While you wait…

Say hello in the #faraday-tech-series channel on the pybamm slack

Visit www.pybamm.org

Follow @pybamm_ on twitter for updates

PyBaMM

Faraday Tech Series Part 2
09/06/2020

UNIVERSITY OF OXFORD
UNIVERSITY OF MICHIGAN
UCL
Imperial College London
WMG THE UNIVERSITY OF WARWICK
THE FARADAY INSTITUTION

How to Install (Linux and Mac)

Visit www.pybamm.org/getting-started for detailed instructions

We recommend that you install into a virtual environment. On Linux and Mac:

1. python3 -m venv env
2. source env/bin/activate
3. pip install pybamm

How to Install (Windows)

Visit www.pybamm.org/getting-started for detailed instructions

We recommend that you install into a virtual environment. On Windows:

1. python -m venv env
2. env\Scripts\activate.bat
3. pip install pybamm

For advanced features in Windows (like extra solvers) we recommend the use of Windows Subsystem for Linux (instructions online)

Case Study 1

We will see how changing a lot of parameters every time clutters the code

Instead of changing all the geometric parameters in every script

```
param = pybamm.ParameterValues(chemistry=pybamm.parameter_sets.Chen2020)

param["Negative current collector thickness [m]"] = 24E-6
param["Negative electrode thickness [m]"] = 170.4E-6
param["Separator thickness [m]"] = 24E-6
param["Positive electrode thickness [m]"] = 151.2E-6
param["Positive current collector thickness [m]"] = 32E-6
param["Cell capacity [A.h]"] = 10
param["Typical current [A]"] = 10
param["Current function [A]"] = 10
```

We define a new cell parameter set that we can call for our simulations

```
my_chemistry = pybamm.parameter_sets.Chen2020
my_chemistry["cell"] = "my_new_cell"

param = pybamm.ParameterValues(chemistry=my_chemistry)
```

Case Study 2

We will show an example of changing a key parameter and observing the effect on the battery performance.

The thickness of the anode will be changed and the effect on battery capacity with be shown. Results are collected and displayed interactively in the notebook.

An example of how to process the simulations in a loop and in parallel will be shown.

Contributing

If you would like to work on something in private please get in touch!

Create an issue where new proposals can be discussed before any coding is done

Create a branch of PyBaMM on your own fork where all changes will be made

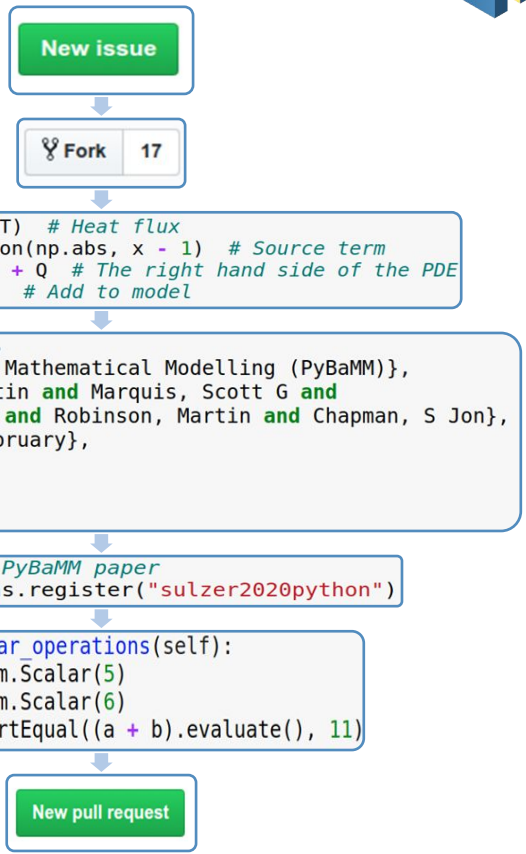
Implement your work

Add relevant references to CITATIONS.txt

Register these in the parts of the code that use your paper

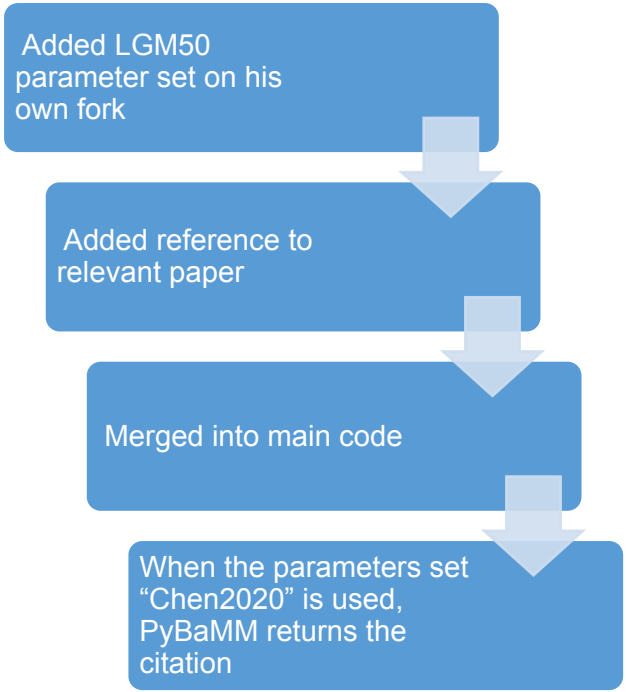
Test your code!

Create a pull request to merge into the main code when you are ready for your code to be publicly available



How to ensure your contribution is credited

Example: Ferran Brosa Planella, WMG




```python
import pybamm

model = pybamm.lithium_ion.SPMe()
param = pybamm.ParameterValues(chemistry=pybamm.parameter_sets.Chen2020)
sim = pybamm.Simulation(model, parameter_values=param)
sim.solve()

pybamm.print_citations()
```

```
@article{Chen2020,
author = {Chen, Chang-Hui and Brosa Planella, Ferran and O'Regan, Kieran and Gastol, Domini
ka and Widanage, W. Dhammika and Kendrick, Emma},
title = {{Development of Experimental Techniques for Parameterization of Multi-scale Lithiu
m-ion Battery Models}},
journal = {Submitted for publication},
year = {2020}
}
```

PyBaMM generates BibTeX entries for the papers cited in the script

Ongoing projects

- EIS and nonlinear EIS
- Degradation mechanisms (SEI, Li plating)
- Mechanics coupled to temperature and lithiation
- Other geometries (jelly rolls, packs, multiple layer cells)
- Other chemistries (LiS)
- Spatially structured electrodes
- Parameter estimation, sensitivity analysis
- Tools to analyse experimental data and characterise batteries
- Improvements to framework

Upcoming sessions:

- 11th June: IP from a commercial perspective