

PyBaMM - Python Battery Mathematical Modelling

A flexible, open-source python package for development, comparison, and fast solutions of physics-based battery models.

Valentin Sulzer, Scott G Marquis, Robert Timms, Tom Tranter,
Ferran Brosa Planella, Jacqueline Edge, Martin Robinson, Thibault Lestang, S Jon Chapman

ABSTRACT

As the worldwide battery modelling community grows, there is a clear need for software that uses modern software engineering techniques to facilitate cross-institutional collaboration on battery research. The Python package PyBaMM addresses this need by being

- Open-source and freely available
- Flexible with a modular design
- Well-tested by employing test-driven software development practices

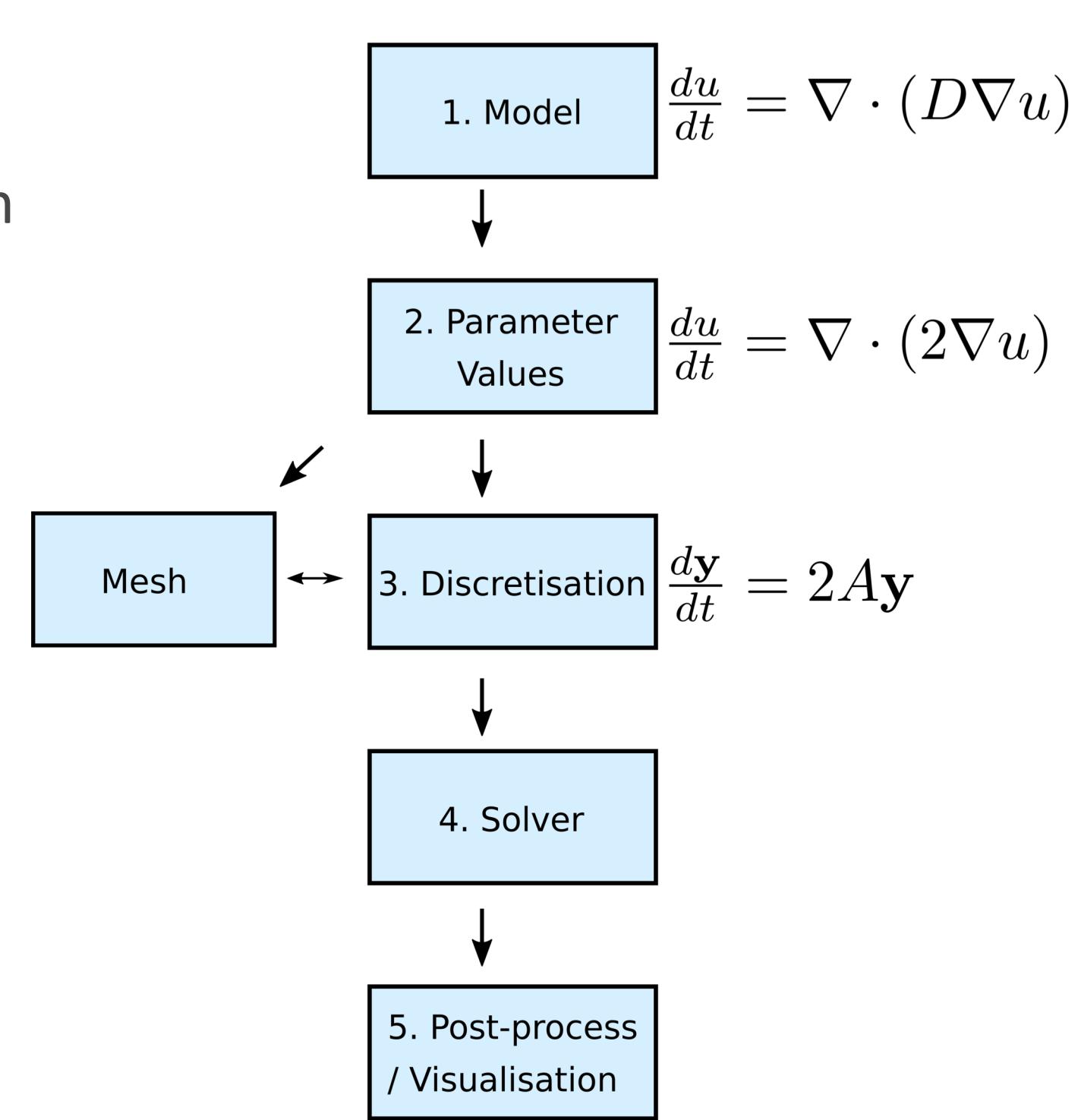
PyBaMM allows users to easily:

- Change model parameters
- Run drive cycles/standard experimental routines (CCCV, GITT, etc)
- Add additional physics (model equations entered symbolically)
- Choose from several state-of-the-art numerical solution methods (with access to analytic Jacobians)
- Achieve fast simulation times (Doyle-Fuller-Newman model with $\sim 10^5$ states solved in **under one second** for a 1C constant current discharge)

PIPELINE

By employing a pipeline process, PyBaMM separates the statement of model equations, from the inputted parameters, choice of spatial discretization, and choice of time solver. This modular approach allows:

- Model equations to be easily changed to account for new physics
- Adjoint to be calculated for parameter estimation routines
- Spatial discretizations to be switched in/out without having to rewrite the model
- Multiple time solvers to be interfaced to in a common manner



FAST SOLUTIONS

Millisecond solve times for a 1C discharge using the CasADI solver (n grid points per domain)

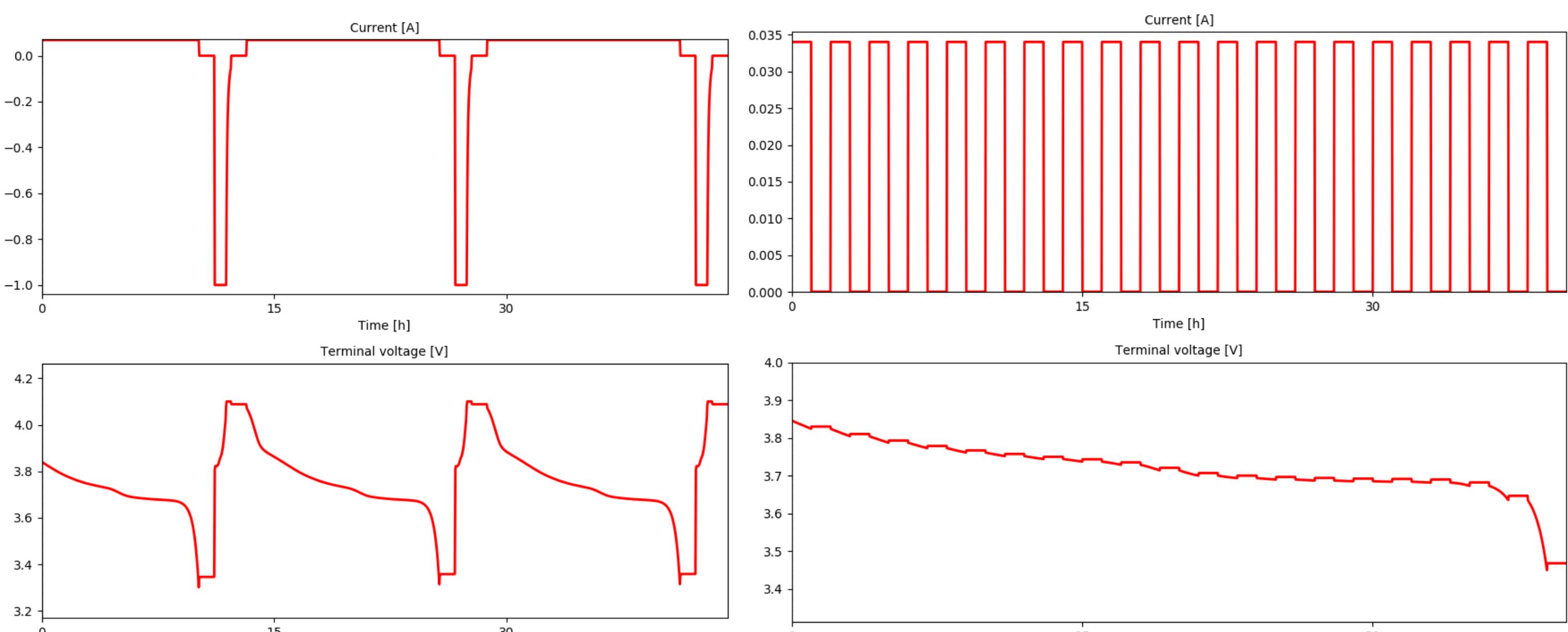
Single particle model (SPM)			Doyle-Fuller-Newman (DFN)		
Grid points	States	Time (ms)	Grid points	States	Time (ms)
10	21	3	10	281	50
20	41	3	20	961	90
40	81	4	40	3521	250
80	161	5	80	13441	930
160	321	9	160	52481	4010

CUSTOM EXPERIMENTAL PROCEDURES

PyBaMM allows users to enter custom experimental procedures in text form using the **Experiment** object:

```
pybamm.Experiment(["Discharge at C/10 for 13 hours", "Rest for 1 hour", "Charge at 1 A until 4.1 V"])
```

Below, we display the results of running CCCV and GITT routines on the DFN model and using the built-in PyBaMM plot function.



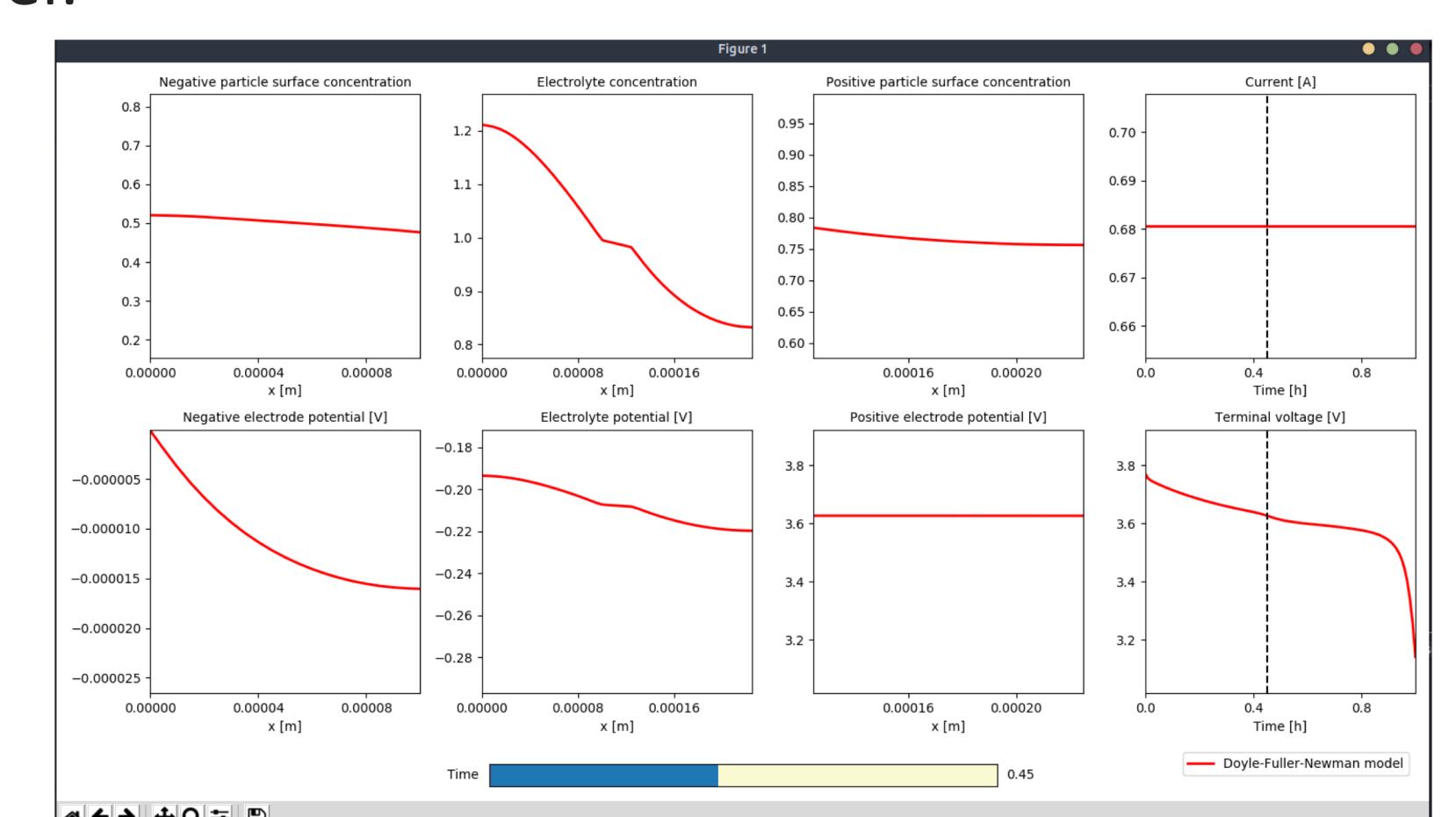
SIMPLE INSTALLATION

Just like other python packages, PyBaMM can be easily installed on **Windows**, **Mac**, and **Linux** with just one command:

```
pip install pybamm
```

With just 5 lines of python code, we can run, solve, and plot the results for a 1C discharge of the classical Doyle-Fuller-Newman Model:

1. `import pybamm`
2. `model = pybamm.lithium_ion.DFN()`
3. `sim = pybamm.Simulation(model)`
4. `sim.solve()`
5. `sim.plot()`



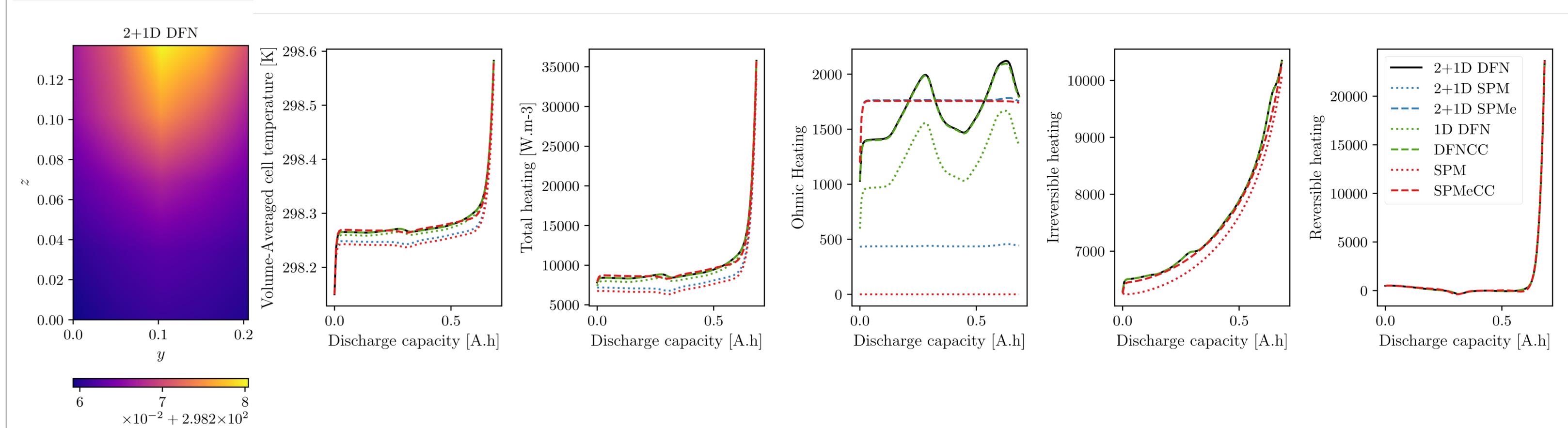
Flexible interactive plotting interface

THERMAL MODELLING OF LITHIUM-ION POUCH CELLS

The inbuilt modelling capabilities of PyBaMM includes:

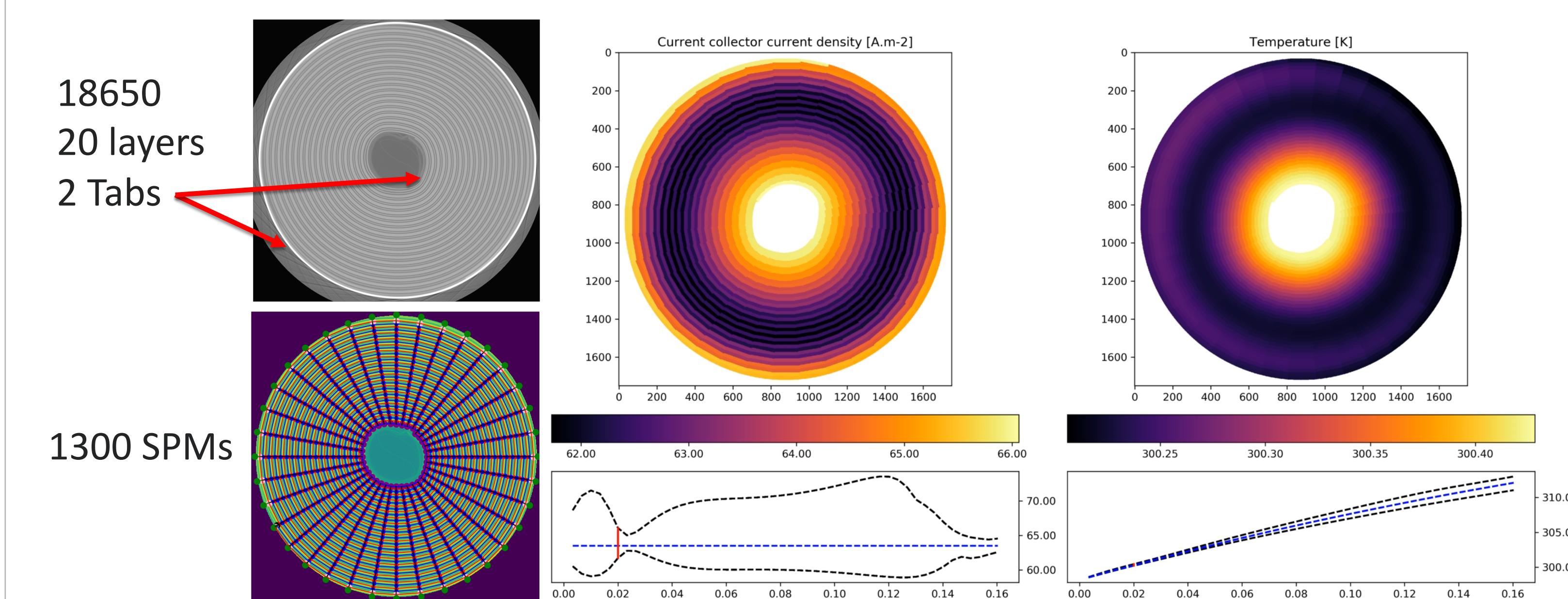
- Full-order (Doyle-Fuller-Newman) and reduced-order (e.g. Single Particle Model) lithium-ion battery models
- Fully-coupled thermal extensions to these models
- Versions of models in 1D, 2D, and 3D

Thermal results for an array of PyBaMM models of a single-layer pouch cell are shown below.



MODELLING OF JELLYROLL CELLS

- Tomography images are used to define equivalent circuit models with flexible placement of tabs and heat sources and sinks. PyBaMM is coupled with other software to compute equivalent resistances and generate heat sources for electrochemical and thermal analysis.



REFERENCES

- V. Sulzer, S.G. Marquis, R. Timms, M. Robinson and S.J. Chapman. Python Battery Mathematical Modelling (PyBaMM). *Submitted to Journal of Open Research Software*.
- S.G. Marquis, V. Sulzer, R. Timms, C.P. Please, and S.J. Chapman. An asymptotic derivation of a single particle model with electrolyte. *Journal of The Electrochemical Society*, 166(15):A3693–A3706, 2019.
- V. Sulzer, S.J. Chapman, C.P. Please, D.A. Howey, and C.W. Monroe. Faster lead-acid battery simulations from porous-electrode theory: II. Asymptotic analysis. *Journal of The Electrochemical Society*, 166(12):A2372–A2382, 2019