Week 15 - 2020 05 16

What I did

This week I watched the gromacs guest lecture presentation. I used some of the knowledge in this to optimize our performance on the benchmarks a little bit. I didn't do much, and this was mostly because of me focusing on bert a lot this week.

We also got a new benchmark, the stmv one. I put the data of that run in the same page as benchmark lignocellulose, because meaningful names be damned.

Please refer to `gromacs summary`, `Benchmark Lignocellulose` and the nvidia presentation on box for more info.

TLDR:

- Gromacs 2020 has a lot of gpu centric optimizations, mostly catered for multi gpu, single node (like the dgx!)
    - Have to enable them explicitly, since they are new and not fully user tested (look at the summary/presentation for more)
        - `export GMX_GPU_DD_COMMS=true`
            - This communicates directly between gpus (using cuda direct and nvlink and such) for the halo exchanges twixt PP tasks
        - `export GMX_GPU_PME_PP_COMMS=true`
            - GPU coms for pp ↔ pme tasks
        - `export GMX_FORCE_UPDATE_DEFAULT_GPU=true`

        - Puts all of the updating constraints (i.e. set new particles positions on the gpu rather than cpu)
        - can also use `-update gpu` option to mdrun (equivalent)
    - `-nb gpu -bonded gpu -pme gpu`
        - This tells mdrun to put all of the force calculations on the gpu
- They basically moved the entire pipeline to the gpu, which means that the cpu almost doesn't do anything.
- They use nvlink to communicate directly 'twixt the gpus, bypassing the cpu.
- On multi node, we need to use a patch (https://gitlab.com/gromacs/gromacs/-/tree/cuda-aware-mpi) and compile with a cuda aware mpi, so that some of the communication benefits are transferred to multi node.

Also, since the cellulose benchmark doesn't use pme calculations, it doesn't benefit so much from the new gpu improvements.

In general, with pme (like stmv), use 1/4 of your resources for pme ranks, and the rest for pp ranks (i.e. on dgx use 1 v100 for pme, and 3 for pp)

What I will do

- Investigate the above options more
- experiment with them
- Look at the cellulose benchmark and try to improve its performance more.

Problems

Best current results

Normal builds, ie intel 2018, mpi

```
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
tar -xzf gromacs-2020.2.tar.gz
cd gromacs-2020.2
mkdir build
cd build

cmake ..  -DREGRESSIONTEST_DOWNLOAD=ON \
 -DGMX_SIMD=AVX2_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
```

Results

Aa Name	≡ Description	# MPI Ranks	# NS/day	# Nodes	# OMP Threads Per Rank	≡ benchmark
build v1		4	1.288	1	7	lignocellulose-rf.tpr
build v1		6	1.299	1	4	lignocellulose-rf.tpr
build v1	-nstlist 100	6	1.299	1	4	lignocellulose-rf.tpr
build v1	-nstlist 1	6	0.544	1	4	lignocellulose-rf.tpr
build v1		8	1.316	1	3	lignocellulose-rf.tpr
build v1	-nstlist 50	12	1.111	1	2	lignocellulose-rf.tpr
build v1		12	0.688	1	2	lignocellulose-rf.tpr
build v1	-nstlist 100	12	1.27	1	2	lignocellulose-rf.tpr
build v2		4	1.343	1	7	lignocellulose-rf.tpr

Aa Name	≡ Description	# MPI Ranks	# NS/day	# Nodes	# OMP Threads Per Rank	≡ benchmark
build v1	-nstlist 200	12	1.185	1	2	lignocellulose-rf.tpr
v2		6	1.36	1	4	lignocellulose-rf.tpr
v2	nsteps 4000	4	1.746	1	7	lignocellulose-rf.tpr
v2	nsteps 20000	4	1.804	1	7	lignocellulose-rf.tpr
Build V3->V100		4	2.487	1	7	lignocellulose-rf.tpr
Build V3->V100		6	2.503	1	4	lignocellulose-rf.tpr
Build V3->V100	-nstlist 50	4	2.583	1	7	lignocellulose-rf.tpr
Build V3->V100	-nstlist 150	4	2.603	1	7	lignocellulose-rf.tpr
Build V3->V100	-nstlist 100	4	2.652	1	7	lignocellulose-rf.tpr
Build V3->V100	-nstlist 100 -dlb yes	4	2.553	1	7	lignocellulose-rf.tpr
build v1		8	2.09	2	7	lignocellulose-rf.tpr
build v1		16	1.85	2	3	lignocellulose-rf.tpr
build v1		28	2.037	2	2	lignocellulose-rf.tpr
Build K40 CPU		4	0.606	1	7	lignocellulose-rf.tpr
Build K40 CPU		6	0.551	1	4	lignocellulose-rf.tpr
buildV8		4	1.29	1	7	lignocellulose-rf.tpr
buildV8		6	1.29	1	4	lignocellulose-rf.tpr

Aa Name	≡ Description	# MPI Ranks	# NS/day	# Nodes	# OMP Threads Per Rank	≡ benchmark
v100 gcc7.5 with avx2_256	-nstlist 100	4	2.654	1	7	lignocellulose-rf.tpr
v100 same as buildv3, with avx2_256	-nstlist 100	4	2.667	1	7	lignocellulose-rf.tpr
v7 avx2_256	-nstlist 100 -rdd 1	4	2.675	1	7	lignocellulose-rf.tpr
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	4	4	1	7	lignocellulose-rf.tpr
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	1	4.244	1	28	lignocellulose-rf.tpr

Results

Aa Name	≡ Description	# MPI Ranks	# NS/day	# Nodes	# OMP Threads Per Rank	≡ benchmark

Name	Description	MPI Ranks	NS/day	Nodes	OMP Threads Per Rank	benchmark
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	1	10.873	1	28	stmv.tpr
Untitled						