

GROMACS - Bug #2845

critical box fluctuations when using GPUs

01/28/2019 12:31 PM - Ramon Guixa-Gonzalez

Status: Closed	
Priority: High	
Assignee:	
Category: mdrun	
Target version: 2019.4	
Affected version - extra info: 2019, 2016	Difficulty: uncategorized
Affected version: 2018	
Description	
<p>Systems become extremely unstable (i.e. very large box fluctuations) after 80-100 ns until they eventually deform (i.e. membrane abnormally enlarged, big water pores go through, and protein collapses). None of the following actions helps:</p> <ol style="list-style-type: none">1 - Increase/decrease certain parameters within the mdp file (e.g. tau-p).2 - Re-build and re-equilibrate the system.3 - Use of different GROMACS versions, namely 5.4, gromacs2016 and gromacs2018.4 - Use of different computers and different GPUs models.5 - Use of a different system (other membrane proteins of the same family with different ligands). <p>Sooner or later, same thing happens over and over again in every protein-membrane system. Intriguingly, systems never crash (i.e. simulations keep on running despite of the observed artifacts). Thus, the gromacs+CHARMM36+GPUs combination inevitably induces protein-membrane systems to collapse.</p> <p>The only workaround so far is running the system using only CPUs.</p>	
Related issues:	
Related to GROMACS - Bug #2867: abnormal box fluctuations on GPUs still there	Closed

Associated revisions

Revision 4576f802 - 02/01/2019 03:53 PM - Berk Hess

Fix incorrect LJ repulsion force switching on GPUs

When using a CUDA or OpenCL GPU, the coefficient for the second order term for the LJ repulsion in the force (not energy) switching function, called 'A' in the manual, had the wrong sign in both the force-only kernels and the force+energy kernels.
Note that the dispersion force switching was correct.

Fixes #2845

Change-Id: Ib5c250a1f370d4c0a4fd652bf6efa03e70e18748

Revision de27733a - 02/04/2019 02:26 PM - Berk Hess

Fix incorrect LJ repulsion force switching on GPUs

When using a CUDA or OpenCL GPU, the coefficient for the second order term for the LJ repulsion in the force (not energy) switching function, called 'A' in the manual, had the wrong sign in both the force-only kernels and the force+energy kernels.
Note that the dispersion force switching was correct.

Cherry-picked from 2019

Fixes #2845

Change-Id: Ib5c250a1f370d4c0a4fd652bf6efa03e70e18748

Revision c131d790 - 02/20/2019 08:59 AM - Berk Hess

Fix incorrect LJ repulsion force switching on GPUs

When using a CUDA or OpenCL GPU, the coefficient for the second order term for the LJ repulsion in the force (not energy) switching function, called 'A' in the manual, had the wrong sign in both the force-only kernels and the force+energy kernels.

Note that the dispersion force switching was correct.

Cherry picked from 2019

Fixes #2845

Change-Id: lb5c250a1f370d4c0a4fd652bf6efa03e70e18748

Revision 8fc3cc55 - 09/21/2019 12:01 PM - Berk Hess

Fix incorrect shift forces for CMAP

The shift force indices were inverted for the second and third atom in the CMAP term, leading to incorrect virial and pressure contributions when these atoms resided in different periodic images.

Fixes #2845 and #2867

Change-Id: l1946a1d375d6c62e4e6d23ee25b92b42a3d4a6f7

Revision 771d4c99 - 09/21/2019 12:37 PM - Berk Hess

Fix incorrect shift forces for CMAP

The shift force indices were inverted for the second and third atom in the CMAP term, leading to incorrect virial and pressure contributions when these atoms resided in different periodic images.

Fixes #2845 and #2867

Also prepares branch for another point release.

Change-Id: l1946a1d375d6c62e4e6d23ee25b92b42a3d4a6f7

History

#1 - 01/28/2019 02:38 PM - Mark Abraham

Thanks for the report. I've started GPU and CPU runs to see what I reproduce.

What is the source of your membrane parameters and protocol? Is that known to be stable ie. not undergo phase transitions?

Can you please let us also have the full topology file and supporting files, so we can build our own .tpr if we need to?

#2 - 01/28/2019 04:22 PM - Berk Hess

How long did you run a CPU run? Also more than 100 ns?

Membranes are/were not stable at constant surface tension in some version of the CHARMM force field, but I forgot with one(s) that was/were.

#3 - 01/28/2019 05:40 PM - Berk Hess

Issue 2007 affected this configuration, but that was fixed in version 5.1.4, in a patch release of 2016 and is not present in the 2018 release. But maybe this was not fixed correctly.

Can you attach a md.log file of a 2018 GPU run that produced incorrect results?

#4 - 01/28/2019 10:45 PM - Ramon Guixa-Gonzalez

- File *simfiles.zip* added

Mark Abraham wrote:

Thanks for the report. I've started GPU and CPU runs to see what I reproduce.

What is the source of your membrane parameters and protocol? Is that known to be stable ie. not undergo phase transitions?

Can you please let us also have the full topology file and supporting files, so we can build our own .tpr if we need to?

Thanks for the response Mark:

- I build the system using the CHARMM-GUI web tool, so the source of my membrane parameters is the last stable CHARMM release. No phase transition expected at all (pure POPC at 310 K)
- Please find attached the extra files you would need to build your own tpr

#5 - 01/28/2019 10:49 PM - Ramon Guixa-Gonzalez

Berk Hess wrote:

How long did you run a CPU run? Also more than 100 ns?
Membranes are/were not stable at constant surface tension in some version of the CHARMM force field, but I forgot with one(s) that was/were.

Dear Berk,

- The CPU run was run for 500ns
- I have been using CHARMM with the same parameters for more than 6 years and never came across any problem like this. Since I use the CHARMM-GUI interface to generate my systems, the version I am using is always the last stable release (i.e. in this case, C36m).

#6 - 01/28/2019 10:50 PM - Ramon Guixa-Gonzalez

- File *logConcat* added

Berk Hess wrote:

Issue 2007 affected this configuration, but that was fixed in version 5.1.4, in a patch release of 2016 and is not present in the 2018 release. But maybe this was not fixed correctly.

Can you attach a md.log file of a 2018 GPU run that produced incorrect results?

Please find attached here the log file of the system I uploaded

#7 - 01/29/2019 03:16 PM - Szilárd Páll

Berk Hess wrote:

Issue 2007 affected this configuration, but that was fixed in version 5.1.4, in a patch release of 2016 and is not present in the 2018 release. But maybe this was not fixed correctly.

Unless the issue that caused [#2007](#) was not correctly identified, to me it seems that the fix on 5.1 was appropriate.

#8 - 01/31/2019 10:52 PM - Gerrit Code Review Bot

Gerrit received a related patchset '1' for Issue [#2845](#).
Uploader: Berk Hess (hess@kth.se)
Change-Id: gromacs~release-2019~1b5c250a1f370d4c0a4fd652bf6efa03e70e18748
Gerrit URL: <https://gerrit.gromacs.org/9066>

#9 - 01/31/2019 10:53 PM - Berk Hess

- Category set to *mdrun*
- Status changed from *New* to *Fix uploaded*
- Target version set to *2018.6*
- Affected version - extra info set to *2019, 2016*

I am quite sure the fix I uploaded fixes the issue, but please test.

#10 - 01/31/2019 10:56 PM - Berk Hess

I did actually test myself that this fix significantly reduces the difference in pressure between CPU and GPU at step 0.

#11 - 02/04/2019 02:30 PM - Berk Hess

- Status changed from *Fix uploaded* to *Resolved*

Applied in changeset [4576f802ac81b07a49c7c4ad21d3bf15831979a7](#).

#12 - 02/04/2019 02:30 PM - Gerrit Code Review Bot

Gerrit received a related patchset '1' for Issue [#2845](#).
Uploader: Paul Bauer (paul.bauer.q@gmail.com)
Change-Id: gromacs~release-2016~1b5c250a1f370d4c0a4fd652bf6efa03e70e18748
Gerrit URL: <https://gerrit.gromacs.org/9083>

#13 - 02/04/2019 02:53 PM - Gerrit Code Review Bot

Gerrit received a related patchset '1' for Issue [#2845](#).
Uploader: Paul Bauer (paul.bauer.q@gmail.com)
Change-Id: gromacs~release-2018~1b5c250a1f370d4c0a4fd652bf6efa03e70e18748
Gerrit URL: <https://gerrit.gromacs.org/9084>

#14 - 02/07/2019 11:45 AM - Berk Hess

Applied in changeset [de27733a387922033b116bf7024c595e5b1e99a8](#).

#15 - 02/20/2019 09:30 AM - Berk Hess

Applied in changeset [c131d790b48da10f5fcb9e4aabee0d730b76724](#).

#16 - 02/22/2019 10:46 AM - Ramon Guixa-Gonzalez

- File 2016_6.zip added

Ramon Guixa-Gonzalez wrote:

Systems become extremely unstable (i.e. very large box fluctuations) after 80-100 ns until they eventually deform (i.e. membrane abnormally enlarged, big water pores go through, and protein collapses). None of the following actions helps:

- 1 - Increase/decrease certain parameters within the mdp file (e.g. tau-p).
- 2 - Re-build and re-equilibrate the system.
- 3 - Use of different GROMACS versions, namely 5.4, gromacs2016 and gromacs2018.
- 4 - Use of different computers and different GPUs models.
- 5 - Use of a different system (other membrane proteins of the same family with different ligands).

Sooner or later, same thing happens over and over again in every protein-membrane system. Intriguingly, systems never crash (i.e. simulations keep on running despite of the observed artifacts). Thus, the gromacs+CHARMM36+GPUs combination inevitably induces protein-membrane systems to collapse.

The only workaround so far is running the system using only CPUs.

Berk Hess wrote:

Applied in changeset [c131d790b48da10f5fcb9e4aabee0d730b76724](#).

Berk Hess wrote:

I am quite sure the fix I uploaded fixes the issue, but please test.

Dear Berk,

I tested the 2016.6 release (<http://manual.gromacs.org/2016.6/ReleaseNotes/release-notes.html#fix-incorrect-lj-repulsion-force-switching-on-gpus>) and I am afraid this is not fixed...
It took more than 300 ns, but box fluctuations seem to be there again. Please check the files I have attached

Am I doing something wrong?

Thanks for the help again

R

#17 - 02/22/2019 11:15 AM - Paul Bauer

- Status changed from Resolved to Closed

#18 - 02/22/2019 01:18 PM - Berk Hess

- Related to Bug #2867: abnormal box fluctuations on GPUs still there added

#19 - 02/22/2019 01:23 PM - Berk Hess

Could you attach a box plot for a long cpu run, so we can see what you think is normal and abnormal?

#20 - 02/22/2019 03:43 PM - Ramon Guixa-Gonzalez

- File boxzCpu.jpeg added

Berk Hess wrote:

Could you attach a box plot for a long cpu run, so we can see what you think is normal and abnormal?

Sure, here you go. 1 microsecond of a CPU run

Berk Hess wrote:

Could you attach a box plot for a long cpu run, so we can see what you think is normal and abnormal?

Sure, here you go Berk. 1-microsecond cpu run

#21 - 02/22/2019 08:09 PM - Ramon Guixa-Gonzalez

- File boxProtein-free.png added

Ramon Guixa-Gonzalez wrote:

Berk Hess wrote:

Could you attach a box plot for a long cpu run, so we can see what you think is normal and abnormal?

Sure, here you go. 1 microsecond of a CPU run

Berk Hess wrote:

Could you attach a box plot for a long cpu run, so we can see what you think is normal and abnormal?

Sure, here you go Berk. 1-microsecond cpu run

By the way, one thing that is now puzzling me is the fact that if I build the same system but protein-free, and simulate it under the same conditions, there are no box fluctuations whatsoever even using GPUs...? Picture attached. Any hint?

#22 - 02/23/2019 11:15 PM - Berk Hess

Have you continued the 2016.6 GPU run? It could be that the system with protein is just on the limit of the stability with the force field. So maybe the 2016.6 GPU results are just "unlucky", but the system remains at the same box size for most of the time?

#23 - 02/24/2019 01:32 PM - Ramon Guixa-Gonzalez

- File 2016.6_400ns.jpeg added

- File 2016.6XYZ.jpeg added

Berk Hess wrote:

Have you continued the 2016.6 GPU run? It could be that the system with protein is just on the limit of the stability with the force field. So maybe the 2016.6 GPU results are just "unlucky", but the system remains at the same box size for most of the time?

I continued the 2016.6 GPU run and, as you can see in the pictures, the system is basically getting squeezed along the z dimension, so back to the beginning.

#24 - 02/24/2019 09:11 PM - Berk Hess

Are you 100% sure that you are running 2016.6 with the fix?

I checked that the fix corrected the energy and the pressure, so the fix actually fixes a bug that occurred with your run setup. I can not exclude that there is a second bug, but I have no clue where that could be. The only special code with force-switching on GPUs is the few lines where the fix is.

Could you attach the md.log file for the GPU 2016.6 run?

#25 - 02/25/2019 09:54 AM - Ramon Guixa-Gonzalez

- File run1Concat.log added

Berk Hess wrote:

Are you 100% sure that you are running 2016.6 with the fix?

I checked that the fix corrected the energy and the pressure, so the fix actually fixes a bug that occurred with your run setup. I can not exclude that there is a second bug, but I have not clue where that could be. The only special code with force-switching on GPUs is the few lines where the fix is.

Could you attach the md.log file for the GPU 2016.6 run?

Well, I am running the 2016.6 version recently released (<http://manual.gromacs.org/documentation/2016.6/download.html>). There, you state this is fixed right? (<http://manual.gromacs.org/documentation/2016.6/ReleaseNotes/index.html>).

Please find attached the md.log file for the GPU 2016.6 run.

#26 - 03/01/2019 02:56 AM - Ramon Guixa-Gonzalez

Ramon Guixa-Gonzalez wrote:

Berk Hess wrote:

Are you 100% sure that you are running 2016.6 with the fix?

I checked that the fix corrected the energy and the pressure, so the fix actually fixes a bug that occurred with your run setup. I can not exclude that there is a second bug, but I have not clue where that could be. The only special code with force-switching on GPUs is the few lines where the fix is.

Could you attach the md.log file for the GPU 2016.6 run?

Well, I am running the 2016.6 version recently released (<http://manual.gromacs.org/documentation/2016.6/download.html>). There, you state this is fixed right? (<http://manual.gromacs.org/documentation/2016.6/ReleaseNotes/index.html>).

Please find attached the md.log file for the GPU 2016.6 run.

Hi there, have you figured what the problem is here? I only have access to a GPU cluster, so since this issue came up I am basically stuck...

#27 - 03/06/2019 11:07 AM - Berk Hess

This is an issue with high priority, as the setting says. But I have little clue where to look. Normally I would test things myself. But since this happens after so long time, and not without the protein, that is impractical.

The only suggestion I have is to run with a plain cut-off instead. Using a plain cut-off of 1.11 nm gives the same potential and force at distances between 0 and 1 nm and only differs in the switching region. This should not significantly change the properties of the system. Whether this is stable or not should also give us hints about this issue.

#28 - 03/06/2019 11:08 AM - Berk Hess

PS I meant a plain cut-off with shifted potential, which is the default potential modifier with the Verlet cutoff scheme.

#29 - 03/18/2019 01:12 PM - Berk Hess

- Status changed from Closed to Feedback wanted

- Target version changed from 2018.6 to 2019.2

Do you already have results with a 1.11 nm cut-off without switch?

#30 - 03/18/2019 08:19 PM - Ramon Guixa-Gonzalez

Berk Hess wrote:

Do you already have results with a 1.11 nm cut-off without switch?

Sorry, I have not been able to launch this test yet.

Would you mind confirming the lines I need to have in my mdp in order to perform this run?

This is what I currently use:

cutoff-scheme = Verlet
coulombtype = pme
rcoulomb = 1.2

Should I just use this:

cutoff-scheme = Verlet
coulombtype = Cut-off
rcoulomb = 1.1

Would this be all?

Thanks

R

#31 - 03/18/2019 11:10 PM - Berk Hess

No, still use PME.

Change:

rcoulomb=1.11

rvdw=1.11

vdw-modifier=potential-shift (which is the default)

This should also run faster.

You seem to be running with the default PME fourier spacing. You could have used a coarser grid with rcoulomb=1.2. But I would leave it as is with the 1.11 nm cut-off.

#32 - 03/29/2019 12:42 PM - Ramon Guixa-Gonzalez

- File *plainCutoff.zip* added

Berk Hess wrote:

No, still use PME.

Change:

rcoulomb=1.11

rvdw=1.11

vdw-modifier=potential-shift (which is the default)

This should also run faster.

You seem to be running with the default PME fourier spacing. You could have used a coarser grid with rcoulomb=1.2. But I would leave it as is with the 1.11 nm cut-off.

Hi Berk,

I have results with a 1.11 nm cut-off without switch and the issue is still there I am afraid...
I have attached the files, please have a look.

Ramon

#33 - 04/01/2019 12:07 PM - Mark Abraham

- Status changed from *Feedback wanted* to *In Progress*

Berk's away this week, but I may get time to explore further

#34 - 04/09/2019 02:05 PM - Berk Hess

Could you attach the log file as well?

#35 - 04/10/2019 11:09 AM - Ramon Guixa-Gonzalez

- File *runLog* added

Berk Hess wrote:

Could you attach the log file as well?

Sure, here you go Berk

#36 - 04/15/2019 09:30 AM - Paul Bauer

- Target version changed from 2019.2 to 2019.3

@Berk and @Ramon, any progress with identifying what is happening there?
Bumped for now to next point release.

#37 - 04/15/2019 09:55 AM - Berk Hess

Still no clue. Now we only know that it has nothing to do with the force-switch functionality.

#38 - 04/15/2019 11:32 AM - Ramon Guixa-Gonzalez

Ramon Guixa-Gonzalez wrote:

Berk Hess wrote:

Could you attach the log file as well?

Sure, here you go Berk

Paul Bauer wrote:

@Berk and @Ramon, any progress with identifying what is happening there?
Bumped for now to next point release.

Berk Hess wrote:

Still no clue. Now we only know that it has nothing to do with the force-switch functionality.

Same thing. I don't know what else to check here...

#39 - 04/26/2019 07:54 AM - Berk Hess

I am (still) wondering as to why this happens only after 300 ns. Are there slow rearrangements in the system?
Can you restart the simulation from a checkpoint file or trajectory frame close to 300 ns to reproduce the issue much quicker and also check what happens with a CPU run? If so, we might be able to find what the issue is.

#40 - 04/29/2019 01:35 PM - Ramon Guixa-Gonzalez

Hi there, ok, I will do that. Will keep you posted

#41 - 05/09/2019 11:38 AM - jose flores-canales

I have seen similar simulation box deformations for POPC + protein simulations with Charmm36 FF. These deformations are observed using Gromacs 2018.6 in CPU version (I observe similar deformations for a POPC + CHOL + Protein MD runs in both CPU and GPU versions, while one repeat on CPU does not show deformations after 1000 ns with 15 restarts, two other CPU repeats show deformation as early as 220 ns. One GPU run shows deformations around 400 ns).

For the POPC + Protein system, in two CPU repeats I have noticed that these deformations (deformations begin near 600 ns or 1000 ns) with runs restarted every 24 hours (the maximum walltime available in one of the clusters I have access, approx. 18 restarts for 1000 ns). For a single GPU repeat of the same system with 96 hours max. walltime, I haven't seen these deformations after 1000 ns (5 restarts).

I don't know if this helps in any way, since sometimes I get lucky with some runs.

#42 - 05/20/2019 01:45 PM - Berk Hess

So it doesn't seem to be clear yet that there is an actual code issue here. It could be that the force field has some unwanted state that is close in free-energy to the intended state.

One question now is what the statistical certainty is of this happening in GPU and not is CPU, or even the other way around now?

As I said before, the most useful would be to have a checkpoint close before a fluctuation occurs.

#43 - 05/21/2019 12:30 PM - Ramon Guixa-Gonzalez

- File bug0.zip added

- File bug1.zip added

Berk Hess wrote:

So it doesn't seem to be clear yet that there is an actual code issue here. It could be that the force field has some unwanted state that is close in free-energy to the intended state.

One question now is what the statistical certainty is of this happening in GPU and not in CPU, or even the other way around now?

As I said before, the most useful would be to have a checkpoint close before a fluctuation occurs.

Hi Berk, well, in my experience, I only see these artifacts in GPU, but it is true that I could only test it in CPU twice.

As you suggested, I have run my system starting from a checkpoint close before major fluctuation occurs. I have attached two zips (bug0.zip and bug1.zip) with all files for this run. As shown in plot boxz.jpeg, fluctuations immediately start in this run. Plot boxz.jpeg shows what happens before the checkpoint (taken at around 178ns).

#44 - 05/21/2019 12:41 PM - Ramon Guixa-Gonzalez

- File newSys0.zip added

- File newSys1.zip added

In addition, I decided to run a different system (different protein and completely different membrane composition) to rule out that the test system I am using is somehow not ill-defined. As you see in the files I have attached (newSys0.zip and newSys1.zip), critical fluctuations also arise before 200 ns. I experience the same fluctuations in GPUs regardless of the system and in 100% of the runs then.

I hope the checkpoint simulation I ran above is useful for further testing, because I don't really know what else should I check at this point.

#45 - 05/23/2019 03:18 PM - jose flores-canales

- File box_dimension_pressure_YZ.png added

- File repro_bug.tar.gz added

I attach here a plot of the box dimensions X and Z in nanometers, jointly with the YZ pressure graph. The pressure in bar units was scaled to appear in the same plot (divided by 25). There seems to be a correlation between changes in the box dimensions and the pressure at YZ.

I also attach the files necessary to run again the simulation and the output files from the failed run using a GPU with Gromacs 2018.6 (including the files to plot the graphs). The system is a protein in a pure POPC bilayer. This system in particular had box distortions on two GPU repeats and it runs ok on CPU for 2 repeats of up to 1000 ns. It uses semiisotropic NPT conditions with Parrinello-Rahman $\tau_p = 5$, and Langevin thermostat.

#46 - 06/05/2019 11:41 AM - David van der Spoel

Isn't this just the result of having a too short τ_p (5 ps) with semiisotropic scaling? Since relaxation is system size dependent the "common knowledge" that a 5 ps time constant is long enough may not hold. Try increasing it to 20 ps.

#47 - 06/05/2019 02:43 PM - Berk Hess

I missed that you had uploaded checkpoint files two weeks ago. I have reproduced the issue and have been running many simulations to find leads to what is wrong, but didn't find any hint up till now.

Do you know if the issue seems related to running non-bonded interactions or PME on the GPU?

#48 - 06/07/2019 11:10 AM - Berk Hess

- File state_save0.cpt added

- File compare.pdf added

I ran a GPU simulation from the checkpoint you provided and started runs from a later checkpoint at time 179558.6 ps with $\text{box}_z = 9.3686$.

The plot shows that with GPU runs box_z fluctuates significantly and has spikes downward. But also, the CPU run, which runs much slower, shows a drop to 8.5. So this is likely not a GPU issue.

My hypothesis is still that the force field has a second free-energy minimum at state compressed in z and expanded along x/y.

#49 - 06/08/2019 01:43 PM - jose flores-canales

- File tests_system_gromacs.png added

Thank you for taking a look on this problem. I am currently running tests with larger barostat τ . I tested another system with Gromacs 5.1.4 on CPU and it resulted in similar box perturbations. I attach a graph with results of the same system with Gromacs 2018.6 on GPU or CPU platforms. I used the default auto settings for the pme and nb options in Gromacs 2018.6.

#50 - 06/10/2019 08:33 AM - Berk Hess

- Status changed from In Progress to Resolved

- Target version deleted (2019.3)

My CPU continuation runs has dips down to $\text{box_z}=8.2$ nm.

So our conclusion is that this is a force field issue (which only appears after long simulation times, so likely slow rearrangements are involved finally leading to rapid changes).

#51 - 06/11/2019 11:29 AM - Ramon Guixa-Gonzalez

David van der Spoel wrote:

Isn't this just the result of having a too short τ_P (5 ps) with semiisotropic scaling? Since relaxation is system size dependent the "common knowledge" that a 5 ps time constant is long enough may not hold. Try increasing it to 20 ps.

sorry, I was away, I'll give long τ_P s a try, thanks!

#52 - 06/11/2019 11:30 AM - Ramon Guixa-Gonzalez

Berk Hess wrote:

I missed that you had uploaded checkpoint files two weeks ago. I have reproduced the issue and have been running many simulations to find leads to what is wrong, but didn't find any hint up till now.

Do you know if the issue seems related to running non-bonded interactions or PME on the GPU?

Hey Berk, well, not a clue, really...

#53 - 06/11/2019 11:51 AM - Ramon Guixa-Gonzalez

Berk Hess wrote:

I ran a GPU simulation from the checkpoint you provided and started runs from a later checkpoint at time 179558.6 ps with $\text{box_z}=9.3686$. The plot shows that with GPU runs box_z fluctuates significantly and has spikes downward. But also, the CPU run, which runs much slower, shows a drop to 8.5. So this is likely not a GPU issue.

My hypothesis is still that the force field has a second free-energy minimum at state compressed in z and expanded along x/y.

Ok...

Berk Hess wrote:

My CPU continuation runs has dips down to $\text{box_z}=8.2$ nm.

So our conclusion is that this is a force field issue (which only appears after long simulation times, so likely slow rearrangements are involved finally leading to rapid changes).

Well, ok, this was also what I initially suspected. I will try to report it to the CHARMM force field community to see if they are aware. On the meanwhile, I will run more tests, namely:

- 1) one of the failing simulations on NAMD
- 2) one at longer τ_P (20)
- 3) one using an older release of the CHARMM force field

Thanks Berk

#54 - 06/21/2019 12:35 PM - Ramon Guixa-Gonzalez

- File *longTau.jpeg* added

David van der Spoel wrote:

Isn't this just the result of having a too short τ_P (5 ps) with semiisotropic scaling? Since relaxation is system size dependent the "common knowledge" that a 5 ps time constant is long enough may not hold. Try increasing it to 20 ps.

Hi David, attached the results of a system run under a long τ_P (20ps). As you see in the pic I have attached, the box remains somewhat stable up to 400 ns, then starts fluctuating slightly to eventually collapse after half a microsecond. Just wanted to report it so that you are aware.

#55 - 07/30/2019 03:06 PM - Alexander Vogel

I can confirm this bug with GROMACS 2019.3. I have several simulations of a GPCR+Membrane system that get ripped apart after 100-200ns. But already before that unnaturally large fluctuations are seen that manage to recover however. After some time the system violently extends in x-y dimension (which is the membrane plane) and collapses in the z-dimension. I have two simulations of identical coordinates that were started with different random seeds but otherwise identical parameters and both get ripped apart. Interestingly, both systems have very similar box dimensions after the "event" although the disrupted membrane structure is very different between the two. Since I didn't notice the error immediately and the simulations did not crash, they continued for several hundred ns (first simulation for 450ns, second simulation for 900 ns). There still seem to be some pretty large fluctuations during that time but no violent changes as before.

I'll start a third version of the system today. A colleague of mine reported similar issues but I didn't have time to look at it yet.

I setup the simulation with charmm-gui and run them on an AMD 1700X + Nvidia RTX2080Ti. If more info is needed please tell me what you need.

Alexander Vogel

#56 - 09/21/2019 11:36 AM - Berk Hess

- Status changed from Resolved to Fix uploaded

- Target version set to 2019.4

#57 - 09/22/2019 12:20 PM - Berk Hess

- Status changed from Fix uploaded to Resolved

Applied in changeset [8fc3cc55a0a0721fc9c336f142a33025d18b4d5f](#).

#58 - 09/24/2019 08:45 AM - Berk Hess

Applied in changeset [771d4c99c5aca3ac2e1b4892417898b6686d045a](#).

#59 - 09/24/2019 03:18 PM - Paul Bauer

- Status changed from Resolved to Closed

Files

bug.zip	34.1 MB	01/28/2019	Ramon Guixa-Gonzalez
simfiles.zip	1010 KB	01/28/2019	Ramon Guixa-Gonzalez
logConcat	2.72 MB	01/28/2019	Ramon Guixa-Gonzalez
2016_6.zip	43.4 MB	02/22/2019	Ramon Guixa-Gonzalez
boxzCpu.jpeg	863 KB	02/22/2019	Ramon Guixa-Gonzalez
boxProtein-free.png	111 KB	02/22/2019	Ramon Guixa-Gonzalez
2016.6_400ns.jpeg	497 KB	02/24/2019	Ramon Guixa-Gonzalez
2016.6XYZ.jpeg	554 KB	02/24/2019	Ramon Guixa-Gonzalez
run1Concat.log	2.26 MB	02/25/2019	Ramon Guixa-Gonzalez
plainCutoff.zip	44.2 MB	03/29/2019	Ramon Guixa-Gonzalez
runLog	3.24 MB	04/10/2019	Ramon Guixa-Gonzalez
bug0.zip	5.6 MB	05/21/2019	Ramon Guixa-Gonzalez
bug1.zip	45.9 MB	05/21/2019	Ramon Guixa-Gonzalez
newSys0.zip	8.32 MB	05/21/2019	Ramon Guixa-Gonzalez
newSys1.zip	41.9 MB	05/21/2019	Ramon Guixa-Gonzalez
box_dimension_pressure_YZ.png	54.9 KB	05/23/2019	jose flores-canales
repro_bug.tar.gz	11.5 MB	05/23/2019	jose flores-canales
state_save0.cpt	1.65 MB	06/07/2019	Berk Hess
compare.pdf	8.01 KB	06/07/2019	Berk Hess
tests_system_gromacs.png	1.04 MB	06/08/2019	jose flores-canales
longTau.jpeg	440 KB	06/21/2019	Ramon Guixa-Gonzalez