PIGLE Starter Guide

SM Lambrick
August 2022

This guide goes through the steps of installing then getting started running PIGLE (Particles Interacting in Generalized Langevin Equation simulator). It is a practical instruction set on using the software and does not go into the theory or the science of the method, refer to the associated publication for information on the method.

What is PIGLE

PIGLE (Particles Interacting in Generalized Langevin Equation simulator) is a simulator of adsorbate particle dynamics on surfaces. The adsorbed particles may move in 4 dimensions: x, y, z, and rotations. The simulation of the z and rotational motion is optional. In addition adsorbate-adsorbate interactions may be included. The simulator models the *Generalized Langevin Equation*,

$$m\mathbf{x}_i = -\nabla V[\mathbf{x}_i(t)] - m\int_{-\infty}^{t} \gamma_i(t-t')\mathbf{x}_i(t')\mathrm{d}t' + \zeta(t) + \sum_{j\neq i} F_{ij},$$

which includes a time dependent friction term in addition to the ordinary Langevin Equation.

Units

In general the units of PIGLE are:

> ps for time (ps-1 for inverse time) Å (Angstroms) for spatial distances (Å-1 for inverse distances) meV for energy

Downloading/installing

Prerequisits

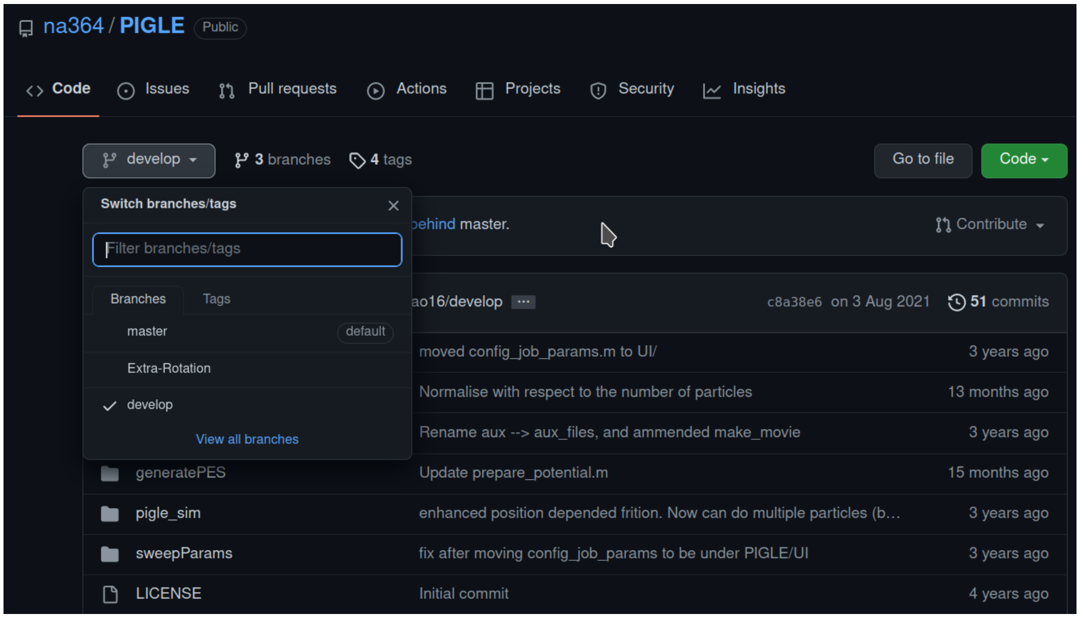
PIGLE uses Matlab and Simulik therefore a local install of both is needed. Refer to the Mathworks website for instruction on the installation of Matlab and Simulink.

The following toolboxes are needed in addition to the base install of Matlab+Simulink:

- Parallel Computing Toolbox
- MATLAB Parallel Server
- Polyspace Bug Finder
- Simulink Coder
- Simulink Compiler
- If on Linux a supported compiler

Getting PIGLE

The permanent record of PIGLE is stored in a Zenodo repository while the up-to-date development version is kept on github. At present it is recommended to use the version on GitHub and to use the branch `develop`. Click on the dropdown menu to see the list of available branches to choose `develop`.



moved config_job_params.m to UI/ 3 years ago
Normalise with respect to the number of particles 13 months ago
Rename aux --> aux_files, and ammended make_movie 3 years ago
generatePES Update prepare_potential.m 15 months ago
pigle_sim enhanced position depended fritions. Now can do multiple particles (b… 3 years ago
sweepParams fix after moving config_job_params to be under PIGLE/UI 3 years ago
LICENSE Initial commit 4 years ago

Figure 1: Changing branch on GitHub. It is recommended to use the ‘develop’ branch.

Once on the correct branch the code may be downloaded by either clicking the “Code” dropdown menu and selectring “download zip” or by running the git command
`git clone https://github.com/na364/PIGLE.git`
followed by
`git checkout develop` if one is using the git command line.

Running PIGLE

The basic steps in running PIGLE are:

1. (optional) Generate a potential and save it as a `mat` file.
2. Configure the parameters for your adsorbates in the `.m` files under the subfolder “UI”
3. Run the main script `run_pigle.m`
4. (optional) Use additional scripts for plotting and analysis.

When you first download PIGLE the simulation will be set up for the motion of CO molecules on a Cu(111) surface that are measured by the Cambridge He3 Spine-Echo machine. Therefore you can test run PIGLE before changing the parameters for your system (see setting parameters below).

The main script

To run a simulation simply run the `run_pigle.m` script.

Setting parameters

With the exception of the potential energy surface the parameters for a PIGLE simulation are specified in three `.m` files found in the "UI" subfolder. Modify the parameters by changing the value of Matlab variables in these files.

There are many parameters that are 'true/false' or 'on/off' options, in general these use the C convention of `1` being true/on and `0` being false/off.

List of files including parameters:

File	Purpose
`pigle_wrapper_params.m`	Non-scientific general parameters, e.g. plotting.
`pigle_ui.m`	Key scientific simulation parameters.
`pigle_ui_surface_params.m`	Parameters for the surface and the adsorbates.
`params_for_function_prepare_potential.m`	Specifying a potential for PIGLE to interpolate.
`prepare_params_for_interactions.m`	Define adsorbate-adsorbate interactions.
`params_for_pos_depended_spatial_friction.m`	Can be used to create a spatial dependence of the friction.
`params_for_pos_depended_spatial_theta_friction.m`	Can be used to create a spatial dependence of the rotational friction.
`config_job_params.m`	HPC use parameters.

Wrapper parameters

`pigle_wrapper_params.m`

The wrapper parameters are related to the running and output of the simultion, they do not change what is actually simulated. These parameters are also explained in comments in the `.m` file.

Parameter	Possible values	Description
`isISF`	1/0	should an ISF be calcuated?
`isSave`	1/0	should the results be saved?
`ISF2save`	list, 1/2	save the 1-incoherent ISF, 2-coherent
`toPlot`	1/0	should plots of the results be made
`reduceData`	2/1/0	reduce the amount of data saved (0=keep all)
`clearParams`	1/0	clear previous parameters file

Simulation parameters

`pigle_ui.m`

Here the main simulation parameters are defined.

Overall simulation parameters The first section of parameters define the overall simulation to be performed, some of these are scientific and some are practical. Key *scientific* options are should *rotations be included*, should *z motion be included*, and the *initial momentum* of the adsorbate (0 or thermal).

Parameter	Possible values	Description
`z_enabled`	1/0	include z motion of adsorbates?
`dKz_include_in_isf`	1/0	???
`theta_enables`	1/0	include adsorbate rotations?
`zero_p_init`	1/0	set the initial momentum to be 0 or thermal
`N_runs`	+ve integer	how many runs of the simulation to perform
`run_parallel`	1/0	should parallel computing be used?

Delta K & azimuth The next set of parameters specify the the ΔKs to be use in the simulation and two azimuths to simulate. The ΔKs are specified in a Matlab array of all desired ΔK in units of Å-1 and the azimuths are defined by 2 element arrays that define azimuthal directions using the crystolographic directions.

Beam PIGLE is initially set up to simulate the Cambridge He3 Spin-Echo, therefore the beam parameters, total scattering angle and beam incidence wavevector, only need to be changed if you want to simulate a different machine.

Time and simulation steps The time and simulation steps may be set, however some of them may be modified in some way by the simulation.

TODO: I don't understand these bits

Surface parameters

`pigle_ui_surface_params.m`
`params_for_function_prepare_potential.m`

Throught PIGLE the initialism **PES** is used to refer to a potentail energy surface.

There are two ways to specify the adsorbate-surface potential in PIGLE: either the potential may be generated externally, saved as a `.mat` file then imported into the correct structure; or heights of specific poits on the potential surface may be specified and then PIGLE will interpolate using Fourier components in between them.

PIGLE interpolated PES Built into PIGLE is an interpolation function to generate a PES for a close packed hexagonal surface. Four heights and two slopes are needed to specify a close packed surface, they are demonstrated on the figure below.

These six parameters need to be specified as a column vector in the file `params_for_function_prepare_potential.m` as `pot_strct(i).V`, where `pot_strct(i)` is the a Matlab structure containing all the information on the potential, `i` means we are setting the potential for the ith species on the surface, and `.V` means we are specifying the potential values.

If z motion is being considered then the parameters for a Morse potential,

$$V(r) = D_e\left[1 - e^(- a(r - r_e))\right]^2,$$

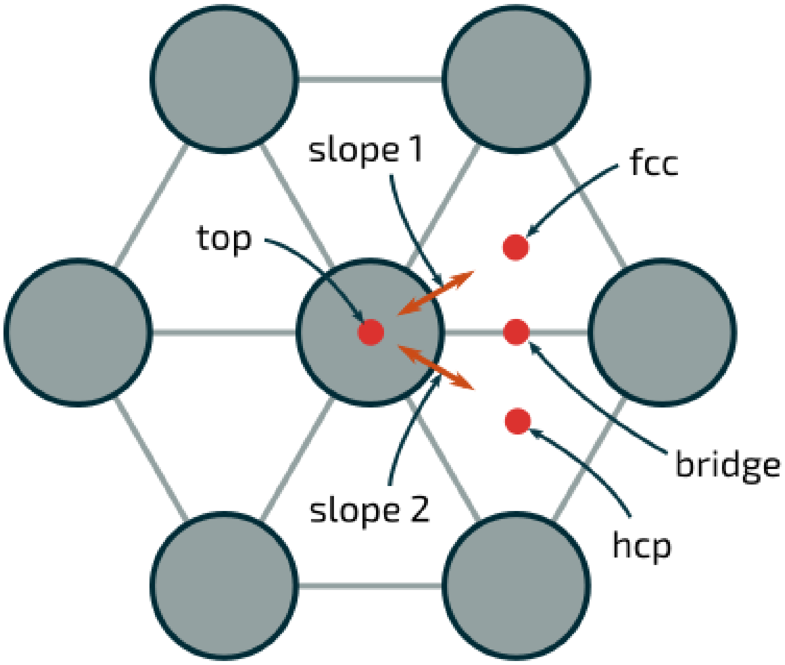


Figure 2: The locations of the six potential values needed for PIGLE to interpolate a PES.

need to be specified at each point as well as the potential height. In the specification of the potential all **potential values are in meV** and all **position values are in Angstroms**.

The total code for specifying a potential is shown below,

```
pot_strct(i).ref_De = [top; slope1; slop2; bridge; hcp; fcc];
pot_strct(i).V = [top; slope1; slop2; bridge; hcp; fcc];
pot_strct(i).a = [top; slope1; slop2; bridge; hcp; fcc];
pot_strct(i).r_e = [top; slope1; slop2; bridge; hcp; fcc];

pot_strct(i).is_potval = [1 0 0 1 1 1];
pot_strct(i).f_2D = @hexagonal6interp;
```

`hexagonal6interp` is the function that performs Fourier interpolation between the given potential values to produce a PES and `is_potval` should not need to be edited.

If more than one adsorbate species is being simulated then PES need to be defined for every adsorbate species and added to the array of potentials `pot_strct`.

A very similar specification to that used for the PES is used to define position dependent spatial friction in the file `params_for_pos_depended_spatial_friction.m`.

Specifying your own PES . . .

A PIGLE exercise

This is a very short exercise in running a PIGLE simulation and looking at the data. A more detailed exercise in PIGLE can be found as assignments 13 and 14 part of the Cambridge Atom Scattering Centre Educational Package, alternatively you may wish to get on with simulating your system after this brief introduction.