flowws$_{u}nit_{c}ell$

Release 0.1.0

May 05, 2020

Contents:

1 Installation **3**

2 API Documentation **5**
2.1 Usage 5
2.2 Modules 6
2.3 Indices and tables 7

Python Module Index **9**

Index **11**

`flowws-unit-cell` is a set of modules to identify crystalline unit cells. At a high level, this analysis proceeds in 4 steps:

1. Manually select a clean, single grain of well-ordered particles
2. Select three vectors specifying the periodic directions of the crystal
3. Project the observations into the unit cell and cluster the resulting coordinates
4. Detect the space group and center the system accordingly

`flowws-unit-cell` implements this workflow interactively in the desktop or jupyter notebook as a set of modules using flowws-analysis.

CHAPTER 1

Installation

Install `flowws-unit-cell` from PyPI:

```
pip install flowws-unit-cell
```

Alternatively, install it from source:

```
pip install git+https://github.com/glotzerlab/flowws-unit-cell.git#egg=flowws-unit-
↪cell
```

CHAPTER 2

API Documentation

Browse more detailed documentation online or build the sphinx documentation from source:

```
git clone https://github.com/glotzerlab/flowws-unit-cell
cd flowws-unit-cell/doc
pip install -r requirements.txt
make html
```

2.1 Usage

The modules provided in this project are designed to be used as part of a live flowws-analysis workflow, sandwiched between data sources and interactive backends and/or data sinks. Consult the flowws documentation or the flowws-examples project for information on how to create and execute workflows.

A suggested workflow:

- Read in data (`flowws_analysis.Pyriodic` for simple test data, otherwise read in a file with `flowws_analysis.GTAR` or `flowws_analysis.Garnett`)
- Select a clean crystalline grain (`flowws_analysis.Selection` and `flowws_analysis.Plato`)
- Compute the RDF (`flowws_freud.RDF`)
- Select directions and distances to form the basis of the unit cell (*`flowws_unit_cell.BasisSelection`*)
- Project all particles into the smaller unit cell and cluster (*`flowws_unit_cell.Projection`*)
- Center the system according to the automatically-detected space group (*`flowws_unit_cell.CenterSpaceGroup`*)
- Visualize the resulting unit cell (`flowws_analysis.Plato`)
- Display plots and add interactivity (`flowws_analysis.ViewNotebook` or `flowws_analysis.ViewQt`)
- Optionally, save the final unit cell (`flowws_analysis.SaveGarnett`)

2.2 Modules

class flowws_unit_cell.**BasisSelection**(**args*, ***kwargs*)

Select directions and distances to form the basis vectors for a unit cell.

This stage produces two visuals: a bond orientational order diagram that can be used to select symmetric directions, and a "cylindrical RDF" that measures bonds along each given direction. Together, these can be used to select the direction and length of the three basis vectors for the unit cell.

Parameters

- **orientations** – Quaternions specifying orientations for basis vectors (in the (0, 0, 1) direction)
- **r_max** – Maximum distance to consider for binds to select for the unit cell basis vectors
- **angle_tolerance** – Angle tolerance for selecting bonds (in degrees)
- **rdf_bins** – Number of bins to use for cylindrical RDF
- **x_direction** – Candidate direction to take as the x direction in the final unit cell
- **x_min** – Minimum distance to take bonds from the cylindrical RDF for the x basis vector
- **x_max** – Maximum distance to take bonds from the cylindrical RDF for the x basis vector
- **y_direction** – Candidate direction to take as the y direction in the final unit cell
- **y_min** – Minimum distance to take bonds from the cylindrical RDF for the y basis vector
- **y_max** – Maximum distance to take bonds from the cylindrical RDF for the y basis vector
- **z_direction** – Candidate direction to take as the z direction in the final unit cell
- **z_min** – Minimum distance to take bonds from the cylindrical RDF for the z basis vector
- **z_max** – Maximum distance to take bonds from the cylindrical RDF for the z basis vector

run(*scope*, *storage*)

Display the interactive direction- and distance-selection visuals.

class flowws_unit_cell.**CenterSpaceGroup**(***kwargs*)

Attempt to automatically detect the space group of the system and center it accordingly.

Parameters

- **minimum_distance** – Precision with which to merge points transformed by space group transformations
- **use_types** – Use type information when centering the unit cell

run(*scope*, *storage*)

Detect the space group and center the system.

class flowws_unit_cell.**Projection**(***kwargs*)

Project the system using a previously-calculated set of basis vectors.

This stage wraps the system into the box specified by a given box matrix and clusters the particles using sklearn. Either a density map or the final cluster centers can be visualized.

Parameters

- **color_scale** – Scale to adjust colors by in the reduced unit cell density map
- **minimum_distance** – Distance scale to use to cluster projected unit cell particles
- **density** – If True, plot a density map instead of clustered points

run (*scope*, *storage*)
Project and cluster the particles.

2.3 Indices and tables

- genindex
- modindex
- search

Python Module Index

f

flowws_unit_cell, 6

Index

B

`BasisSelection` (*class in flowws_unit_cell*), 6

C

`CenterSpaceGroup` (*class in flowws_unit_cell*), 6

F

`flowws_unit_cell` (*module*), 6

P

`Projection` (*class in flowws_unit_cell*), 6

R

`run()` (*flowws_unit_cell.BasisSelection method*), 6
`run()` (*flowws_unit_cell.CenterSpaceGroup method*), 6
`run()` (*flowws_unit_cell.Projection method*), 6