Transition Path Sampling

Introduction

In this tutorial the useage of the TIPSI package is demonstrated. TIPSI is an application of rare event sampling, designed to sample many possible transition paths in a molecular system with two stable states, where transitions between the two states are extremely rare. MORE INFORMATION TPS and GROMACS.

We use the THING TIPSI. Hoogsteen and Watson-Crick config.

This tutorial is written for TIPSI version 0.1, which relies on GROMACS version 4.5.4 to do molecular simulation. Git is required to pull the repository with the demonstration files. For visualizing structures, we recommend the VMD package. This tutorial is written assuming you work on a Linux systemWe assume some knowledge in working with Linux (`cd` and `ls` command), as well as a basic understanding of how to work with GROMACS. There will be a cheatsheet in the appendix of this tutorial. This version of the tutorial assumes you are working on Carbon, the clustercomputer of the Computational Chemistry group at the Universiteit van Amsterdam.

Stukje over hoe het opgezet is naast het protocol, welk field en alles.

Preparatory Analysis

To run an analysis TIPSI, as we need to supply an intial trajectory to TIPSI in which the transition takes place at least once. We also need to define the stable states between which the transition takes place in a quantitative way. To get an initial trajectory, we do a regular analysis of the molecular system first.

First, we log into the Carbon cluster. Make sure you are connected to the UvA network or VPN, and enter

```
ssh user@carbon.science.uva.nl
```

When logged into the account, pull the tutorial repository to your system, which can be done by

```
git clone https://dgoldsb@bitbucket.org/dgoldsb/tipsi-tutorial.git
```

which will copy the tutorial directory as a new directory in your current folder. Exploring the folder shows the following subfolders:

```
bin documents input output README.md
```

The folder `bin` contains some analysis tools we will use later, as well as the `TIPSI` distribution. All input files required for this tutorial are included in `input`, and output will be written into `output`. The `documents` directory contains supporting reading material.

We enter the directory `input` and explore, which shows the following 8 files:

```
dipeptala.pdb emw.mdp highT.mdp posre.mdp
submit_highT submit_md em.mdp md.mdp
```

The `.gro` file contains the structure of the molecule we simulate, in our case dipeptide analine. To view this structure, copy the file to your own PC using a secure copy command, and view the structure in VMD. A nice piece of software that can help in copying files is FileZilla, which provides a GUI for secure copy.

Already added solvent, did an energy minimaztion and contrained run. Now that we have relaxed the solvent we can run the MD simulation. Typically when preparing for TIPSI we run at two temperatures, room temperature (298K) and a higher temperature (400K). Here we do room only, otherwise denatures completely. The simulation at room temperature shows the stable state at the temperature we are interested in. At the higher temperature, the system will transition more easily to the second stable state. This simulation will provide the initial transition trajectory, and also helps us define the two stable states. We run these by entering:

```
grompp -f run.mdp -c conf.gro -p topol.top -o md.tpr
qsub submit\_md
```

The output will be put in the output directory.

Running TIPSI

To tell TIPSI how to define the stable states, we have to set up a parameter (.par) file. To find if there are stables states between which a transition can take place, we need to express our trajectory in terms of ome collective variable. In this example, we use the radius of gyration and the dihedral angle over atoms (5, 7, 9, 15).

First, to analyze our initial run, go to the output folder of the high temperature run. Now, run

```
g_gyrate -f highT.xtc -s highT.tpr -o gyr_highT.xvg
```

to generate a dataset containing the radius of gyration for each frame. Then, define all dihedral angles by running

```
mk_angndx -s highT.tpr -n angle.ndx -type dihedral
```

followed by

```
g_angle -f highT.xtc -od angles.ndx -ov dihed_highT.xvg
-type dihedral
```

with option 8 (5, 7, 9, 15) to generate a dataset of the dihedral angles. We can visualize how often each combination of these two variables is visited by running

```
python ../../bin/generate2Dbins.py gyr_highT.xvg 1 dihed_highT.xvg
1 0.2 0.3 -180 -40 100 100 Gyration Dihedral GyrDihed
```

This script will generate a set of plots in .pdf format. Transfer the plots to your computer and view them. You will see that there are two combinations of a radius of gyration and dihedral angle that are visited a lot by the simulation. If we repeat this with the lowT results, we see that there are few transitions between the states. As such, they can be considered semi-stable, and are our A and B state for TIPSI. Note down the ranges for both variables at which the molecule is in a semi-stable state.

Go to the ./bin directory, and look at the tps.par file. In this file, we define the maximum length of a transition path, as well as when the system is in state A, state B, or inbetween (I). Confirm that we confirmed the state in accordance with the range that you noted down. There is little documentation so far on how to include other variables in the parameter file, so some alternative options are included in an appendix. Make sure to fill in your own directory in the parameter file for the location of the .gro file, as we enter the full path here, not the relative path.

We included a group of atoms in out parameter file, namely protein. We must define this custom group in an .ndx file to pass this group to TIPSI. Go to the input directory, and run

```
make_ndx -f lowT.tpr
```

If you open this file with more, you will see that groups are defined in blocks, and that the name of the block is specified by a header in square brackets. The produced file allows you to use any of the GROMACS standard groups. Custom groups can be made manually, and are referred to by their header. The index file of choice is included in the run_ tipsi file. We edit this file slightly to contain just the groups that we are interested in: the protein group, and the group for the dihedral angle. Open index.ndx, and delete the blocks in index.ndx that are not the protein group, and rename the file index-pr-dh.ndx. Then, add

```
[ Dihedral ]
   5    7    9   15
```

and save the file in the input directory.

Finally, we need to copy our initial trajectory to our input directory. Change your directory to your high temperature MD output, and exectute

```
cp highT.trr ../../input-dipeptala/highT.trr
```

Now that we set up the parameter file, we need to edit our run script to take the correct input. Open the file run_ tipsi, and check if the .tpr file (for the simulation setting) and the .trr file (for the trajectory) are set correctly. Run TIPSI by entering

```
qsub submit_tipsi
```

Analyzing Results from TIPSI

Linux Cheatsheet

- **Display current directory**: `pwd`
- **View contents of the current directory**: `ls`
- **Change the directory**: `cd ./your/directory/here`

  Note: the currect directory is referred to as `./`, the directory in which your current directory resides is `../`
- **Copy a directory/file**: `cp ./target ./new/location/`

  Note: the option `-r` (recursive) is required for directories, this implies all the contents should be copied
- **Delete a directory/file**: `rm ./target`

  Note: again the recursive option is required for directories, and mind that deleted directories cannot be recovered.
- **Read a file**: `more ./target` or `head ./target` or `tail ./target`
- **Make a file executable**: `chmod +x ./target`

GROMACS Cheatsheet

- **Check the temperature of a simulation**: `gmxdump -s target.md | grep ref_t`

TIPSI Cheatsheet

In general, the way to specify the atom targets is in a numpy array. Groups can be specified in an index file, included in the run tipsi script (single line). Structure of a par file is discussed... as well as the functions in TIPSI.

Making a .par file

Defining groups

```
# Make sure to change to your own directory!
coords     x0  = /home/dylan/tipsi-tutorial/input-dipeptala/posre.gro

par        dh  = dihdeg(frame[5,:],frame[7,:],frame[9,:],frame[51,:])
init       ra  = rgyration(x0$Protein)
expression rg  = rgyration(frame$Protein)
```

Alternatively

```
# Initialize group of atom
init acc = numpy.array((5, 7, 13, 18))

parameter rg  = rgyration(frame[acc,:])
```

Calculator options

The available calculater function. List of coordinates or specific coordinates, sometimes specific atom role. Below a selection, for more explore `functions.py` in the `./bin/tipsi/calculator` folder.

- **Description**: `function with arguments`

NOTE THAT GYRATION IS WITHOUT MASS