

A decorative graphic featuring various colored circles (teal, lime green, orange, pink, yellow) and dashed lines of different colors (teal, yellow, green) arranged in a circular pattern around the central text.

Forecasting Critical Design Review

By Gordon Li, Jaimie Obatake,
Brianna Sundberg, and Austin
Tasato



Overview

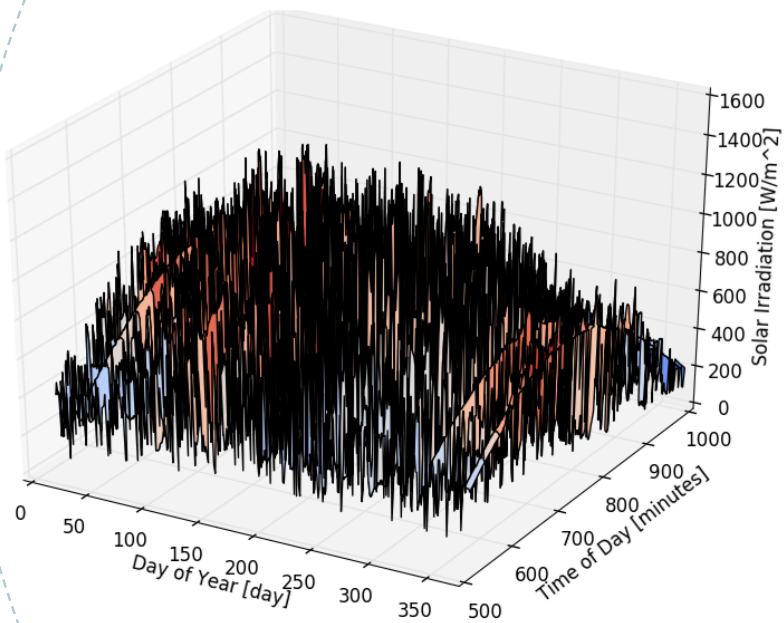
- Block Diagram
- Algorithms:
 - 3-D Plot
 - Error Detection & Correction
 - Least Squares
- Progress since PDR
- Yet to Finish
- Questions



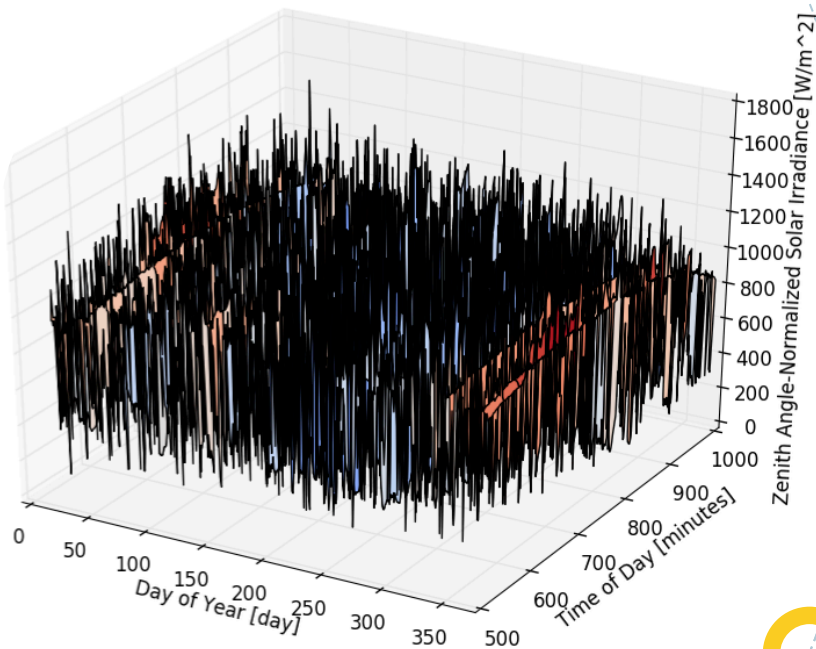
3D Plot Algorithm

- ⦿ **Problem:** How can we create a 3D surface plot of the data?
 - ⦿ Solar irradiance
 - ⦿ Time
 - ⦿ Day
- ⦿ **Process:** `reshape()` data, figure out surface grid interval, etc.

3D Plots : La Ola, Lanai

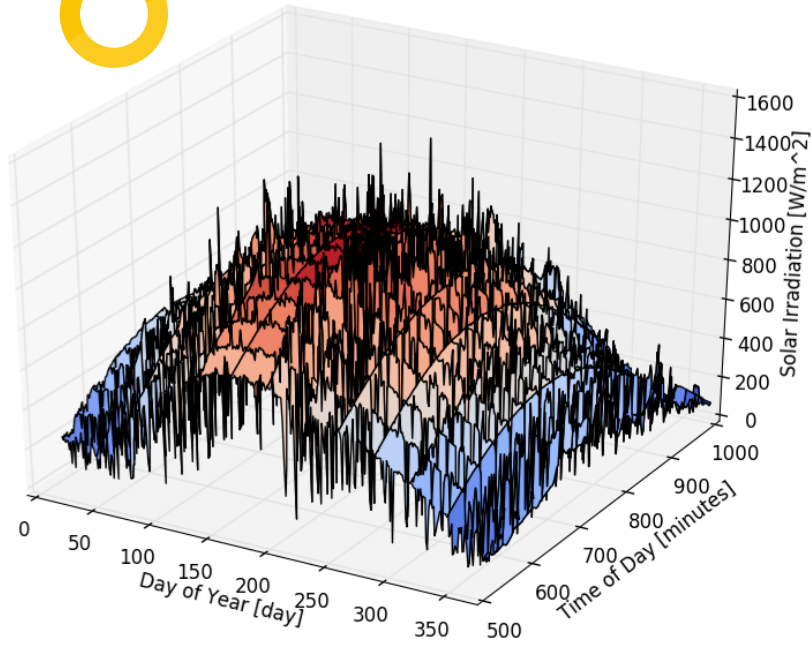


Not Normalized

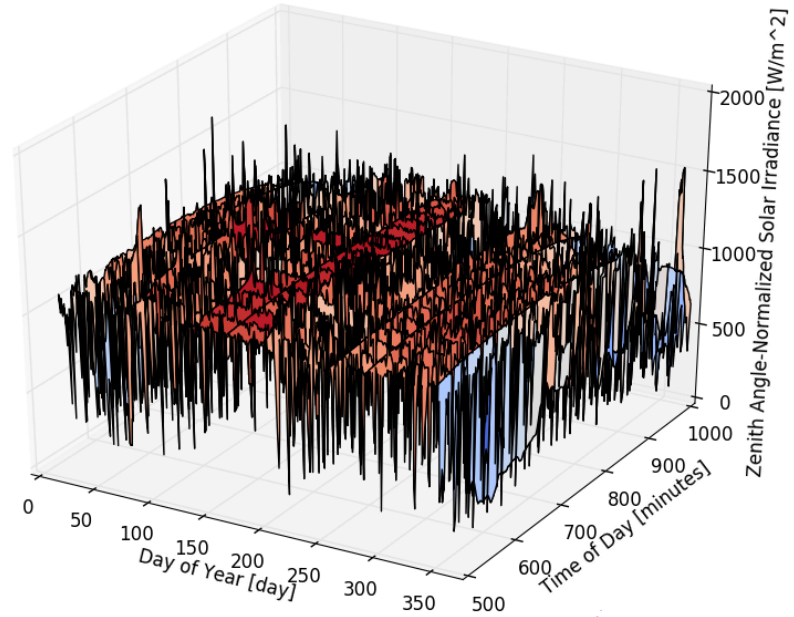


Normalized

3D Plots : Milford, Utah



Not Normalized



Normalized

Block Diagram of System



Error Detection and Correction

`check_samples()`

- ⦿ Checking for continuous time samples & filling in missing data
- ⦿ **“Why is 10:00AM and 10:01 is missing?”**
- ⦿ Corrects by adding in missing samples, and deleting duplicates

`check_errors()`

- ⦿ Checking & replacing erroneous solar irradiance data
- ⦿ **“Why is there a spike that’s 10x the surrounding data at 9:59AM?”**
- ⦿ Replaces irradiance samples with previous value

Least Squares Algorithm

- Batch algorithm
- 5 minute decimation
- Zenith angle normalization
- Creating X matrices (taps)
- Weights vector
 - $W = (XX^T)^{-1} XD$
- Root Mean Squared Error
 - RMSE =
- Currently testing with k-folds and tap filters

$$\frac{1}{m} \sum (D - Y)^2$$



Progress Since Preliminary Design Review

- 3D visualization
- Error correction functions
- Data processing
- Tap filters
- Least Squares algorithm
- Testing/training models (Various)



Yet to Finish

- Finish Least-Squares (fixing errors, etc.) ●
- Make more robust error detection and correction functions (zenith angle)
- Train/test with k-folds/validation
- Seasonal models
- New algorithms
- Documentation



Any Questions?