GROMACS 5.1.2
Installation Best Practices

HPC ADVISORY COUNCIL
NETWORK OF EXPERTISE
BEST PRACTICES

1. Introduction:
The following best practices document is provided as courtesy of the HPC Advisory Council.

2. Application Description:
GROMACS is a versatile package to perform molecular dynamics, i.e. simulate the Newtonian equations of motion for systems with hundreds to millions of particles. It is primarily designed for biochemical molecules like proteins, lipids and nucleic acids that have a lot of complicated bonded interactions, but since GROMACS is extremely fast at calculating the non-bonded interactions (that usually dominate simulations) many groups are also using it for research on non-biological systems, e.g. polymers. More information on GROMACS can be found at the following web site: http://www.gromacs.org.

3. Version Information:
This guideline is based on GROMACS 5.1.2. The source code can be downloaded from http://www.gromacs.org/Downloads.

4. Prerequisites:

4.1 Hardware:

The instructions from this best practice have been tested on the HPC Advisory Council, Dell™ PowerEdge™ R730 32-node cluster.

- Dual Socket Intel® Xeon® 14-core CPUs E5-2697A V4 @ 2.60 GHz
- Mellanox ConnectX-4 EDR 100Gb/s InfiniBand adapters
- Mellanox Switch-IB SB7700 36-Port 100Gb/s EDR InfiniBand switches
- NVIDIA Tesla K40 and K80 GPUs

4.2 Software:

a) OS: Red Hat Enterprise Linux 6.5
b) GNU Compiler for Linux; 4.7.x or higher
c) Required software for GPU computing
   - NVIDIA CUDA 6.5 or 7.0
   - CUDA-aware MPI, such as:
     - Mellanox HPC-X MPI Toolkit v1.3 (based on Open MPI 1.8.x)
     - Open MPI 1.8.x
     - MVAPICH2-GDR
   - cmake 2.8.x or higher

5. Building GROMACS

5.1 CPU only version

CPU version can be built with either single or double precision.

```
MPI=hpcx
MPI=impi
module load intel/compiler/2016.3.210
if [ "$MPI" == "impi" ]; then
        module load intel/impi/5.1.3.210
        export I_MPI_CC=icc
```

```
        export I_MPI_CXX=icpc
        export I_MPI_FC=ifort
        export I_MPI_F90=ifort
elif [ "$MPI" == "hpcx" ]; then
        module load hpcx/icc-2016
        export OMPI_MPICC=icc
        export OMPI_MPICXX=icpc
        export OMPI_MPIFC=ifort
        export OMPI_MPIF90=ifort
fi

BASE=$PWD

rm -fr $BASE/build $BASE/install-$MPI
mkdir $BASE/build
cd $BASE/build

BUILD_FLAGS="-DGMX_FFT_LIBRARY=mkl -DMKL_LIBRARIES=-mkl -DMKL_INCLUDE_DIR=$MKLR
OOT/include \
        -DGMX_SIMD=AVX2_256 -DGMX_BUILD_MDRUN_ONLY=on \
        -DGMX_DEFAULT_SUFFIX=OFF -DGMX_BINARY_SUFFIX=_$MPI \
        -DCMAKE_INSTALL_PREFIX=$BASE/install-$MPI \
        -DCMAKE_C_COMPILER=mpicc -DCMAKE_CXX_COMPILER=mpicxx -DGMX_MPI=on"

cmake .. $BUILD_FLAGS
make -j 16
make check
make install
```

5.2 GPU version

GPU version is available in only single precision.

```
# Intel compilers 2015 does not support CUDA 6.x, but CUDA 7.x
module load gnu/4.9.3
module load cmake/3.2.3
source /opt/intel/composer_xe_2015.3.187/bin/compilervars.sh intel64

CVER=7.0
module use /application/hpcx-v1.3.336-icc-MLNX_OFED_LINUX-3.0-1.0.1-redhat6.5-x
86_64/modulefiles
module load hpcx-ompi-mellanox-v1.8_vanilla_cuda$CVER
export OMPI_MPICXX=icpc
export OMPI_MPICC=icc
export OMPI_MPIF77=ifort
export OMPI_MPIF90=ifort

export PATH=/usr/local/cuda-$CVER/bin:$PATH
export CFLAGS="-I/usr/local/cuda-$CVER/include"
export LDFLAGS="-L/usr/local/cuda-$CVER/lib64"
export LD_LIBRARY_PATH=/usr/local/cuda-$CVER/lib64:$LD_LIBRARY_PATH
```

```
BASEDIR=$PWD
rm -fr $BASEDIR/build
mkdir $BASEDIR/build
cd $BASEDIR/build

FLAGS="-DGMX_GPU=ON "
FLAGS+="-DGMX_DOUBLE=OFF "
FLAGS+="-DCUDA_TOOLKIT_ROOT_DIR=/usr/local/cuda-$CVER "
FLAGS+="-DCMAKE_INSTALL_PREFIX=$BASEDIR/install/GPU-ompi.SP "

cmake .. $FLAGS -DGMX_MPI=ON -DGMX_CPU_ACCELERATION=AVX2_256 -DGMX_FFT_LIBRARY=
fftw3 -DCMAKE_C_COMPILER=icc -DCMAKE_CXX_COMPILER=icpc -DGMX_PREFER_STATIC_LIBS
=ON -DGMX_FFT_LIBRARY=mkl

make -j 32
make install
```

5.3 Building Mellanox HPC-X (based on Open MPI) with CUDA support

```
export HPCX_HOME=/application/hpcx-v1.3.336-icc-MLNX_OFED_LINUX-3.0-1.0.1-redha
t6.5-x86_64
module use $HPCX_HOME/modulefiles
module load hpcx
source /opt/intel/composer_xe_2015.3.187/bin/compilervars.sh intel64

CVER=7.0
export PATH=/usr/local/cuda-$CVER/bin:$PATH
export LD_LIBRARY_PATH=/usr/local/cuda-$CVER/lib64:$LD_LIBRARY_PATH

./configure --prefix=${HPCX_HOME}/ompi-mellanox-v1.8_vanilla_cuda$CVER \
            --with-slurm --with-pmi \
            --with-cuda=/usr/local/cuda-$CVER

make all
make install
```

6. Running GROMACS with CPU

6.1 CPU benchmark with Mellanox HPC-X (based on Open MPI)

```
module load hpcx/icc-2016
mpirun -report-bindings -mca coll_fca_enable 0 -mca coll_hcoll_enable 1 -mca co
ll_hcoll_np 0 -mca pml yalla -mca mtl_mxm_np 0 -mca btl_openib_if_include mlx5_
0:1 -x MXM_RDMA_PORTS=mlx
5_0:1 -bind-to core -x MXM_TLS=ud,shm,self -np 112  <path to GROMACS executable
s>/mdrun_hpcx  -s topol -noconfout -nsteps 10000 -g <logfile>
```

6.2 CPU benchmark with Intel MPI

```
module load intel/compiler/2016.3.210
```

```
module load intel/impi/5.1.3.210
mpiexec -IB -genv MV2_USE_APM 0 -genv I_MPI_OFA_ADAPTER_NAME mlx5_0 -genv I_MPI
_OFA_NUM_PORTS 1 -perhost 32 -ppn 32  -envall -np 112  <path to GROMACS executa
bles>/mdrun_impi  -s topol -noconfout -nsteps 10000 -g <logfile>
```

7. Running GROMACS with CPU

7.1 GPU benchmark using Mellanox HPC-X (based on Open MPI)

```
module load gnu/4.9.3
source /opt/intel/composer_xe_2015.3.187/bin/compilervars.sh intel64
export LD_LIBRARY_PATH=/usr/local/cuda-7.0/bin:$LD_LIBRARY_PATH
module use /application/hpcx-v1.3.336-icc-MLNX_OFED_LINUX-3.0-1.0.1-redhat6.5-x
86_64/modulefiles
module load hpcx-ompi-mellanox-v1.8_vanilla_cuda7.0

mpirun -hostfile hostfile.txt -np 8 --report-bindings -mca btl_sm_use_knem 1 -m
ca coll_fca_enable 0 -mca coll_hcoll_enable 0 -mca mtl ^mxm -mca pml ob1 -mca b
tl openib,sm,self -mca btl_openib_if_include mlx5_0:1 --bind-to none  <path to
Gromacs executables>/mdrun_mpi -s topol-pme
```