Benchmark lignocellulose

Builds

On the website(https://hpcadvisorycouncil.atlassian.net/wiki/spaces/HPCWORKS/pages/1453719580/Gromacs+Challenge), the suggest the following build:

```
wget http://ftp.gromacs.org/pub/gromacs/gromacs-2020.2.tar.gz
tar xfz gromacs-2020.2.tar.gz
cd gromacs-2020.2
# <load compilers and MPI>
mkdir build
cd build
cmake .. -DGMX_FFT_LIBRARY=mkl -DMKL_LIBRARIES=-mkl -DMKL_INCLUDE_DIR=$MKLROOT/include \
        -DGMX_SIMD=AVX2_256 \
        -DGMX_MPI=ON \
        -DGMX_BUILD_MDRUN_ONLY=on \
        -DBUILD_SHARED_LIBS=on \
        -DCMAKE_INSTALL_PREFIX=<install path> \
        -DCMAKE_C_COMPILER=mpicc -DCMAKE_CXX_COMPILER=mpicxx
make -j 16 install
```

Results

V1

▼ **Build** script, referred to as V1

```
wget http://ftp.gromacs.org/pub/gromacs/gromacs-2020.1.tar.gz
tar -xzf gromacs-2020.1.tar.gz

cd gromacs-2020.1

# need intel 18, because gromacs doesn't support 19
module load cmake/3.14.3
module load gcc/5.5.0
source /home/mbeukman/intel/2018/parallel_studio_xe_2018.4.057/psxevars.sh
module load cuda/10.0

mkdir build
cd build

cmake .. -DGMX_BUILD_OWN_FFTW=ON -DREGRESSIONTEST_DOWNLOAD=ON \
-DCMAKE_INSTALL_PREFIX=/home/mbeukman/otherBenches/gromacs/v1GPU/install \
-DCMAKE_C_COMPILER=icc -DCMAKE_CXX_COMPILER=icpc -DGMX_MPI=on -DGMX_GPU=on -DGMX_FFT_LIBRARY=mkl 2>&1 | tee cmake.log
make -j28 2>&1 | tee make.log
make check 2>&1 | tee makecheck.log
make install 2>&1 | tee makeinstall.log
```

▼ makeEnv script

```
module load cmake/3.14.3
module load gcc/5.5.0
export GMX_MAXBACKUP=1000
source /home/mbeukman/intel/2018/parallel_studio_xe_2018.4.057/psxevars.sh
module load cuda/10.0
export PATH=$PATH:/home/mbeukman/otherBenches/gromacs/v1GPU/install/bin
```

- I'm using the following **run** script

```
source ../gromacs-2020.1/makeEnv.sh

procs=${1:-4}
args=${2:""}
filename=run-$procs-${args/ /-}".log"

mpirun -np $procs gmx_mpi mdrun -s lignocellulose-rf.tpr  -gpu_id 01  -nsteps 400 $args  2>&1 | tee logs/$filename
```

V2 →

- Build script

```
wget http://ftp.gromacs.org/pub/gromacs/gromacs-2020.1.tar.gz
tar -xzf gromacs-2020.1.tar.gz

cd gromacs-2020.1

# need intel 18, because gromacs doesn't support 19
module load cmake/3.14.3
module load gcc/5.5.0
source /home/mbeukman/intel/2018/parallel_studio_xe_2018.4.057/psxevars.sh
module load cuda/10.0

mkdir build
cd build

cmake .. -DGMX_BUILD_OWN_FFTW=ON -DREGRESSIONTEST_DOWNLOAD=ON \
-DCMAKE_INSTALL_PREFIX=/home/mbeukman/otherBenches/gromacs/v2GPU/install \
-DCMAKE_C_COMPILER=icc -DCMAKE_CXX_COMPILER=icpc -DGMX_THREAD_MPI=on -DGMX_GPU=on -DGMX_FFT_LIBRARY=mkl 2>&1 | tee cmake.lo
make -j28 2>&1 | tee make.log
make check 2>&1 | tee makecheck.log
make install 2>&1 | tee makeinstall.log
```

- makeEnv script

```
module load cmake/3.14.3
module load gcc/5.5.0
source /home/mbeukman/intel/2018/parallel_studio_xe_2018.4.057/psxevars.sh
module load cuda/10.0
export PATH=$PATH:/home/mbeukman/otherBenches/gromacs/v2GPU/install/bin
```

- Run script

```
source ../makeEnv.sh

procs=${1:-4}
args=${2:-""}
filename=run-$procs-${args// /-/g}".log"

gmx mdrun -ntmpi $procs -s lignocellulose-rf.tpr  -gpu_id 01  -nsteps 400 $args  2>&1 | tee logs/$filename
```

V3 → On the V100

- Build script

```
module load cmake/3.14.3
module load gcc/5.5.0
source /home/mbeukman/intel/2018/parallel_studio_xe_2018.4.057/psxevars.sh
module load cuda/10.0

mkdir build
cd build

cmake ..  -DREGRESSIONTEST_DOWNLOAD=ON \
 -DGMX_SIMD=AVX_256 \
 -DCMAKE_INSTALL_PREFIX=`pwd`/../install \
 -DCMAKE_C_COMPILER=icc -DCMAKE_CXX_COMPILER=icpc \
 -DGMX_MPI=on -DGMX_GPU=on -DGMX_FFT_LIBRARY=mkl 2>&1 | tee cmake.log
make -j28 2>&1 | tee make.log
make check 2>&1 | tee makecheck.log
make install 2>&1 | tee makeinstall.log
```

▼ Run script

```
module load cmake/3.14.3
module load gcc/5.5.0
source /home/mbeukman/intel/2018/parallel_studio_xe_2018.4.057/psxevars.sh
module load cuda/10.0
source ../install/bin/GMXRC
which gmx_mpi

procs=${1:-4}
args=${2:-""}
filename=run-$procs-${args// /-/g}".log"

mpirun -np $procs gmx_mpi mdrun -s lignocellulose-rf.tpr  -gpu_id 0 -nsteps 400 $args  2>&1 | tee logs/$filename
```

V8 - K40 optim

▼ Build

```
cd gromacs-2020.2
module load cmake/3.14.3
module load gcc/5.5.0
source /home/mbeukman/intel/2018/parallel_studio_xe_2018.4.057/psxevars.sh
module load cuda/10.0

mkdir build
cd build
export CC=icc
export CXX=icpc
cmake ..  -DREGRESSIONTEST_DOWNLOAD=ON \
 -DGMX_SIMD=AVX2_256  -DGMX_BUILD_OWN_FFTW=ON \
 -DCMAKE_INSTALL_PREFIX=`pwd`/../install \
 -DCMAKE_C_COMPILER=icc -DCMAKE_CXX_COMPILER=icpc \
 -DGMX_MPI=on -DGMX_GPU=on -DGMX_FFT_LIBRARY=fftw3 2>&1 | tee cmake.log
make -j28 2>&1 | tee make.log
make check 2>&1 | tee makecheck.log
make install 2>&1 | tee makeinstall.log
```

▼ Run

```
module load cmake/3.14.3
module load gcc/5.5.0
source /home/mbeukman/intel/2018/parallel_studio_xe_2018.4.057/psxevars.sh
module load cuda/10.0
source ../install/bin/GMXRC
which gmx_mpi

procs=${1:-4}
args=${2:-""}
filename=run-$procs-${args// /-/g}".log"

mpirun -np $procs gmx_mpi mdrun -s lignocellulose-rf.tpr  -gpu_id 01 -nsteps 400 $args  2>&1 | tee logs/$filename
```

If not specified, I used nsteps=400

Results

Name	Description	benchmark	Nodes	MPI Ranks	OMP Threads Per Rank	NS/day
v7 avx2_256	-nstlist 100 -rdd 1	lignocellulose-rf.tpr	1	4	7	2.675

Name	Description	benchmark	Nodes	MPI Ranks	OMP Threads Per Rank	NS/day
v100 same as buildv3, with avx2_256	-nstlist 100	lignocellulose-rf.tpr	1	4	7	2.667
v100 gcc7.5 with avx2_256	-nstlist 100	lignocellulose-rf.tpr	1	4	7	2.654
Build V3→V100	-nstlist 100	lignocellulose-rf.tpr	1	4	7	2.652
Build V3→V100	-nstlist 150	lignocellulose-rf.tpr	1	4	7	2.603
Build V3→V100	-nstlist 50	lignocellulose-rf.tpr	1	4	7	2.583
Build V3→V100	-nstlist 100 -dlb yes	lignocellulose-rf.tpr	1	4	7	2.553
Build V3→V100		lignocellulose-rf.tpr	1	6	4	2.503
Build V3→V100		lignocellulose-rf.tpr	1	4	7	2.487
build v1		lignocellulose-rf.tpr	2	8	7	2.09
build v1		lignocellulose-rf.tpr	2	28	2	2.037
build v1		lignocellulose-rf.tpr	2	16	3	1.85
v2	nsteps 20000	lignocellulose-rf.tpr	1	4	7	1.804
v2	nsteps 4000	lignocellulose-rf.tpr	1	4	7	1.746
v2		lignocellulose-rf.tpr	1	6	4	1.36
build v2		lignocellulose-rf.tpr	1	4	7	1.343
build v1		lignocellulose-rf.tpr	1	8	3	1.316
build v1		lignocellulose-rf.tpr	1	6	4	1.299
build v1	-nstlist 100	lignocellulose-rf.tpr	1	6	4	1.299
buildV8		lignocellulose-rf.tpr	1	4	7	1.29
buildV8		lignocellulose-rf.tpr	1	6	4	1.29
build v1		lignocellulose-rf.tpr	1	4	7	1.288
build v1	-nstlist 100	lignocellulose-rf.tpr	1	12	2	1.27
build v1	-nstlist 200	lignocellulose-rf.tpr	1	12	2	1.185
build v1	-nstlist 50	lignocellulose-rf.tpr	1	12	2	1.111
build v1		lignocellulose-rf.tpr	1	12	2	0.688

Name	Description	benchmark	Nodes	MPI Ranks	OMP Threads Per Rank	NS/day
Build K40 CPU		lignocellulose-rf.tpr	1	4	7	0.606
Build K40 CPU		lignocellulose-rf.tpr	1	6	4	0.551
build v1	-nstlist 1	lignocellulose-rf.tpr	1	6	4	0.544
v7V100	mpirun -np $procs gmx_mpi mdrun -s lignocellulose-rf.tpr -noconfout -gpu_id 0 -nb gpu -bonded gpu -nstlist 400 -nsteps 2000 -resetstep 1000	lignocellulose-rf.tpr	1	4	7	4
v7V100	mpirun -np $procs gmx_mpi mdrun -s lignocellulose-rf.tpr -noconfout -gpu_id 0 -nb gpu -bonded gpu -nstlist 400 -nsteps 2000 -resetstep 1000 + all environment variables	lignocellulose-rf.tpr	1	1	28	4.244

Results

Name	Description	benchmark	Nodes	MPI Ranks	OMP Threads Per Rank	NS/day
v7V100	mpirun -np $procs gmx_mpi mdrun -s lignocellulose-rf.tpr -noconfout -gpu_id 0 -nb gpu -bonded gpu -nstlist 400 -nsteps 2000 -resetstep 1000 -notunepme	stmv.tpr	1	1	28	10.873
Untitled						

Most options for -rdd, -dd, and -dds don't change anything noticably

TODO

Stuff still to do

Read:

- ☑ ~~http://manual.gromacs.org/current/user-guide/environment-variables.html#performance-and-run-control → not much useful~~
- ☑ ~~http://manual.gromacs.org/current/user-guide/system-preparation.html#tips-and-tricks~~
- ☑ ~~http://manual.gromacs.org/current/user-guide/cmdline.html#commands-by-name~~
- ☑ ~~http://manual.gromacs.org/current/user-guide/mdrun-performance.html~~
  - ☑ ~~Use -DGMX_SIMD~~

```
The SIMD intrinsic code is compiled by the compiler. Technically, it is possible to compile different levels of acceleratio
```

  - ☑ ~~Consider building mdrun configured with GMX_SIMD=AVX2_256 instead of GMX_SIMD=AVX512 for better performance in GPU accelerated or highly parallel MPI runs. → NA I don't think the dgxs have the correct instruction set.~~
  - ☑ ~~Compile with GMX_SIMD=AVX2_256~~
  - □ **GMX_THREAD_MPI is good for single nodes → i.e. DGX**
  - □ **The number GPUs used has to match the number of of MPI processes (or thread-MPI threads) the simulation is started with.**

- □ While the automated CPU-GPU load balancing always attempts to find the optimal cut-off setting, it might not always be possible to balance CPU and GPU workload. This happens when the CPU threads finish calculating the bonded forces and PME faster than the GPU the non-bonded force calculation, even with the shortest possible cut-off. In such cases the CPU will wait for the GPU and this time will show up as Wait GPU local in the cycle and timing summary table at the end of the log file. → we're getting some of this

☑ ~~http://manual.gromacs.org/current/install-guide/index.html#gmx-gpu-support →~~

- ☑ ~~different GPU kernels~~

☑ ~~http://manual.gromacs.org/documentation/5.1/user-guide/index.html~~

☑ ~~http://manual.gromacs.org/documentation/5.1/onlinehelp/gmx-mdrun.html~~

□ `Generally, there is no advantage in using MKL with GROMACS, and FFTW is often faster.` BUT,

- □ With PME GPU offload support using CUDA, a GPU-based FFT library is required → Seems like fftw3 doesn't help when running on GPUs?

☑ ~~Compile FFTW yourself~~