pvlib python 2022 update

Kevin Anderson (NREL)
Will Holmgren (DNV)
Cliff Hansen (Sandia)
Mark Mikofski (DNV)
Adam R. Jensen (DTU)
Anton Driesse (PV Performance Labs)

PV Performance Modeling and Monitoring Workshop
Salt Lake City, Aug 23, 2022
## Contents

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>What is pvlib?</td>
</tr>
<tr>
<td>2</td>
<td>New model functions and ModelChain features</td>
</tr>
<tr>
<td>3</td>
<td>New weather data sources</td>
</tr>
<tr>
<td>4</td>
<td>Documentation updates</td>
</tr>
<tr>
<td>5</td>
<td>Automated testing</td>
</tr>
<tr>
<td>6</td>
<td>Community growth</td>
</tr>
<tr>
<td>7</td>
<td>Future: pvlib 1.0</td>
</tr>
</tbody>
</table>
What is pvlib?

A python library for PV performance modeling that is **community-driven, free, open-source, and well-documented**

**Modeling Toolbox**

Stand-alone models for:
- Atmosphere
- Solar position
- Transposition
- Bifacial
- Temperature
- Clear-sky
- Snow
- Soiling
- Shading
- I-V curves
- Inverters
- IAM
- ...and more!

**System Energy Yield**

Weather-to-power following the PVPMC workflow

- Customizable end-to-end PV system modeling (ModelChain)
- Completely scriptable and automatable by design

**Data I/O**

Batteries-included data import:
- TMY
- EPW
- NSRDB
- PVGIS
- CAMS
- ECMWF MACC
- SURFRAD
- SOLRAD
- MIDC
- BSRN
- UO SRML
- NOAA USCRN

Online documentation: [https://pvlib-python.readthedocs.io](https://pvlib-python.readthedocs.io)
New models (v0.7.0 – v0.9.2)

**pvlib.bifacial**
- infinite_sheds

**pvlib.iam**
- martin_ruiz
- martin_ruiz_diffuse
- marion_diffuse

**pvlib.temperature**
- faiman
- fuentes
- ross
- noct_sam
- prilliman transient model

**pvlib.snow**
- Marion model

**pvlib.soiling**
- Kimber model
- Humboldt State model

**pvlib.shading**
- sky_diffuse_passias

**pvlib.inverter**
- fit_sandia
- sandia_multi
- pvwatts_multi

**pvlib.ivtools**
- fit_sde_sandia
- fit_sdm_cec_sam
- fit_sdm_desoto
- fit_pvsyst_sandia
- fit_desoto_sandia

**pvlib.spectrum**
- spectrl2

**pvlib.scaling**
- Wavelet variability model

**pvlib.tracking**
- slope-aware backtracking

Major refactor for non-homogeneous systems, e.g.:

- **Different subarray orientations:**
  - Multiple orientations (e.g. rooftop)
  - Part tracking, part fixed-tilt

- **Different subarray electrical properties:**
  - e.g. part 300W, part 305W

- **New classes:**
  - Array
  - FixedMount, SingleAxisTrackerMount

Thanks to Will Vining!
[https://github.com/wfvining](https://github.com/wfvining)
Google Summer of Code 2021: pvlib.iotools

GSoC project adding data import for:

- **BSRN**: global ground station network (WRMC)
- **CAMS**: satellite-based irradiance
- **McClear**: satellite-based clear-sky irradiance
- **PVGIS**: various satellite/reanalysis data
- **MERRA2**: reanalysis weather (NASA)
- **ERA5**: reanalysis weather (ECMWF)

- Participants work with an open-source organization on a 10 week programming project
- Good fit for students, but any open-source beginner can apply
- [https://summerofcode.withgoogle.com/](https://summerofcode.withgoogle.com/)
- Get in touch for 2023!

Thanks to Adam Jensen! [https://github.com/adamrjensen/](https://github.com/adamrjensen/)
Other Code Changes (v0.7.0 – v0.9.2)

- pvlib.forecast was deprecated in v0.9.1
  - To be removed altogether in some future release

- Partial model reorg by modeling topic
  - pvlib.iam, pvlib.inverter, etc

- Consistency improvements, especially in pvlib.iotools

... and many other enhancements and bug fixes

pvlb “cookbook” -- small self-contained scripts for various modeling tasks, intended as a starting point for your own code.

Wanna make cool plots like this one? Go check out the example gallery!

Each model function has a page with:
- Brief model description
- Inputs: description, data types, units
- Outputs: description, data types, units
- Published reference(s) for the model
- Links to other relevant functions
- Links to relevant gallery examples
- Other notes as needed

Several hundred model-level pages, all built automatically from in-code documentation!
Interactive tutorials for PV system performance:
- Modeling concepts
- Implementation in pvlib
- Previous tutorials at PVSC and PyData Global

The next one is here, tomorrow afternoon!
Led by Silvana Ovaitt, don’t miss it!

PVSC 2021
Source material: https://github.com/PVSC-Python-Tutorials/PVSC48-Python-Tutorial

PyData Global 2021
Youtube recording: https://www.youtube.com/watch?v=sweUakFc3l8
Source material: https://github.com/PVSC-Python-Tutorials/pyData-2021-Solar-PV-Modeling
**Correctness Testing**

- Comprehensive coverage of the entire package
- Essential to avoid accidentally introducing bugs in development
- pvlib has >10k LOC for testing alone

![pytest passing](https://example.com)
![codecov 97%](https://example.com)

**Speed Benchmarks**

- Monitor execution speed of test cases to detect performance regressions over time
- Uses airspeed velocity (asv): [https://asv.readthedocs.io](https://asv.readthedocs.io)
- Benchmark Results:
  - [https://pvlib-benchmarker.github.io/pvlib-benchmarks/](https://pvlib-benchmarker.github.io/pvlib-benchmarks/)
Google Group (user discussion, announcements)
- 650+ members
- Roughly quadrupled since 2019 workshop
- [https://groups.google.com/g/pvlib-python](https://groups.google.com/g/pvlib-python)

GitHub (code development)
- 700+ pull requests
- Code contributions from 80+ people
- [https://github.com/pvlib/pvlib-python](https://github.com/pvlib/pvlib-python)
This software is made possible by contributions from people like you. You can help!

**pvlib 1.0?**

**Does 1.0.0 come right after 0.9.*?**
- No. Expect 0.10.0 to come next. There is no ETA for pvlib 1.0.0 yet 😊

**What does 1.0 mean?**
- A declaration that pvlib is no longer “beta” (whatever that means)
- Mostly, no more changes that break people’s code (until 2.0, anyway)

**What needs to happen before 1.0?**
- Change all the things we want to change:
  - Package-wide consistency in naming (mostly there already, but still room for improvement)
  - Fill in some modeling gaps: transformer losses, direct shading, etc
  - Rewrite/reorg the docs to follow an intentional strategy instead of the current ad-hoc “pile of info”
- **What else? We’d love to get your feedback!** Come to the pvlib user discussion tomorrow, 3pm
Thank You

www.github.com/pvlib/pvlib-python
https://pvlib-python.readthedocs.io
NREL/PR-5K00-83420

This work was authored in part by Alliance for Sustainable Energy, LLC, the manager and operator of the National Renewable Energy Laboratory for the U.S. Department of Energy (DOE) under Contract No. DE-AC36-08GO28308. Funding provided by the U.S. Department of Energy’s Office of Energy Efficiency and Renewable Energy (EERE) under Solar Energy Technologies Office (SETO) Agreement Numbers 34348. The views expressed in the article do not necessarily represent the views of the DOE or the U.S. Government. The U.S. Government retains and the publisher, by accepting the article for publication, acknowledges that the U.S. Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for U.S. Government purposes.