Python Quantitative STEM simulation

- Free software: GPL V3 or later

1.1 Features

- TODO
Installation

At the command line:
$ easy_install pyqstem

Or, if you have virtualenvwrapper installed:
$ mkvirtualenv pyqstem
$ pip install pyqstem
Usage

To use PyQSTEM in a project:

```python
import qstem
```
Source Reference
Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given. You can contribute in many ways:

## 5.1 Types of Contributions

### 5.1.1 Report Bugs


If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

### 5.1.2 Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” is open to whoever wants to implement it.

### 5.1.3 Implement Features

Look through the GitHub issues for features. Anything tagged with “feature” is open to whoever wants to implement it.

### 5.1.4 Write Documentation

PyQSTEM could always use more documentation, whether as part of the official PyQSTEM docs, in docstrings, or even on the web in blog posts, articles, and such.

### 5.1.5 Submit Feedback

The best way to send feedback is to file an issue at https://github.com/msarahar/pyqstem/issues.

If you are proposing a feature:
• Explain in detail how it would work.
• Keep the scope as narrow as possible, to make it easier to implement.
• Remember that this is a volunteer-driven project, and that contributions are welcome :)

## 5.2 Get Started!

Ready to contribute? Here’s how to set up `pyqstem` for local development.

1. Fork the `pyqstem` repo on GitHub.
2. Clone your fork locally:
   
   ```bash
   $ git clone git@github.com:your_name_here/pyqstem.git
   ```

3. Install your local copy into a virtualenv. Assuming you have `virtualenvwrapper` installed, this is how you set up your fork for local development:
   
   ```bash
   $ mkvirtualenv pyqstem
   $ cd pyqstem/
   $ python setup.py develop
   ```

4. Create a branch for local development:

   ```bash
   $ git checkout -b name-of-your-bugfix-or-feature
   ``

   Now you can make your changes locally.

5. When you’re done making changes, check that your changes pass flake8 and the tests, including testing other Python versions with tox:

   ```bash
   $ flake8 pyqstem tests
   $ python setup.py test
   $ tox
   ``

   To get flake8 and tox, just pip install them into your virtualenv.

6. Commit your changes and push your branch to GitHub:

   ```bash
   $ git add .
   $ git commit -m "Your detailed description of your changes."
   $ git push origin name-of-your-bugfix-or-feature
   ```

7. Submit a pull request through the GitHub website.

## 5.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.
3. The pull request should work for Python 2.6, 2.7, and 3.3, and for PyPy. Check [https://travis-ci.org/msarahan/pyqstem/pull_requests](https://travis-ci.org/msarahan/pyqstem/pull_requests) and make sure that the tests pass for all supported Python versions.
5.4 Tips

To run a subset of tests:

```
$ python -m unittest tests.test_pyqstem
```
6.1 Development Lead

- Michael Sarahan <msarahan@gmail.com>

6.2 Contributors

None yet. Why not be the first?
History

7.1 0.1.0 (2014-07-04)

- First release on PyPI.
• genindex
• modindex
• search