GROMACS - Bug #1645

Difference in energy with Verlet scheme due to PME dipole correction

11/19/2014 09:07 AM - David van der Spoel

Status:	Closed		
Priority:	Normal		
Assignee:	David van der Spoel		
Category:	mdrun		
Target version:	4.6.8		
Affected version - extra info:		**Difficulty:**	uncategorized
Affected version:	4.6		

Description

When computing the energy for a system of shell-waters, that is with ptype S, the energy is different in the 4.6 series and onwards, when using the Verlet scheme, compared to the group scheme. Will upload example shortly.
One issue that helps to get the first step (before minimization) correct is to comment out the following in relax_shell_flexcon (line 1010):

```
if (inputrec->cutoff_scheme == ecutsVERLET)
{
            /* put_atoms_in_box_omp(fr->ePBC, state->box, md->homenr, state->x); */
        }
```

Associated revisions

Revision 53526cfe - 11/24/2014 09:29 AM - David van der Spoel

Added warnings for ewald-geometry and surface-epsilon

ewald-geometry and surface-epsilon require the system dipole,
which will be incorrect when charge groups with net charge cross pbc.
grompp now checks and warns for this.
Refs #1645.

Change-Id: I02e317cbddb47256f942312ec53c5bab2b13be2a

History

#1 - 11/22/2014 02:41 PM - David van der Spoel

- File dimer.tgz added

Attached example shows that the Ecoul recip is the culprit. A water dimer is simulated for 0 steps. If the dimer is located inside the box group and verlet yield the same result, if one of the molecules is on the edge of the box the Ecoul recip gets a strange very high value after two iterations (note that the patch above takes care of differences due to the first do_force call).

Group:
LJ (SR)   0   --   0   0 (kJ/mol)
Coulomb (SR)   0   --   0   0 (kJ/mol)
Coul. recip.   0.0419108   --   0   0 (kJ/mol)
Polarization   5.97909e-05   --   0   0 (kJ/mol)
Potential   0.0419706   --   0   0 (kJ/mol)
Kinetic En.   0.00202031   --   0   0 (kJ/mol)

Verlet:
LJ (SR)   0   --   0   0 (kJ/mol)
Coulomb (SR)   ~~4.11256 -~~   0   0 (kJ/mol)

```
Coul. recip.         110.869       --        0         0  (kJ/mol)
Polarization      5.97914e-05      --        0         0  (kJ/mol)
Potential            106.756       --        0         0  (kJ/mol)
Kinetic En.       0.00202491       --        0         0  (kJ/mol)
```

#2 - 11/22/2014 04:11 PM - David van der Spoel

- File dimer-nopol.tgz added

- Subject changed from Difference in energy for shell systems with Verlet scheme to Difference in energy systems with Verlet scheme

- Priority changed from Normal to High

The issue is not reserved to shell systems as we thought at first. Attached is an example without shells where the reciprocal energy is incorrect as well.

#3 - 11/22/2014 04:19 PM - David van der Spoel

- Subject changed from Difference in energy systems with Verlet scheme to Difference in energy for shell systems with Verlet scheme

- Priority changed from High to Normal

OK, sorry it happens only for shells anyway.

#4 - 11/22/2014 06:06 PM - David van der Spoel

It is somewhat more complicated. Due to some historic reason the mdp file has a surface correction for PME. This triggers an issue in the Verlet code, namely that the total system dipole needs to be computed. This only works correctly when molecules are intact. Calling put_atoms_in_box_omp breaks this. Removing the call to this function and having a good start structure fixes this for the first round.

In subsequent calls to do_force I can prevent that mu is computed again by changing the force_flags like:

```
/* Try the new positions */
        do_force(fplog, cr, inputrec, 1, nrnb, wcycle,
                 top, mtop, groups, state->box, pos[Try], &state->hist,
                 force[Try], force_vir,
                 md, enerd, fcd, state->lambda, graph,
                 fr, vsite, mu_tot, t, fp_field, NULL, bBornRadii,
                 force_flags & ~GMX_FORCE_NS & ~GMX_FORCE_STATECHANGED);
```

However, this means that the total system dipole is not updated during the minimization and that the total energy is not correct.

So how does one in the Verlet framework make molecules whole again?
If that is not possible the dipole will always be incorrect, right?

#5 - 11/23/2014 09:35 AM - David van der Spoel

- File spc-surf.tgz added

- Subject changed from Difference in energy for shell systems with Verlet scheme to Difference in energy with Verlet scheme due to ewald_geometry = 3dc

It seems there is a real issue with Verlet and PME. Since the molecules are broken the total system dipole is incorrect and hence the PME dipole correction can not be computed. The attached test system, spc-surf shows that for a system with ewald_geometry = 3dc
the dipole and energy of normal SPC water diverge quickly (within 20 steps in double precision). The dipole correction to the energy is way too high, e.g.

```
group/mdrun0.debug:dipcorr =    0.000     0.000   -22.673
verlet/mdrun0.debug:dipcorr =    0.000     0.000  -193.160
```

It should be noted that this is a pathetic case in some sense since there is no vacuum in the box and hence molecules are broken in the normal direction too which they would not normally be.
The dipoles in the X and Y direction as expected also different:

```
group/mdrun0.debug:mutot   =   -0.021    -0.495    -0.084
verlet/mdrun0.debug:mutot  =   -2.301    -1.260    -0.714
```

#6 - 11/23/2014 09:46 AM - Gerrit Code Review Bot

Gerrit received a related patchset '1' for Issue #1645.
Uploader: David van der Spoel (davidvanderspoel@gmail.com)
Change-Id: I4c756e835f35834d474c6372ee942b2037c34e28
Gerrit URL: https://gerrit.gromacs.org/4232

#7 - 11/23/2014 10:27 AM - David van der Spoel

- Subject changed from Difference in energy with Verlet scheme due to ewald_geometry = 3dc to Difference in energy with Verlet scheme due to PME dipole correction

Digging somewhat more I now found issues also with the default ewald_geometry = 3d and Verlet when the epsilon_surface is not 0 (infinity). E.g. when using 65 for SPC (which happens to be the magic SPC dielectric constant) one gets after some steps:

```
group/mdrun0.debug:Total dipole correction: Vdipole=0.294524
verlet/mdrun0.debug:Total dipole correction: Vdipole=11.3957
```

Since the dipole correction also influences the force this is a real problem, even though many would not use it. I guess we just have to turn off the dipole correction completely with Verlet.
We have:

```
group/mdrun0.debug:dipcorr =   -0.061    -1.016    -0.423
verlet/mdrun0.debug:dipcorr =   -6.345    -2.571    -0.424
```

and the code does the following:

```
if (dipole_coeff != 0)            {
for (j = 0; (j < DIM); j++)                {
f[i][j] -= dipcorrA[j]*chargeA[i];
}
}
```

In other word large spurious forces will be added to the atoms.

#8 - 11/23/2014 11:05 AM - Berk Hess

These are not Verlet scheme issues, but occur with non-neutral charge groups.

The 3dc geometry should only be used without pbc in z, we should at least add a warning for this. I also used it for water slab in the middle of the box with full pbc. Then you only have minor issues when a single water passes the border.

A non-zero surface epsilon could work with neutral molecules, but without charge groups we will need to correct for molecules split over pbc.

#9 - 11/23/2014 12:02 PM - Gerrit Code Review Bot

Gerrit received a related patchset '1' for Issue #1645.
Uploader: David van der Spoel (davidvanderspoel@gmail.com)
Change-Id: I02e317cbddb47256f942312ec53c5bab2b13be2a
Gerrit URL: https://gerrit.gromacs.org/4233

#10 - 11/23/2014 06:06 PM - Berk Hess

I would be nice to fix the actual issue. But to do this for the general case would require graph like code, which we already have, combined with domain decomposition. This is difficult and might be expensive. Doing it for molecules up to the size of the cut-off radius would not be too hard. We could add an "interaction"-type dipole to the DD reverse top which then automatically gets assigned to the domain where all coordinates of the molecule are available. The question is how often the surface_epsilon option is used (correctly).

#11 - 11/23/2014 09:48 PM - David van der Spoel

The bulk dipole correction drops off quickly with box size, therefore it can probably be neglected in most cases. The surface correction is more important but not a problem in practice either if the liquid is in the center of the box. So is it really worth the effort?

#12 - 11/26/2014 11:41 AM - Berk Hess

The dipole and surface epsilon correction are identical, except that surface epsilon works on all 3 dimensions.
The dipole correction is not really problematic, since molecules should not cross pbc in z.
The surface epsilon only makes sense when only molecules with no net charge are mobile and any charged molecules do not cross pbc. If there is a need for such cases, we could implement a correct treatment for the Verlet scheme.

#13 - 06/22/2015 06:03 PM - Erik Lindahl

- Status changed from New to Closed

4233 has been merged and solved the main problem, and the remaining point appears to mostly be a theoretical issue. Closing for now, but if somebody working on shells wants to spend time on improving it we can reopen.

Files

dimer.tgz	110 KB	11/22/2014	David van der Spoel
dimer-nopol.tgz	6.96 KB	11/22/2014	David van der Spoel
spc-surf.tgz	44 KB	11/23/2014	David van der Spoel