

Solar Array Power Modeling

Don Walker, Colin J. Mann, Pilar Espinet Gonzales, Jann Grovogui, Yao Lao, Simon H. Liu

4/27/23

© The Aerospace Corporation, 2022

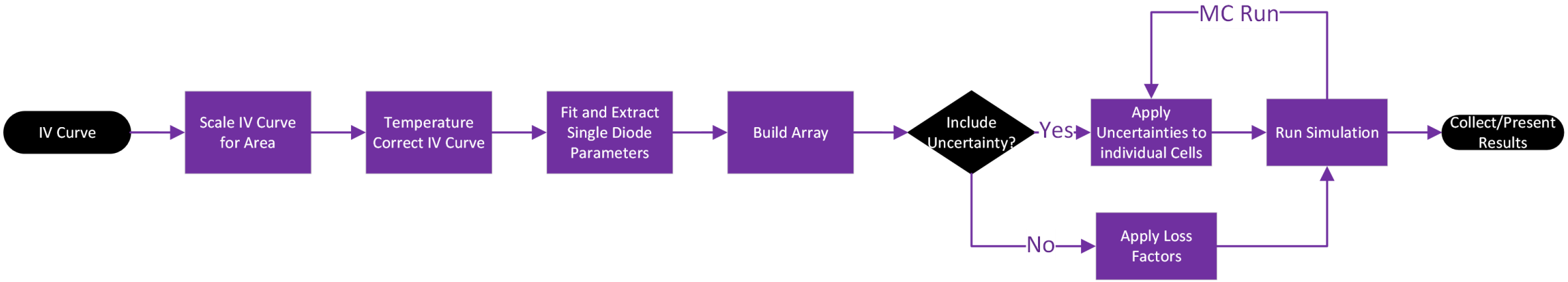
Goals

- Accurately predict array performance

- Develop standard modeling capability
  - *Best Practices*
  - *Step by step procedure*
  - *Model rooted in measured data*

- Incorporate statistical values wherever possible
  - *AE9/AP9 stats*
  - *Test and Measurement Stats*
  - *Reliability*

Array Modeling Process Flow

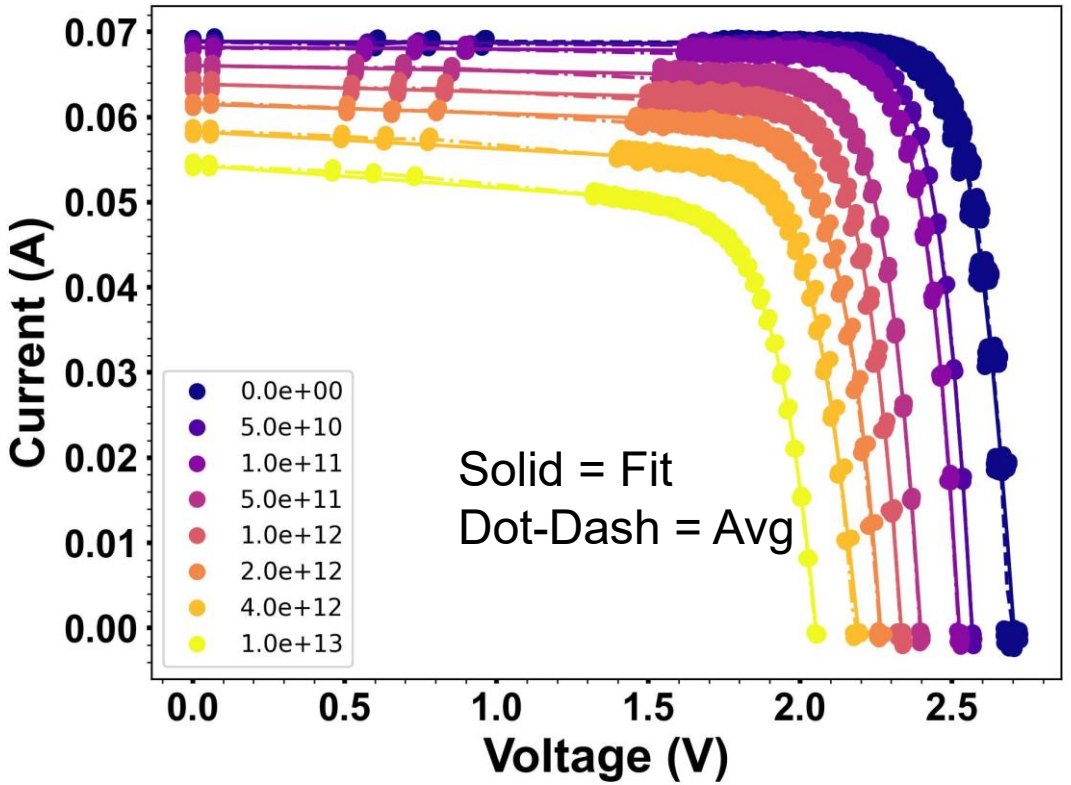


IV Curve

- Data is from Aerospace 3 MeV proton radiation tests at Auburn
  - *Data was annealed to 60C Results (results available from earlier report*

- Averaging IV curves
  1. *All IV curves grouped together*
  2. *Currents and voltages are binned using voltage bins*
  3. *Current and voltage are averaged yielding current and voltage error in the IV curve average*
  4. *Average IV curve is then fit using single diode equation*

$$I = I_{ph} - I_o e^{\left[\frac{q(V+IR_S)}{nkT}\right]} - \frac{V+IR_S}{R_{sh}}$$

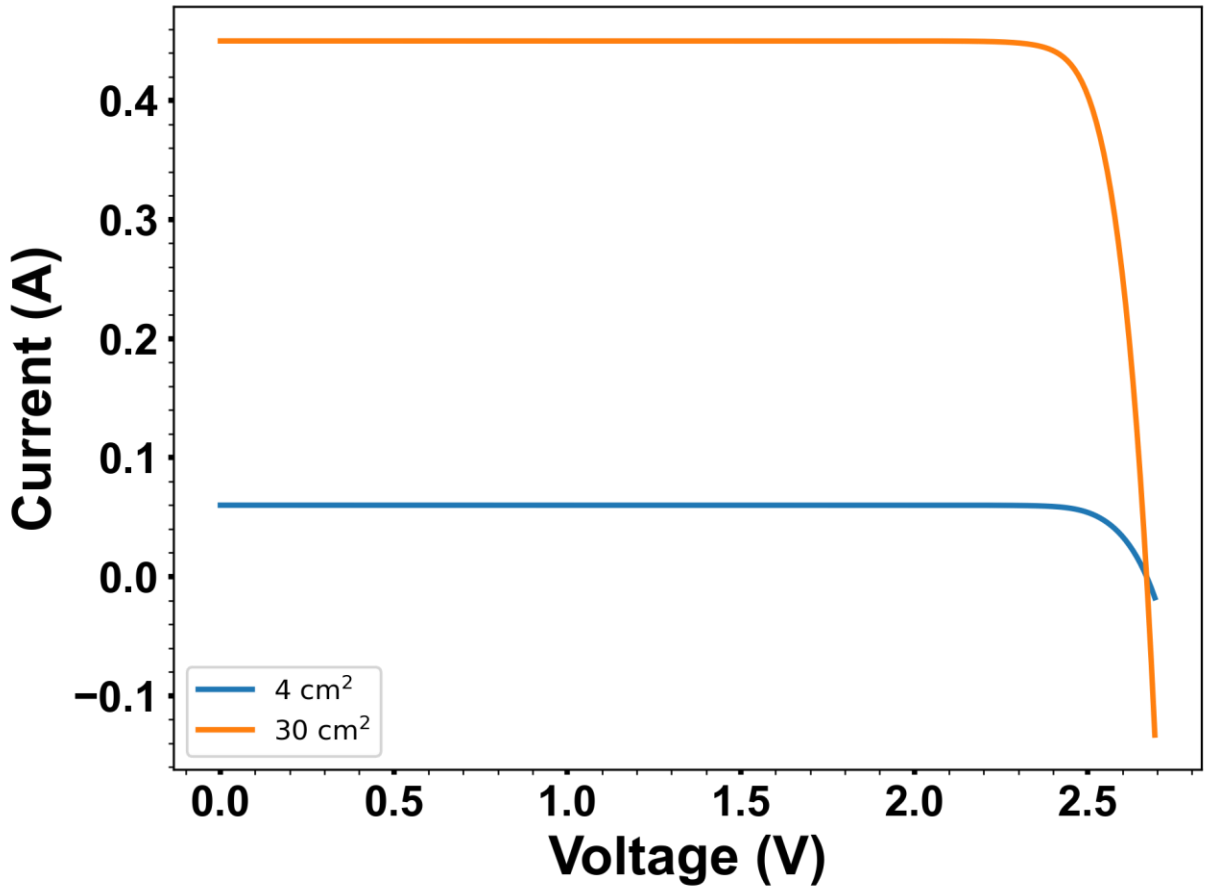


Fluence	I_{ph}	I_0	n	R_s	R_{sh}	r^2
0.E+00	0.0689	6.34E-20	2.5090	1.0381	6828.3	0.9963
5.E+10	0.0689	1.46E-15	3.1420	0.3825	5783.6	0.9998
1.E+11	0.0682	4.53E-15	3.2063	0.3729	4608.4	0.9997
5.E+11	0.0661	4.03E-14	3.2869	0.4781	1593.0	0.9998
1.E+12	0.0639	4.75E-13	3.5134	0.4534	1061.2	0.9998
2.E+12	0.0615	1.14E-10	4.3490	0.0991	929.1	0.9987
4.E+12	0.0585	4.87E-14	3.0434	1.4296	498.2	0.9979
1.E+13	0.0544	2.81E-10	4.1668	0.4982	383.0	0.9996

Scale IV Curve by Area

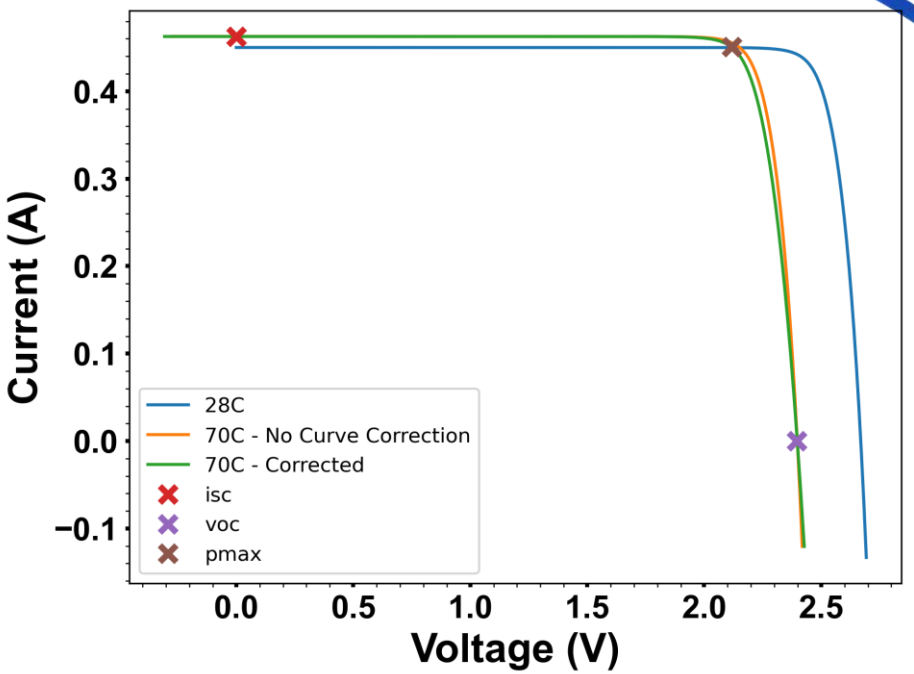
- Radiation data was collected using 4cm^2 solar cell data with, whereas the program cells have an area of 30cm^2

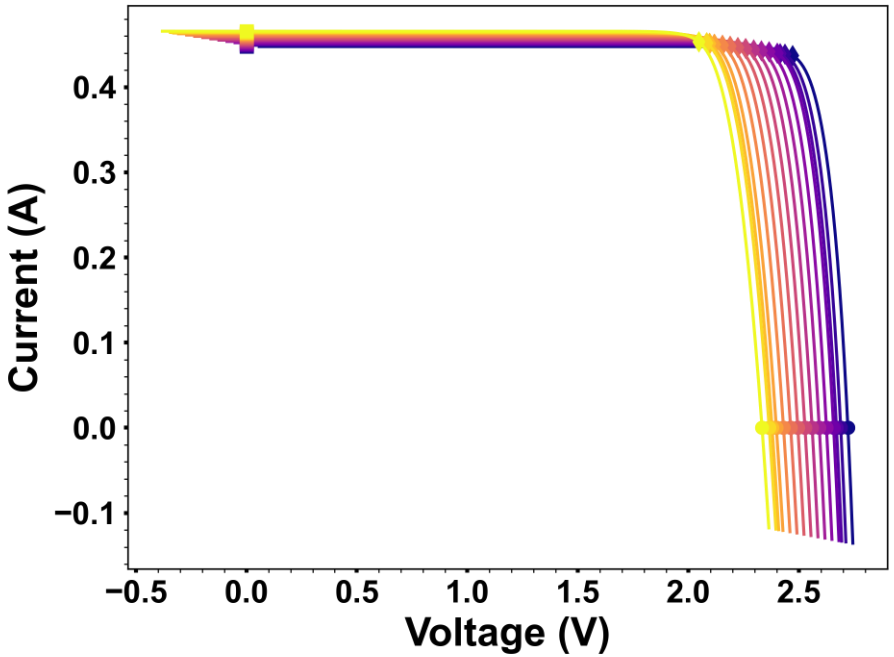
- To scale by area, the current values of the IV curve are multiplied the increase in area. This is representative of putting many 4cm^2 cells in parallel with each other
  - $I(V)_s = \left(\frac{I(V)}{A_m}\right) \times A_s$
  - *V_{oc} Doesn't change upon area scaling*
  - *Series resistance decreases following Kirchhoff's Rules*
    - R_s of 4cm^2 cell is 0.45Ω and when the cell is scaled to 30cm^2 the series is 0.06Ω



Temperature Correct IV Curve

- IV curve corrected for temperature using Procedure 1 of the IEC 60891
  - *Can translate IV curve for temperature and intensity, but we only use it for temperature translation*
  - $I_T = I_i + I_{sc_i}(G_s - 1) + \alpha(\Delta T)$
  - $V_T = V_i - R_s(I_T - I_i) - \kappa I_t(\Delta T) + \beta(\Delta T)$
  - I_T, V_T = *Temperature corrected current and voltage*
  - I_i, V_i = *Initial current and voltage*
  - α, β = *Current and voltage temperature coefficient*
  - ΔT = *Delta between initial and final temperature*
  - I_{sc_i} = *Short circuit current of initial IV curve*
  - R_s = *Series resistance*
  - G_s = *Intensity scaling factor*
  - κ = *curve correction factor*
    - Found through fitting initial IV curve and minimized on the error of the calculated meta data (I_{sc}, V_{oc}, I_{max}, V_{max}, and P_{max})





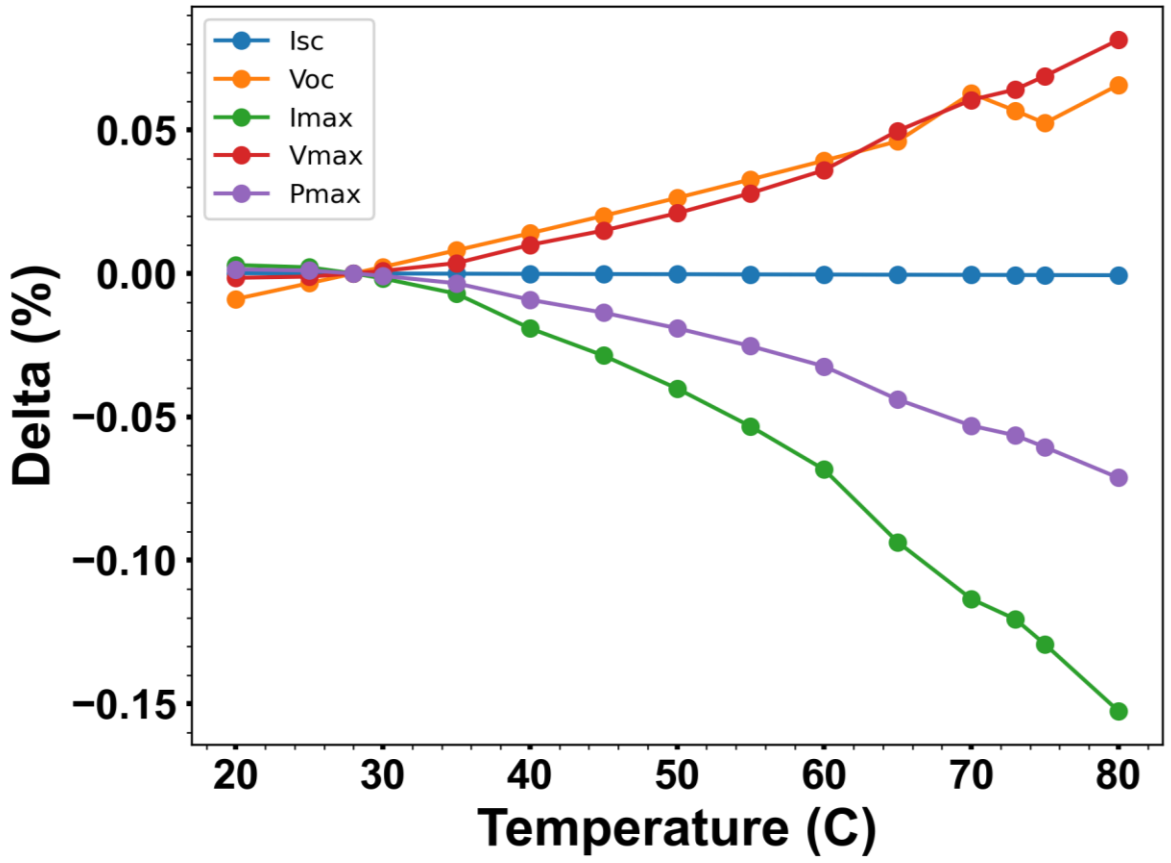
Validation of Temperature Correcting IV Curves

- The extracted meta data (I_{sc}, V_{oc}, I_{max}, V_{max}, and P_{max}) from the temperature corrected IV curves ($meta_{IV\text{-}TC}$) is compared to the meta data at the starting temperature that has been temperature corrected only using temperature coefficients ($meta_{TC}$)

- $meta_{IV\text{-}TC}$ were extracted using ASTM E1036-15R19
  - *If I_{sc} not within a certain voltage range of 0V is linear extrapolated to 0V*
  - *If V_{oc} not within a certain current range of 0A is linearly extrapolated to 0A*
  - *I_{max}, V_{max}, and P_{max} are found by taking a polynomial about the knee of the curve and the root is found.*

- $meta_{TC}$ is derived by taking the meta data at $T_{initial}$ corrected the meta data for the target temperature using only temperature coefficients

Correction error is less than 0.06% from 20C -80C.



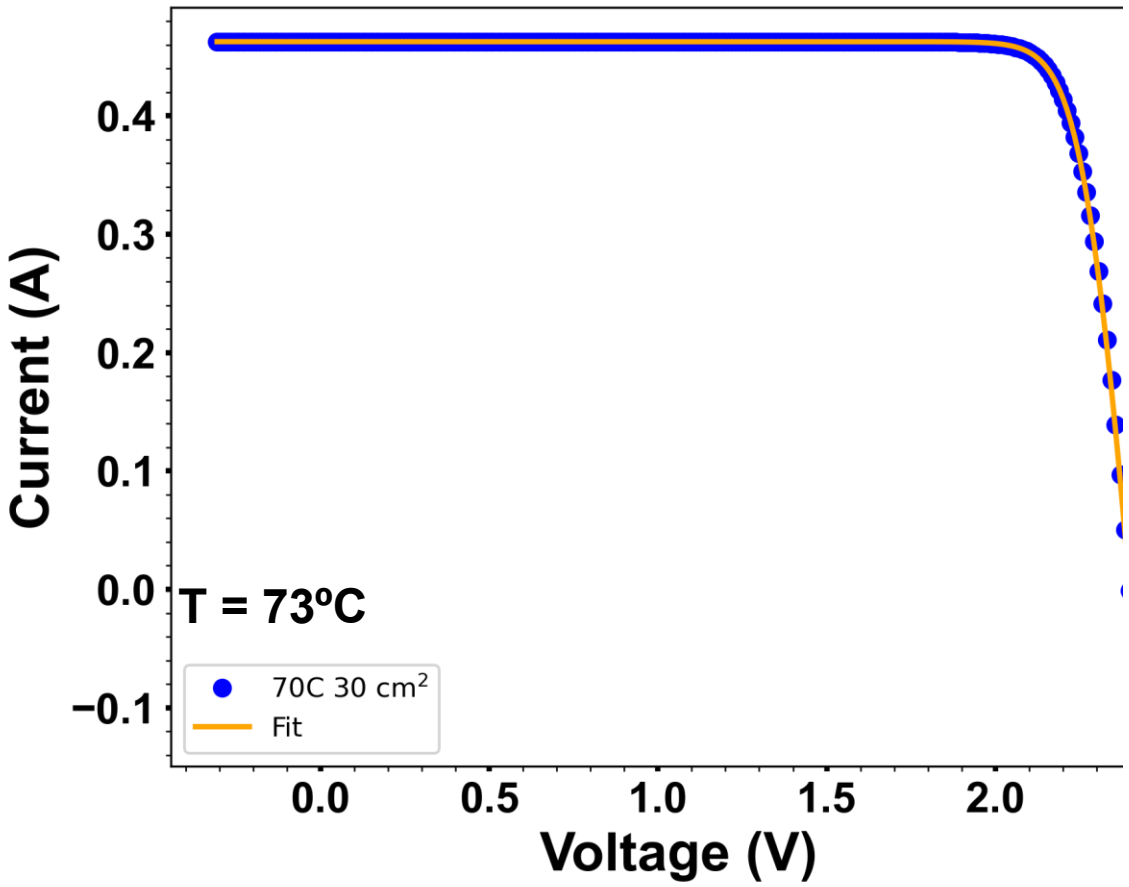
Fit and Extract Single Diode Parameters

Results

- Single diode model was used as it allows for safe extrapolation and interpolation of an IV curve
  - *If subjunction IV data is available those should be used*

- The parameters can be used to numerically calculate the IV curve and as inputs into spice circuit model simulations that are used

- The five parameters needed for the single diode equation are the photocurrent (I_{ph}), saturation current (I_0), ideality factor (n),), series resistance (R_s), and shunt resistance (R_{sh})



I_{ph} (A)	I_o (A)	n	R_s Ω	R_{sh} Ω	r^2
0.463	1.17e-19	2.15	0.17	1.33e5	0.999

Fit and Extract Single Diode Parameters

Methods

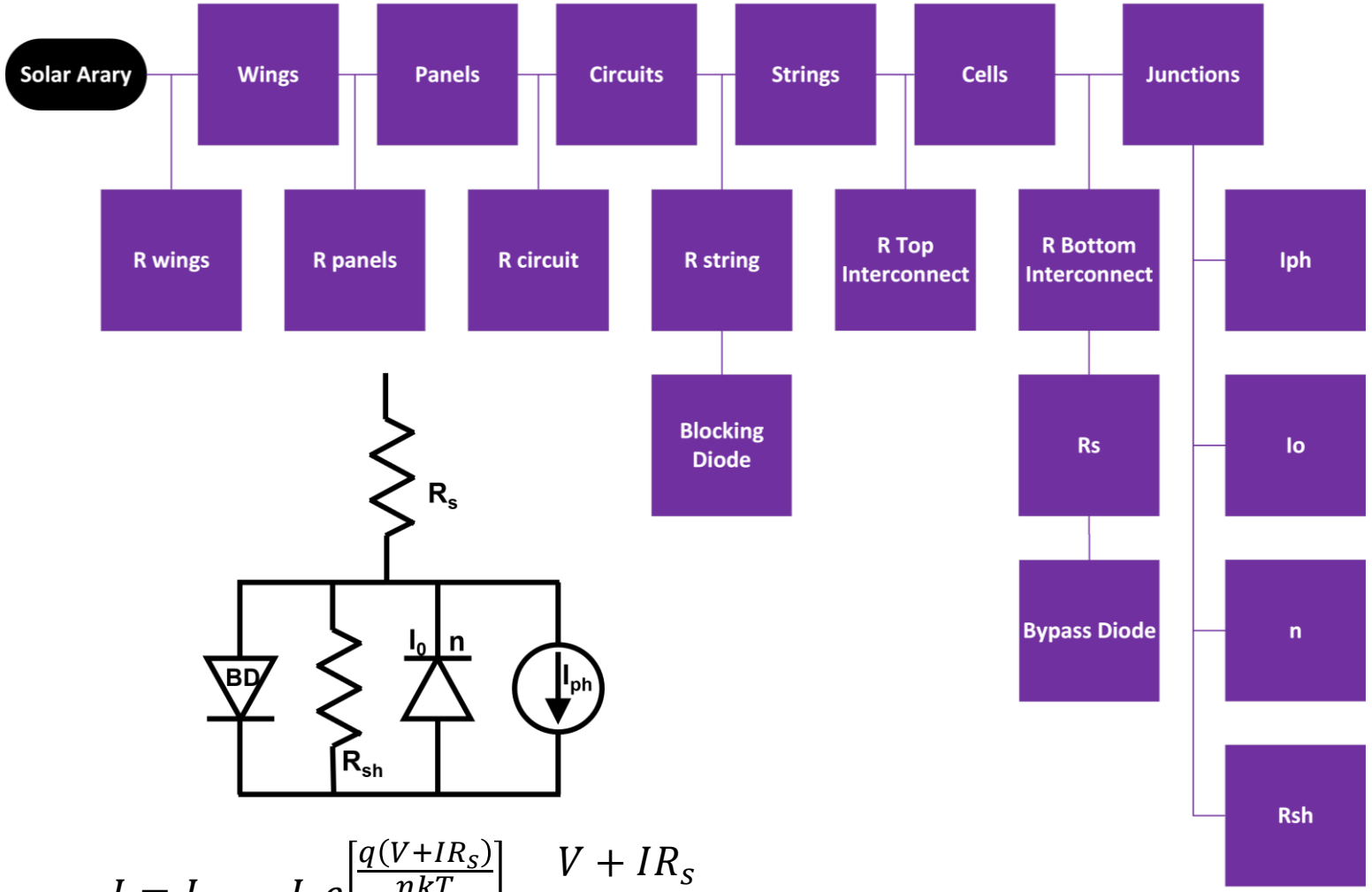
- We explored two methods to extract the parameters (there are over a 100)

1. Zhang Model (Journal of Applied Physics, 110, 064504 (2011)
   - *Only involves extracting three parameters (n, R_s, R_{sh}) and I_{ph} and I_0, are calculated*
   - *The inputs are limited to the I_{sc}, V_{oc}, and a method to derive initial guesses for n, R_s, R_{sh}.*
   - *Produces values that are reasonable and trend in some direction and requires the user to have little knowledge in guessing the input*
   - *Achieves r^2 fits of 0.999*
   - *Easy to implement in python, matlab, etc.*

2. Toledo Model (Renewable Energy, 72, 2014, Pages 125-133)
   - *Non-iterative, analytical method that doesn't require fitting*
   - *The inputs include I_{sc}, V_{oc}, I_{max}, V_{max}, and one (I,V) point between P_{max} and V_{oc}. The initial R_{sh} is also needed and is determined by the slope of the IV curve near I_{sc}, but upon derivation of the 4 of the 5 parameters, R_{sh} can be calculated*
   - *Produce r^2 fits of 0.98 to 0.999, but sometimes the 5 parameters do not correlate with one can visually interpret from the iv curve or what you would expect for radiation damage*
   - *Easy to implement in MS Excel*

Build Array: psim

- The solar array model is built using psim
  - *psim is a python library that builds the solar cell array and generates a spice netlist that is then run using ngspice*
  - *Ngspice was chose as it is open source and free. The netlist generated can be run any spice simulator with little to no modification*
- The fundamental building element of the model is the diode equation and the solar cell IV curve



$$I = I_{ph} - I_o e^{\left[\frac{q(V+IR_s)}{nkT}\right]} - \frac{V + IR_s}{R_{sh}}$$

Include Uncertainty?

- Loss factors for Current were grouped into 3 Groups
  - *Static*
    - Factors that are not affected by time varying components
    - These factors remain constant from the start of the mission
  - *Time Dependent*
    - Factors that change over time
    - Considered linear in the simplest case, but are more than likely nonlinear
  - *Uncertainties*
    - Uncertainties are treated as 1 sigma
- Totals
  - *This row shows how one would calculate total loss factors at end-of-life (EOL), but could also be applied at any given point in time*
  - *Static losses are multiplied to arrive at a total static lost factor*
  - *Time dependent total losses are also multiplied and the aphelion was used as the "worst case" EOL time dependent loss factor*
  - *Uncertainties are treated as uncorrelated, thereby we propagated the error by taking the square root of the sum of squares.*
    - Results in every solar cell in the array having different IV curves

Loss Factor	Type of Loss	%Loss
Static Loss	**Installation Losses**	0.98
	Off Pointing	0.995
Time Dependent	**Earth Sun Distance**	0.967-1.033
	Coverglass Darkening	1-0.95
	Adhesive Darkening	1-0.99
	Contamination	1-0.95
	Micrometeoroid	1-0.999
Uncertainties	**Cell Measurements**	±0.05
	Coverglass Darkening	±0.02
	Adhesive Darkening	±0.001
	Contamination	±0.05
	Micrometeoroid	0
Totals	**Static (EOL)**	0.975
	Time Dependent (EOL)	0.863
	Uncertainty	0.073
	Total Loss Factors (EOL)	0.841±0.073

Include Uncertainty? - Yes

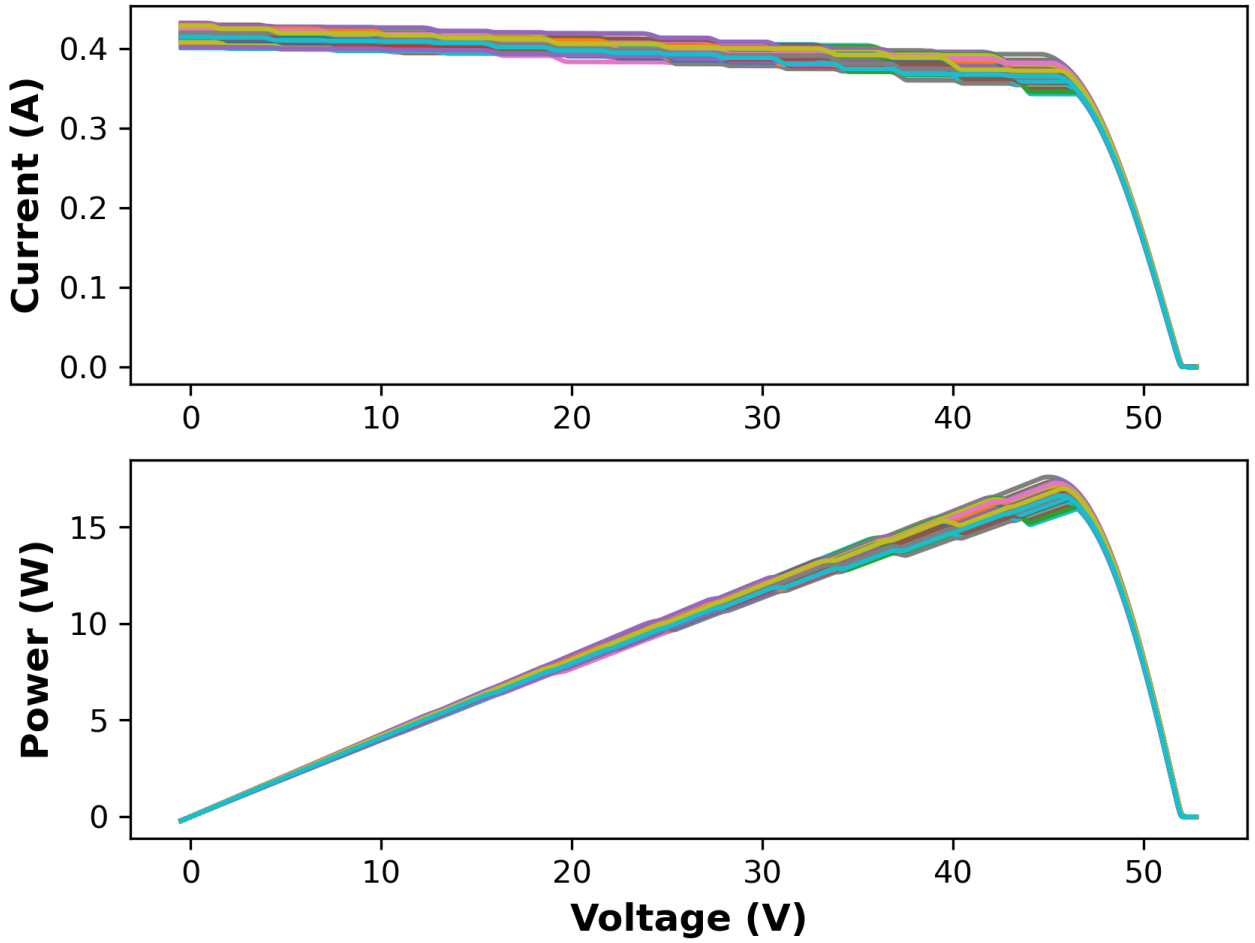
Monte Carlo Simulation

- The uncertainty derived in the previous chart is applied to I_{ph} of each solar cell

- Using a random normal distribution each cell's I_{ph} is adjusted resulting in a slightly different IV curve.
  - *Because the model is based on the diode equation, any increase or decrease in the current generate results in slightly higher and lower meta parameters.*
  - *The figure to the right represents 200 strings of 22 solar cells*

- To calculate the power of the array those 200 are simply added together.

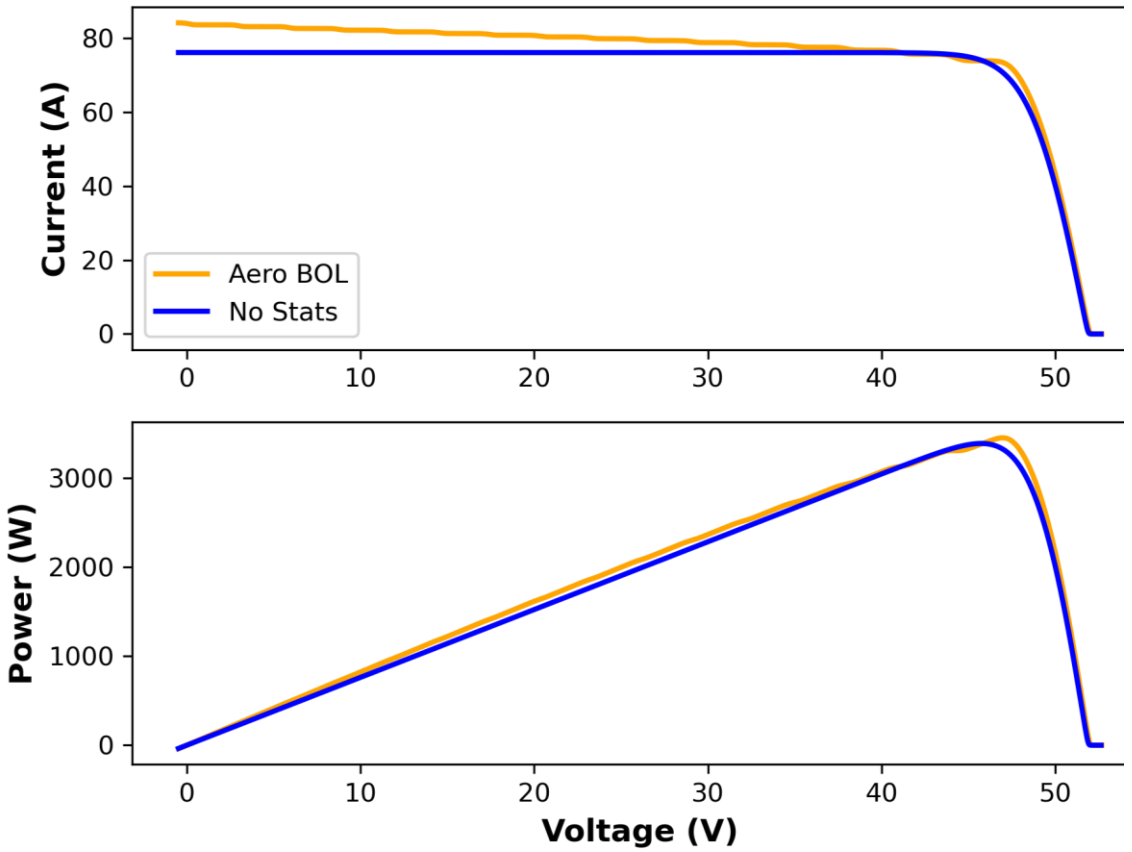
- We repeat the generation of the plot to the right over and over to arrive at an average power



Result of MC Runs

Monte Carlo Simulation

- Results presented are for 20 runs
- For the MC runs all the variability basically averages out over the array
  - *Randomly placing cells with photocurrents ranging over ±7% does not alter the power of the array very much as seen by the low standard deviations of the meta parameters*
- Slope of the IV curve is less square due to the variability being added to each string and the effect of the bypass diodes
- One effects use case of this MC method is randomizing failures of various components in an array



	WC	MC Run	std
I_{sc} (A)	76.14	84.00	0.07
V_{oc} (V)	52.38	52.62	0.06
I_{max} (A)	74.06	73.61	0.16
V_{max} (V)	45.72	46.84	0.02

Conclusion

- We have developed a model and procedure for calculating array power that allows the incorporation of variability in performance and degradation factors at the cell level
- The model and method can be rooted in measured or simulated IV curves
- In our example we only randomized the loss in current but the variability in cell voltage can be adjusted using the saturation current
  - *Variability in the radiation environment can be applied using Ae9/Ap9 and solar cell radiation degradation models*
- Cells can be mapped to a real array or binned to arrive at the best array configuration
- We are working or open sourcing this to The Aerospace github page.
  - *Please suggest any features you would like!*