyp-disp: insufficient memory
- tests/o2_ccca/o2_ccca: createfile: failed ga_create size/nproc bytes:

Failed tests for parallel version.

- tests/h2o-b3lyp-disp/h2o-b3lyp-disp: insufficient memory
- (tests/oh2/oh2: to be failed)
- tests/qmmm_grad0/qmmm_grad0: insufficient memory
- tests/lys_qmmm/lys_qmmm: insufficient memory
- tests/ethane_qmmm/ethane_qmmm: insufficient memory
- tests/qmmm_freq/qmmm_freq: insufficient memory
- tests/h2o-b3lyp-disp/h2o-b3lyp-disp: insufficient memory
- tests/dmo_tddft_cd_velocity/dmo_tddft_cd_velocity: minor numerical error (1.620 vs 1.693; Rotatory Strength)
- tests/k6h2o/k6h2o: k6h2o.err: No such file or directory

MD tests.

- tests/na_k/nak_md: ?
- tests/crown/crown_md: Unable to read previous box pair list
- tests/ethanol/ethanol_md: ethanol_md.err: No such file or directory
- tests/ethanol/ethanol_ti: Unable to read previous box pair list
- tests/had/had_em: 0: Dimension mwm too small 0
- tests/had/had_md: Unresolved atom types in fragment HEM
- tests/prep/a3n: Unresolved atom types in fragment HEM

Notes

- Intel compiler version shows slightly better performance than GCC version in many cases (especially for large ones). However, Intel version is terribly slow in some limited cases. We thus employed GCC this time.
- MKL is better than OpenBLAS+Scalapack in terms of performance.
 - This does not depend on compiler type (GCC/Intel).
- (CUDA version was built as a separated binary.)