GROMACS - Bug #2980

taskassignment fails with unit tests when GPUs and custom number of ranks is used

06/17/2019 05:28 PM - Szilárd Páll

Status:	New		
Priority:	Normal		
Assignee:			
Category:	mdrun		
Target version:			
Affected version - extra info:	master	**Difficulty:**	uncategorized
Affected version:	2019.3		

Description

```
$ bin/mdrun-test -ntmpi 3
[...]

Opened /home/pszilard/projects/gromacs/gromacs-master/build_gcc73_cuda92/src/programs/mdrun/tests/
Testing/Temporary/PmeTest_ReproducesEnergies_spc-and-methanol_PmeOnCpuTune.edr as single precision
 energy file
Last energy frame read 20 time    0.020         Reading file /home/pszilard/projects/gromacs/groma
cs-master/build_gcc73_cuda92/src/programs/mdrun/tests/Testing/Temporary/PmeTest_ReproducesEnergies
.tpr, VERSION 2020-dev-20190606-ec71536 (single precision)
Can not increase nstlist because an NVE ensemble is used
Using 3 MPI threads
Using 1 OpenMP thread per tMPI thread

-------------------------------------------------------
Program:     mdrun-test, version 2020-dev-20190606-ec71536
Source file: src/gromacs/taskassignment/taskassignment.cpp (line 256)
Function:    std::vector<std::vector<gmx::GpuTaskMapping> >::value_type gmx::runTaskAssignment(con
st std::vector<int>&, const std::vector<int>&, const gmx_hw_info_t&, const gmx::MDLogger&, const t
_commrec*, const gmx_multisim_t*, const gmx::PhysicalNodeCommunicator&, const std::vector<gmx::Gpu
Task>&, bool, PmeRunMode)
MPI rank:    0 (out of 3)

Inconsistency in user input:
There were 3 GPU tasks found on node racoon, but 2 GPUs were available. If the
GPUs are equivalent, then it is usually best to have a number of tasks that is
a multiple of the number of GPUs. You should reconsider your GPU task
assignment, number of ranks, or your use of the -nb, -pme, and -npme options,
perhaps after measuring the performance you can get.

For more information and tips for troubleshooting, please check the GROMACS
website at http://www.gromacs.org/Documentation/Errors
```

The above can't run without passing -gputasks; luckily the GMX_GPUTASKS environment variable can be used with unit tests too (which don't have the command line option), but this would require passing -nb/-pme which can't be done:

```
$ GMX_GPUTASKS="001" bin/mdrun-test -ntmpi 3
[...]

This run will generate roughly 0 Mb of data

There were 3 notes
Reading file /home/pszilard/projects/gromacs/gromacs-master/build_gcc73_cuda92/src/programs/mdrun/
tests/Testing/Temporary/PmeTest_ReproducesEnergies.tpr, VERSION 2020-dev-20190606-ec71536 (single
precision)

-------------------------------------------------------
Program:     mdrun-test, version 2020-dev-20190606-ec71536
Source file: src/gromacs/taskassignment/decidegpuusage.cpp (line 132)
```

```
Function:    bool gmx::decideWhetherToUseGpusForNonbondedWithThreadMpi(gmx::TaskTarget, const std:
:vector<int>&, const std::vector<int>&, gmx::EmulateGpuNonbonded, bool, bool, bool, int)

Inconsistency in user input:
When you use mdrun -gputasks, -nb and -ntmpi must be set to non-default
values, so that the device IDs can be interpreted correctly. If you simply
want to restrict which GPUs are used, then it is better to use mdrun -gpu_id.
Otherwise, setting the CUDA_VISIBLE_DEVICES environment variable in your bash
profile or job script may be more convenient.

For more information and tips for troubleshooting, please check the GROMACS
website at http://www.gromacs.org/Documentation/Errors
-------------------------------------------------------
```