

Citation:

Please cite the following papers when using the PACE force field,

- 1) Han, W.; Schulten, K. *J. Chem. Theory Comput.* **2012**, *8*, 4413.
- 2) Han, W.; Wan, C.-K.; Jiang, F.; Wu, Y.-D. *J. Chem. Theory Comput.* **2010**, *6*, 3373.

Please also cite the following paper as PACE is coupled with the MARTINI force field,

- 3) Marrink, S. J.; Risselada, H. J.; Yemov, S.; Tieleman, D. P.; de Vries, A. H. *J. Phys. Chem. B* **2007**, *111*, 7812.

Finally, if PACE is applied to model membrane protein systems, please also cite the following paper,

- 4) Wan, C.-K.; Han, W.; Wu, Y.-D. *J. Chem. Theory Comput.* **2011**, *8*, 300.

Prerequisites:

The following software is required for modeling protein systems with PACE:

- 1) GROMACS 3.x is required to prepare necessary files for simulation setup. GROMACS 4.x is strongly recommended for actual simulations as this version has gained much improved performance in parallel simulations. The installation guideline of GROMACS is detailed at www.gromacs.org. We present, in the end of this tutorial, a step-by-step instruction for users to quickly compile GROMACS (version 3.x) that is enough for system setup but not optimized for production simulation;
- 2) Python 2.4 or later is needed. To install Python, please visit www.python.org;
- 3) A C compiler is required. A GNU C compiler is recommended and will be assumed throughout this tutorial;
- 4) VMD, a visualization software, is optional but highly recommended.

Installation:

Assuming that a GROMACS 3.x has been installed at `/home/xxx/gromacs` and you have downloaded PACE at `/home/xxx/pace/`, please follow the steps below to complete the installation:

- 1) Copy all the files in `/home/xxx/pace/FF/` to `/home/xxx/gromacs/share/gromacs/top/`
- 2) Append one more line in `/home/xxx/gromacs/share/gromacs/top/FF.dat` as follows:

```
ffPACE_1.3vdW    PACE 1.3vdW
```

and increase the count displayed in the first line of the file by one;

- 3) Compile a program called `genPairPACE13` which is needed to modify topology files to incorporate PACE parameters:

```
cd /home/xxx/pace/programs/genpair/  
make
```

A binary file called `genPairPACE13` should be generated in the same directory;

- 4) Assuming that your working directory is `/home/xxx/workdir`, please copy the following files to the working directory:

```
/home/xxx/pace/programs/genpair/insert_param.py  
/home/xxx/pace/programs/genpair/genPairPACE13  
/home/xxx/pace/mdp/*.mdp
```

Getting started for simulations:

Once the installation is all set, you should be able to build up simulation systems with PACE in the same way as used to setup normal GROMACS simulations, except for a few variations for PACE. Shown below is a simple example of how to setup a PACE simulation starting from PDB. We have assumed here that users are currently located at the working directory `/home/xxx/workdir/` and that the environment variable `PATH` includes the path to the GROMACS binaries

```
/home/xxx/gromacs/bin.
```

- 1) Download a PDB file from www.rcsb.org to the working directories. Throughout this tutorial, we will use `3GB1.pdb` as an example;
- 2) Construct topology and coordinate files according to the downloaded PDB file:

```
pdb2gmx -f 3GB1.pdb -o 3GB1-pace.pdb -p draft.top -ter  
pdb2gmx -f 3GB1.pdb -o 3GB1-pace.gro -ter
```

Please note:

- (a) `pdb2gmx` will let user choose force fields for system setup, please make sure that PACE 1.3 is chosen;
- (b) Current PACE supports various protonation states of Glu, Asp and His. These protonation states can be specified with `pdb2gmx` using option `-glu`, `-asp` and `-his`, respectively;
- (c) Current PACE also supports both charged and capped termini. To construct charged termini, users need only to provide coordinates of an uncapped sequence and choose option 0 when asked to select terminus states. To build a capped terminus, both Ace for N-terminus and

- NMe for C-terminus, users need to provide coordinates of heavy atoms for these termini as separate residues and then choose the option `none` when asked to choose terminus states;
- (d) `pdb2gmx` is applied twice to generate two files needed for simulations. For the first time, it will generate a PDB file `3GB1pace.pdb` where atoms appear in the exactly same order as those do in the topology file `draft.top`. `3GB1-pace.pdb` will be used later as a reference to generate a special topology file including PACE parameters for the system. For the second time, `pdb2gmx generate coordinates (3GB1-pace.gro)` for simulation.
 - (e) Please make sure that the same options for building topologies are chosen when `pdb2gmx` is applied twice as discussed above.

- 3) Generate topology files for PACE. Recall that in the working directory, there are two programs called `genPairPACE13` and `insert_param.py`, the former generating special topology file containing PACE parameters for the system and the latter combining the normal topology file (`draft.top`) and the special topology file to create a topology file for actual simulations. To use `genPairPACE13`, users first need to find out how many atoms (`count_atom`) and residues (`count_residue`) there are in `3GB1-pace.pdb`, and then run the program as follows:

```
./genPairPACE13 count_atom count_residue 3GB1-pace.pdb  
1 > 3GB1.patch
```

where `3GB1.patch` is the special topology file. Please note that there is an option of `0/1` right after the name of a PDB file such as in the current case `3GB1-pace.pdb`. This option tells the program the status of protein termini with `1` for charged termini and with `0` for capped ones. Then one can combine `3GB1.patch` with `draft.top` as follows:

```
Python insert_param.py 3GB1.patch draft.top > 3GB1pace.top
```

Finally, a minor change need to be made to the resulting `3GB1-pace.top`, simply replacing the line

```
#include "spc.itp"
```

with

```
#include "cgWater.itp"
```

Otherwise, a lot of error information would pop up in the next step.

- 4) Starting from this step, the building procedure becomes the same as instructed in standard GROMACS documents, though there are a few variations in setup parameters. The setup of the example system is detailed as below:

Set the size of simulation box:

```
editconf -f 3GB1-pace.gro -o pace.gro -c -bt cubic -d 1.2
```

Solvate proteins:

```
genbox -cp 3GB1-pace.gro -cs cg216water.itp -p 3GB1-pace.top -  
vdwd 0.235 -o 3GB1pace-sov.gro
```

- 5) Energy minimization and pre-equilibration. In this stage, you need to carry out a series of simulations, using the mdp files provided, to get a .gro file that can be used to build the .tpr file for production runs. For each of these simulations, you need to do the following steps:

```
grompp -v -f xxx.mdp -c from_last_run.gro -p 3GB1-pace.top -o  
current_sim.tpr
```

```
mdrun -s current_sim.tpr -c for_next_run.gro
```

The first one of these simulations starts with 3GB1-pacesov.gro, generated in the last step. Each of the simulations will generate the coordinate files as the input for building the next simulation. We suggest an order, by mdp files to be used, as follow to perform simulations:

```
em-posres.mdp ; minimization with positional restraints on proteins  
pr.mdp ; short simulations with positional restraints on proteins  
em.mdp ; minimization without restraints pr.mdp  
full300.mdp ; for production run at 300 K
```

After the last step involving full300.mdp, a .tpr file will be generated for production runs. This .tpr file can be applicable for both GROMACS 3.x and 4.x. In case that GROMACS 4.x is used for simulation, please add additional options as follow:

```
mdrun ... (the same options as 3.x) -rdd 1.9 -dds 0.9
```

A quick installation of GROMACS 3.x

If users have already GROMACS 4.x installed, GROMACS 3.x is only needed to generate files for PACE simulations and users may not need to install an optimized version of GROMACS 3.x. Below introduces a quick way to install a version of GROMACS 3.x just for system setup.

First, please create a directory for installing 3.x,
`/home/xxx/gromacs`

Decompress source code files from `gromacs.tar`,
`programs/ tar -zxf gromacs.tar`

At the decompressed directory, compile and build
GROMACS:

```
cd gromacs-3.3.1
```

```
./configure --with-fft= fftpack --prefix=/home/xxx/gromacs
```

```
make
```

```
make install
```

After that, all the needed binaries should be installed at
`/home/xxx/gromacs/bin/`