Tutorial:
Rare Event Sampling with FRESHS and FFS using the example of a 1D particle in a double well potential.

Kai Kratzer

April 23, 2014

Contents

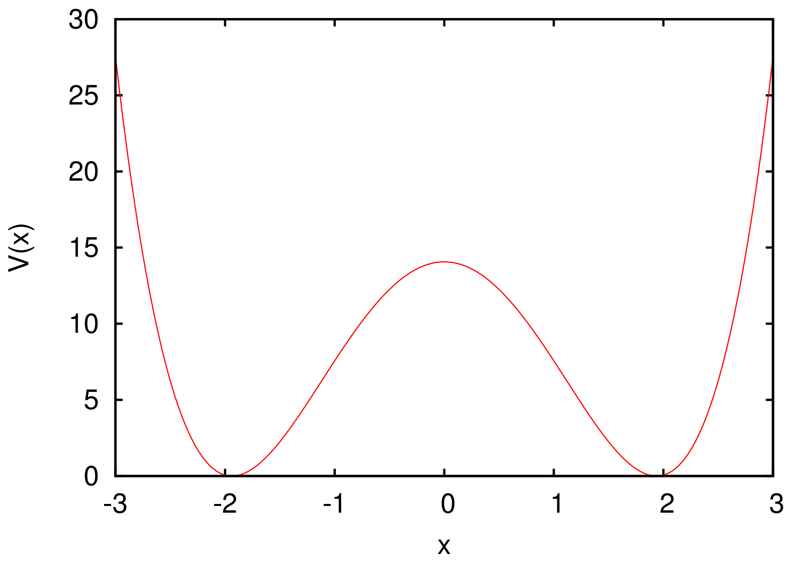

Figure 1: Double well potential.

1 Simulation problem

In this tutorial we perform FFS simulations of a 1D particle in a double well potential using FRESHS [1]. For the simulations, we use a velocity verlet integrator and a Langevin thermostat, set up in a custom python harness script. The potential (see also fig. 1) from which the forces are derived is given by

$$V(x) = x^4 - 7.5x^2 + 14.0625. \tag{1}$$

The particle is set up at the left-hand side of the barrier with $x < -1.5$. We are interested in pushing the particle over the barrier using FFS. Therefore, we set $\lambda_A = -1.5$ and $\lambda_B = 1.5$. We use the automatic optimized interface placement [2] with exploring scouts and a target probability of $p = 0.5$ to set up the scenario automatically. All values are pre-defined in the configuration scripts, but feel free to play around.

2 Getting started

1. Clone the latest FRESHS code from github:

```
git clone https://github.com/freshs/freshs.git
```

2. Navigate to

```
freshs/server
```

3. Have a look at the configuration file

```
server-python-ffs_tutorial.conf
```

and make changes if desired.

4. Start the server for this tutorial with

```
python main_server.py -c server-python-ffs_tutorial.conf
```

5. Open a second terminal and navigate to

```
freshs/harnesses/python-ffs_tutorial
```

   Open the job_script and look if the physics is calculated correctly. Then, navigate to

```
freshs/client
```

6. Start the client with

```
python main_client.py -c client-python-ffs.conf
```

The client starts the harness script specified in its configuration file which then performs the calculation of the physics.

Wait for the simulation to finish. If you can't wait, download a ready populated database for this example[1]. Note that for this small example we have a lot of communication overhead, because the runs are very short. In a 'real' simulation the computation time should take the largest amount of time.

3 Postprocessing

3.1 Database view

If you have the 'sqlitebrowser' or the firefox 'SQLite Manager' plugin installed, then you could use them to have a look at the raw data which comprises the output database, which contains the state at the end of the fragment, as well as enough information to re-generate it (i.e. a pointer to the parent fragment, and an RNG seed). You can also use the commandline tool 'sqlite3' to manipulate the database directly, but don't forget to do a backup first!

The database can be already analyzed and viewed during the simulation. Note, that you should not write to the database during the simulation, handle with care. If the simulation fails at a certain stage, data can be modified/deleted, and the simulation can be resumed from the last state in the database using the '-r' flag of the server.

3.2 Analysis scripts

The folder 'scripts' in your FRESHS directory contains analysis scripts which can be used directly to e.g. backtrack successful runs or to serve as templates for own analysis scripts.

[1] `http://www.freshs.org/dw/lib/exe/fetch.php?media=tutorial:2014-04-22_19-24-53_configpoints.sqlite.gz`

3.2.1 Backtracking successful runs

Change to the ‘freshs/scripts’ folder and run the following script:

```
python ffs_python-tutorial-analysis.py /tmp/DB/<timestamp>_configpoints.sqlite
```

The output is written to a folder called ‘OUTPUT/timestamp’. Change to this directory, start gnuplot and plot the tree graph to visualize the traces which lead to the points on the last known interface:

```
load "tree_success.gnuplot"
```

If you’d like to plot the pathways and configuration points, you could do it like this:

```
plot "trace0.dat" u 1:4 w p, "trace0.path" w l
```

3.2.2 Interface statistics

To plot statistical information about the simulation, e.g. a histogram of the runtime of the clients or a histogram of the calculation steps, run

```
./ffs_interfaces_statistics.py /tmp/DB/<timestamp>_configpoints.sqlite
```

The output is written again to the folder called ‘OUTPUT/timestamp’. Change to this directory, start gnuplot and plot the histograms:

```
load "histo_runtime.gnuplot"
load "histo_calcsteps.gnuplot"
```

3.2.3 Energy landscape

The energy landscape can be extracted from the distribution of the order parameter during the simulation and by weighting with the particular transition probabilities [3]:

```
python ffs_python-tutorial-energy-profile.py /tmp/DB/<timestamp>_configpoints.sqlite
```

The output is written again to the folder called ‘OUTPUT/timestamp’. Change to this directory, start gnuplot and plot the energy landscape and the potential:

```
V(x)=(x**4.0)-7.5*(x**2)+14.0625+c
fit V(x) "data_dG.dat" via c
plot "data_dG.dat", V(x)
```

References

[1] K. Kratzer, J. T. Berryman, A. Taudt, and A. Arnold. The Flexible Rare Event Sampling Harness System (FRESHS). *Comp. Phys. Comm.*, 2014.

[2] K. Kratzer, A. Arnold, and R. J. Allen. Automatic, optimized interface placement in forward flux sampling simulations. *J. Chem. Phys.*, 138(16):164112, 2013.

[3] C. Valeriani, R.J. Allen, M.J. Morelli, D. Frenkel, and P.R. ten Wolde. Computing stationary distributions in equilibrium and nonequilibrium systems with forward flux sampling. *J. Chem. Phys.*, 127, 114109, 2007.