Tutorial 1. SIRAH forcefield in GROMACS

Simulation of a coarse grained DNA molecule in explicit solvent

By Matias Machado
Mail any comment or suggestion to *spantano@pasteur.edu.uy*

This tutorial shows how to use the SIRAH forcefield to perform a coarse grained (CG) simulation of a double stranded DNA in explicit solvent (called WatFour, WT4) in four simple steps: 1 download; 2 map; 3 solvate and; 4 run. The main references for this tutorial are: *A Coarse Grained Model for Atomic-Detailed DNA Simulations with Explicit Electrostatics.* Dans PD, Zeida A, Machado MR, Pantano S. JCTC, 2010, 6:1711. *Another Coarse Grain Model for Aqueous Solvation: WAT Four?* Darré L, Machado MR, Dans PD, Herrera FE, Pantano S. JCTC. 2010, 6:3793. We strongly advise you to read these articles before starting the tutorial.

Software required
GROMACS 4.5.5 or later version properly installed in your computer. A molecular visualization program, we recommend VMD (freely available at: *http://www.ks.uiuc.edu/Research/vmd/*)

Previous knowledge required
How to perform a standard atomistic molecular dynamic simulation with GROMACS.

Hands on
0) Download the file *sirah.gmx.tgz* from *www.sirahff.com* and uncompress it into your work directory:

```
tar -xzvf sirah.gmx.tgz
```

You will get a folder *sirah.ff/* containing the forcefield definition, a set of useful tools in *sirah.ff/tools/* and the required material to perform the tutorial in *sirah.ff/tutorial/1/*.

Make a new folder for this tutorial:

```
mkdir tutorial1; cd tutorial1
```

Create the following symbolic links in the folder *tutorial1*:

```
ln -s ../sirah.ff .
ln -s ../sirah.ff/residuetypes.dat .
ln -s ../sirah.ff/specbond.dat .
```

1) Map the atomistic structure of a dodecamer DNA to its CG representation:

```
./sirah.ff/tools/CGCONV/cgconv.pl\
 -i sirah.ff/tutorial/1/dna.pdb\
 -o dna_cg.pdb
```

The input file *dna.pdb* contains all the heavy atoms composing the DNA molecule, while the output *dna_cg.pdb* preserves a few of them. Please check both PDB structures using VMD and a text editor:

```
vmd -m sirah.ff/tutorial/1/dna.pdb dna_cg.pdb
```

Notice: This is the basic usage of the script *cgconv.pl*, you can learn other capabilities from its help:

```
./sirah.ff/tools/CGCONV/cgconv.pl -h
```

From now on it is just normal GROMACS stuff!

2) Use Gromacs' pdb2gmx to convert your PDB file into Gromacs format:

```
pdb2gmx -f dna_cg.pdb -o dna_cg.gro -merge all
```

Choose SIRAH forcefield and then SIRAH solvent model.

Notice: Merging both DNA chains is convenient when planning to apply restraints between them.

During long simulations of DNA, capping residues may eventually separate. If you want to avoid this effect, which is called helix fraying, add Watson-Crick (WC) restraints at terminal base pairs. To add these restraints edit *topol.top* to include the file *WC_RST.itp* at the end of the *[moleculetype]* section:

Topology without WC restraints	Topology with WC restraints

`; Include Position restraint file`
`#ifdef POSRES`
`#include "posre.itp"`
`#endif`
	`; Include Position restraint file`
	`#ifdef POSRES`
	`#include "posre.itp"`
	`#endif`
	
	`; Watson-Crick restraints`
	`#include "./sirah.ff/tutorial/1/WC_RST.itp"`

3) Solvate the system

Define the simulation box of the system:

```
editconf -f dna_cg.gro -o dna_cg_box.gro -bt octahedron -d 2
```

Add WT4 molecules:

```
genbox -cp dna_cg_box.gro -cs sirah.ff/wt416.gro -o dna_cg_solv.gro
```

Update *[molecules]* section in *topol.top* to include the number of added WT4 molecules:

Hint! If you forget to read the number of added WT4 molecules from the output of genbox, then use the following command line to get it

```
grep -c WP1 dna_cg_solv.gro
```

Topology before update	Topology after update

`[ molecules ]`
`; Compound  #mols`
`DNA_chain_A  1`
	`[ molecules ]`
	`; Compound  #mols`
	`DNA_chain_A  1`
	`WT4  3179`

Add CG counterions:

```
grompp\
 -f sirah.ff/tutorial/1/em_CGDNA.mdp\
 -p topol.top\
 -c dna_cg_solv.gro\
 -o dna_cg_solv.tpr

genion -s dna_cg_solv.tpr -o dna_cg_ion.gro -np 38 -pname NaW
```

When prompted, choose to substitute ions for WT4 molecules.

Notice: The available ionic species in SIRAH forcefield are: Na+ (NaW), K+ (KW) and Cl- (ClW). One ion pair (e.g. NaW-ClW) each 34 WT4 molecules renders a salt concentration of ~0.15M.

Update *[molecules]* section in *topol.top* to include the 38 NaW ions and the correct number of WT4.

Before running the simulation it may be a good idea to visualize your molecular system. CG molecules are not recognized by molecular visualizers and will not display correctly. To fix this problem you may generate a PSF file of the system using the script *top2psf.pl*:

```
./sirah.ff/tools/top2psf.pl -i topol.top -o dna_cg.psf
```

Notice: This is the basic usage of the script *top2psf.pl*, you can learn other capabilities from its help:

```
./sirah.ff/tools/top2psf.pl -h
```

Use VMD to check how the CG system looks like:

```
vmd dna_cg.psf dna_cg_ion.gro -e sirah.ff/tools/sirahradii.tcl
```

Notice: VMD assigns default radius to unknown atom types, the script *sirahradii.tcl* sets the right ones.

4) Run the simulation

You can find typical input files for energy minimization, equilibration and production runs in *em_CGDNA.mdp*, *eq_CGDNA.mdp* and *md_CGDNA.mdp*, respectively. Please check carefully the input flags therein.

Create an index files adding a group for WT4 and NaW:

```
echo -e "r WT4 | r NaW\nq\n" | make_ndx -f dna_cg_ion.gro -o dna_cg.ndx
```

Make a new folder for the run:

```
mkdir run; cd run
```

Energy Minimization:

```
grompp\
 -f ../sirah.ff/tutorial/1/em_CGDNA.mdp\
 -p ../topol.top\
 -n ../dna_cg.ndx\
 -c ../dna_cg_ion.gro\
 -o dna_cg_em.tpr

mdrun -deffnm dna_cg_em &
```

Equilibration (NPT):

```
grompp\
 -f ../sirah.ff/tutorial/1/eq_CGDNA.mdp\
 -p ../topol.top\
 -n ../dna_cg.ndx\
 -c dna_cg_em.gro\
 -o dna_cg_eq.tpr

mdrun -deffnm dna_cg_eq &
```

Production (100ns):

```
grompp\
 -f ../sirah.ff/tutorial/1/md_CGDNA.mdp\
 -p ../topol.top\
 -n ../dna_cg.ndx\
 -c dna_cg_eq.gro\
 -o dna_cg_md.tpr

mdrun -deffnm dna_cg_md &
```

That’s it! Now you can check the simulation using VMD:

```
vmd ../dna_cg.psf dna_cg_md.xtc
```