Gromacs - Bug #1633
mdrun -nsteps -1 reports silly numbers
10/30/2014 11:50 PM - Mark Abraham

**Status:** Closed  
**Priority:** Low  
**Assignee:** Szilárd Páll  
**Category:** mdrun  
**Target version:** 5.0.4  
**Affected version:** 5.0.2

**Description**
The execution of mdrun is OK, but the reporting on stdout (and similarly in the log file) has:

```plaintext
GROMACS:      GROMACS_Cray-XC30-ARCHER_cname_Cray-XC30-ARCHER, VERSION 5.0.2
Executable:   -XC30-ARCHER_cname_Cray-XC30-ARCHER.exe
Library dir:  romacs/top
Command line: GROMACS_Cray-XC30-ARCHER_cname_Cray-XC30-ARCHER -deffnm bench -noconfout -nsteps -1 -maxh 0.04 -resetstep 1000 -gcom 100
```

Number of hardware threads detected (48) does not match the number reported by OpenMP (1).
Consider setting the launch configuration manually!
Reading file bench.tpr, VERSION 5.0.3-dev-20141020-bf6deb3 (single precision)
Changing nstlist from 10 to 20, rlist from 1 to 1.028

The number of OpenMP threads was set by environment variable OMP_NUM_THREADS to 2
Overriding nsteps with value passed on the command line: -1 steps, -0.003 ps

This .tpr is using a 2.5fs time step, so there are two problems with the final string shown above.

I have no idea what Cray/EPCC are doing to produce 48 and 1 in the mismatch reported higher up.

**Related issues:**
- Related to Gromacs - Feature #1122: Allow to force pinning
  - Blocked, need info
  - 01/17/2013

**Associated revisions**
- Revision ac6556c4 - 12/10/2014 08:49 PM - Szilárd Páll
  - Fix nstep command line override print
    - The commit addresses two issues:
      - printing negative simulation length with "-nsteps 1"
      - eliminates rounding when converting a non-integer time-step value from fs to ps units.

11/25/2015
I have no idea what Cray/EPCC are doing to produce 48 and 1 in the mismatch reported higher up.

This is not necessarily something Cray or EPCC misconfigured, it can also be a sign of incorrect launch config.

I think this commonly seem issue is related to OpenMP initialization (and likely pinning) happening outside of mdrun. Did you set the threads per rank/task flag for the job scheduler? While not a proper fix, aprun -cc none will likely work around the warning.

Target version changed from 5.0.3 to 5.0.4

While looking at node sharing setups I managed to reproduce this issue by simply using taskset on the mdrun process (and telling mdrun to pin). E.g.

```
$ taskset 0x1 $gmx mdrun -ntmpi 1 -ntomp 2
```

There is room for improvement, I'd say we should:

- improve the message by including a hint on what can be causing this;
- making the message upon encountering non-default affinity with "-pin auto" much more prominent to emphasize that incorrect affinity settings can cause severe performance loss and that the correct way to run mdrun with external affinities is to explicitly set "-pin off";
- revisiting my previously not very successful attempt to allow affinity overriding (see #1122).
#4 - 12/09/2014 10:36 PM - Szilárd Páll
- Related to Feature #1122: Allow to force pinning added

#5 - 12/10/2014 05:48 PM - Szilárd Páll
- Status changed from New to In Progress
- Assignee set to Szilárd Páll

#6 - 12/10/2014 05:57 PM - Gerrit Code Review Bot
Gerrit received a related patchset '1' for Issue #1633.
Uploader: Szilárd Páll (pall.szilard@gmail.com)
Change-id: f1aac7e0f4e8e37f3e9777fa4eaa79744f3ccd65
Gerrit URL: https://gerrit.gromacs.org/4292

#7 - 12/15/2014 09:45 PM - Mark Abraham
- Status changed from In Progress to Closed