GROMACS Benchmarking in GPU V100 (including PLUMED)

HPC IIT Delhi:Performance GROMACS19 *V100=skylake

```
#PBS -l select=1:ncpus=$ntomp:ngpus=1:centos=skylake
Mpirun -np 1  gmx_mpi mdrun -ntomp $ntomp -nsteps 100000 -v -nb gpu -pme gpu -bonded gpu -gcom 100 -deffnm
nvt5
```

-nsteps: overrides number of steps in .mdp file. Used for benchmarking/testing runs.

-nb gpu
-pme gpu
-bonded gpu
To offload the corresponding calculation in the GPU.

-gcom 100
To rduce the communication frequency, to get better performance.

np > 1 is not supported for PME GPU Offloading

*Kutzner et al. used time step fo 2 fs for GROMACS benchmarking, whereas we have used 1 fs.

Kutzner et al. *J. Comput. Chem.* **2019,** 40**,** 2418-2431.

HPC IIT Delhi:Performance GROMACS19 *V100=skylake

System: 3CLpro ~ 52 K atoms

np	Node [core=20]	gpu*	$ntomp [5P X core]##	Performance (ns/day)	(GPU+CPU/min)/ (Rs/day)
1	4	1 (V100)	4 (20 P)	77	70P/ 1008 Rs
1	**8**	**1 (V100)**	**8 (40 P)**	**89**	**90P/ 1290 Rs**
1	12	1 (V100)	12 (60 P)	93	110P/ 1584 Rs
1	**20**	**1 (V100)**	**20 (100 P)**	**95**	**150P/ 2160 Rs**

As in each GPU node 20 processors in CPU

##
EXPORT OMP_NUM_THREADS=$ntomp

GPU: 50 p/min
CPU: 5p/core min
1 Day= 1440 min

```
#PBS -l select=1:ncpus=$ntomp:ngpus=1:centos=skylake
Mpirun -np 1  gmx_mpi mdrun -ntomp $ntomp -nsteps 100000 -v -nb gpu -pme gpu -bonded gpu -gcom 100 -deffnm nvt5
```

HPC IIT Delhi:Performance GROMACS19 with PLUMED*V100=skylake

Plumed.dat:
#Add keyword in starting of plumed file to print time consumed in individual CV calculation
DEBUG DETAILED_TIMERS

In PBS script
#Add keyword in the PBS submission script. The PLUMED_OMP_THREADS value will be same as OMP_NUM_THREAS value
PLUMED_OMP_THREADS=

```
#PBS -l select=1:ncpus=$ntomp:ngpus=1:centos=skylake
Mpirun -np 1  gmx_mpi mdrun -ntomp $ntomp -nsteps 100000 -v -nb gpu -pme gpu -bonded gpu -gcom 100 -deffnm
nvt5 -plumed plumed.dat
```

GROMACS19 with PLUMED: COORDINATION NUMBER CV Optimization

Plumed.dat:

#Add keyword in starting of plumed file to print time consumed in individual CV calculation

DEBUG DETAILED_TIMERS

#Coordination Number calculation consumes a lot of time, if calculate for solvent like water. In such cases, to reduce calculation time use neighbour list.

LIGsolv: COORDINATION GROUPA=Lig_heavy GROUPB=OW R_0=0.4 NLIST NL_STRIDE=100 NL_CUTOFF=1.0

#To tunne the value of NL_STRIDE run a setup with printing in COLVAR file

LIGsolv1: COORDINATION GROUPA=Lig_heavy GROUPB=OW R_0=0.4
LIGsolv2: COORDINATION GROUPA=Lig_heavy GROUPB=OW R_0=0.4 NLIST NL_STRIDE=10 NL_CUTOFF=1.0
LIGsolv3: COORDINATION GROUPA=Lig_heavy GROUPB=OW R_0=0.4 NLIST NL_STRIDE=50 NL_CUTOFF=1.0
LIGsolv4: COORDINATION GROUPA=Lig_heavy GROUPB=OW R_0=0.4 NLIST NL_STRIDE=100 NL_CUTOFF=1.0

#Plot and use the NL_STRIDE value giving performance similar to without NLIST case.

https://www.plumed.org/doc-v2.5/user-doc/html/performance-optimization.html

GROMACS19 with PLUMED Benchmarking with COORDINATION NUMBER CV

np	Node [core=20]	gpu*	$ntomp [5P X core]##	PLUMED NLIST	Performance (ns/day)	(GPU+CPU/min)/ (Rs/day)
1	20	1 (V100)	20 (20 P)	Without NLIST	7.5	**150P/ 2160 Rs**
1	20	1 (V100)	20 (20 P)	NLIST STRIDE 10	18	**150P/ 2160 Rs**
1	20	1 (V100)	20 (20 P)	NLIST STRIDE 50	29	**150P/ 2160 Rs**
1	20	1 (V100)	20 (20 P)	NLIST STRIDE 100	35.5	**150P/ 2160 Rs**
1	12	1 (V100)	12 (60 P)	NLIST STRIDE 100	35	110P/ 1584 Rs
1	**8**	**1(V100)**	**8 (40 P)**	NLIST STRIDE 100	**34.5**	**90P/ 1290 Rs**

OMP_NUM_THREADS = PLUMED_NUM_THREADS

System: CBL:AVI
~1 lakh atoms