

Gromacs 2025.2 with CUDA support

Webpage

<http://www.gromacs.org/>

Version

2025.2

Build Environment

- GCC 13.3.1 (gcc-toolset-13)
- CUDA 12.8 Update 1
- Open MPI 4.1.8 (CUDA-aware)
- cmake 3.31.6
- python 3.12.9 (from conda env. prepared on Mar 10, 2025)
- openblas 0.3.29-lp64
- cuDNN 9.10.1
- cuDSS 0.5.0
- cuSPARSElt 0.7.1

Files Required

- gromacs-2025.2.tar.gz
- regressiontests-2025.2.tar.gz
- (Some of files will be downloaded during the installation.)

Build Procedure

LibTorch 2.7.0 (briefly)

Source code downloaded for CPU version was also used for GPU version.

```
[user@ccfep pytorch]$ mkdir build && cd build
[user@ccfep build]$ module purge
[user@ccfep build]$ . /apl/conda/20250310/conda_init.sh
(base) [user@ccfep build]$ export CUDSS_ROOT_DIR=/apl/cudss/0.5.0/libcudss-linux-x86_64-0.5.0.16_cuda12-archive
(base) [user@ccfep build]$ export CUSPARSELT_ROOT_DIR=/apl/cusparselt/0.7.1/libcusparselt-linux-x86_64-0.7.1.0-archive
(base) [user@ccfep build]$ export TORCH_CUDA_ARCH_LIST="8.0"
(base) [user@ccfep build]$ module load cuda/12.8u1 cudnn/9.10.1-cuda12 cudss/0.5.0.16-cuda12 cusparselt/0.7.1 openblas/0.3.29-lp64 gcc-toolset/13
(base) [user@ccfep build]$ cmake3 .. \
  -GNinja \
  -DBLAS=OpenBLAS \
  -DBUILD_FUNCTORCH=OFF \
  -DBUILD_PYTHON=False \
  -DBUILD_TEST=True \
  -DCMAKE_BUILD_TYPE=Release \
  -DCMAKE_INSTALL_PREFIX=/apl/libtorch/2.7.0/cu128 \
  -DCMAKE_PREFIX_PATH="/apl/cuda/12.8u1;/apl/cudnn/9.10.1/cudnn-linux-x86_64-9.10.1.4_cuda12-archive" \
  -DPython_EXECUTABLE=/lustre/rcs/apl/ap/conda/20250310/envs/gpuenv/bin/python \
  -DTORCH_BUILD_VERSION=2.7.0a0+git1341794 \
  -DUSE_CUDA=ON \
  -DUSE_CUDNN=ON \
  -DUSE_CUDSS=ON \
  -DUSE_CUSPARSELT=ON \
  -DUSE_NUMPY=True
(base) [user@ccfep build]$ ninja -j2
(base) [user@ccfep build]$ ninja -j2 install
```

Tests were performed on ccgpu.

```
(base) [user@ccgpu build]$ ninja -j2 test
```

- Following tests failed.
 - 56 - vec_test_all_types_AVX512 (ILLEGAL) : our CPUs do not support AVX512...
 - 57 - vec_test_all_types_AVX2 (Failed) : ?
 - 144 - ProcessGroupNCCLErrorsTest (Failed) : ?

Gromacs 2025.2

Built and tested on ccgpu.

```
#!/bin/sh

VERSION=2025.2
INSTALL_PREFIX=/apl/gromacs/${VERSION}-CUDA

BASEDIR=/home/users/${USER}/Software/Gromacs/${VERSION}/
GROMACS_TARBALL=${BASEDIR}/gromacs-${VERSION}.tar.gz
REGRESSION_TARBALL=${BASEDIR}/regressiontests-${VERSION}.tar.gz
WORKDIR=/gwork/users/${USER}
REGRESSION_PATH=${WORKDIR}/regressiontests-${VERSION}

FFTW_VER=3.3.10
FFTW_PATH=${BASEDIR}/fftw-${FFTW_VER}.tar.gz

PARALLEL=12
export LANG=C

#-----
umask 0022

module -s purge
module -s load gcc-toolset/13
module -s load openmpi/4.1.8/gcc13-cuda12.8u1
module -s load cuda/12.8u1
module -s load cmake/3.31.6
module -s load openblas/0.3.29-lp64
module -s load cudnn/9.10.1-cuda12
module -s load cudss/0.5.0.16-cuda12
module -s load cusparse/0.7.1

TORCH_DIR=/apl/libtorch/2.7.0/cu128
OPENBLAS_DIR=/apl/openblas/0.3.29/lp64
export CUDNN_ROOT_DIR=/apl/cudnn/9.10.1/cudnn-linux-x86_64-9.10.1.4_cuda12-archive
export CUDSS_ROOT_DIR=/apl/cudss/0.5.0/libcudss-linux-x86_64-0.5.0.16_cuda12-archive
export CUSPARSELT_ROOT_DIR=/apl/cusparse/0.7.1/libcusparse-lt-linux-x86_64-0.7.1.0-archive

#export CUDA_VISIBLE_DEVICES=0,1
unset OMP_NUM_THREADS

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

if [ -d regressiontests-${VERSION} ]; then
  mv regressiontests-${VERSION} regressiontests_erase
  rm -rf regressiontests_erase &
fi

tar xzf ${GROMACS_TARBALL}
tar xzf ${REGRESSION_TARBALL}
cd gromacs-${VERSION}

# single precision, no MPI
mkdir rccs-s
cd rccs-s
```

```

cmake .. \
-DMAKE_PREFIX_PATH="${TORCH_DIR};${OPENBLAS_DIR}" \
-DMAKE_INSTALL_PREFIX=${INSTALL_PREFIX} \
-DMAKE_VERBOSE_MAKEFILE=ON \
-DMAKE_C_COMPILER=gcc \
-DMAKE_CXX_COMPILER=g++ \
-DGMX_MPI=OFF \
-DGMX_GPU=CUDA \
-DGMX_DOUBLE=OFF \
-DGMX_THREAD_MPI=ON \
-DGMX_USE_CUFFTMP=OFF \
-DGMX_NNPOT=TORCH \
-DCAFFE2_USE_CUDNN=ON \
-DCAFFE2_USE_CUSPARSELT=ON \
-DUSE_CUDSS=ON \
-DPython_EXECUTABLE=/usr/bin/python3 \
-DGMX_BUILD_OWN_FFTW=ON \
-DGMX_BUILD_OWN_FFTW_URL=${FFTW_PATH} \
-DREGRESSIONTEST_DOWNLOAD=OFF \
-DREGRESSIONTEST_PATH=${REGRESSION_PATH}
make -j${PARALLEL} && make check && make install
cd ..

```

single precision, with MPI

mkdir rccs-mpi-s

cd rccs-mpi-s

```

cmake .. \
-DMAKE_PREFIX_PATH="${TORCH_DIR};${OPENBLAS_DIR}" \
-DMAKE_INSTALL_PREFIX=${INSTALL_PREFIX} \
-DMAKE_VERBOSE_MAKEFILE=ON \
-DMAKE_C_COMPILER=mpicc \
-DMAKE_CXX_COMPILER=mpicxx \
-DGMX_MPI=ON \
-DGMX_GPU=CUDA \
-DGMX_DOUBLE=OFF \
-DGMX_THREAD_MPI=OFF \
-DGMX_USE_CUFFTMP=OFF \
-DGMX_NNPOT=TORCH \
-DCAFFE2_USE_CUDNN=ON \
-DCAFFE2_USE_CUSPARSELT=ON \
-DUSE_CUDSS=ON \
-DPython_EXECUTABLE=/usr/bin/python3 \
-DGMX_USE_PLUMED=ON \
-DGMX_BUILD_OWN_FFTW=ON \
-DGMX_BUILD_OWN_FFTW_URL=${FFTW_PATH} \
-DREGRESSIONTEST_DOWNLOAD=OFF \
-DREGRESSIONTEST_PATH=${REGRESSION_PATH}
make -j${PARALLEL} && make check && make install
cd ..

```

Tests (Gromacs)

All the tests have passed successfully.

Notes

- Neural Network Potential is available thanks to libtorch 2.7.0.
 - The model should be trained/created with pytorch 2.7.0 according to official document. If you want to use different version of pytorch, you may need to prepare it by yourselves. The above procedure may be helpful.
 - To use GPU for NNP calculations, "export GMX_NN_DEVICE=cuda" is required. (Otherwise, default value "cpu" is used.)
 - In gromacs/2025.2-CUDA module on RCCS system, this is set to "cuda".
 - The actual setting is shown as "GMX_NN_DEVICE environment variable found. Using device: cuda." in the logfile.
 - You can skip compilation of libtorch if binary version can be used. (Unfortunately, we have glibc version mismatch problem...)

- External plumed is also available.
 - PLUMED_KERNEL environment variable is set to libplumedKernel.so of plumed 2.9.3 in gromacs/2025.2 module on RCCS system.