When Verlet cutoff scheme is used mdrun crushes mysteriously if the non-interacting dummy molecule is present in the system. The minimal top file is below. Mdrun crashes despite the fact, that dummy atoms do not interact with anything and should not affect the system in any way:

Steepest Descents:
Tolerance (Fmax) = 1.00000e+00
Number of steps = 5000
Step = 0, Dmax = 1.0e-03 nm, Epot = -nan, Fmax = 8.47202e+02, atom = 607
Segmentation fault (core dumped)

When there are only two atoms in the dummy molecule everything is fine. If the third atom (named T) is added mdrun crashes immediately.

If group cutoff scheme is used everything works in both cases with no problem.

```plaintext
#include "amber99.ff/forcefield.itp"

[ atomtypes ]
; Dummy atoms not interacting with anything!

; name bond_type mass charge ptype          sigma      epsilon
SW   12  0.00000  0.00000   A            0.0        0.0
DU   12  0.00000  0.00000   A            0.0        0.0

[ moleculetype ]
SW 3

[ atoms ]
; nr type resnr residue atom cgnr charge mass
1   SW  1   SW  1   0.0    12.0
2   DU  1   SW  1   0.0    12.0
3   DU  1   SW  T   3   0.0    12.0 ;Remove this atom and it will work!

[ bonds ]
1  2   6   0.0  10000

#include "amber99.ff/tip3p.itp"

[ system ]
water

[ molecules ]
; Compound    nmols
SOL    216
SW    1
```

Related issues:
- Related to GROMACS - Bug #1965: Crash of mdrun with strange error messages: Closed
- Related to GROMACS - Bug #2023: Segfault again with non-interacting atoms and...: Closed
**Associated revisions**

**Revision 3c6ba9c7 - 05/30/2016 07:31 PM - Berk Hess**

Add grompp check for unbound atoms

grompp now print a note for atoms that are not connected by a potential or constraint to any other atom in the same moleculetype, since this often means the user made a mistake.

Refs #1958.

Change-Id: iabb00563c76a9f7954f84d89d1c67d438f2c31ff

**Revision fcc7c4c4 - 05/31/2016 01:52 PM - Berk Hess**

Avoid numerical overflow with overlapping atoms

The verlet kernels did not allow overlapping atoms, even if they were not interacting (in contrast to the group kernels). Fixed by clamping the interaction distance so it can not become smaller than ~6e-4 in single and ~1e-18 in double, and when this number is later multiplied by zero parameters it will not influence forces. The clamping should never affect normal interactions; we would previously crash for distances that were this small.

On Haswell, RF and PME kernels get 3% and 1% slower, respectively. On CUDA, RF and PME kernels get 1% and 2% faster, respectively.

Fixes #1958.

Change-Id: i83b88f0e9ca34dc151a8b907f334a95a1a4301cc

**History**

**#1 - 05/13/2016 07:01 PM - Semen Yesylevskyy**

Such dummy particles are very useful in certain kind of simulations. The originate from complex production system where I fix DU atoms and let SW atoms to interact selectively with certain atom types in the system (a kind of wall invisible to certain atoms and visible to the others). T particles are used as anchors when such wall is moved using the com pulling.

It is possible to workaround this bug by moving T particles to separate moleculetype but such inconsistency between group and verlet schemes is not normal.

**#2 - 05/24/2016 01:30 AM - Gerrit Code Review Bot**

Gerrit received a related patchset ‘1’ for Issue #1958.

Uploader: Erik Lindahl (erik.lindahl@gmail.com)

Change-Id: i83b88f0e9ca34dc151a8b907f334a95a1a4301cc

Gerrit URL: https://gerrit.gromacs.org/5886

**#3 - 05/25/2016 10:11 AM - Berk Hess**

Note that the dummies can overlap between themselves, since they are excluded from each other. The issue here is that all dummies overlap with the last hydrogen of the last water molecule. I think that for your real setup where the dummies do something useful there will be no issue.

**#4 - 05/26/2016 09:43 AM - Semen Yesylevskyy**

In real setup such dummies form a semi-transparent barrier which is invisible for water but visible to other molecules. If overlapping with certain atom types while interacting normally with the others is now Ok then the bug is fixed.

**#5 - 05/26/2016 10:53 AM - Berk Hess**

The actual issue in your setup is that atoms DU and T are not connected in any way. This will not result in a meaningful simulation. If you connect these by a chemical bonds (e.g. bond type 5) or exclude them manually, the system will run fine.

But the issue of non-excluded atoms not being allowed to overlap is still there in general.

It is not clear from your answer is these atoms should be able to overlap with atoms from other molecules. That would be an issue, since you can only exclude interactions within a molecule.

**#6 - 05/26/2016 11:47 AM - Semen Yesylevskyy**

Ok, let me be more specific.

I have a set of atomtypes A1,A2... An which interact with atom type SW normally. And another set B1,B2...Bn which does not interact with SW at all. Types A and B could be in the same molecule or in different molecules. For example head of the lipid is interacting with SW while the tail is not OR lipid1 interacts but lipid2 does not. So atoms B1,B2...Bn can overlap with SW (SW is never in the same moleculetype with A or B). Something like this:

```
[ atomtypes ]
; name bond_type mass charge ptype sigma epsilon
SW  12 0.00000 0.00000 A  0.0  0.0
```

02/23/2020 2/4
DU 12 0.00000 0.00000 A 0.0 0.0

[nonbond_params]

; i j func
SW A1 1 0.85 1e-5
SW A2 1 0.85 1e-5
SW A3 1 0.85 1e-5

; All interactions with B1,B2,B3 are zero by default

[moleculetype]
SW 3
[atoms]
1 SW 1 SW SW 1 0.0 12.0
2 DU 1 SW DU 2 0.0 12.0
3 DU 1 SW T 3 0.0 12.0 ;Pull target particle

[bonds]
1 2 6 0.0 10000

[moleculetype]
LIP1 3
[atoms]
1 A1 ....
2 A2 ....
3 A3 ....

[moleculetype]
LIP2 3
[atoms]
1 B1 ....
2 B2 ....
3 B3 ....

[moleculetype]
LIP3 3
[atoms]
1 A1 .... ; Interacts!
2 B2 .... ; Do not interact!
3 B3 ....

#7 - 05/26/2016 12:03 PM - Gerrit Code Review Bot
Gerrit received a related patchset '1' for Issue #1958.
Uploader: Berk Hess (hess@kth.se)
Change-Id: If33245a96ceae78c9077a825cb22335c853a810
Gerrit URL: https://gerrit.gromacs.org/5893

#8 - 05/26/2016 12:08 PM - Berk Hess
But the tail of the lipid would never come close to Sw, will it? So theoretically there is an issue, but in practice not. Note that the actual nan/inf issue is still due to atoms DU and T not being excluded from each other in moleculetype SW.

PS I just pushed up a change to gerrit that checks if the energies are finite. This should catch most cases and give the user a hint of what is causing the issue. However, it doesn't allow non-interacting atoms to overlap.

#9 - 05/26/2016 01:26 PM - Semen Yesylevskyy
The lipid which is B1,B2... would overlap with SW since there are no interactions between them. The whole story is about allowing A1,A2... to feel the wall made of SW while B1,B2... will go through (and overlap).

#10 - 05/26/2016 03:52 PM - Berk Hess
Ah, now I understand. (but you still need to add the missing exclusions/bond)
Maybe you could use the wall potential option for that, instead of using atoms?

With one non-interacting atom the chance that something goes wrong is around (REAL_MAX^(-1/12))^3 * number-density of the other atoms. In your case this would result in something around 10^-8. So you should be able to run for a reasonable amount of time.

#11 - 05/26/2016 05:15 PM - Semen Yesylevskyy
No, unfortunately we can't use walls since our semi-transparent wall is of complex shape and we want to change its form during the simulation, which means that there is no alternative to atoms. This setup works nicely for group cutoffs and it would be very unfortunate if switch to Verlet will kill this possibility forever. We are using around 2000 dummies, so it will crash for sure.
I'm not a big expert in low-level Gromacs internals, so it is hard for me to understand why it works with group kernels but fails for Verlet.

Anyway, is it possible to introduce some kind of special particles for such semi-transparent walls which can safely overlap with normal atoms? Even if they will slow down simulations due to additional checks for overlaps, non-optimal kernels, etc. this is still much better then having nothing.

#12 - 05/26/2016 05:23 PM - Semen Yesylevskyy
Btw, if the atoms have no charge how close they can be located without the crash in current implementation? May be it is possible to workaround the problem by defining very weak LJ repulsion between them?

Another question. Does it help to define an energy group exclusions between SW and B1,B2... particles?

#13 - 05/26/2016 05:48 PM - Berk Hess
I am thinking if I can come up with an efficient enough solution.

But you can just try to run (when you added the missing exclusion), since as I said, the change of getting NaN is about 10^-8 per atom.

#14 - 05/26/2016 05:56 PM - Berk Hess
If you don't care about a factor 1.5 in performance, you can run double precision. Then the chance that atoms gets too close is 10^-75 per atom, which is completely negligible.

#15 - 05/26/2016 09:29 PM - Semen Yesylevskyy
Well, this is not the best idea to run something, which is known to crash randomly. From the user point of view I'd prefer to run this in a stable way in single precision (probably with performance penalty). I understand that my setup is a bit bizarre and 99% of the users will never use such black magic as semi-transparent particles, but at the same time it is really very useful.

#16 - 05/27/2016 10:02 AM - Erik Lindahl
Hi,

I strongly agree that we cannot have things failing randomly, no matter how small the probability is or what performance penalty it has.

Having said that I think, I think we have a proper solution now in https://gerrit.gromacs.org/#/c/5886/ ;)

#17 - 05/27/2016 04:04 PM - Berk Hess
I am thinking of adding a warning to grompp for atoms in moleculetypes that are not bound to any other atom. Although technically such setups will run, in 99% of the cases this is not what the user intended.

#18 - 05/28/2016 09:25 AM - Gerrit Code Review Bot
Gerrit received a related patchset '1' for Issue #1958.
Uploader: Berk Hess (hess@kth.se)
Change-Id: iabb00563c76a9795484d89d1c67d438f2c31f
Gerrit URL: https://gerrit.gromacs.org/5904

#19 - 05/31/2016 04:25 PM - Berk Hess
- Status changed from New to Resolved

Applied in changeset fcc7c4c4c3891aa2db8cd4590479b5f152e1e6f.

#20 - 06/01/2016 02:07 PM - Mark Abraham
- Status changed from Resolved to Closed
- Target version set to 2016

#21 - 07/28/2016 03:21 PM - Mark Abraham
- Related to Bug #1965: Crash of mdrun with strange error messages added

#22 - 08/05/2016 12:39 AM - Mark Abraham
- Related to Bug #2023: Segfault again with non-interacting atoms and verlet cutoff added

Files
bug.zip  9.54 KB  05/13/2016  Semen Yesylevskyy