

# Challenges of PV Degradation Analysis: PVLIB and Performance Data Analysis

Joshua S Stein

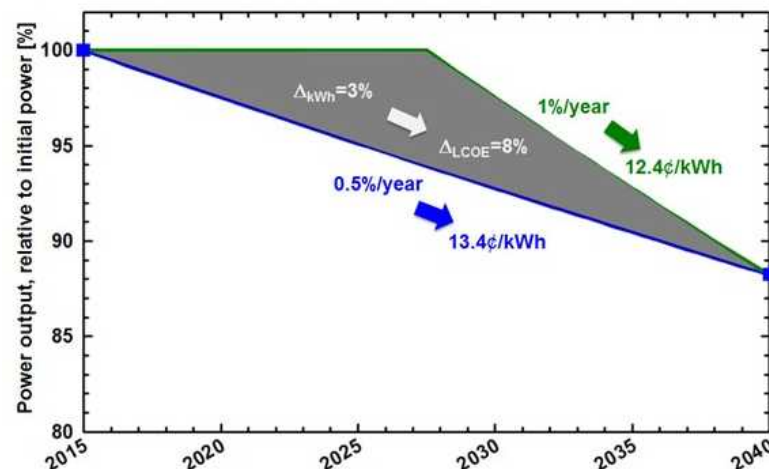
Sandia National Laboratories

# PV Degradation and Performance Modeling

- In order to reach DuraMAT goals of a levelized cost of electricity for PV of less than 3 cents per kilowatt-hour it is necessary to understand and measure PV degradation rates.
- Several of the DuraMAT capabilities are focused on data collection and analysis.
  - DataHub
  - Data Management and Analytics
  - Field Deployment
- Analyzing PV degradation is not really routine or straightforward, especially with outdoor performance data
- This talk will present a current case study underway to measure PV degradation and then introduce an open source toolbox, PVLIB, and show how it can help analyze measured PV performance data.
- Analyses shown here are preliminary and are shown only to better describe general methods used measure degradation.

# Case Study – PV Lifetime Project

- PV degradation is likely neither linear nor constant.
- This fact affects the LCOE.
- PV Lifetime project was started in 2016 to carefully track PV module degradation to better understand its progression over time.
- Common and representative PV modules are purchased on the open market and sent to Sandia, NREL, and FSEC.
- Initial indoor characterization is performed (STC flash testing)
- String-level IV curves are automatically and periodically measured
- Annual re-characterization is performed on sample of modules.
- 605 Modules are currently included in the program. More on the way



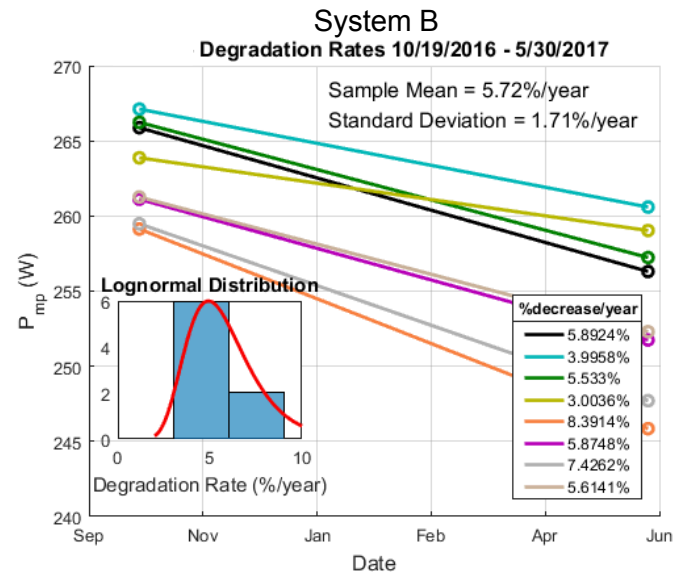
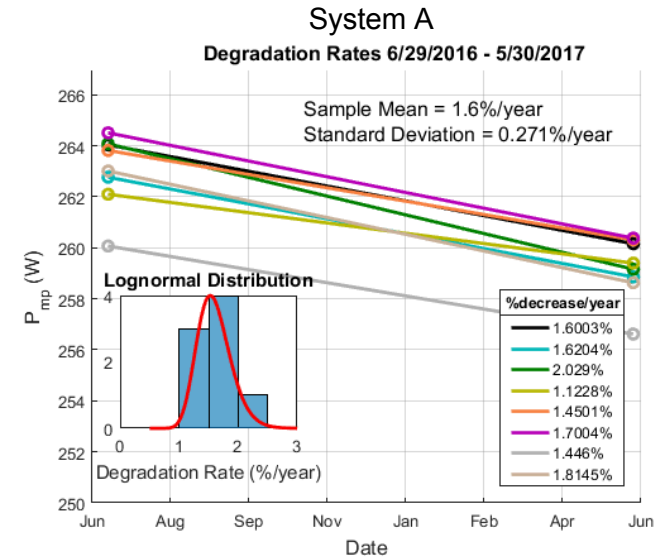
# Test Systems in NM

- System A (4 strings)
- System B (4 strings)
  
- Plane of array irradiance is measured
- IV curves are measured with a Pordis 140A Series II 8-32 Channel In-Line String-level I-V Tracer.
  - Curves taken every 30 min
  - Can be triggered on irradiance, temperature, or other signals (speaks MODBUS)
- DC & AC Current and Voltage measured from inverter every 15 min
- Other nearby systems being monitored for comparison.



# Example Flash Test Results (NM)

- All modules were flash tested at the beginning of the project (after  $\sim 20\text{kWh/m}^2$  of light exposure).
- A sample of 8 modules were reflashed after 7-12 months in the field.
  - System A:  $P_{mp}$  went down  $\sim 1.6\%$  in 1 yr
  - System B:  $P_{mp}$  went down an average of 3.3% from initial flash. If this degradation is constant it would equal  $\sim 5.7\%/yr$ .
- System degradation should be higher than module degradation due to mismatch
  - Lowest performing modules in each string drag others down.

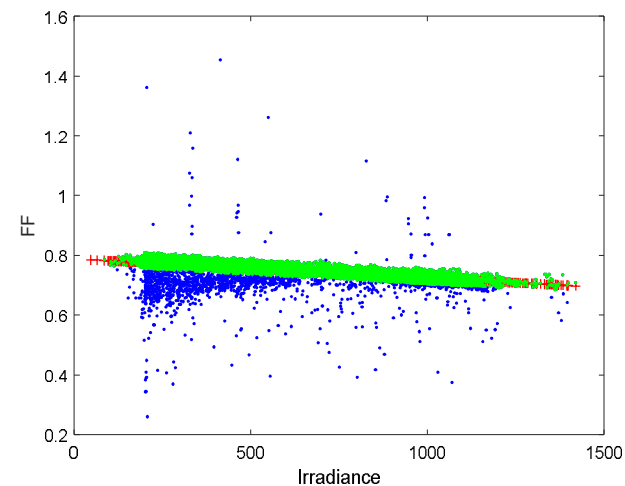


# Outdoor IV Curve Analysis (NM)

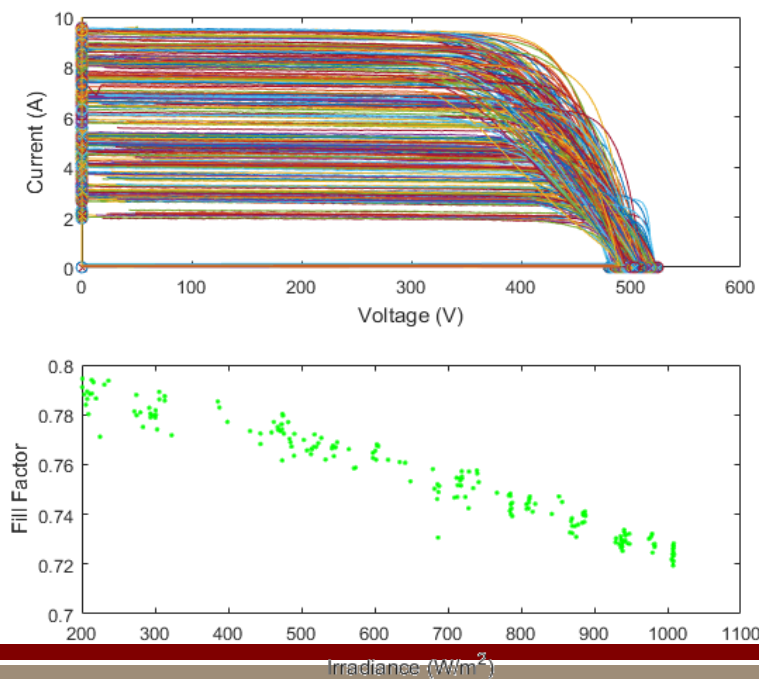
- IV curve measurements began several months after systems were connected.
- IV curves are easy to measure but hard to interpret.
  - They take several seconds to measure and irradiance can vary in that time
  - Capacitive tracers do not go to 0 V or  $I_{sc}$ . There can be noise at low voltage values
    - $I_{sc}$  and  $V_{oc}$  should be calculated by extrapolating the curve (avoiding any noise).
- Shading or soiling can have large effects.
- “Bad” IV curves must be filtered
  - **Fill Factor** is a fairly reliable data quality filter.

# Fill Factor Filter

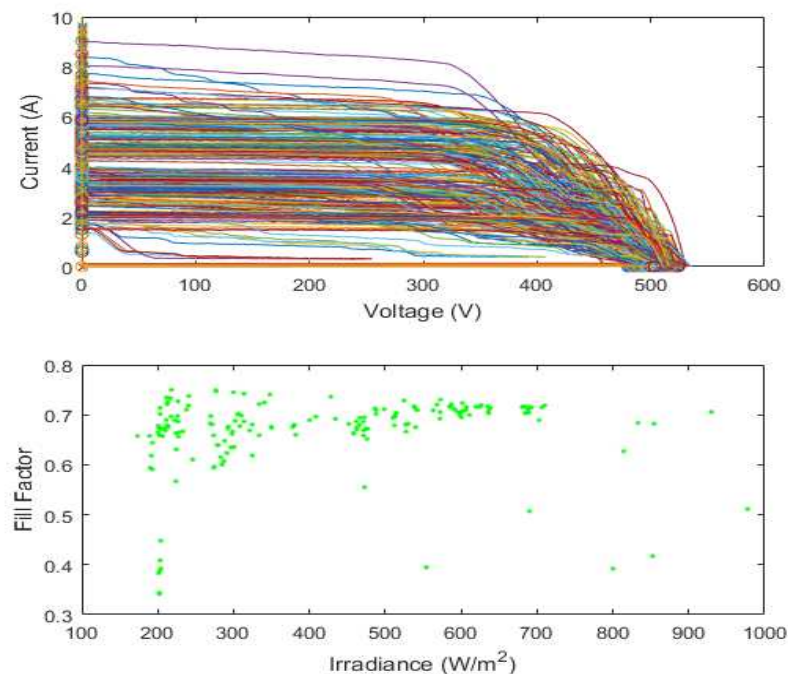
- Fill factor tends to decrease with irradiance.
  - Filtering on this trend helps to remove spurious measurements
    - Shading, irradiance variability



200 IV curves filtered in by FF

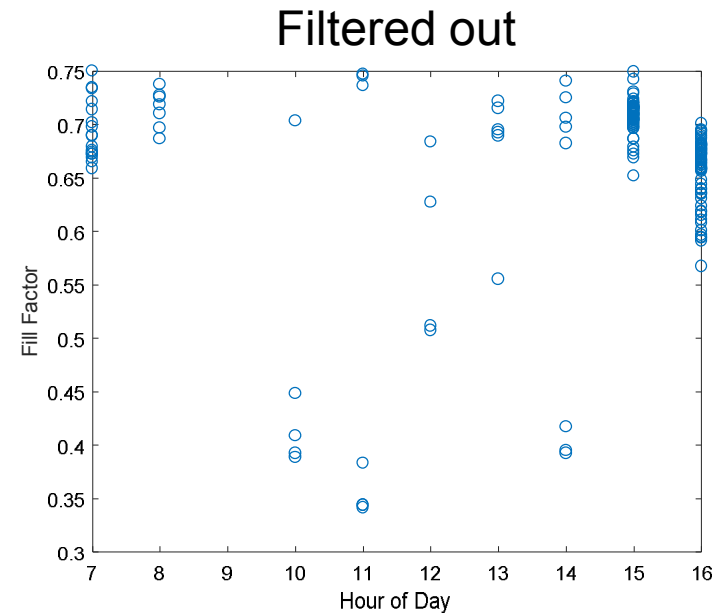
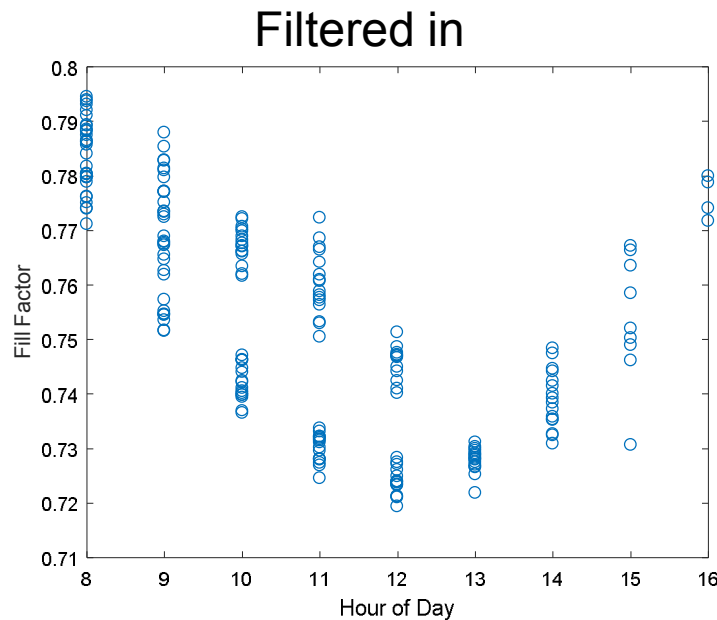


200 IV curves filtered out by FF





# Cause of Fill Factor Deviations?



- Most of the curves that are filtered out are from the start and end of the day
  - This is a time when the arrays are partially shaded.
- A few curves are filtered out in the middle of the day
  - Variable irradiance during trace or human caused shade

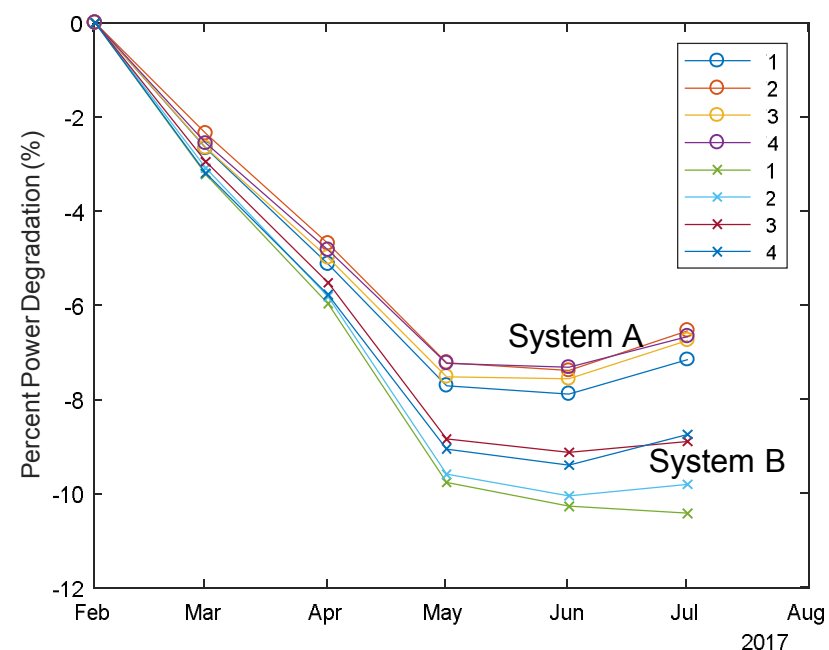
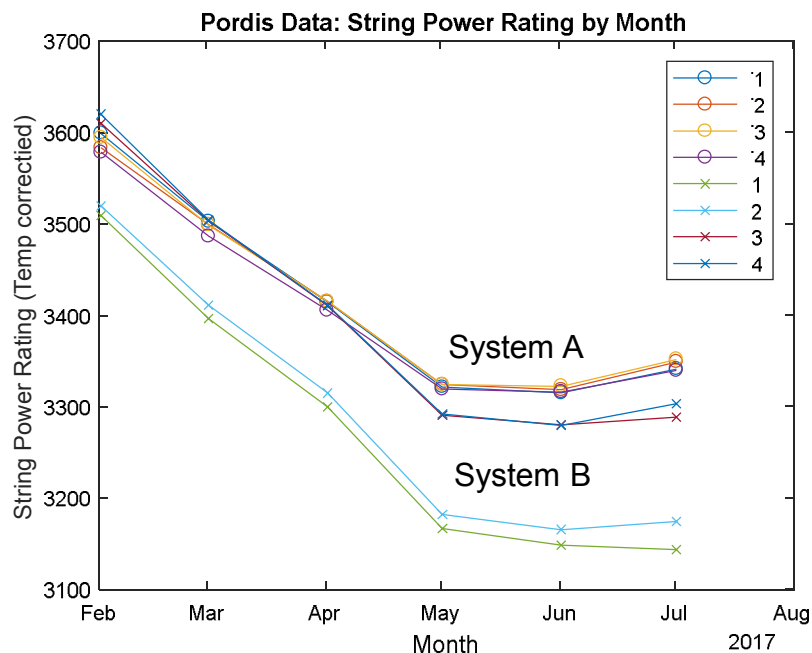


# Example Degradation Calculation

- $P_{mp}$  is calculated for each curve (filtered in)
- Module temperature is used to correct  $P_{mp}$  to 25 deg C
  - $P_{mp,25C} = \frac{P_{mp,meas}}{1 + \frac{\gamma_{Pmp}}{100}(T_m - 25)}$ , where  $\gamma_{Pmp}$  is from spec sheet.
- $P_{mp,25C}$  vs. POA irradiance is fit with a line for a given period (e.g., month)
  - Intercept at 1,000 W/m<sup>2</sup> is taken as “string rated power”
- This rated power is plotted over time
- Similar analyses of  $I_{sc}$ ,  $V_{oc}$ ,  $R_s$ ,  $R_{sh}$  are helpful for identifying the source of the degradation (Not covered in this talk).

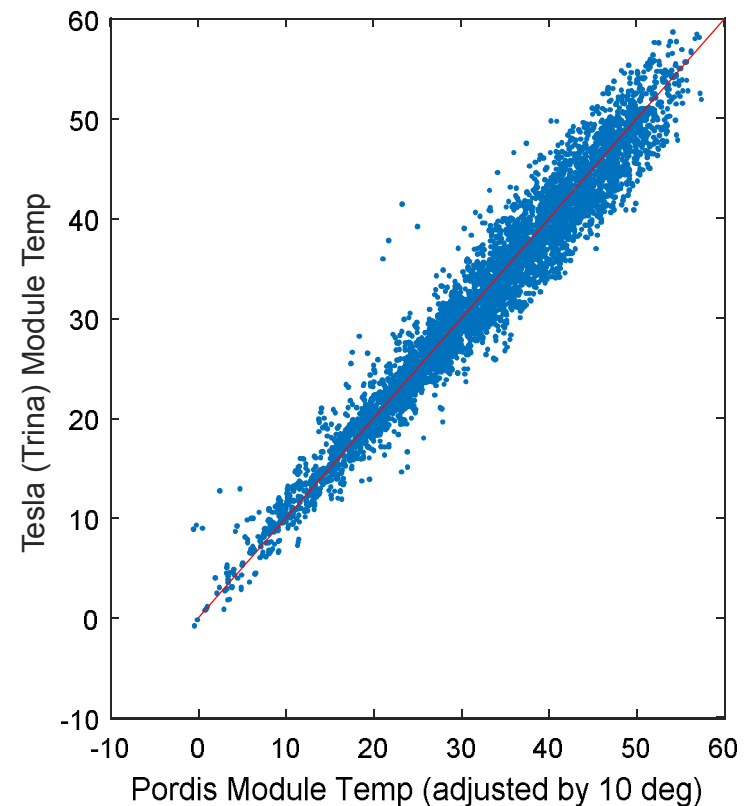
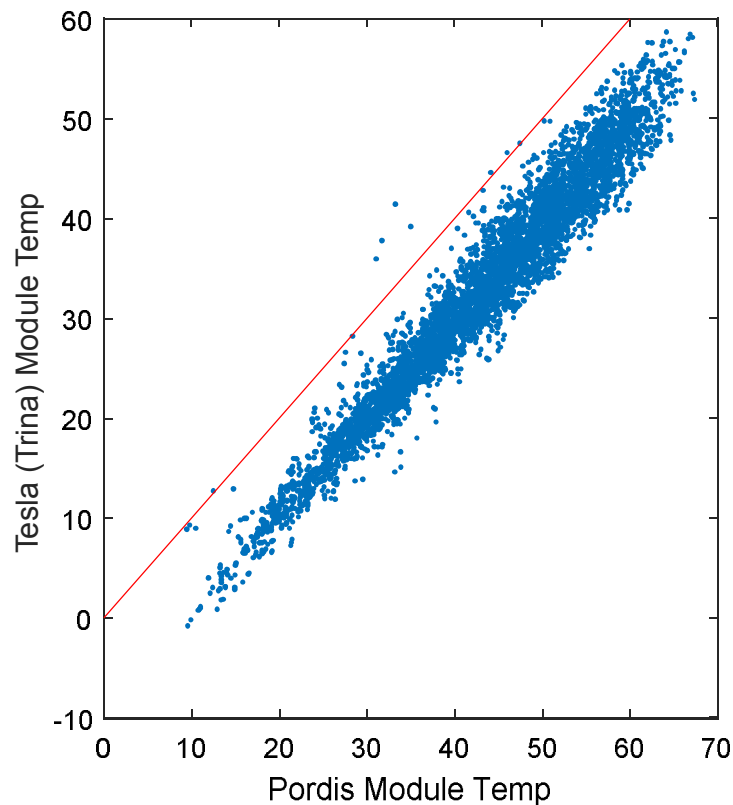
# Preliminary Monthly String Degradation Results

- Monthly string degradation appears to have proceeded until May 2017 when it stops.
  - Systems A and B appear to degrade in a similar pattern.
  - This is suspicious! We must validate this data in case we are observing IV tracer or other input data degradation!



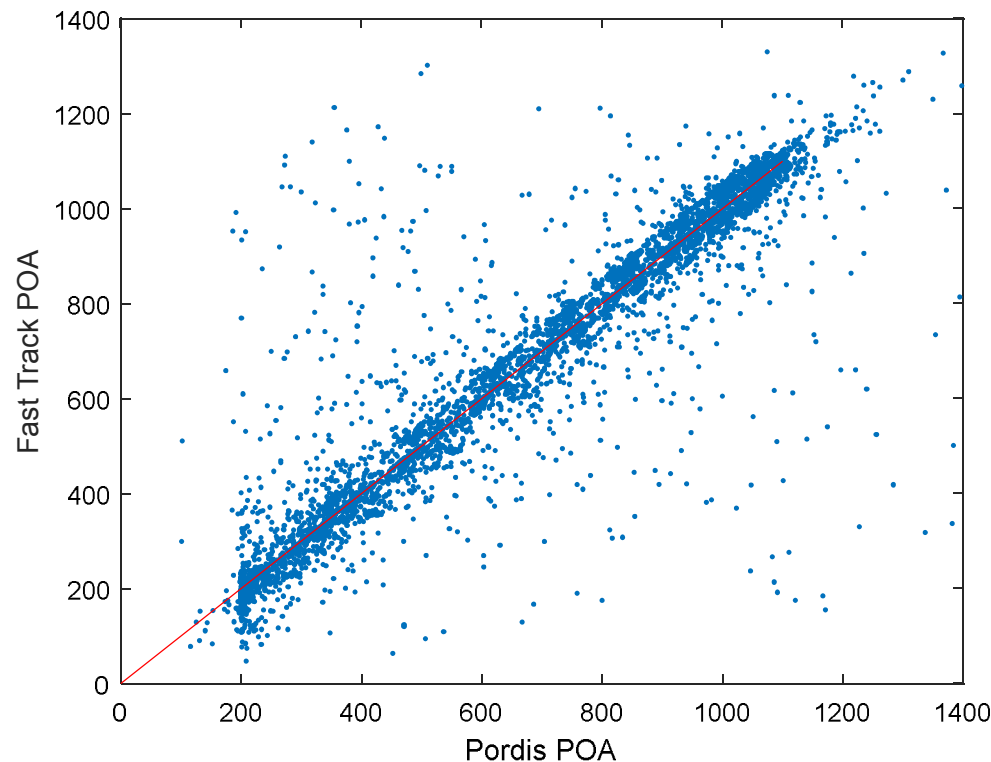
# Check Module Temperatures

- Module temperature data is actually from an uncalibrated reference cell.
  - 10° C offset is apparent. This correction was used in the Pmp temp correction.



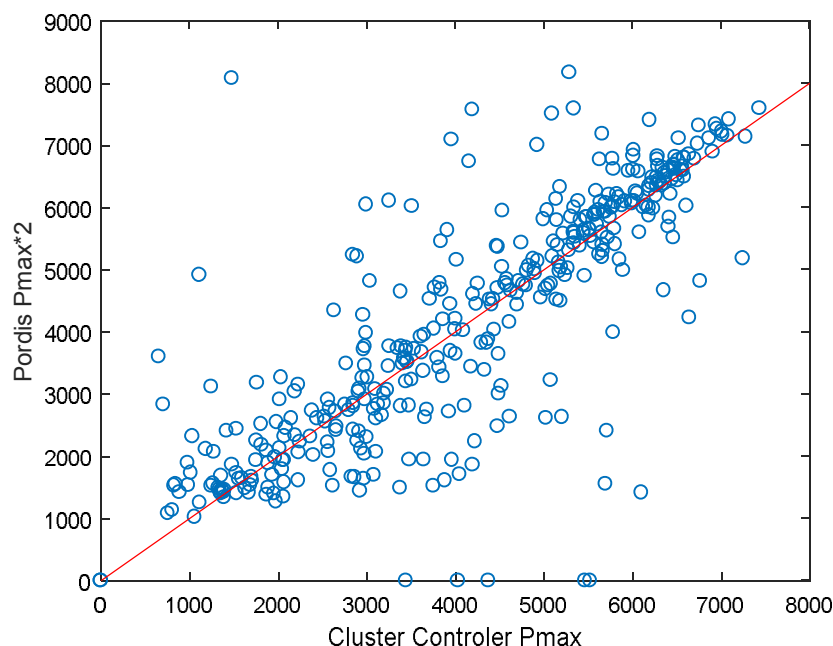
# Check POA Irradiance

- POA irradiance was compared with a calibrated reference cell at same orientation located about 200 m away.
  - Nice match! Little to no bias error.
  - Scatter is commonly observed and due to discrete cloud shadows



# Check Power Readings

- DC current and voltage are measured by the inverter every 15 min by an SMA cluster controller.
- I matched Pordis times with cluster controller times and compared Pmp values
  - Cluster controller may report time averaged values, which would cause scatter.
  - No bias error is evident!



## Preliminary Conclusion

- IV Data appears to be legitimate. Additional checks will be performed before these results will be released publically.

# Alternate Methods of Measuring Degradation

- PVUSA Model fits PV AC output to a function of POA irradiance, ambient temperature (T), and windspeed (WS)

- $P_{AC} = POA(a + b * POA + c * WS + d * T)$

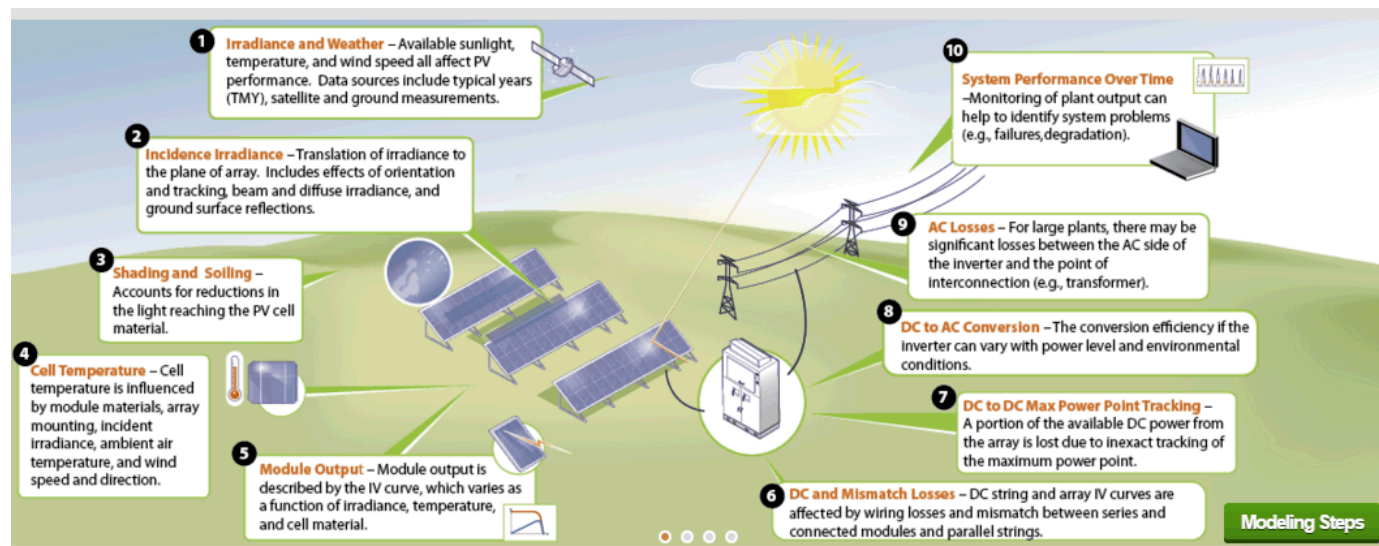
- “MPM” Model [1] based on knowledge of “sensible behaviours” of PV cells and modules.

- $PR_{DC} = c_1 + c_2(T_{mod} - 25) + c_3 \log(POA) + c_4 POA + c_5 WS + \frac{c_6}{POA}$

<u>Dependency</u>	<u>Comments</u>
1. $I_{MAX} \propto POA$	neglecting Spectral, AOI
2. $V_{MAX} \propto \log(POA)$	from single diode model
3. $T_{MOD} \sim T_{AMB} - \text{fn}(WS)$	due to NOCT/NMOT
4. $P_{MAX} \propto (1 + \gamma * (T_{MOD} - 25))$	$P_{MAX}$ temp. coefficient
5. $\Delta P_{MAX} \propto I_{MAX2} * R_{SERIES}$	Eff falls with GI2
6. $R_{SHUNT} \propto 1/\exp(G)$	varies by technology

- These can be “calibrated” each month and then run for “reference conditions” to track degradation.

# Performance Modeling Methods



- Read inputs:
  - Array design (module, string, inverter, mounting, tracking, ground cover, etc.)
  - Weather (irradiance, temperature, wind speed, etc.)
- Translate irradiance to plane-of-array (POA) (or measure directly)
  - Sun position calculation, irradiance model
- Evaluate 'effective' irradiance
  - Angle on incidence effects
  - Spectral effects (air mass correlations or physics models)
- Determine cell temperature (or measure directly)
- Calculate  $I_{mp}$ ,  $V_{mp}$ , and  $P_{mp}$
- Estimate and apply derates (soiling, DC losses, mismatch, array utilization, etc.)
- Model inverter performance ( $P_{ac}$ )
- Compare with measured data.



# PV Performance Modeling Collaborative (PVPMC)



- Model agnostic, focus on algorithms, methods, data, etc.
- Three Pillars of Communication and Collaboration
  - 1. Website** ([PVPMC.sandia.gov](http://PVPMC.sandia.gov)) (>10,000 visits per month)
    - Detailed Modeling Steps (~150 technical webpages)
    - Past workshop presentations (over 300 available for download)
    - Document library, datasets, blog, events, ...
  - 2. Open Source Software**
    - **PVLIB** - Modeling function libraries for Matlab and Python (50+ functions), BSD 3-clause licenses
    - **Wavelet Variability Model** – calculates geographic smoothing of PV plant power.
    - **GridPV** – Matlab code for analysis of distribution systems with PV
  - 3. Workshops**
    - Planning for the 9<sup>th</sup> and 10<sup>th</sup> Workshops in China and US is underway

# PVLIB for Matlab : Irradiance and Weather

- **pvl\_readtmy3, pvl\_readtmy2** – Reads TMY formatted weather files
- **pvl\_getISDdata, pvl\_readISH** – Reads data from the Integrated Surface Database (source of global weather data)
- **pvl\_ephemeris, pvl\_spa** – Calculates Sun position
- **pvl\_extraradiation** – Calculates extraterrestrial radiation
- **pvl\_alt2pres, pvl\_pres2alt** – Converts between altitude and air pressure
- **pvl\_relativeairmass, pvl\_absoluteairmass** – Calculates air mass both relative and absolute
- **pvl\_disc, pvl\_dirint, pvl\_erbs, pvl\_louche, pvl\_orgill\_Hollands, pvl\_reindl\_1, pvl\_reindl\_2** – Estimates DNI from GHI
- **pvl\_clearsky\_haurwitz, pvl\_clearsky\_ineichen** – Calculates clear sky irradiance
- **pvl\_calcPwat** – Estimates precipitable water in the atmosphere from air temperature and relative humidity

Download PVLIB for Matlab (1.3.2) at

➤ [https://pvpmc.sandia.gov/applications/pv\\_lib-toolbox/](https://pvpmc.sandia.gov/applications/pv_lib-toolbox/)

➤ [https://github.com/sandialabs/MATLAB\\_PV\\_LIB](https://github.com/sandialabs/MATLAB_PV_LIB)

Report issues, submit code, etc. via github

# PVLIB for Matlab : Incident Irradiance (POA)

- **pvl\_grounddiffuse** – Calculates ground reflected irradiance
- **pvl\_isotropicsky** – Isotropic model for sky diffuse irradiance on POA
- **pvl\_perez**, **pvl\_reindl1990**, **pvl\_kingdiffuse**, **pvl\_klucher1979**, **pvl\_haydavies1980** - models for sky diffuse irradiance on POA
- **pvl\_getaoi** – Calculates to the angle of incidence on a tilted plane.
  
- **pvl\_detect\_clear\_times** – Identify clear periods in irradiance time series
- **pvl\_detect\_shadows** – Detect the effect of shadows in measured irradiance data (e.g., overhead wires, trees, poles, etc.)

Download PVLIB for Matlab (1.3.2) at

- [https://pvpmc.sandia.gov/applications/pv\\_lib-toolbox/](https://pvpmc.sandia.gov/applications/pv_lib-toolbox/)
- [https://github.com/sandialabs/MATLAB\\_PV\\_LIB](https://github.com/sandialabs/MATLAB_PV_LIB)

Report issues, submit code, etc. via github

# PVLIB for Matlab : PV Models

- **pvl\_sapmmmoduledb, pvl\_SAMLibraryReader\_CECModules, pvl\_SAMLibraryReader\_SNLInverters** – Reads in PV module coefficients
- **pvl\_physicaliam, pvl\_martinruiziam, pvl\_ashraeiam** – Models for estimating incident angle modifiers
- **pvl\_Fsspeccorr** – First Solar model for spectral corrections
- **pvl\_calcp\_params\_Desoto & pvl\_calcp\_params\_CEC & pvl\_calcp\_params\_Pvsyst** – Calculates PV module parameters for the single diode model
- **pvl\_singlediode** – Implements the single diode model
- **pvl\_sapm** – implements the Sandia PV Array Performance Model
- **pvl\_huld** – Implements the Huld performance model
- **pvl\_snlinverter, pvl\_adrinverter** – inverter performance models
- **pvl\_singleaxis** – Calculates single axis tracker positions and angles
- **pvl\_sapmcelltemp** – Calculates PV cell temperatures from air temp, irradiance, and wind speed.

Download PVLIB for Matlab (1.3.2) at

➤ [https://pvpmc.sandia.gov/applications/pv\\_lib-toolbox/](https://pvpmc.sandia.gov/applications/pv_lib-toolbox/)

➤ [https://github.com/sandialabs/MATLAB\\_PV\\_LIB](https://github.com/sandialabs/MATLAB_PV_LIB)

Report issues, submit code, etc. via [github](#)

# pvlib-python

- Object-oriented implementation of PV modeling functions
- Location class:
  - Container for latitude, longitude, altitude, timezone data
  - Methods for solar position, clear sky models, air mass models, TMY readers
- PVSystem class:
  - Container for system description data, e.g.: tilt, azimuth, albedo, module and inverter, system topology
  - Methods for POA irradiance, module and inverter performance models
- ModelChain class:
  - Container for a Location and a PVSystem object
  - Specify choice of models using keywords, e.g., `clearsky_model='ineichen'`
  - Methods for power and energy simulations, e.g., `'prepare_inputs()'`, `'run_model()'`

Pvlib-python v0.5.0 at

➤ <https://github.com/pvlib/pvlib-python>

Report issues, submit code, etc. via github

# PVPMC Workshops

- 1<sup>st</sup> PVPMC Workshop (Albuquerque, NM, Sept. 2010)
- 2<sup>nd</sup> PVPMC Workshop (Santa Clara, CA, May 2013)
- 3<sup>rd</sup> PVPMC Workshop (Santa Clara, CA, May 2014)
- 4<sup>th</sup> PVPMC Workshop (Cologne, Germany Oct 2015)
- 5<sup>th</sup> PVPMC Workshop (Santa Clara, CA, May 2016)
- 6<sup>th</sup> PVPMC Workshop (Freiburg, Germany, Oct 2016)
- 7<sup>th</sup> PVPMC Workshop (Lugano, Switzerland, March 2017)
- 8<sup>th</sup> PVPMC Workshop (Albuquerque, NM, May 2017)
- 9<sup>th</sup> *PVPMC Workshop (Weihai, China, Dec 5-7, 2017)*
- 10<sup>th</sup> *PVPMC Workshop (Albuquerque, NM, May 1-3 2018)*

# Summary and Conclusions

- Degradation rates of PV modules and components affect LCOE and the value of solar PV.
  - Degradation is unlikely to be uniform or constant
- There are many different methods to measure degradation.
  - Indoor flash testing
  - Outdoor performance monitoring (DC or AC measurements, IV curves)
- All methods rely on a stable irradiance reference
  - Regular calibration and cleaning is important.
  - Comparison with nearby measurements is very useful too.
- Filtering of data is important
  - Transient irradiance conditions or partial shading need to be removed.
  - Fill factor seems to be a reliable filter.
- Many valuable analysis functions are available from PVLIB.



# Final Thoughts

- Analysis of PV data is always a challenge
  - What is measured is not always representative
    - E.g., soiling, shading, transient conditions all affect measurements
    - Stability of DUT and uncertainty in the measurement need to be understood
    - Accuracy vs. precision of flash testers
- Data analysis **MUST** be closely linked with the collection of the data and design of the experiment!
  - DuraMAT DataHub **will only work if** the nuances of the data are understood and transferred from the experiment designers and data collectors to the analysts.