How to use GROMACS 4.5.5 with S^2 restraints

Zoltán Gáspári

June 10, 2015

1 Introduction

This document describes the usage of S^2 restraints in GROMACS. These restraints are not included in the official version, rather programmed by myself (Z. Gáspári). Therefore, it is important that the version described here is **not official GROMACS** and should not be treated or considered as such in any way. This document is practice-oriented, for the theoretical details kindly refer to the literature cited. Installation instructions can be found in the file named 'README.TXT' supplied with the package.

The first implementation of S^2 restraints (and, in principle, the MUMO approach described in Richter et al, 2007) in GROMACS was done for version 3.3.1 (described first in Batta et al., 2009 and for the 4.5.5. version in Fizil et al., 2015). The present package contains an implementation of S^2 restraints for GROMACS version 4.5.5. Note that it is not expected that the provided files will be suitable for any earlier or later GROMACS release without modifications. Also note that the package is under development and might contain features not yet extensively tested and you might run into parameters that are not implemented (might be implemented later or are obsolete).

2 About ensemble simulations

S^2 restraints can only be used for multiple structures simlated simultaneously, so a bririef intoduction to ensemble calculations is in place. The official version of GROMACS (as of writing this is 4.5.5) is able to handle ensembles and use NOE and orientation restraints on them. GROMACS should be compiled with mpi support (most binary distributions are compiled without it). See the file 'README.TXT' supplied with the package for more information.

The (most basic) magic command to run a simulation in ensemble mode is:

```
mpirun -np N mdrun -multi N -deffnm <molname>
```

Here the number N controls the number of replicas to simulate, note that it should be supplied twice, once for MPIRUN and once for MDRUN. The option `-multi` tells gromacs to simulate N different replicas and this also causes mdrun to look for files `topol0.tpr`, `topol1.tpr ...` as inputs for each process. Note that for this type of runs you should run grompp as usual but separately for each replica to produce the N input (`tpr`) files[1] It

---

[1]This means that you should **not** use the -n switch for GROMPP, that should be used with distributed calculations where you would like to run a single simulation on multiple nodes. Of course you can run an ensemble calculation with each replica running as more than one process, then you should combine these two modes but note that it might not work with the S^2 extension at the moment.

is important that in order to get meaningful results, your input `tpr` files should contain different starting structures and/or velocities (different starting velocities can be generated by in setting the `gen_vel` option to "yes" and the `gen_seed` to a different number for each replica in the `mdp` files). This can be easily done with the scripts MDP_RANDOMSEED.PL and MULTIGROMACS.PL, as detailed below.

3 Getting started

The directory 'testcalc' supplied with the package contains the files and scripts for a simple parallel simulation. Copy this directory to a place in your home diractory and try the script RUN.SH. If you understand this script, you can easily set up and run your own stuff. The script GETSTRUCTURESATDT.SH might come handy for the analysis of the runs as can extract structure snapshots from all of the trajectories generated. The files containing the snapshots from each trajectory can be used as input for the script G_DTPDBS2ENSEMBLES.PL able to produce ensemble files for each snapshotted time (thus each of these ensembles contains the replicas that were subject to ensemble restraning simultaneously).

4 Perl additions

There are a number of small PERL scripts to ease calculation setup and analysis (in the 'scripts' directory of the package, copy/link them into a location in your PATH or modify your PATH to be able to use them). All programs are GROMACS-like and invoked with `-h` will give information on their usage. The programs are:

- MULTIGROMACS.PL This program helps you set up ensemble simlations as it runs a specified GROMACS command (specified with option `-C`) N times (option texttt-N). Any '#' characters in the commands given will be substituted by the numbers 0..N-1 for each invocation. This program is especially useful for generating the input `topol0.tpr, topol1.tpr ...` files with grompp.

- MDP_RANDOMSEED.PL This program is used to set all the "seed" options in the input `mdp` file to a random number. If the otion is present in the input file, it will be overwritten, if not, it will be added.

- GROMACSRES2TOP.PL This script can be used to add restraints (NOE, S^2 and dihedral) to a GROMACS topology (`top`) file. Invoking with `-h` will give you an idea how to use it.

- G_DTPDBS2ENSEMBLES.PL This program uses the output multi-model pdb files of GETFINALSTRUCTURES.SH and produces ensembles consisting of as many members as replicas used for the simulation and containing structures at each time frame extracted by GETFINALSTRUCTURES.SH.

5 Topology and run control parameters

A section of the example topology file:

```
[ s2_restraints ]
; ai   aj   type   S2   S2err   bondlen
   18   19   1    0.277   0.0   -1.000 ;   2   N:   2   H
```

```
34   35  1    0.454   0.0   -1.000 ;   3   N:   3   H
48   49  1    0.600   0.0   -1.000 ;   4   N:   4   H
58   59  1    0.653   0.0   -1.000 ;   5   N:   5   H
87   88  1    0.524   0.0   -1.000 ;   7   N:   7   H
94   95  1    0.595   0.0   -1.000 ;   8   N:   8   H
```

S2 restraints should be included in the topology file similarly to distance restraints by specifying the atom pairs to be restrained. The S^2 errors are not considered in the present implementation and the `type` parameter should be 1. The `bondlen` parameter is interpreted as follows:

A (nonzero) positive value: the bond length specified will be used for all S^2 calculations regardless of its actual value at the time of evalutaion.

Zero: the bond length at the time of calculation will be used. Note that fluctuations pf the bond length are not taken into account in S^2 calculations (see below).

-1.0: the bond length in the initial structure is used for all subsequent calculations. Note that the bond length depends on the actual force field used. This is the recommended value for this parameter.

Note that the present implementation assumes the use of bond length constraints (e.g. LINCS or SHAKE) as the force calculations do not take into account fluctuations of the bond length (Best and Vendruscolo 2004). The script GROMACSRES2TOP.PL can be used to insert the restraints into the topology file.

A section of the example mdp file:

```
; S2 restraints: No,  Withfit, Withoutfit
s2re                    = Withfit
; Output frequency for S2 values to energy file
nsts2reout              = 100
; S2 force constant (denoted alpha in Best & Vendruscolo), use with care
s2re-fc                 = 1000000
; S2 population size (-1 : all replicas in one population)
s2re-popsize            = 2
; Group to fit for S2 calculation
s2re-fitgrp             = backbone
```

Parameters to be set in the mdp file control the following:

`s2re`: `No` = no S^2 restraining, any S^2 restraints in the topolpgy file are ignored; `Withfit` = for each S^2 calculation step, structures are fitted to the initial structure on the head ode using the group specified with s2-fitgrp; `Withoutfit` = no fitting performed for qssq calculations, results are not expected to be reliable except in special cases.

`nsts2reout`: frequency to write out calculated S^2 values into the log file. (Note that not into the `ener.edr` files!)

`s2re-fc`: the force constant for S^2 calculations (the parameter α in Best & Vendruscolo, 2004).

`s2re-fitgrp`: the group used for superpositioning of the structures when S^2 calculation involves fitting.

`s2re-popsize`: this (yet experimental!) parameter can be used to apply S^2 parameters independently on sub-ensembles. For example, if the total number of replicas is 8 and

s2re-popsize is 4, two 4-membered sub-ensembles (replicas 0-3 and 4-7) will be independently restrained by the S^2 values while not requiring correspondence to the restraints of all of the 8 replicas at any time.

6 Setting up and running a simulation

To run a parallel restrained simulation, standard preparations (position restrained MD & energy minimization etc.) can be done according to the favourite protocol of the user. To run a simulation, use the script run.sh, adjust the molecule name and the number of replicas (parameter np) to your needs. This script requires the programs MDP_RANDOMSEED.PL (to give random initial velocities to each replica) and MULTIGROMACS.PL (a tool to ease up the invocation of the same command for multiple files). A run control file named `full.mdp` is also needed (see above).

7 Additional notes

The present implementation includes experimental support for averaging distance restraints in a pairwise manner over the replicas, similar to the principles described in Richter et al. 2007 (MUMO concept). For this, you only need to make the following setting in the `mdp` file:

```
disre                   = Pairwise
```

Note that no other aspects of the calculation procedure are modified (most importantly r^{-6} averaging is left untouched in this version).

In addition, a simple implementation of the Accelerated Molecular Dynamics (AMD) method (see e.g. Wang & McCammon 2012) has been added for the dihedral energy term (used in Fizil et al., 2015). This feature can be controlled in the mdp file as follows:

```
; Accelerated MD by boosting dihedral potential: No or Yes
AMD                     = yes
; AMD alpha parameter
AMD-alpha               = 0.5
; AMD boost energy for dihedrals
AMD-dihboostE           = 2500
; Frequency of writing AMD information to log file
AMD-out                 = 10
```

For details on these parameters kindly refer to the Wang & McCammon paper (or basically any other review on AMD).

References

A. Fizil, Z. Gaspari, T. Barna, F. Marx, G. Batta (2015): "Invisible" conformers of an antifungal disulfide protein revealed by constrained cold and heat unfolding, CEST-NMR experiments, and molecular dynamics calculations. *Chem. Eur. J.* 21:5136-5144.

R.B.Best, M. Vendruscolo (2004) Determination of ensembles of structures consistent with NMR order parameters. *J. Am. Chem. Soc.* 126:8090-8091.

B. Richter, J. Gsponer, P. Varnai, X. Salvatella, M. Vendruscolo (2007) The MUMO (minimal under-restraining minimal over-restraining) method for the determination of native state ensembles of proteins *J. Biomol. NMR* 37:117-135.

Y. Wang, J. A. McCammon (2012) Accelerated molecular dynamics: Theory, implementation and applications *AIP Conf. Proc.* 1456, doi: 10.1063/1.4730656