Hi,

i found that using g_hbond as in:

```
g_hbond -f traj.xtc -s topol.tpr -n index.ndx -ac -smooth 1
```

gives me a seg-fault (see for example the stdout below)
i did not include/upload any input files because the files i used tend
to be huge. However this should be reproducible anyway as it seems to be a general issue:
I get the same error for a number of widely different systems, e.g., a polymer blend or
a protein in aqueous solution ...

this happens both with single and double precision (both for the
g_hbond and the trajectory versions) and also for both
g_hbond from gmx-4.6.5 and gmx-5.0-beta2 (the latter with single prec
i seem to recall ... gmx -h wouldn't tell me ...)

if i do not use the smooth option but leave everything else
the same then the segfault goes away ...

cheers,

michael

below a typical stdout of g_hbond -f traj.xtc -s topol.tpr -n index.ndx -ac -smooth 1

```
[...]
Select a group: 1
Selected 1: 'Protein'
Select a group: 20
Selected 20: 'Water'
Checking for overlap in atoms between Protein and Water
Calculating hydrogen bonds between Protein (4899 atoms) and Water (23835 atoms)
Found 8389 donors and 8833 acceptors
Making hbmap structure...done.
Reading frame    0 time    0.000
Will do grid-search on 15x15x15 grid, rcut=0.35
Frame loop parallelized with OpenMP using 8 threads.
Last frame      5000 time 2000.000
Found 88942 different hydrogen bonds in trajectory
Found 181742 different atom-pairs within hydrogen bonding distance
Merging hbons with Acceptor and Donor swapped
8389/8389  
- Reduced number of hbons from 88942 to 83886
- Reduced number of distances from 181742 to 181742
Back Off! I just backed up hbnum.xvg to ./#hbnum.xvg.1#
Average number of hbons per timeframe 578.545 out of 3.705e+07 possible
```
Doing autocorrelation according to the theory of Luzar and Chandler.

ACF 83886/83886
Normalization for c(t) = 0.00172792 for gh(t) = 3.45631e-07

WARNING: Correlation function is probably not long enough
because the standard deviation in the tail of C(t) > 0.001
Tail value (average C(t) over second half of acf): 0.0786273 +/- 0.00446663
Segmentation fault (core dumped)

Associated revisions
Revision 3cced2d - 06/23/2015 02:24 PM - Erik Lindahl
Removed buggy -smooth hack in hbond module
As discussed in redmine #1553, this was a hack
added a while ago, but apparently it leads to
segfaults. We should avoid hacks in the first
place, and since nobody has been interested in
fixing it for the last year it's time to kill it.

Fixes #1553.
Change-Id: I94faa3c543bcca4c729753c8b5cda5a5775d0ff

History
#1 - 07/03/2014 10:17 PM - Mark Abraham
- Description updated

#2 - 07/11/2014 11:11 AM - Erik Lindahl
- Assignee set to David van der Spoel

#3 - 07/28/2014 02:41 PM - David van der Spoel
Michael, have you tried to run this without the -smooth option? This is some kind of hack that I implemented at one point in time, but which is
unnecessary once correlation functions are computed properly (which we are working on https://gerrit.gromacs.org/#/c/3471/).
So rather than “fixing”
an old hack I would suggest you run without the smooth option for now and try the new version of code once it is finished.

#4 - 07/28/2014 02:57 PM - Michael Brunsteiner
ok, thanks for the feed back, i'll wait ...

BTW ... when i use g_hbond to analyze h-bond dynamics i keep
getting a warning "Correlation function is probably not long enough"
i guess this is not surprising as i look at H-bond dynamics in a polymer blend
which is very SLOW ... however, i do observe quite a number of switches (bonded<>non-bonded),
on average dozens to hundreds per H-bond ... and if i understood that correctly the algorithm
you discuss in the paper that is cited by g_hbond (see below) was originally designed to deal
with RARE events ... do have any idea how rare an event can be for this to still converge ?

D. van der Spoel, P. J. van Maaren, P. Larsson and N. Timneanu
Thermodynamics of hydrogen bonding in hydrophilic and hydrophobic media

#5 - 07/28/2014 03:32 PM - David van der Spoel
The warning message comes from a rather crude analysis where the second half of the ACF (presumed to be the tail) is averaged and it is assumed it
should go to zero. If it does not the warning is printed. However, due to the way in which the ACF is computed, the convergence is not as good as it
should be, this is why I started to rewrite the code.

#6 - 07/28/2014 03:39 PM - Michael Brunsteiner
ok, thanks ... i'll keep an eye on https://gerrit.gromacs.org/#/c/3471/ ...
cheers,
#7 - 08/17/2014 11:04 PM - Mark Abraham
- Target version changed from 5.0 to 5.x

#8 - 06/22/2015 06:10 AM - Gerrit Code Review Bot
Gerrit received a related patchset '1' for Issue #1553.
Uploader: Erik Lindahl (erik.lindahl@gmail.com)
Change-id: 94faa3c543bcca4c729753c8b5cdaf5a5775d0ff
Gerrit URL: https://gerrit.gromacs.org/4757

#9 - 06/22/2015 10:50 PM - Erik Lindahl
- Status changed from New to Fix uploaded

#10 - 06/29/2015 03:24 PM - Erik Lindahl
- Status changed from Fix uploaded to Resolved

#11 - 06/29/2015 03:24 PM - Erik Lindahl
- Status changed from Resolved to Closed

#12 - 07/11/2016 07:47 PM - Mark Abraham
- Target version changed from 5.x to 5.0.7