Journal: Living Journal of Computational Molecular Science

Title: A Suite of Advanced Tutorials for the GROMOS Biomolecular Simulation Software.

Authors: Bettina Lier, Christoph Öhlknecht, Anita de Ruiter, Julia Gebhardt, Wilfred F. van Gunsteren, Chris Oostenbrink, Niels Hansen

Date: December 15, 2020

Comments of the authors to the comments of the reviewers and of the editor on the manuscript.

Reviewer 1:

Reviewer's comments:
This work will be immensely helpful for advanced users who are new to GROMOS. In my opinion, this is suitable for the journal. I hope there will be updates regarding new software changes and installation requirements with time.

Authors' reply:
We thank the reviewer for the positive evaluation of this tutorial. New releases of GROMOS are made available through the website (www.gromos.net) regularly, with added functionality in the simulation and analysis tools. We hope to time the next release with the publication of this tutorial. The tutorial was constructed such, that additional chapters can be easily added when future releases have new functionalities or tools that could be described.

Reviewer's comments:
From my perspective, the paper did not have any significant grammatical errors and it was easy to read. However, few file names and directory changes are needed (suggested below) to make it easier to follow. In tutorial 1 in section 3.1.2, while the GROMOS coordinate file generation, there are warnings about ignoring certain atoms, which might be a good idea to mention in this manuscript. In section 3.2.2, the topology for the ligand bound to the protein (PLA2_ASA.top) is not present. In section 3.2.4, the input file md_TI_PLA2_ASA_Ca_2Na.imd mentioned in the md_TI subdirectory is not present or not evident. In section 3.2.9, the authors could consider changing the subdirectory correctionASAdGdir to correction/ASA/dGdir. The author should consider mentioning the reason behind their choice of 'lambda' windows for tutorial 2 for readers' sake.

Authors' reply:
Indeed, the warnings need an explanation. Since we are using an NMR structure, the PDB file contains more hydrogen atoms than are needed for GROMOS, as in the GROMOS force field only polar and aromatic hydrogens are explicitly represented. Aliphatic hydrogens are non-existing due to the use of united atoms. The aliphatic hydrogen and carbon atoms are merged to form united atoms which have their own parameters. We added the following text to section 3.1.2:

> "Since the used NMR structure contains more hydrogen atoms than needed by the united-atom GROMOS force field, merging aliphatic hydrogen and carbon atoms into one interaction site, a list of warnings regarding ignored hydrogen atoms is issued, which can be ignored. If the initial structure was determined using X-ray diffraction, missing hydrogen atoms can be generated with the GROMOS program gch as explained in the basic tutorial"

Additionally, in the revised version of the tutorial files we added the @gch flag to the pdb2g96 argument file which ensures that the coordinates of those hydrogen atoms remaining in the configuration file are consistent with the force field within a tolerance of 0.1%. However, this step is usually not critical because the energy minimization procedure carried out in the next step would also take care of this.

The file PLA2_ASA.top is not present indeed, because it is not relevant. We prepare the topologies for the ligand and the protein separately (ASA.top and PLA2.top) and combine them directly together with the ions. All relevant topologies are present in the topo directory. We thank the reviewer for making us aware of this inconsistency in our manuscript, where we now changed "ligand bound to the protein PLA2_ASA.top" to "protein PLA2.top".

We checked the missing file md_TI_PLA2_ASA_Ca_2Na.imd and found that it was present.

We changed the subdirectory correctionsASAdGdir to corrections/ASA/dGdir in the manuscript and thank the reviewer for noticing this inconsistency.

The choice of lambda windows (or lambda points) is a repeatedly discussed point. Here we explained that 11 equally spaced windows are typically a reasonable starting point, especially without pre-knowledge of the curvature of dH/dlambda. The number of points and the spacing is recommended to be adjusted according to the curvature and the error estimates. We further mentioned that in most cases this is not necessary when one uses extended thermodynamic integration with its reweighting scheme. Sometimes, even fewer than 11 points are sufficient. However, for standard TI it is often required to add more points and change the spacing. As we are using X-TI and take advantage of the pre-calculation of intermediate non-simulated points, we will not go into detail on how to adjust the windows here (which is mostly relevant for standard TI).

Reviewer's comments:
The authors could improve the installation instructions. The version of gcc and gsl specified in the instructions are old. The installation should be tried with newer versions of the same. More focus should be put into the details of mpi installation. A lot of the calculations mentioned in this article involve cluster use and require longer cpu time. For instance, in the installation guide for the md++ mpi the use of fftw3.3alpha1 is specified. However, this doesn't seem to be available anymore in the fftw website. I went ahead and used fftw.3.3.8 instead for the mpi installation, as it also had mpi libraries. However, significant improvement was not seen when using the mpi version of md++ over the serial version; I am unsure if it is because of the fftw installation. For ex: using 16 cpus(Intel(R) Xeon(R) CPU E5-2620) took 27 hours to complete one of the unrestrained molecular dynamics. It might be a good idea to include the GPU installation instructions for md++ for longer runs to make it faster. It would also be beneficial to mention the simulation timings for the user to have an approximate idea about the resources.

Authors' reply:
Indeed, the installation guide seems to be somewhat outdated and has been revised in the meantime including GPU installation instructions and references to volume 8 of the GROMOS manuals. In our experience GROMOS works well with the current libraries of gsl (2.6) and fftw (3.3.8) as well. We fear that the reviewer may not have noticed that after compilation with mpi support, the binary md_mpi should be used, rather than just md. This has been made more explicit in the revised installation guide and also in the relevant tutorial files.

Reviewer 2:

Reviewer's comments:
The article could be improved by addressing the following minor comments.

1. *At the beginning of section 3.1. the authors should briefly comment on why restraining of order parameters might practically be relevant and what one will learn with such an analysis?*
2. *In the first sentence in section 3.1.4, the comma should be removed.*
3. *The authors may consider commenting as to why the experimental order parameters in Figure 3 tend to in general be greater than the calculated values?*
4. *The comma on line 6 from the bottom of the first column on p. 16 should be removed (between "now" and "will")*
5. *The binding free energy from the second tutorial is quoted as -32.2 kJ/mol at the end of section 3.3.5 on p.16, while in the second tutorial it was actually given as -32.3 kJ/mol. This should be made consistent.*

Authors' reply:
We thank the reviewer for the suggestion and made the following changes:

1. In section 3.1 we added a short paragraph and point to recent applications of the method: "Such time-averaged restraining enhances the configurational sampling by forcing the molecule to surmount barriers that would, without restraining, only be surmounted rarely, that is, on longer time-scales. Moreover, a possible force-field deficiency hampering the agreement with experiment can be redressed using this restraining technique. In this way configurational ensembles consistent with NMR data can be generated allowing a structural interpretation of experimental observations [14,15]."
2. We removed the comma.
3. In the unrestrained simulations the majority of the calculated values is indeed below the experimental values. Although we observed such a behaviour also in other cases, it seems to be very dependent on the system studied and the force field used (see e.g. Childers and Daggett, J. Phys. Chem. B, 2018, 122, 6673) and is not a general feature. In the restrained simulations a flat bottom of ΔS^2 = ±0.05 was used, meaning that no restraining force was applied if the absolute value of the difference between simulation and experiment was within this boundary. Except for very few cases the difference between the orange dots and the red lines are within this range. In the revised version we amended the text on the flat bottom parameter on page 5: "In the column DSO the flat-bottom parameter of the restraining potential-energy term is set to 0.05. Therefore, no restraining force is applied if the absolute value of the difference between simulation and experiment is smaller than or equal to this value."
4. We removed the comma.
5. In the second tutorial we calculated the binding free energy as -32.3 kJ/mol. In the third tutorial we changed -32.2 kJ/mol to -32.3 kJ/mol at the end of section 3.3.5 for consistency and thank the reviewer for noticing this typo.

Reviewer 3:

Reviewer's comments:
The article is very well written. I only note a typo of "benificial" in section 3.3.3 that should be corrected. References may need additional revising, as "van Gunsteren et al WF" appears frequently and is likely due to some incorrect importing of the references. The PyMOL reference also appears incomplete or otherwise malformatted.

Authors’ reply:
The typo in section 3.3 was corrected. The malformatted references have been fixed.

Reviewer’s comments:
The tutorials run as expected, though given the computationally intense nature of the latter two, in conjunction with very slow simulation speed, I was not able to run everything. I would strongly suggest to the authors to generalize the input run scripts somewhat to avoid constant replacement of binary locations, directory names, etc. The issues can easily be circumvented by searching for (or setting) appropriate environment variables or use of current working directories for locations of files.

Authors’ reply:
We thank the reviewer for the feedback. Rather than forcing the users to set environment variables in a way that the scripts work, we encourage the users to put the GROMOS binary directory in the PATH. We added a sentence to the section “2.2 Software/system requirements”: “In order to use the GROMOS programs without specifying the full path you can add them to your PATH variable, see section 3.2.2 in volume 8 of the GROMOS documentation.” Furthermore, we have revised the scripts to avoid setting absolute paths.

Reviewer’s comments:
One specific issue that I do not feel the authors succeeded in is generalizing their approach to other systems. While there are some notes about the inadequacy of the setup in the HREMD simulations, there is little discussion about pitfalls or considerations for applying these techniques in other systems. A slightly more in-depth treatment of the literature would be useful throughout, as I find the material somewhat sparsely sourced, and all methods are simply delivered matter-of-factly. “This is what is done,” with very little explanation as to why. A really effective tutorial teaches a user to think about the techniques, not simply get an end result in such a highly controlled environment. That is, while the tutorial itself should deliver a predictable outcome with robust methods, it should also include enough discussion for a user to adapt it. It is not clear to me that this will be easily done based on the current presentation.

Authors’ reply:
The reviewer raises an interesting point. We agree that in particular tutorial 1 represents a rather controlled environment in which not much could go wrong. Therefore, the revised version of tutorial 1 now points to two recent applications of the order parameter restraining method (refs 14,15) that clearly go beyond a highly controlled environment and extensively discuss the properties of the method. The systems treated in tutorials 2 and 3 are rather complex already and represent typical research questions that appear for example in drug design. However, the methods discussed in tutorials 2 and 3 have been employed in other research settings as well. To provide additional context for the reader we added the following references to tutorial 2

[21]C. Ohlknecht, D. Petrov, P. Engele, C. Kröß, B. Sprenger, A. Fischer, N. Lingg, R. Schneider, C. Oostenbrink
Enhancing the promiscuity of a member of the Caspase protease family by rational design
Proteins 88 (2020) 1303 - 1318
doi: 10.1002/prot.25950

and to tutorial 3

[41] G. Nagy, C. Oostenbrink and J. Hritz
Exploring the Binding Pathways of the 14--3--3ζ Protein: Structural and free--energy profiles revealed by Hamiltonian Replica Exchange Molecular Dynamics with distance field distance restraints

Reviewer's comments:
Installing the software was quite burdensome and only became possible by disabling shared libraries, which I did not find anywhere in the INSTALL file as being a solution. Moreover, the dependencies are positively ancient. The tested GSL version is over a decade old, and no modern Linux machine will come with this option via package managers, and it may even be difficult to install from source. The Mac installation instructions became obsolete in 2007. If the authors desire for more users to adopt GROMOS as their simulation software of choice, its build system must be modernized. AMBER and OpenMM install easily with conda, CHARMM requires no external dependencies and only a Fortran compiler, and GROMACS has an elegant cmake build system that automatically determines almost everything a user needs (and then finds it) to optimize the build. GROMOS installation is lagging well behind all of these other software choices.

Authors' reply:
We are aware of the fact that users at times find it difficult to install GROMOS. In our experience GROMOS works with the most recent versions of the gsl (2.6) and the fftw (3.3.8) libraries. We have adjusted the installation guide of the new release. With this release, the disable-shared option becomes the default, which can still be overruled by the user at the time of configuration.

Reviewer's comments:
The tutorials rely on a user knowing all of the contents of the .arg input files, mentioning that other tutorial material is available in Chapter 8 of the GROMOS manual. Having no access to this material, most of the input files were unclear to me and I cannot assess their correctness. They give output that is consistent with the expectations of the tutorial, but I wonder if even experienced users will know every single input keyword and whether they are being properly applied. Rather than approaching the tutorial simply from the perspective of "run this command, get this output," I think more general discussion of the contents of the input files is required. Some are well commented, others are not, with very cryptic values and nomenclature. Perhaps this reflects mostly on my lack of familiarity with the software, but in general, I would find this kind of setup overwhelming for trying to learn new programs and employing them reliably.

Authors' reply:
Indeed, this is a tutorial that aims at more advanced users of the program. The basic tutorial is part of the GROMOS manuals and is available free of charge at the GROMOS website (after registration). Users who downloaded GROMOS from the website, also have access to the program guide (volume 5 of the manual) with more extensive documentation of the input flags. Furthermore, after compilation of the code, one can generate local documentation using doxygen. In the revised version of the tutorial, we have explicitly pointed at these sources of further information in "2 Prerequisites".

Reviewer's comments:
In Section 3.1.1, the authors mention that check_top can be used to verify the contents of the topology, noting however that the program may not catch every inconsistency. This kind of remark is very troubling, and this is a prime opportunity for the authors to describe exactly what a user should expect and be careful of when running the program. Please elaborate here.

Authors' reply:
The reviewer points at a clumsy formulation. The program check_top will capture most of the common problems with topology. However, we do not want

to raise the expectation that if check_top agrees with the topology, this means that the topology is correct. One can still have inappropriate parameters assigned etc. We have rephrased the indicated text and now also point explicitly at volume 5 of the GROMOS manual where all 34 types of logical checks that check_top performs are listed explicitly. We think that it goes too far to list all of these in the current tutorial.

Reviewer's comments:
In Section 3.1.3, the authors note that the protein "has to be relaxed by energy minimization" before being embedded in the solvent. This is an example of what I noted elsewhere - a declarative statement that something must be done this way, when in fact I do not find such an operation generally useful in my own work or reported very frequently in the literature. If the authors find some benefit in performing this operation, it should be discussed rather than simply presented as a matter of absolute necessity. In this same section, it needs to be noted that the path/program name needs to be edited in em_GB3.run (as is noted for other run scripts later, but is omitted here).

Authors' reply:
We thank the reviewer for pointing at an example of the earlier statement. We recommend such a minimization to adjust the protein from its initial configuration to the force field that is being used, mostly the bond-lengths that may be slightly different in the force field that was used for the structure elucidation. This is mostly done, because subsequent minimizations of the solvent typically involve position restraints on the protein. If the protein itself was not minimized yet, restraining to a conformation that is not fully in agreement with the bond lengths in the force fields sometimes leads to unnecessary strain. The reviewer is right though, that this is not an absolute necessity. We have tried to describes our recommended procedures in more detail to the explanations why we choose to do things. Experienced users can then always determine to follow these recommendations or not. In lines 2 to 4 of section 3.1.3 (now lines 76 to 8) it was mentioned that the file em_GB3.run needs to be edited to adapt it to the concrete environment.

Reviewer's comments:
In Section 3.3.2, the authors describe a penalty associated with the ligand crossing the protein, set via PROTEINOFFSET. They give a value of 15 (no units) and mention that this value should be "large enough," without any discussion of what this value is in practice, its units, or how one decides on an appropriate magnitude. Please elaborate as I think this is a key consideration as it will likely affect ligand sampling. If users are led to believe that some generic value is appropriate, or they are otherwise not given sufficient reference discussion to choose their own value, they may be misled.

Authors' reply:
Once again, we thank the reviewer for pointing out this unclarity. The penalty technically is an offset to the calculated distances over the grid, to ensure that a grid points within the protein are never considered for the shortest path towards the reference point. It therefore has the unit of length. The simulations are not very sensitive to the exact choice of this value, as the ligand is basically not supposed to be pulled through the protein. We have adjusted the explanation of this parameter in the revised version of the manual.

Reviewer's comments:
I suggest explicitly adding units of nm to the values of DISC and DISH in Section 3.3.2.

Authors' reply:
We have added the units to the values in 3.3.2.

Editorial Office:

Editor’s comments:
An additional editorial point that I wanted to bring up as a more longer-term question is whether there are plans to make the more "basic" tutorials for GROMOS that are referenced (ref (6)) available online as well (the pdf manual that I can access online appears to be quite old). Having the more basic tutorials easily accessible and also updated will enable a broader audience to eventually benefit from these more advanced tutorials; perhaps it is your intention to ultimately expand this tutorial to include the ore basic ones as well, but I wanted to bring it up to get your thoughts on this.

Authors’ reply:
The basic tutorial is available online free of costs via the gromos website. We have now made this more explicit in the current tutorial. We have so far not included the basic tutorial in the one for LiveCoMS because we fear that it will be very difficult to maintain both version with respect to version changes or new releases. Based on our experiences with the current tutorial, we will consider to place the basic tutorial only on LiveCoMS, such that a single document needs to be maintained.