PVLIB Python 2015

William F. Holmgren[1], Robert W. Andrews[2], Antonio T. Lorenzo[3], Joshua S. Stein[4]

[1]Dept. of Atmospheric Sciences, Univ. of Arizona, Tucson, AZ; [2]Heliolytics, Toronto, ON;
[3]College of Optical Sciences, Univ. of Arizona, Tucson, AZ; [4]Sandia National Laboratories, Albuquerque, NM

Introduction to PVLIB

The PVLIB Toolbox is a well-established MATLAB library for photovoltaic modeling and analysis. It was originally developed at Sandia National Laboratories and has been expanded by contributions from members of the Photovoltaic Performance and Modeling Collaboration (PVPMC).

Why port PVLIB to Python?

Python is elegant, easy to read and write, portable across platforms, free and open source, and it has a large scientific computing community. The scientific Python stack enables the use of a single language for the entire data collection, processing, and analysis workflow, which can result in faster development with fewer bugs. Python provides a powerful alternative to MATLAB and R.

Andrews et. al. introduced the PVLIB-Python toolbox in 2014 and outlined its three main principles:

- Take advantage of the Python programming language, to ensure free access to academic and commercial users.
- Designed for collaborative development; backed by a rigorous method to include the contributions of authors and researchers.
- Backed by a full testing and validation suite to ensure stability of the package and to allow for validation of model results against real-world performance data.

The PVLIB-Python source code is hosted on GitHub.

Getting Started with PVLIB

1. Install the Anaconda Python distribution.
2. Create a new conda environment:
   `conda create –n pvlib`
3. Activate the environment:
   `source activate pvlib`
4. Install pvlib
   Latest release: `pip install pvlib`
   Development: `pip install git+https://github.com/pvlib/pvlib-python.git`
5. See documentation and tutorials for help and examples

Examples of PVLIB-Python models

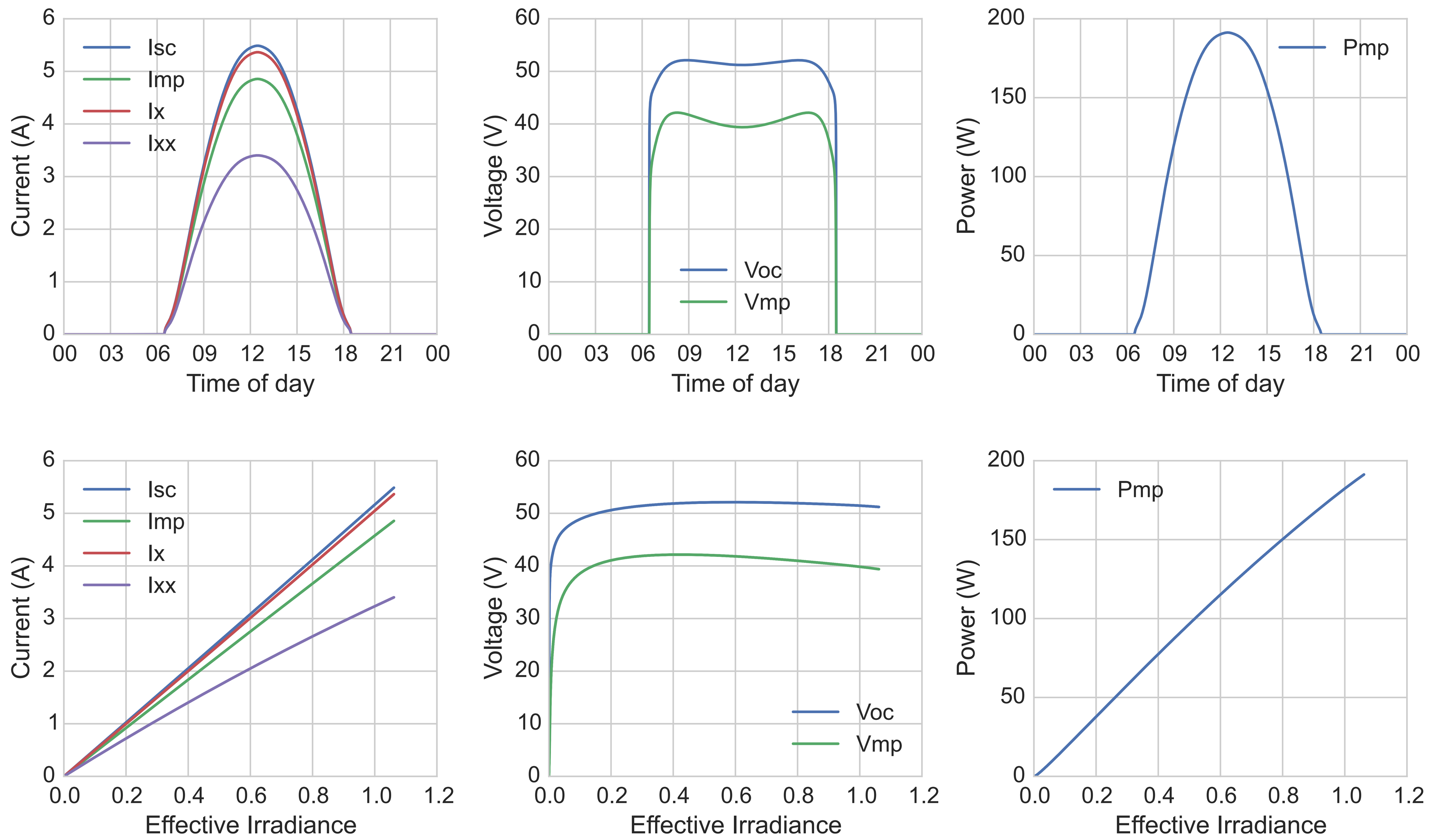

Fig. 1: Simulation of a fixed-tilt PV system on April 1, 2015, in Tucson, AZ, using PVLIB-Python's implementation of the Sandia Array Performance Model. Currents, voltages, and maximum power are plotted as a function of time of day and effective irradiance. PVLIB-Python was used to load the Sandia Module Database from NREL's website, calculate solar position, clear sky data, airmass, cell temperature, module temperature, and finally run the Sandia Array Performance Model in 9 lines of code. Detailed simulation parameters may be found at this poster's GitHub repository: https://github.com/pvlib/pvsc2105

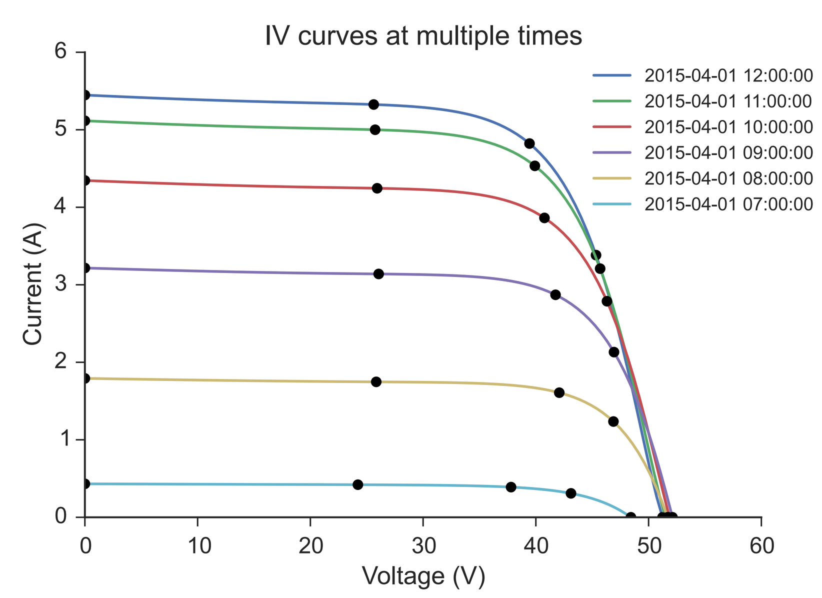

Fig. 2: IV curves at different times using the same parameters as Fig. 1. Points represent the 5 points of the SAPM: Isc, Voc, Pmp, Ix, Ixx.

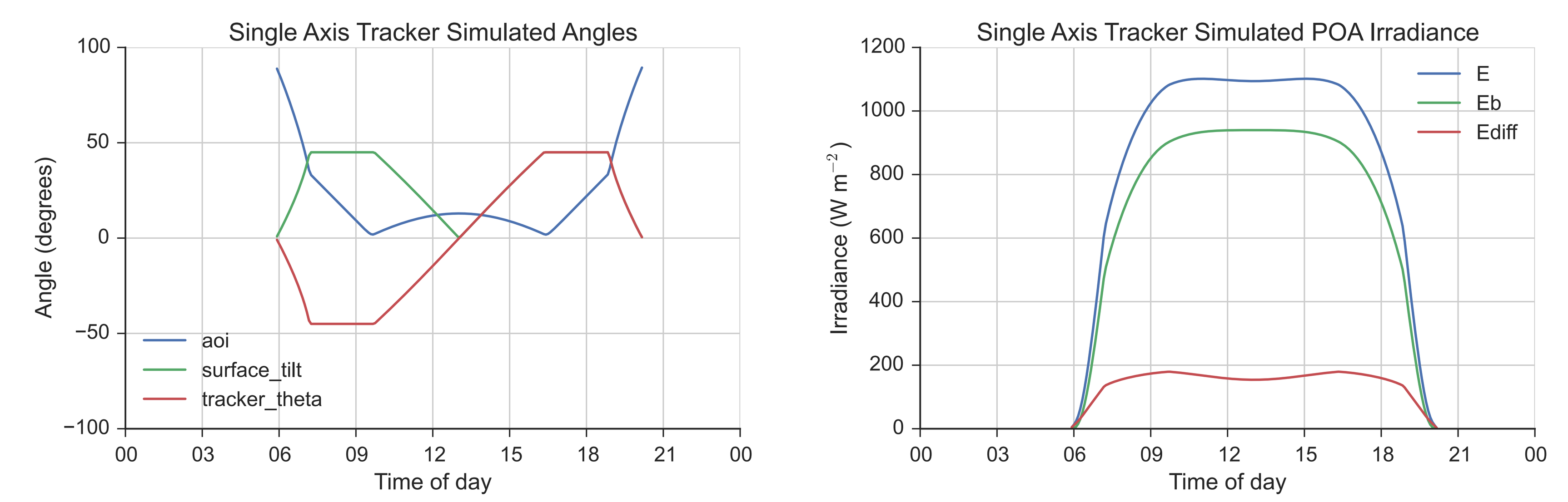

Fig. 3: PVLIB-Python simulation of a single axis tracker, with backtracking, located near Albuquerque, NM, for June 1, 2015. This example simulation used the Ineichen model to generate clear sky DNI, GHI, and DHI, the Hay-Davies model to generate the diffuse plane of array irradiance, and an isotropic ground diffuse model with an albedo of 0.25.

Acknowledgements

The authors gratefully acknowledge Sandia National Laboratories for the initial development of PVLIB-MATLAB and PVLIB-Python and the ongoing contributions of many others to the project. A list of PVLIB-Python contributors may be found on the GitHub repository and in the online documentation. WFH thanks the Department of Energy (DOE) Office of Energy Efficiency and Renewable Energy (EERE) Postdoctoral Research Award for support. ATL thanks the University of Arizona Renewable Energy Network for support.

42nd IEEE Photovoltaics Specialists Conference
June 14-19, New Orleans, LA

holmgren@email.arizona.edu
https://github.com/pvlib/pvlib-python