GROMACS - Bug #2830

DLB issue with 2D/3D domain decomposition

01/13/2019 02:07 AM - Roland Schulz

<table>
<thead>
<tr>
<th>Status</th>
<th>Closed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority</td>
<td>Normal</td>
</tr>
<tr>
<td>Assignee</td>
<td>Berk Hess</td>
</tr>
<tr>
<td>Category</td>
<td>mdrun</td>
</tr>
<tr>
<td>Target version</td>
<td>2019.1</td>
</tr>
<tr>
<td>Affected version - extra info:</td>
<td>2019</td>
</tr>
<tr>
<td>Affected version:</td>
<td>2019</td>
</tr>
<tr>
<td>Difficulty</td>
<td>uncategorized</td>
</tr>
</tbody>
</table>

Description

Using the lignocellulose_rf benchmark (version from prace, md5sum 592f8fbcc77e7df2e221d6068b3c966b6b) with 2560 ranks the 2019 version fails with:
step 200: The domain decomposition grid has shifted too much in the Z-direction around cell 20 5 1.

With GCC 7.1 or ICC 18u2 on SKL or KNL. Also happens with GMX_DLB_BASED_ON_FLOPS=1 or -dds .75. Also whether using -dd 128 20 1 or -dd 40 8 8 (default) doesn't matter. It is fine with fewer ranks or with 2018.3.

```
CC=gcc CXX=g++ cmake .. -DGMX_MPI=on -DGMX_SIMD=AVX_512_KNL -DGMX_HWLOC=no ibrun ~/gromacs/gcc7.3/bin/gmx_mpi mdrun -s lignocellulose-rf.tpr -nsteps 3000 -noconfout
```

Associated revisions

Revision ca3e8f89 - 01/23/2019 10:32 AM - Berk Hess
Fix error with 2D/3D DLB

With 2D or 3D dynamic domain decomposition with dynamic load balancing, mdrun would exit with a fatal error when a cell size was limited.
This bug was introduced in commit 49367d45.

Fixes #2830

Change-Id: If36fcc2ddbb45c0855c78a2767b1d8562584b76f

History

#1 - 01/13/2019 02:13 AM - Roland Schulz
- Description updated

#2 - 01/22/2019 12:30 AM - Roland Schulz
@Berk: Do you have a time to look at this and/or a suggestion of how to debug this?

#3 - 01/22/2019 12:14 PM - Berk Hess
I assume this uses update groups. I haven't changed much between 2018 and 2019 in the DLB code. Looking at the git log I actually don't see any functional change at all. So this might be an old bug which gets triggered more easily with update groups, either because this enables more out of sync running, or because this, I think, changes the DLB limits.
You can at least check if running with GMX_NO_UPDATEGROUPS makes the issue go away.

#4 - 01/23/2019 06:51 AM - Roland Schulz
It isn't related to update groups. The input doesn't use vaites or constraints.
The commit which introduced the issue is commit: 49367d4 (previous commit is fine and it shows the error). Reviewing the commit I don't see the issue. Any idea?

#5 - 01/23/2019 09:47 AM - Berk Hess
I looked at the commit twice, but couldn't see any issue.
But now I managed to reproduce this with 400 waters on 12 ranks.
#6 - 01/23/2019 10:36 AM - Gerrit Code Review Bot
Gerrit received a related patchset ‘1’ for Issue #2830.
Uploader: Berk Hess (hess@kth.se)
Change-Id: gromacs~release-2019~If36fcc2ddbb45c0855c78a2767b1d8562584b76f
Gerrit URL: https://gerrit.gromacs.org/9022

#7 - 01/23/2019 10:36 AM - Berk Hess
- Subject changed from DLB issue with 2560 ranks to DLB issue with 2D/3D domain decomposition
- Category set to mdrun
- Status changed from New to Fix uploaded
- Assignee set to Berk Hess
- Target version set to 2019.1

#8 - 01/23/2019 11:15 AM - Roland Schulz
Can confirm this fixes it.

#9 - 01/23/2019 12:15 PM - Berk Hess
- Status changed from Fix uploaded to Resolved

Applied in changeset ca3e8f897aeef036cdd6119229f3d3f789aadce24845.

#10 - 01/23/2019 09:21 PM - Roland Schulz
- Status changed from Resolved to Closed

BTW: Unit tests for the DLB code would be very nice.

#11 - 01/24/2019 09:57 PM - Berk Hess
Indeed would unit tests be nice. I always thought it would be hard to test the strange conditions under which things can easily go wrong. But actually it is not so hard. Just use a 2D and 3D decomposition and put nearly all load in one or two, well chosen, cells.