GROMACS - Bug #2131
mdrun hangs upon "-nsteps " or "-maxh" trigger with more than 20 MPI processes

03/06/2017 02:47 PM - Jan-Philipp Machtens

Status: Closed
Priority: Normal
Assignee: 
Category: mdrun
Target version: 2016.3
Affected version - extra info: GROMACS 2016.2
Affected version: 2016.2
Difficulty: uncategorized

Description
Dear all,
In standard MD simulations with 20 or more MPI processes in total, my mdrun hangs (GROMACS 2016.2), when either "-nsteps " or "-maxh " should trigger mdrun termination.
I extensively tested on CPU-only nodes (2x E5-2680 v3 each)
(1) mdrun using thread MPI on a single node
(2) mdrun compiled up-to-date ParaStation MPI across 1,2,4, or 5 nodes

Summary:
Mdrun did not terminate upon a -nsteps/-maxh trigger, whenever the total number of MPI threads or MPI processes across all nodes was equal/larger than 20, irrespective of the number of OpenMP threads per MPI/process.
I tested GROMACS 5.1.x versions, GROMACS 2016, and GROMACS 2016.1, and it appears that this issue is specific to GROMACS 2016.2.

When GROMACS hangs, the output looks like this:

$-----------------------------------------------------$  
GROMACS: gmx mdrun, version 2016.2
Executable: /home/XXX/XXX/bin/gromacs-2016.2-threadMPI/bin/gmx
Data prefix: /home/XXX/XXX/bin/gromacs-2016.2-threadMPI
Working dir: /work/XXX/XXX/test-norestraint
Command line:
 gmx mdrun  -nsteps 300  -ntomp 2
Running on 1 node with total 24 cores, 48 logical cores 
Hardware detected:
 CPU info:
  Vendor: Intel
  Brand: Intel(R) Xeon(R) CPU E5-2680 v3 @ 2.50GHz
  SIMD instructions most likely to fit this hardware: AVX2_256
  SIMD instructions selected at GROMACS compile time: AVX2_256
  Hardware topology: Basic
Reading file topol.tpr, VERSION 5.1.4 (single precision)
Note: file tpx version 103, software tpx version 110
Overriding nsteps with value passed on the command line: 300 steps, 1.2 ps
Will use 16 particle-particle and 8 PME only ranks
This is a guess, check the performance at the end of the log file
Using 24 MPI threads
Using 2 OpenMP threads per tMPI thread
starting mdrun 'Protein'
300 steps,  1.2 ps.
step 40 Turning on dynamic load balancing, because the performance loss due to load imbalance is 1 3.0 %.
Writing final coordinates.
$-----------------------------------------------------$

Does anyone know a solution?
Many thanks in advance!!!

Jan-Philipp  ==========================================================
Dr. Jan-Philipp Machtens
Computational Neurophysiology Group
Institute of Complex Systems - Zelluläre Biophysik (ICS-4)
Forschungszentrum Jülich, Germany  ==========================================================

<table>
<thead>
<tr>
<th>Related issues:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Related to GROMACS - Task #2134: assess whether Jenkins is testing multi-rank...</td>
<td>Closed</td>
</tr>
<tr>
<td>Related to GROMACS - Task #1781: re-design benchmarking functionality</td>
<td>Accepted</td>
</tr>
<tr>
<td>Related to GROMACS - Bug #2041: mdrun -resetstep can finish too early</td>
<td>Closed</td>
</tr>
</tbody>
</table>

Associated revisions

Revision 66ec44e6 - 03/09/2017 04:08 PM - Szilárd Páll
Fix mdrun hanging upon exit with sep PME ranks

Commit 1d2d95e introduced a check and early return to skip printing perf stats when no valid wallcycle data was collected (due to missed reset). However, as the validity of wallcycle data does not get checked/recorded on separate PME ranks, mdrun deadlocks before exit in collective comm that PME ranks never enter.

This change fixes the hang by refactoring the printing code to use a boolean rather than an early return. This means the normal code path is unaffected in all cases (only the simulation master can ever write reports), and the case where it is invalid to write a report (premature termination) works correctly because all ranks communicate the data for the report that is never written (and efficiency is not of concern in this case).

Fixes #2131
Change-Id: If8b0813444d0b00a1a9a4a21d30fc8655c52752a

History

#1 - 03/06/2017 02:48 PM - Jan-Philipp Machtens
The title should be "mdrun hangs upon "-nsteps " or "-maxh" trigger with 20 or more MPI processes"

#2 - 03/07/2017 03:56 AM - Gerrit Code Review Bot
Gerrit received a related patchset '1' for Issue #2131.
Uploader: Szilárd Páll (pall.szilard@gmail.com)
Change-Id: gromacs~release-2016~If8b0813444d0b00a1a9a4a21d30fc8655c52752a
Gerrit URL: https://gerrit.gromacs.org/6499

#3 - 03/07/2017 03:58 AM - Szilárd Páll
- Status changed from New to Accepted

Confirmed. Thanks for the report!

Unfortunately, 2016.2 has a bug that gets triggered whenever PME ranks are used. A fix is being worked on, until then use -npme 0 if the performance loss is acceptable or revert to 2016.1 if you need to do parallel runs.

#4 - 03/07/2017 03:58 AM - Szilárd Páll
- Status changed from Accepted to In Progress

#5 - 03/07/2017 02:34 PM - Mark Abraham
- Related to Task #2134: assess whether Jenkins is testing multi-rank runs appropriately added

#6 - 03/07/2017 02:56 PM - Mark Abraham
- Related to Task #1781: re-design benchmarking functionality added

#7 - 03/07/2017 04:48 PM - Mark Abraham
- Description updated
#8 - 03/07/2017 04:54 PM - Szilárd Páll
- Target version set to 2016.3

#9 - 03/08/2017 04:19 PM - Mark Abraham
- Related to Bug #2041: mdrun -resetstep can finish too early added

#10 - 03/09/2017 04:06 PM - Mark Abraham
- Status changed from In Progress to Resolved

#11 - 03/10/2017 09:41 AM - Szilárd Páll
Applied in changeset 66ec44e6ffaf2a72b17a01aa16cd451bf7968217.

#12 - 03/13/2017 11:06 AM - Mark Abraham
- Status changed from Resolved to Closed