SOLAR ENERGY GENERATION FORECASTING AND POWER OUTPUT OPTIMIZATION OF UTILITY SCALE SOLAR FIELD

A Thesis

presented to

the Faculty of California Polytechnic State University,

San Luis Obispo

In Partial Fulfillment

of the Requirements for the Degree

Master of Science in Mechanical Engineering

by

Byungyu Kim

June 2020

COMMITTEE MEMBERSHIP

TITLE: Solar Energy Generation Forecasting and Power Output Optimization of Utility Scale Solar Field

AUTHOR: Byungyu Kim

DATE SUBMITTED: June 2020

COMMITTEE CHAIR: Jacques Belanger, Ph.D. P.E.
Assistant Professor of Mechanical Engineering
PG&E Professor of Nuclear and Renewable Energy

COMMITTEE MEMBER: Andrew Davol, Ph.D.
Professor of Mechanical Engineering

COMMITTEE MEMBER: Dale Dolan, Ph.D.
Professor of Electrical Engineering

ABSTRACT

Solar Energy Generation Forecasting and Power Output Optimization of Utility Scale Solar Field

Byungyu Kim

The optimization of photovoltaic (PV) power generation system requires an accurate system performance model capable of validating the PV system optimization design. Currently, many commercial PV system modeling programs are available, but those programs are not able to model PV systems on a distorted ground level. Furthermore, they were not designed to optimize PV systems that are already installed. To solve these types of problems, this thesis proposes an optimization method using model simulations and a MATLAB-based PV system performance model. The optimization method is particularly designed to address partial shading issues often encountered in PV system installed on distorted ground. The MATLAB-based model was validated using the data collected from the Cal Poly Gold Tree Solar Field. It was able to predict the system performance with 96.4 to 99.6 percent accuracy. The optimization method utilizes the backtracking algorithm already installed in the system and the pitch distance to control the angle of the tracker and reduces solar panels partial shading on the adjacent row to improve system output. With pitch distances reduced in the backtracking algorithm between 2.5 meters and 3 meters, the inverter with inter-row shading can expect a 10.4 percent to 28.9 percent increase in power production. The implementation and calibration of this optimization method in the field this spring was delayed due to COVID-19. The field implementation is now expected to start this summer.

ACKNOWLEDGMENTS

I would like to thank my family and friends for being there for me throughout this journey; I appreciate and love every one of you. With the support and love from all of you guys, I was able to finish my graduate school.

I want to thank my advisors, Dr. Belanger and Dr. Davol for giving me this great opportunity to learn about every aspects of solar energy generation and learn what it means to work together with helpful, caring, and supportive advisors. Despite the closure of campus, I really enjoyed our weekly meetings over Zoom this last quarter.

I want to thank the projector sponsors, REC solar and Cal Poly RSCA Research Grant Program for providing me with the scholarship and giving me a hands-on opportunity in renewable energy and fueling my enthusiasm in the clean, renewable energy industry.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure Page

LIST OF ABBREVIATIONS

AC	Alternate current
Cal Poly	California Polytechnic State University, San Luis Obispo
DC	Direct Current
DHI	Direct Horizontal Irradiance
DNI	Direct Normal Irradiance
GCR	Ground Cover Ratio
GHI	Global Horizontal Irradiance
GPM	Green Power Monitor
IAM	Incident Angle Modifier
IV	Current vs Voltage
kWh	Kilowatt per Hour
MW_{ac}	Megawatt AC
MW_{dc}	Megawatt DC
MPPT	Maximum Power Point Tracking
NOCT	Nominal Operating Cell Temperature
NREL	National Renewable Energy Laboratory
NSRDB	National Solar Radiation Database
POA	Plane of array
PV	Photovoltaic
PV_LIB	PV_LIB Toolbox
SAT	Single Axis Tracker
STC	Standard testing conditions
TMY	Typical Meteorological Year

Chapter 1

INTRODUCTION

1.1 Background

Due to advances in solar photovoltaic (PV) technology, reduction of cost and growing environmental consciousness, deployment of solar energy has increased throughout the different types of installations from individual homes to large utility-scale. Photovoltaic power plants are being recognized for their great potential, not only in its renewable energy production, but also for their benefits to the economy and environment [1]. With its increase in popularity, the cost of solar energy has been decreasing due to the improvements in the manufacturing process and government policies around the world. It is reported that the price of solar energy has fallen by 99 percent over the last four decades [2].

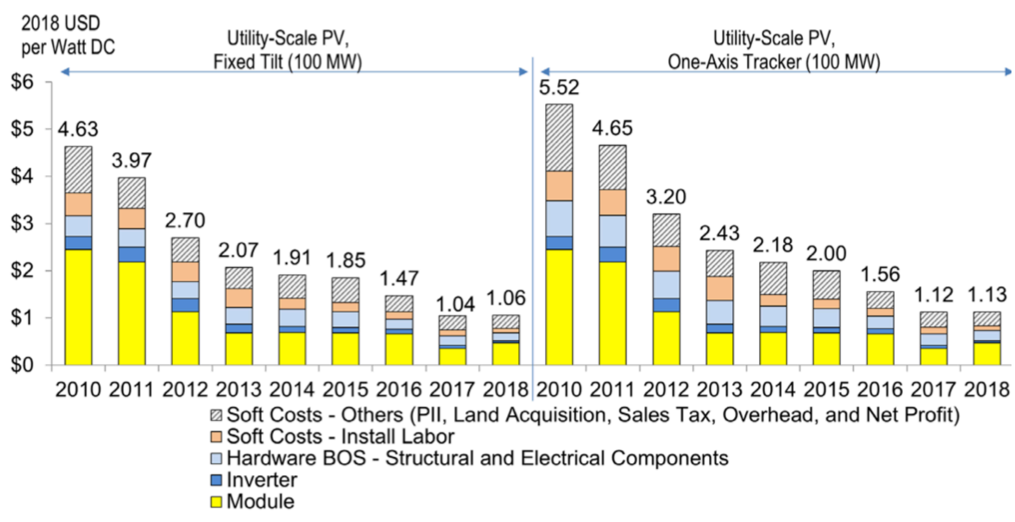


Figure 1.1: Decreasing Cost of Utility Scale PV System from 2010 to 2018

Figure 1.1 [3] shows the price of utility scale PV system with single-axis tracker decreased 79.5 percent from 2010 to 2018. Like any electricity generation provider, that the current solar industry has a narrow margin for profit, and any small improvements in energy generation can be financially beneficial.

With the rapidly increasing number of PV system installations, it is important to be able to quantify how much energy can be generated and delivered to the grid. By accurately forecasting the energy generation of PV systems, grid managements is made easier and grid overloading can be prevented. That information can also be used to raise awareness of any underperforming system. Furthermore, improvements and optimization in energy production can be financially beneficial, especially for utility-scale solar fields. This thesis provides a series of analysis tools characterizing the generation-forecasting and optimization solutions considered for the Cal Poly Gold Tree Solar Farm.

1.2 Objective

The purpose of this thesis is to create a computer model of an existing PV system experiencing inter-row shading and predict the behavior and performance of the PV system with different backtracking inputs. Currently, the utility-scale solar field owned by the California Polytechnic State University-San Luis Obispo is underperforming due in large part to inter-row shading. Inter-row shading is created by the uneven ground profile of the site. That uneven ground can elevate a row of PV modules relative to an adjacent row, which can potentially cast a shadow on that adjacent row. To reduce shading on PV modules, and improve energy production, pitch distance input needs to be adjusted in the backtracking algorithm. With extensive research on system behavior and an accurate computer model of the PV array, optimum inputs can be determined to reduce shading, and improve system performance. This thesis will provide a contribution towards accurate PV system performance prediction for those systems installed on uneven terrain with inter-row shading. This thesis will also serve as a solar system modeling software manual for the many students that will be continuing in that line of research in the future.

1.3 Limitations

It is important to note that values from existing commercial PV system modeling software is based on theoretical data generated with some assumptions; therefore, the results do not always reflect actual data. For example, energy forecasted from software such as PVsyst depends on the meteorological data that is designed to represent a typical year, and the user-defined losses to the system like soiling or panel's degradation. It is important to note that there is no method to accurately identify different losses, so the estimated losses from PVsyst will not always be equal to the real losses. Due to the closure of campus during the spring quarter, implementation of the improved parameters and verification of the model is delayed to the summer and fall of 2020.

Chapter 2

DESCRIPTION OF GOLD TREE SOLAR FARM

2.1 System Description

In a partnership with REC Solar and Cal Poly, REC Solar built an18.5-acre single axis tracker (SAT) solar field that is projected to generate more than 11 million kWh per year with a capacity of 4.5 MW_{ac} or 5.657 MW_{dc}. According to REC Solar, the company in charge of design, construction, and maintenance of the solar facility, Cal Poly Solar Field will generate enough to power more than 1,000 homes, or about 25 percent of Cal Poly's total needs [4]. This project was built under a power-purchase agreement between Cal Poly and REC Solar, which permits REC Solar to construct and maintain the solar field on Cal Poly's property, while Cal Poly pays a lower rate for the electricity. In addition to the savings on annual utility costs, this project allows a hands-on learning opportunity for students interested in renewable energy and in solar fields in particular.

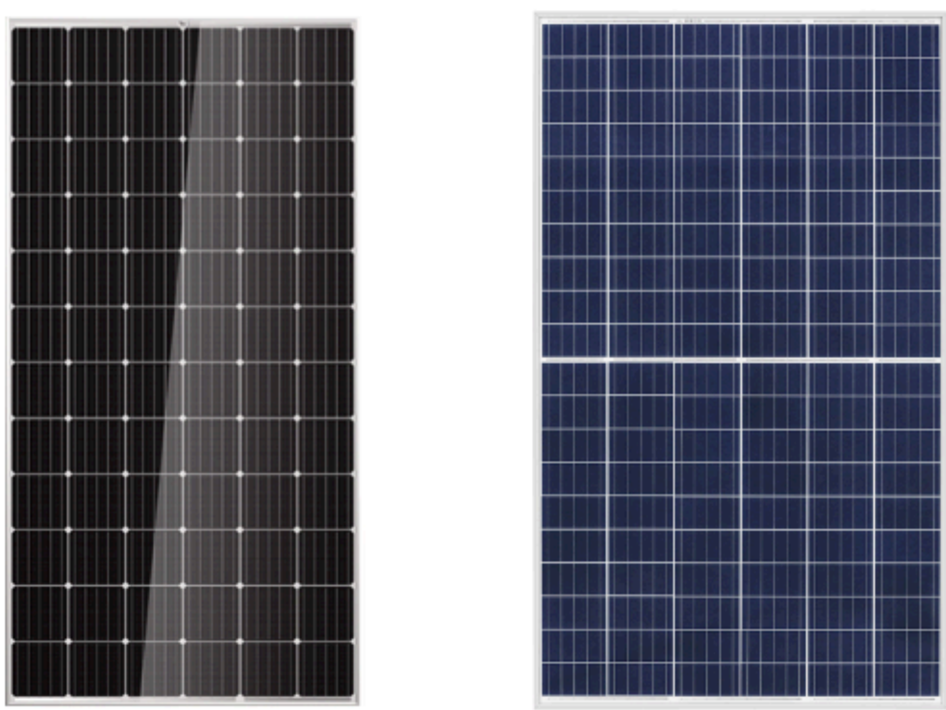
Figure 2.1: Aerial Views of Gold Tree Solar Farm

Figure 2.1 [4] displays two different aerial views of the Cal Poly Solar Farm. The view on the right shows changing slopes of the field and describes how PV modules are installed along the surface of the sloped field.

2.2 System Parameters

2.2.1 PV Modules

Cal Poly Solar Farm uses 16,379 ground-mounted PV modules, 8,664 modules are produced by Trinia Solar, and 7,733 modules are produced by REC Solar. Trinia Solar modules consist of monocrystalline full-cell panels, while REC Solar modules consist of polycrystalline half-cut cell modules. Half-cut cell is a full-cut cell that is cut into two equal pieces, which reduces the internal resistance, increases overall power, and improves the module's performance in reduced irradiance conditions, e.g., when shaded [5]. Figure 2.2 represents the PV modules used in the solar farm, and the REC Solar PV module has a white line across the panel representing the half-cut cells.



(a) (b)

Figure 2.2: Pictures of PV Modules Used in Cal Poly Solar Field: (a) Trinia Solar PV Modules with Full-Cut Cells (b) REC Solar PV Module with Half-Cut Cells.

2.2.2 Tracker

Both modules are installed on a single axis tracker from Array Technologies, which follows the sun across the sky with a tracking angle range of -52° to +52° with the possibility of backtracking. The trackers are equipped with a 2 HP motor that can change the tracking angle of up to 26 rows, and it is powered by the energy from the grid. There is a total of 12 trackers on the solar field; therefore, the solar farm is divided into 12 different zones. The location of the zones and the number of PV array rows per zone are specified in Appendix A.

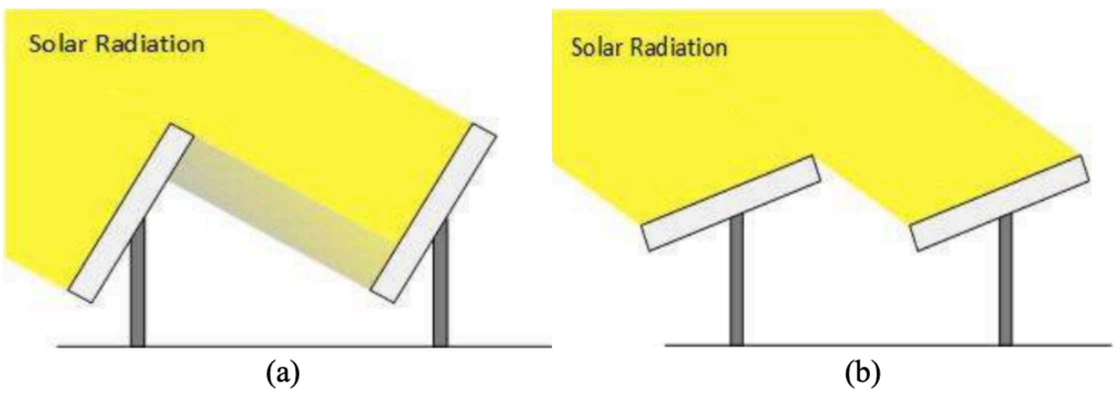


Figure 2.3: Solar Radiation on PV modules: (a) shading on PV module without backtracking (b) PV modules with backtracking reduces shading.

Backtracking is a tracking algorithm implemented on the motion of the arrays, and it minimizes the power loss of the system due to shading by considering the ground coverage ratio (GCR) and row spacing of the solar arrays [6]. Figure 2.3 [7] shows the benefit of backtracking, where SAT can minimize shading on PV modules in the adjacent rows by reducing the angle of the tracker. A disadvantage of implementing backtracking is that the solar panels are not directly receiving the solar irradiance. Solar panels generate the maximum amount of power when the solar irradiance is perpendicular to the surface of the panels. However, with backtracking, solar panels are restricted from facing the sun by flattening the tracker angle earlier, which can reduce the amount of generated power due to the reduced amount of irradiance reaching the surface of the solar panels through the atmosphere.

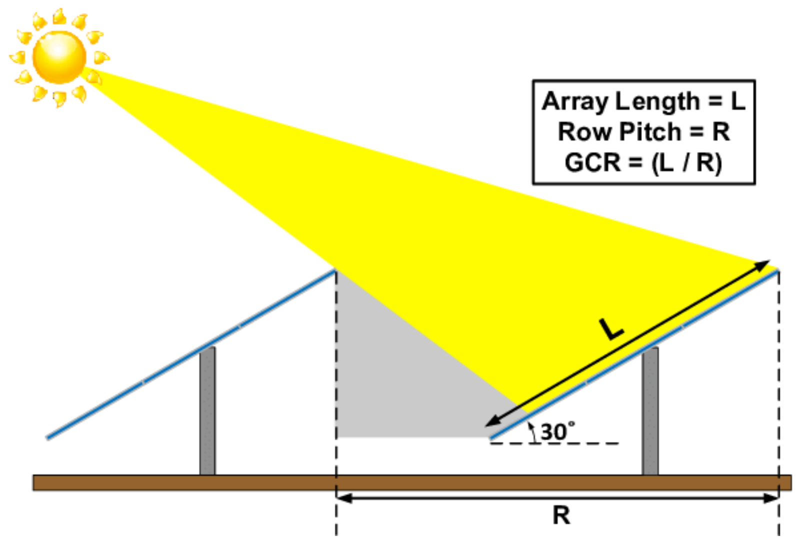


Figure 2.4: Ground Coverage Ratio is the ratio of module height to the pitch distance (L/R). Inter-row shading is directly related to GCR.

The backtracking algorithm is based on the ground coverage ratio, as mentioned previously. As shown in figure 2.4 [8], the ground coverage ratio is the ratio of module height to the pitch distance, where L is the module length, and R is the pitch distance of the modules. This ratio determines the range of tracker angle, and it is directly related to the inter-row shading. When the pitch distance decreases, it makes GCR value higher. Higher GCR value will increase the inter-row shading because the panels are closer to each other. It is beneficial to have a larger pitch distance to reduce inter-row shading, but it is also important to space out the solar panels to optimize the power generation in relation to according to the size of land.

2.2.3 Inverter

There are seventy-five Yaskawa Solectria 60TL string inverters used for both full-cut cell and half-cut cell PV modules on the Cal Poly Solar Farm. Each inverter is linked to three maximum power point tracking (MPPT) inputs, where each input consists of 4 strings that are connected in parallel. Each string consists of 19 modules that are connected in series, and the individual rows of PV modules on the Cal Poly Solar Farm represents a single MPPT input for the corresponding inverter. Multiple MPPT inputs allows the inverter to split the strings of PV modules to maximize the amount of power received from different shading conditions [9].

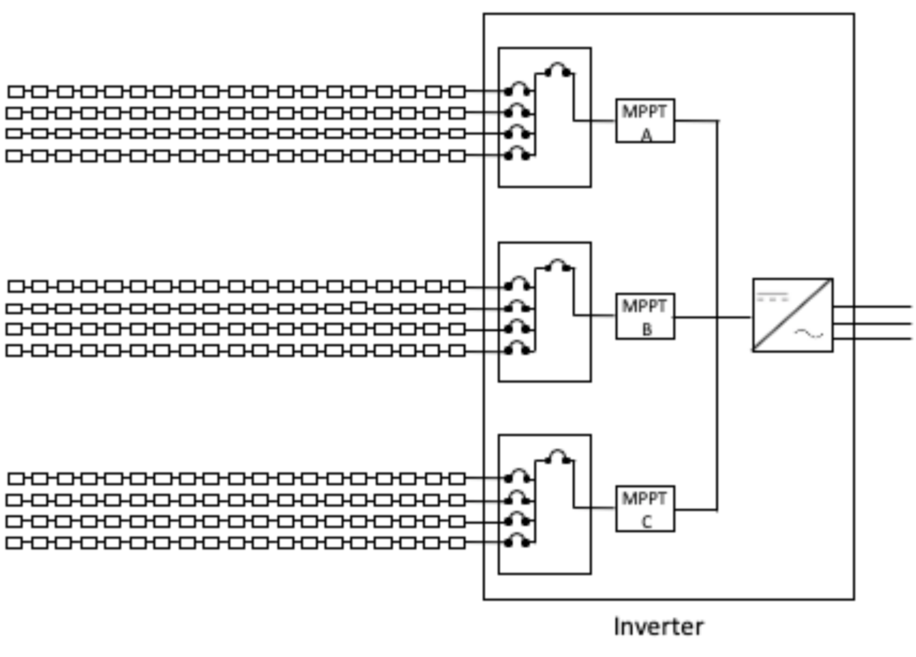


Figure 2.5: Single Line Drawing of an Inverter with 3 MPPT

Figure 2.5 represents the single line drawing of an inverter with three MPPT inputs similar set up as the inverters in the Gold Tree Solar Farm. This figure visually represents the relationship between strings of modules and the inverter. This figure visually represents the relationship between strings of modules and the inverter. There are 12 strings connected for an inverter with full-cut modules, and 11 strings for an inverter with half-cut modules.

2.2.4 Summary of Parameters

Currently, Cal Poly Gold Tree Solar Farm's energy generation is lower than the expected amount projected at the time of the construction and stated by REC Solar in section 2.1. It is important to note that the program used to generate the expected energy generation is not capable of processing the complicated topography of the solar field and assumes there is little to no shading on the PV modules. The table below summarizes geographical information, components used for the Cal Poly Gold Tree Solar Farm, and ground coverage information. More technical information regarding PV modules, inverter, and trackers can be found in Appendix B.

Table 2.1: Detailed Summary of Cal Poly Solar Farm

<table>
<tr><td>Location</td><td colspan="2">(Latitude, Longitude): (35.32° N, -120.69° W)
Elevation: 156 m
San Luis Obispo, California, US</td></tr>
<tr><td>System Capacity</td><td colspan="2">DC: 5.7 MW
AC: 4.5 MW</td></tr>
<tr><td>Module Types</td><td>Trina Solar TSM-DE 14A (II), 345 W
Amount: 8664 panels</td><td>REC Solar Twinpeak 2s 72 Series, 345 W
Amount: 7733panels</td></tr>
<tr><td>Mounting, Tracking</td><td colspan="2">Ground mounting with single axis tracker (E-W)
Tracker: Array Technologies Inc. Duratrack HZ V3
Tracker range of motion: -52° to +52°</td></tr>
<tr><td>Inverter type and Quantity</td><td colspan="2">Yaskawa Solectria PVI-60TL
Amount: 75 inverters</td></tr>
<tr><td>DC/AC ratio</td><td colspan="2">1.26</td></tr>
<tr><td>Row Spacing</td><td colspan="2">3.35 m</td></tr>
<tr><td>Ground Coverage Ratio</td><td colspan="2">58.5 %</td></tr>
</table>

2.3 Green Power Monitor Portal

The Green Power Monitor (GPM) Portal is a website developed by Green Power Monitor for REC Solar, and it is a web based real-time performance monitoring program for the Gold Tree Solar Farm. Since the operation of the solar farm began, GPM Portal has been collecting and storing data related to different aspects of the solar field. The main data used for this project are current, voltage, power and energy generated from the system, tracking angle of the arrays, plane of array irradiance (POA), and global horizontal irradiance. Instructions on navigating through GPM Portal and downloading data can be found in Appendix C.

Chapter 3

PVSYST

3.1 Program Description

PVsyst is a computer software developed by Geneva University in Switzerland for the study of photovoltaic systems. This program is commonly used by architects, engineers, and researchers in the photovoltaic industry. PVsyst can only operate on Windows, but users with Mac or Linux can use Remote Desktop or a virtual running machine to run the program [10]. In PVsyst, the user is capable of simulating grid-connected, stand-alone, solar-powered pumping systems, and DC grid systems. PVsyst also provides users with a large database of meteorological data, and individual system components, which makes designing and simulating PV systems convenient. The program's main operating language is English, but the report can be exported in English, French, Italian, Spanish, and German. The scope of the program includes feasibility, 3D shading analysis, loss analysis, and system yield of the complete PV systems. Detailed instructions on setting up and simulating photovoltaic projects on PVsyst can be found in Appendix D.

3.2 Purpose and Importance

For this study, PVsyst was used to evaluate the changes in the range of tracking angle and the annual energy produced concerning different pitch distances. To validate the accuracy of PVsyst, simulated energy produced was compared to the actual energy produced on October 31st, 2019, found in GPM Portal. This date was chosen after visiting the site and speculating the module conditions. On October 31st, the sky was clear throughout the day, and modules were recently cleaned, reducing the loss due to soiling.

To ensure similar irradiance conditions for simulation in PVsyst, measured global horizontal irradiance (GHI) data from the GPM portal was used to compute direct normal irradiance (DNI) and direct horizontal irradiance (DHI) using the Perez Model [11]. Then those meteorological data were imported to PVsyst as a weather file for simulation. With accurate insolation data and surface conditions of modules, PVsyst can accurately predict the behavior of the arrays tracking motion and expected energy output of individual inverters of the solar farm.

3.3 Strength and Weakness

After using PVsyst for this study, there were some identifiable strengths and weaknesses for this program. One of the strengths of PVsyst is the availability of databases for weather and components such as PV modules and inverters. With a variety of options for components, building the system for simulation is fast and convenient. Simulation windows are well-organized and displayed for users to easily navigate to any aspect of the system including near-shading, orientations, and system configuration.

The second strength of the PVsyst is its accuracy and depth of analysis for system simulations. Simulations in PVsyst are done in hourly steps for the entire year, given that weather data is available in hourly steps. Users can view power and energy generated from the PV system in the same resolution as the simulation. Also, PVsyst is known to be commonly used in the industry, and reports generated from PVsyst are used for banks and financiers for revenue generation. This shows the public dependency on PVsyst for its accuracy and detailed analysis of PV systems.

One of the weaknesses of PVsyst is that for near shading analysis, PVsyst cannot simulate backtracking on an uneven orientation of the PV modules. The PVsyst team is currently working on implementing backtracking on a sloped surface, but for now, it is not possible to model rough terrain surfaces with tracking modules. This limits the shading simulation to flat ground, but it is important to note that fixed-tilt arrays can be simulated on uneven terrain. Because PVsyst could not simulate backtracking on uneven terrain, accurate shade analysis and simulation of the Cal Poly Solar Farm through PVsyst was not possible.

The second weakness of PVsyst is its lack of current weather data in the database. Although recent meteorological data is available, PVsyst requires users to purchase them within the program. Most of the available meteorological data are measured in 2014 or older. But for this study, current data was required for comparison with simulation results. Other weaknesses include PVsyst not allowing users to input more than two decimal points for PV array location and various input parameters. Limiting the number of decimal points of input parameters results in imprecise simulation results and higher error values between modeled and actual energy generated from the system.

Chapter 4

PV_LIB TOOLBOX

4.1 Program Description

PV_LIB Toolbox is a set of well-documented functions for simulating the performances of PV projects. This program was developed at Sandia National Laboratory, which a group of PV professionals called Photovoltaic Performance Modeling Collaborative (PVPMC). This group's main goal is to bring traceability and transparency to the process of PV system modeling and encourage third party validation of existing algorithms [12]. This program is available in both MATLAB and Python for free on their website, and the website also includes detailed descriptions of all the built-in functions. Examples for users are available to utilize and understand the function's capabilities. For this study, the MATLAB version of the PV_LIB Toolbox was used.

4.2 Built-in Functions

There are 7 different categories of built-in functions in PV_LIB: Example Scripts, Time & Location Utilities, Irradiance, and Atmospheric Functions, Irradiance Translation Functions, Irradiance Analysis Functions, Photovoltaic System Functions, Functions for parameter estimation for PV module models, and Numerical Utilities. With multiple combinations of built-in functions and a TMY3 file, users can evaluate irradiance, weather, and performance of a single-axis tracking PV system. Test Script provided by PV_LIB demonstrates how to use the built-in functions to model a PV system. Some of the built-in functions used for this study are explained further in the following section.

4.2.1 pvl_calcparams_PVsyst

This built-in function uses PV module modeling used by PVsyst to calculate five parameters for current-voltage (IV) curves [13,14,15]. This function was used to find IV curve parameters for both Trina 345W TSM DE-14A(II) and REC 345W Twinpeak 2S 72 panels, which were used in the Cal Poly Solar Farm. Module parameters were found in the module database of PVsyst, and the cell temperature and the effective irradiance are provided by the user.

Inputs

- **S** - The effective irradiance (in W/m^2) absorbed by the module. S must be ≥ 0. May be a vector of the same size as Tcell. Due to a division by S in the script, any value equal to 0 will be set to 1E-10.
- **Tcell** - The average cell temperature of cells within a module in C. Tcell must be ≥ -273.15. May be a vector of the same size as S.
- **alpha_isc** - The short-circuit current temperature coefficient of the module in units of A/C (or A/K).
- **ModuleParameters** - a struct with parameters describing PV module performance at reference conditions. The ModuleParameters struct must contain (at least) the following fields:
- **ModuleParameters.gamma_ref** - diode (ideality) factor at reference conditions (unitless).
- **ModuleParameters.mugamma** - temperature dependence of gamma (1/C).
- **ModuleParameters.IL_ref** - Light-generated current (or photocurrent) in amperes at reference conditions. IL is referred to as Iphi in some literature.
- **ModuleParameters.I0_ref** - diode reverse saturation current in amperes at reference conditions.
- **ModuleParameters.Rsh_ref** - shunt resistance at reference conditions (ohms)
- **ModuleParameters.Rsh0** - shunt resistance at zero irradiance (ohms).
- **ModuleParameters.Rshexp** - exponential factor defining decrease in Rsh with increasing effective irradiance.
- **ModuleParameters.Rs_ref** - series resistance at reference conditions (ohms).
- **ModuleParameters.eG** - The energy bandgap at reference temperature (eV), must be >0.
- **Sref** - Optional reference effective irradiance in W/m^2. If omitted, a value of 1000 W/m^2 is used.
- **Tref** - Optional reference cell temperature in C. If omitted, a value of 25 C is used.

Output:

- **IL** - Light-generated current in amperes at irradiance=S and cell temperature=Tcell.
- **I0** - Diode saturation curent in amperes at irradiance S and cell temperature Tcell.
- **Rs** - Series resistance in ohms at irradiance S and cell temperature Tcell.
- **Rsh** - Shunt resistance in ohms at irradiance S and cell temperature Tcell.
- **nNsVth** - modified diode (ideality) factor at irradiance S and cell temperature Tcell. nNsVth is the product of the usual diode (ideality) factor gamma, the number of series-connected cells in the module Ns, and the thermal voltage Vth of a cell in the module at a cell temperature of Tcell.

Figure 4.1: Inputs and Outputs of pvl_calcparams_PVsyst

A detailed explanation of the inputs and outputs for pvl_calcparams_PVsyst is provided in figure 4.1 [16]. Multiple module parameters are required to accurately find the current and voltage behavior of the module. Light-generated current, diode saturation current, series resistance, shunt resistance, and modified diode factors are the output of this function, and they are used to calculate the IV curve for solar panels in user-defined cell temperature and effective irradiance.

4.2.2 pvl_singlediode

This built-in function solves the single diode equation

$$I = I_L - I_0 \left(\exp\left(\frac{V+IR_s}{nNsVth}\right) - 1\right) - \frac{V+IR_s}{R_{sh}} \quad (4.1)$$

for current and voltage using the outputs from pvl_calcparams_PVsyst, mentioned in the previous subsection [17]. After specifying the desired number of data points, this function outputs a result of current and voltage describing the entire IV curve with MPPT points labeled for each irradiance.

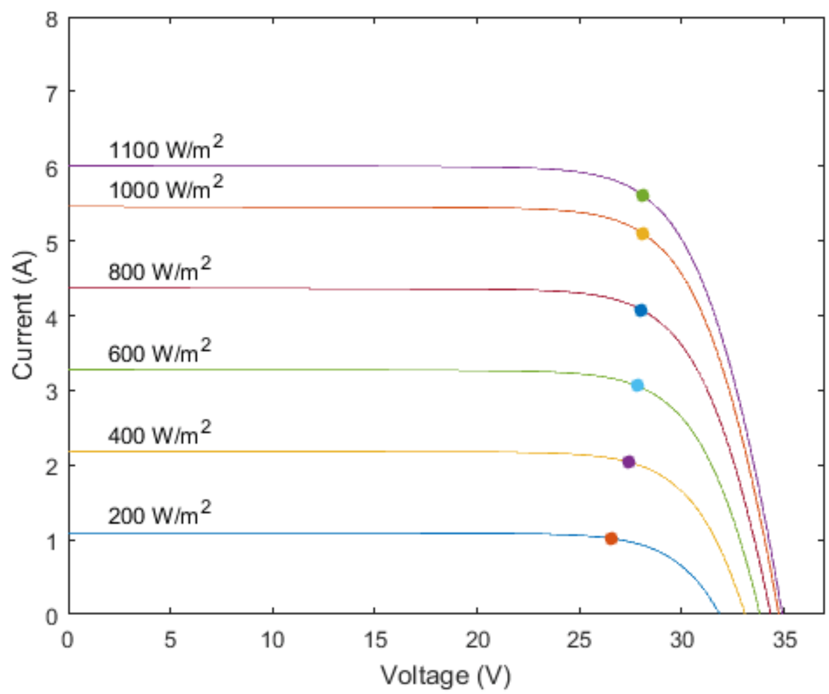


Figure 4.2: Example IV Curve Using pvl_singlediode

An example of an IV curve generated using the pvl_singlediode function is shown in figure 4.2. Users can either use a single irradiance or multiple irradiance conditions to identify different MPPT points at each irradiance. For this study, this function was to use to identify the MPPT current and voltage of panels used in the Cal Poly Solar Farm.

4.2.3 pvl_ephemeris

pvl_ephemeris calculates the position of the sun using location and time specified by the user. This built-in function outputs suns' azimuth angle, elevation angle, apparent elevation angle, and solar time. The location of the sun's position changes daily, and knowing the exact position is essential for tracking algorithms, so that the PV systems can output a maximum amount of power. For more accurate results, atmospheric pressure and temperature can be inputted, but if those two values are not defined, the built-in function assumes pressure to be 1 atm and temperature to be 12 degrees Celsius.

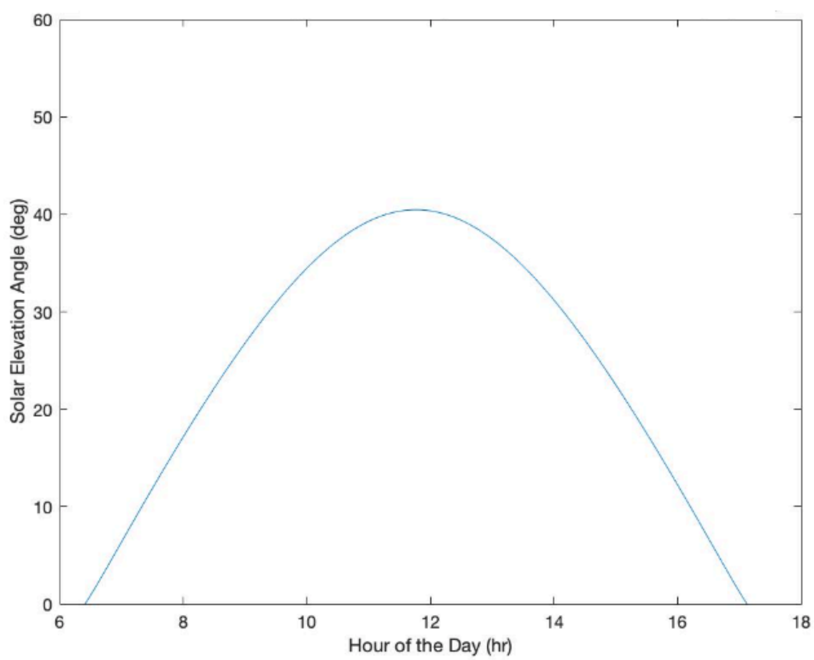


Figure 4.3: Solar Elevation Angle in San Luis Obispo Using pvl_ephemeris

Figure 4.3 shows the solar elevation angle in October of 2019 in San Luis Obispo generated by using pvl_ephemeris. Knowing the elevation angle of the sun throughout the day is important because it is related to the amount of irradiance on the panel. Lower solar elevation results in less irradiance on the panel due to the thicker airmass.

4.2.4 pvl_singleaxis

This built-in function utilizes zenith angle, azimuth angle, axis tilt angle, axis azimuth angle, and ground coverage ratio to output rotation angle of tracker, angle of incidence, surface tilt angle, and surface azimuth angle. Zenith angle and azimuth angle of the sun can be found using pvl_ephemeris as mentioned in the previous subsection. Users can decide to use backtracking by defining GCR for the system. All the outputs, including the tracking angle, is calculated using equations found in "Tracking and Back-Tracking" by Lorenzo, E [18].

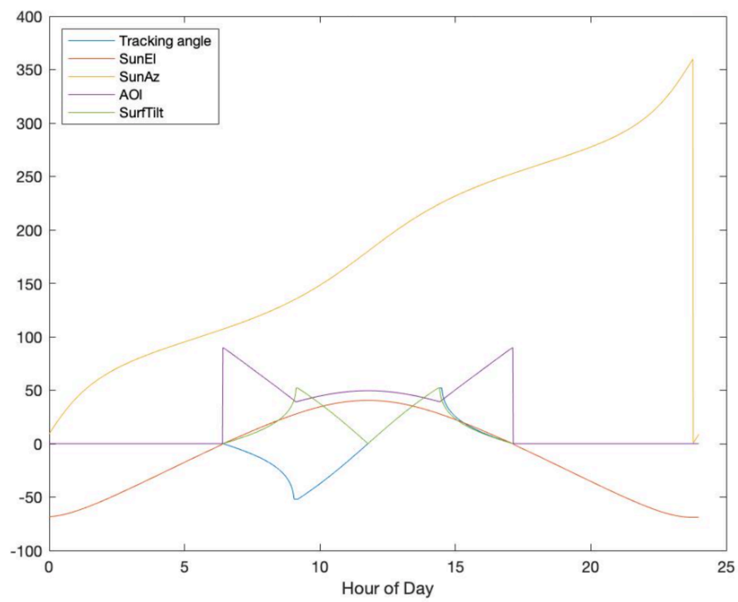


Figure 4.4: Behavior of Single Axis Tracker in San Luis Obispo Modeled Using pvl_singleaxis

Figure 4.4 represents the different output created from pvl_singleaxis for a PV system with backtracking on October 31st in San Luis Obispo. Using this built-in function, the behavior of the tracker and an exact rotation angle can be modeled. It is also important to note that many trackers used in the industry follow the same backtracking algorithm found in this built-in function.

4.3 Strength and Weakness

There are many strengths to PV_LIB, but its greatest strength is the well documented built-in functions. All the built-in functions have well-explained purpose, inputs, outputs, and they also provide examples on how to use them for the users. Documentation and examples allow users to utilize the functions with confidence and even allow users to manipulate and alter the code. As mentioned in the mission statement for PVPMC, they made this program transparent and traceable for users to understand the process of modeling PV systems and included their sources for every calculation done in each built-in function. PV_LIB also works closely with the National Renewable Energy Laboratory (NREL) [19] and acquires module and inverter data from System Advisor Model (SAM) [20], which shows a national institution's support for this program. Other strengths include its cost-effectiveness and versatility. Unlike other expensive PV system modeling programs, PV_LIB is completely free to use, and it also comes in two programming languages, MATLAB and Python.

A few weaknesses to this program would be in shifting the time of the sunrise and sunsets due to the daylight savings time. San Luis Obispo's highest solar elevation was at 1 pm on August 27th, but pvl_ephemeris had difficulties when the time was shifted manually. Subsequently, any built-in functions that utilize elevation angle and azimuth angle also had issues with time shift. These issues were also found in different PV modeling programs and the only assumption for this issue is that the sun's highest elevation is normalized to 12 pm for all of these programs. Other than this minor issue, PV_LIB did not have any other weaknesses.

Chapter 5

EFFECT OF SHADING

5.1 Effect on System Performance

Small shading on a solar panel can significantly affect the performance and the power output of a PV array. Because individual solar cells are connected in series, the maximum possible current for the entire panel is limited to the lowest current produced by any cell on that solar panel.

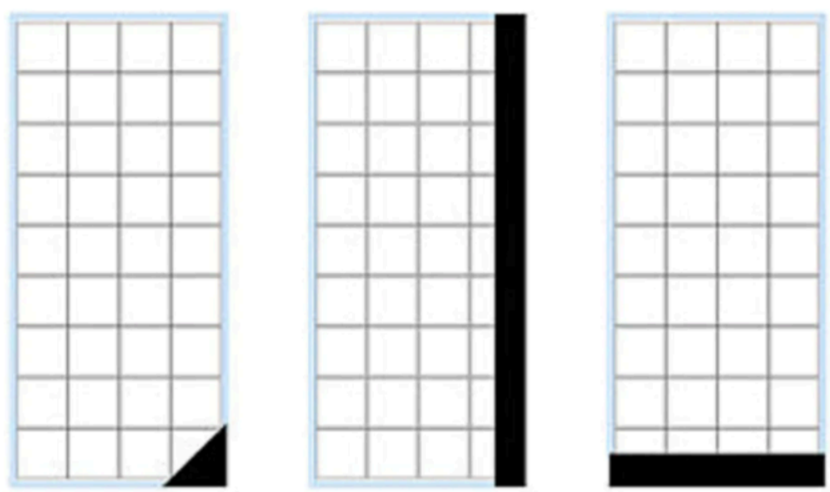


Figure 5.1: Partial Shading on Solar Panels

Figure 5.1 [21] represents three solar panels with different partial shading conditions on each panel. It is important to note that all of the panels below are producing exactly half of the power that is possible, since the partial shade is covering half of the PV cell on each panel. This figure shows the significant effect of a small partial shading on the power generation of solar panels.

To prevent significant power loss from small amount of shading, bypass diodes are installed on every solar panel to redirect the flow of current and minimize power loss. Most of the industrial size solar panels are divided into three substrings with one bypass diode per substring. When there is partial shading on one of the substrings, the corresponding bypass diode activates to prevent significant loss of overall power. However, when a bypass diode is activated, that panel loses a third of the power, since the bypass diode redirects the current to avoid the substring with a shaded cell. Instead of producing half of the possible power without a bypass diode, that solar panel can produce about 66% of the power with an activated bypass diode.

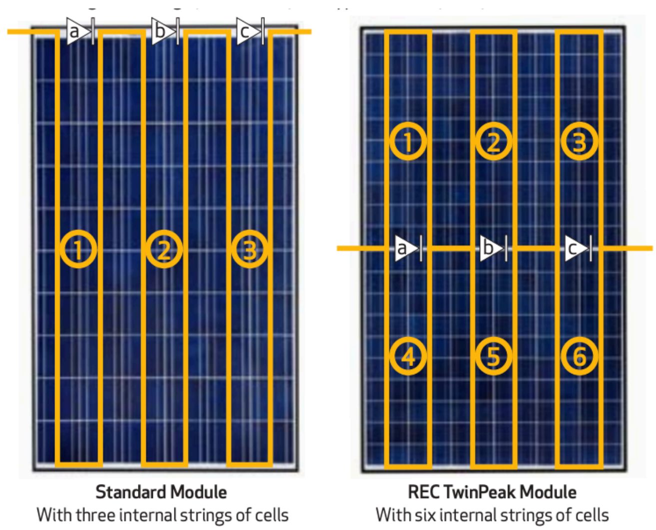
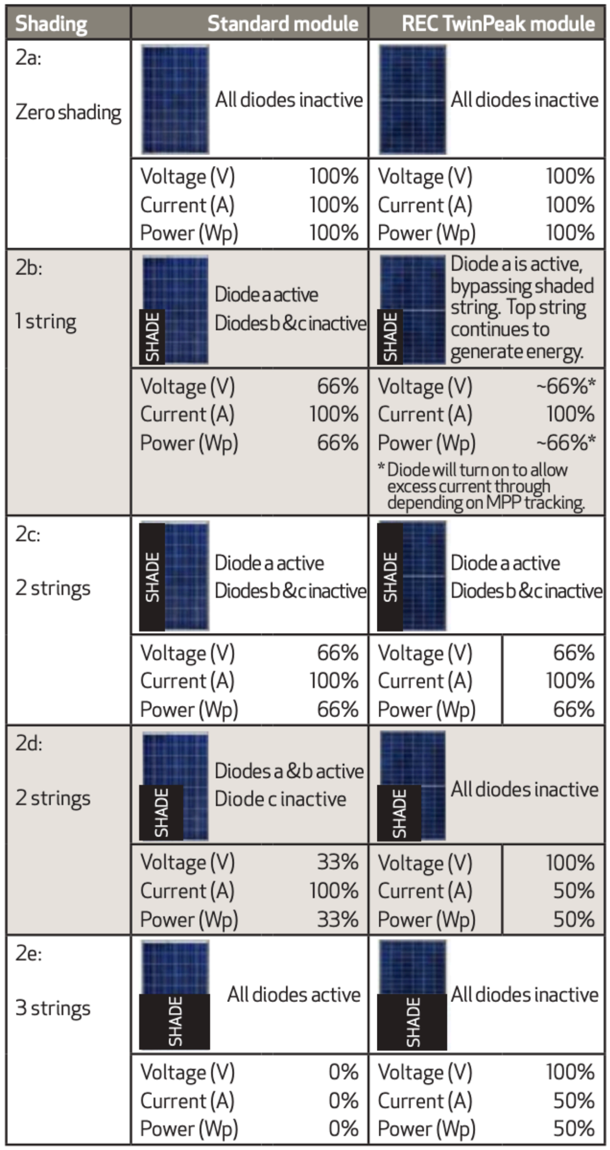


Figure 5.2: Substring and Bypass Diode Layout of a Full-Cut Cell Module (left) and a Half-Cut Cell Module (right)

Figure 5.2 [22] represents two types of solar panels, a full-cut cell module on the left and a half-cut cell module on the right. Individual substrings are labeled with numbers, and bypass diodes are labeled with the alphabet on both modules. It is important to note that the half-cut cell module has twice as many substrings as the full-cut cell module, but it has the same amount of bypass diodes. Half-cut modules are divided into two groups of PV cells, with one group placed on top of the other group. Those groups are connected in parallel to create one solar panel, which means current splits when it enters the solar panel and recombines when it exits the solar panel. The benefit of having solar panels divided into top and bottom is, when the bottom half of the solar panel is shaded by the adjacent row, only the performance of the bottom half is affected [23].



Shading	Standard module	REC TwinPeak module
2a: Zero shading	All diodes inactive	All diodes inactive
	Voltage (V) 100% Current (A) 100% Power (Wp) 100%	Voltage (V) 100% Current (A) 100% Power (Wp) 100%
2b: 1 string	SHADE Diode a active Diodes b & c inactive	SHADE Diode a is active, bypassing shaded string. Top string continues to generate energy.
	Voltage (V) 66% Current (A) 100% Power (Wp) 66%	Voltage (V) ~66%* Current (A) 100% Power (Wp) ~66%* * Diode will turn on to allow excess current through depending on MPP tracking.
2c: 2 strings	SHADE Diode a active Diodes b & c inactive	SHADE Diode a active Diodes b & c inactive
	Voltage (V) 66% Current (A) 100% Power (Wp) 66%	Voltage (V) 66% Current (A) 100% Power (Wp) 66%
2d: 2 strings	SHADE Diodes a & b active Diode c inactive	SHADE All diodes inactive
	Voltage (V) 33% Current (A) 100% Power (Wp) 33%	Voltage (V) 100% Current (A) 50% Power (Wp) 50%
2e: 3 strings	SHADE All diodes active	SHADE All diodes inactive
	Voltage (V) 0% Current (A) 0% Power (Wp) 0%	Voltage (V) 100% Current (A) 50% Power (Wp) 50%

Figure 5.3: Behavior of a Full-Cut Cell Module (left) and a Half-Cut Cell Module (right) Under Different Shading Conditions

Behavior of a full-cut cell module and a half-cut cell module are represented in the figure 5.3 [22]. A half-cut cell module can produce at least 50% of the power even when the entire bottom half of the panel is shaded. While the full-cut cell module typically loses all power due to the horizontal shading from an adjacent row.

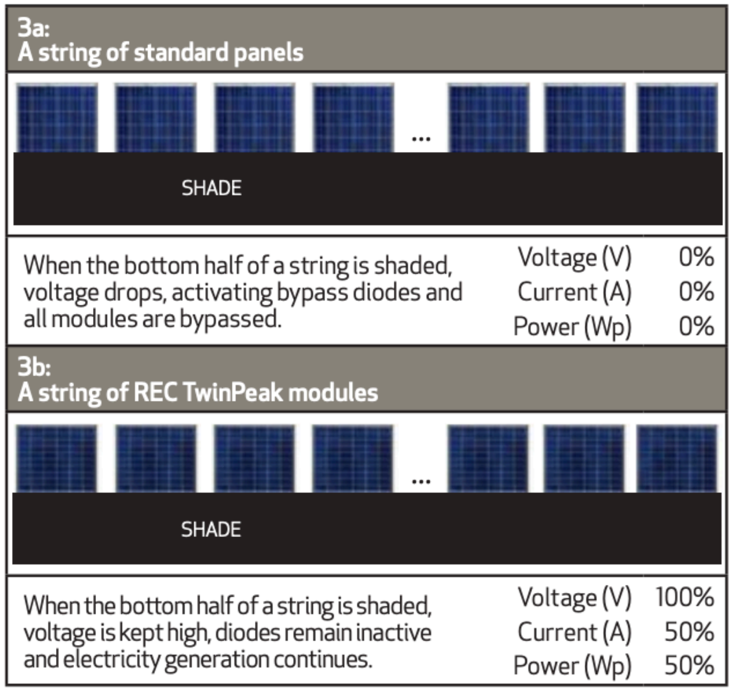


Figure 5.4: Behavior of a Full-Cut Cell Module (top) and a Half-Cut Cell Module (bottom) Under Horizontal Shading

The advantage of the half-cut module is further explained in figure 5.4 [22], where the performance of both full-cut modules and half-cut modules in a string is affected by the horizontal shading across all the panels. As shown in the figure, the power output of full-cut modules in a string is eliminated by activated bypass diodes due to shading, while strings of half-cut modules are producing half of the power. It is evident that half-cut modules are advantageous during inter-row shading, but the smallest amount of shading will have a significant impact on the power output of both types of solar panels [22]. It is important to note that the bypass diodes activate when any cell in a corresponding substring is covered more than thirty percent of its area by shading [24]. Therefore, a horizontal shade that is covering roughly four percent of a solar panel's surface is enough to activate all the bypass diodes and eliminate any power output from a panel.

5.2 Shading on the Gold Tree Solar Farm

5.2.1 Shade Measurements

Due to uneven ground levels and elevation of the rows, some of the zones in the Gold Tree Solar Farm are experiencing inter-row shading. Inter-row shading impacts the performance of the solar panels, especially for the full-cut cell modules, as mentioned in the previous section. Most of the inter-row shading occurs when the sun's elevation is low, such as during sunrise and sunset. Rows installed on zones with a sloped ground experience inter-row shading the most. For example, zone Z1 has a slope that faces East, so there isn't any adjacent shading during sunrise. However, most of the strings in zone Z1 are affected by the inter-row shading during sunset as shown in figure 5.5.

Figure 5.5: Inter-row Shading on Solar Panels Connected to Inverter 55 facing North (left) and Inverter 25 facing South (right) taken on 4/23/20

Figure 5.5 shows the inter-row shading on the modules during sunset on April 23^{rd} of 2020. After visiting the site, the performance of inverter 55 (full-cut cells) and inverter 25 (half-cut cells) were chosen to be modeled due to the significant amount of shading.

Height, length, and shift measurements of shade at two separate times were taken for the three rows corresponding to inverter 55 and inverter 25. Shade shift represents the distance of the shade from the edge of the first solar panel. It is caused by uneven ground levels between two adjacent rows, such as the zone of inverter 25 and the azimuth of the sun. Shade measurements are important for modeling PV array's performances, since they can impact the performance of the panels, as explained in the previous section.

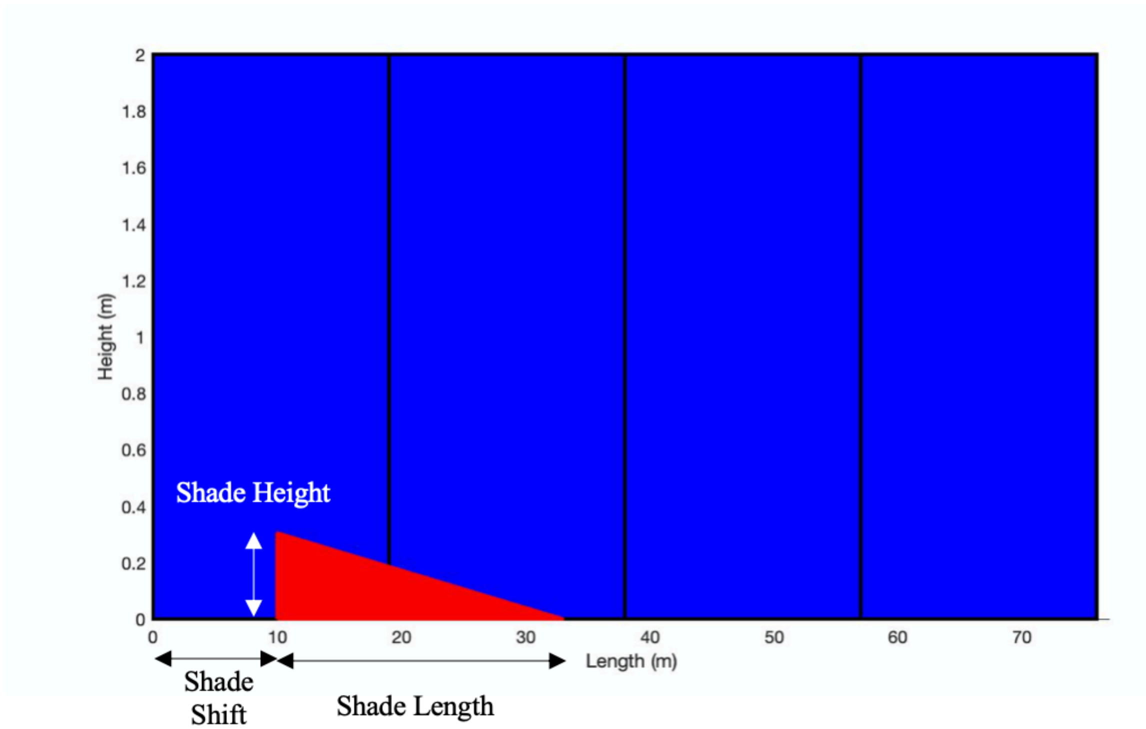


Figure 5.6: Defining Height, Length, and Shift Measurements of the Shade. The Red Triangle is the inter-row shade.

Figure 5.6 was created to visually represent the height, length, and shift measurements of the shade on a row of solar panels. Blue rectangles shown in the figure represents the row of solar panels, and the red triangle represents the inter-row shading on the panels. Shade height and length are a measurement of the height and length of the triangle. The shade shift represents the distance of the shade from the closest edge of the panel. All the measurements are in meters, and they are used as inputs for the PV performance model. Shade measurements for both inverter 55 and inverter 25 are summarized in table 5.1 and 5.2.

Table 5.1: Shade Height, Length, and Shift Length on Inverter 55

Inverter 55 on 4/23/20						
Time	**4:37 pm**			**5:17 pm**		
Row	**1**	**2**	**3**	**1**	**2**	**3**
Shade Height (m)	0.127	0.1397	0.1397	0.3048	0.3048	0.2921
Shade Length (m)	26	29	23	33	35	34
Shade Shift (m)	0	0	0	0	0	0

Table 5.2: Shade Height, Length, and Shift Length on Inverter 25

Inverter 25 on 4/23/20						
Time	**5:31 pm**			**5:38 pm**		
Row	**1**	**2**	**3**	**1**	**2**	**3**
Shade Height (m)	0.1778	0.1524	0.1016	0.1905	0.1778	0.127
Shade Length (m)	20	29	23	33	35	34
Shade Shift (m)	0.4953	0.5334	0.4572	0.4445	0.508	0.4318

5.2.2 Performance of Shaded PV Strings

After visiting the Gold Tree Solar Farm and taking shade measurements for both inverters, MPPT current and voltage of each row for those inverters were found using the GPM portal. Although all rows experience a similar cell temperature and irradiance at an instance, rows perform differently depending on the amount of shade exposure. It is important to note that the inverters used in the Gold Tree Solar Farm have 3 MPPT inputs, and each row equals one MPPT input. Because the GPM portal does not track individual power produced by each MPPT input, power is calculated by multiplying MPPT current and voltage for each MPPT input. The actual performance data collected from the field was compared to the modeled data to validate the result.

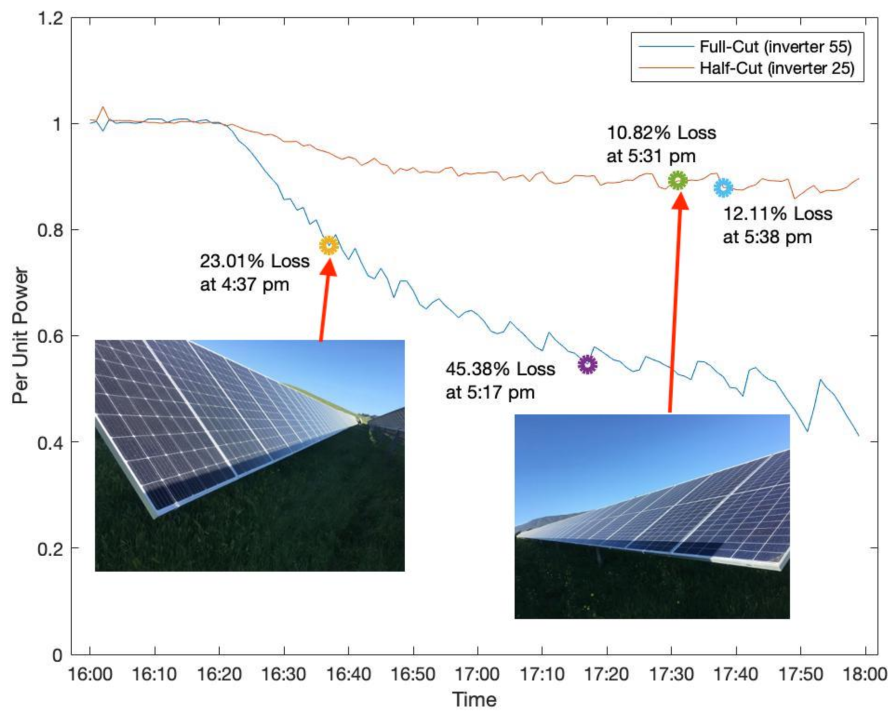


Figure 5.7: Per Unit Power of Full-Cut Module and Half-Cut Module from 4 pm to 6 pm on 4/23/20

Figure 5.7 shows the per-unit output of full-cut module (inverter 55) and full-cut module (inverter 25) to represents the photovoltaic system performance during the hours when the elevation of the sun is decreasing. Per-unit output is the normalized power output of both types of modules. Within an hour, the inverter with full-cut modules experiences a 50 percent decrease in power output, and the inverter with half-cut modules experiences a 12 percent decrease in power output due to inter-row shading. Even a slight inter-row shading during sunset, as shown in the figure with the photos of inter-row shading, is heavily responsible for the reduction in energy generation of the Cal Poly Gold Tree Solar Farm.

Chapter 6

PERFOMRANCE MODELING OF INTER-ROW SHADED PANELS

6.1 Introduction

This PV system performance modeling tool is written in MATLAB, and it models the performance of solar panels with a triangular shading. It is capable of calculating MPPT current, voltage, and power output from a single MPPT input of an inverter from the Cal Poly Gold Tree Solar Farm. Also, a few assumptions were used while modeling the MPPT point of the panels. All working panels were assumed to be operating at the same MPPT point and ignoring mismatch caused by uneven soiling and module degradation within a string.

6.2 Input

There are several inputs required for the program: height, width, and shift of the triangular shadow, ambient temperature, and effective irradiance. These factors impact the MPPT point of the IV curve in different ways. The height, width, and shift of the triangular shadow impacts the performance of the solar panels and activation of the bypass diodes by restricting the flow of the current. The temperature of the cell is inversely related to the open-circuit voltage of the panels, since solar cells are more efficient in colder temperatures. The effective irradiance is directly related to the short-circuit current. Measured data from table 5.1, table 5.2 and table 6.1 was used as inputs for the system performance model.

Table 6.1: Ambient Temperature and POA of Inverter 55 and 25

Inverter	55		25	
Time (4/23/20)	4:37 pm	5:17 pm	5:31 pm	5:38 pm
Ambient Temperature (C)	18	18	18.1	18.1
POA Irradiance (W/m^2)	832	650	608	600

6.3 Module Characteristics

Using thermal properties of Trina TSM-DE14A(II) and REC Twinpeak 2S 72 solar panels from PVsyst database and PV_LIB built-in functions called pvl_calcparams_PVsyst, MPPT current and voltage of the solar panels are determined and used to calculate overall power output. Because full-cut cell modules and half-cut cell modules have different thermal properties and behave differently to shading, there are separate MATLAB models for each type of module.

6.4 Cell Temperature and Direct Irradiance Calculations

The cell temperature sensor on the Gold Tree Solar Farm was malfunctioning, so the ambient temperature was used to estimate the cell temperature. The equation shown below was used to calculate the cell temperature from the ambient temperature given as a user input.

$$T_{cell} = T_{ambient} + \left((T_{NOTC} - 20) * \frac{Effective\ Irradiance}{800} \right) \tag{6.1}$$

It is important to note that this equation is an estimation of cell temperature [25], and it does not take wind velocity and other factors that contribute to change in cell temperature into account. Ambient temperature and effective irradiance are given as the input for users.

Temperature at nominal operating cell temperature (NOCT) for moderately efficient modules is considered to 48 degrees [26].

Finding the correct effective irradiance required a reverse engineering. The measured effected irradiance is not adequate to be used, because the pyranometer is installed on a tracker in zone Y4. Each zone has a different backtracking angle and ground elevation due to the topography of the field, which results in a different direct irradiance from the sun. Therefore, finding the effective irradiance on the modules nearby using the reverse engineering method was more accurate than the measured POA from the GPM portal.

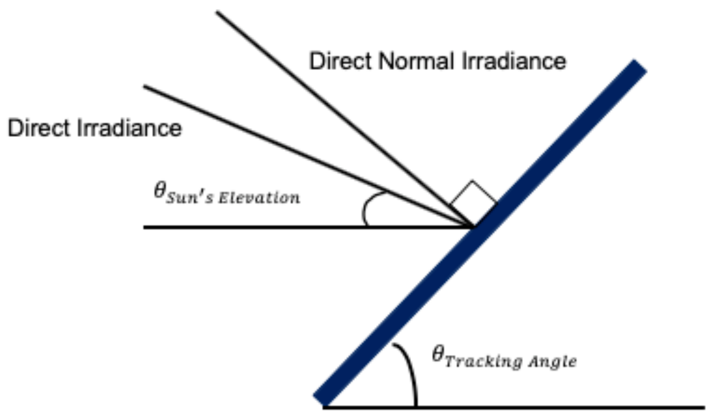


Figure 6.1: Direct Normal Irradiance on a Solar Panel Relative to the Sun's Elevation and the Tracking Angle

The direct irradiance from the sun was calculated by solving I_{Direct} in equation 6.2 using the POA irradiance and tracking angle of zone Y4 from the GPM portal. Suns elevation angle was found using pvl_ephemeris from PV_IB Toolbox.

$$DNI = I_{Direct} * \sin\left(\theta_{Sun's\ Elevation} + abs(\theta_{Tracking\ Angle})\right) \tag{6.2}$$

Then direct normal irradiance of the different zones was found using equation 6.2 with the I_{Direct} of the sun and respective $\theta_{Tracking\ Angle}$ of each zone. It is important to note that distortion in the ground level can also affect the amount of irradiance received by the panel at each zone.

6.5 Decimal Place Adjustment for Shifting Shade

To accurately represent the effect of the shade shift in the model, a series of statements were made to adjust the length of the shade shift. These statements were needed because the modeling program separates the panel into three substrings and sets each substring as a unit space. Since the shade shift is not aligned with the unit space, the program breaks down the shift length to decimal places to find the starting point of the shade with respect to the unit space.

Shifting shade statements find which substring the shade is starting from and detects when the shade is covering more than thirty percent of the substring. If the shade is covering more than thirty percent of the substring, the program neglects the unit space. Otherwise, the program counts the unit space that the shade is starting from, since it does not affect the performance of the string. These statements read up to the thousandth decimal place of the shade shift.

6.6 Maximum Current Calculation

This MATLAB model detects when the shadow is covering thirty percent of the cells on the bottom row and assigns zero to the substring to represent its inactivity due to shading. If the shade is covering less than thirty percent of the cell, an estimate of the current is calculated based on the ratio of the shade height. The equation used to calculate the current of the shaded panel based on the ratio of the shade height is shown below.

$$Current_{Shaded} = Current_{mppt} * \left(1 - \frac{Shade\ Height - 0.0364}{0.1524}\right) \tag{6.3}$$

0.0364 m shown in the equation is the summation of the width of aluminum edge guard, and the white space between the cells on the bottom, and 0.1524 m represents the total length of a full-cut cell. The fraction represents the percentage of the solar cell that is shaded. Then one is subtracted by the fraction to calculate the percentage of the cell's maximum current output. That number is multiplied to the MPPT current at specified irradiance and cell temperature to calculate the current output of a shaded panel.

The same process was used to calculate for the current of half-cut modules using equation 6.4.

$$Current_{Shaded} = \frac{Current_{MPPT}}{2} * \left(1 - \frac{Shade\ Height - 0.0254}{0.0762}\right) \tag{6.4}$$

Half of the MPPT current is multiplied to the ratio because the current is divided into halves for the half-cut modules. The top half of the module is assumed to be operating at its full capacity except when the shade is covering the first substring. In this situation, the bypass diode is on for both top and bottom substring to reflect the behavior of the panel shown in figure 5.3.

The program individually checks the height of the shade on 228 substrings and assigns the current that the substring can produce according to its logic. Then the program finds the maximum current produced by worst shaded solar cells for each string and considers that current to be the maximum current that the string can produce.

6.7 Finding Single MPPT Point

To find the MPPT point at a specified time, an IV curve depicting the row of PV modules is needed. This IV curve represents the behavior of the shaded PV modules, and allows the model to predict the MPPT point. First, the program filters out the shaded panels, and counts active panels in each string. Then, the data points of the IV curves are generated using the number of active panels in each string and a PV_LIB built-in function called pvl_singlediode. The generated current data is multiplied by the ratio between the shaded maximum current and MPPT current without shading to adjust the current data. Finally, the current data points for each string are added together to make a single IV curve for the row. Complete MATLAB models for full-cut cell and half-cut cell can be found in Appendix E and Appendix F.

Chapter 7

RESULTS

7.1 PVsyst Results

When compared to the annual energy produced by an inverter without shading, an inverter with inter-row shading loses about 9.6 MWh per year. Currently, there are many inverters in the Cal Poly solar farm that are affected by the inter-row shading, and they are largely responsible for the solar farm's inability to generate the expected energy output. To reduce inter-row shading, changing the pitch distance in the tracking system algorithm to backtrack earlier is considered. Before implementing this appraoch, loss of energy from backtracking earlier as compared to the amount of energy lost from inter-row shading was studied.

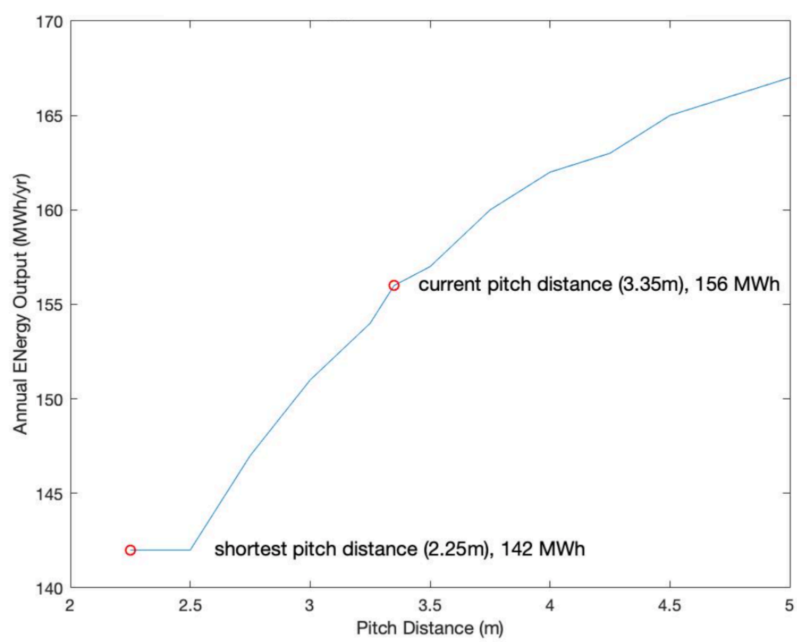


Figure 7.1: Annual Energy Produced by Different Pitch Distances in PVsyst

Figure 7.1 represents PVsyst simulated data of annual energy produced by an inverter from the Gold Tree Solar Farm on different pitch distances. With a larger pitch distance, the panels can be tracking and facing the sun for a longer period of time and generate more power than panels forced to go on backtracking mode due to the proximity of the neighboring row. As shown in the figure, this pitch distance can be directly related to the energy production, until the pitch distance is between 2.25 meters and 2.5 meters. At those pitch distances, the energy production plateaus, because the software assumes that the panels are too close together and they would create inter-row shading if they were actually tracking the sun. Also, if the pitch distance introduced in the software is longer than 2.75 meters, loss due to inter-row shading is greater than the loss due to shorter pitch distances. Pitch distances greater than 3.35 meters will result in larger inter-row shading and significantly reduce the power production by not backtracking fast enough.

It is important to note that, in PVsyst simulations, the rows of PV modules are separated by the user-defined pitch distance. On the other hand, the Gold Tree Solar Farm was built with a pitch distance of 3.35 meters for every row. To change the backtracking behavior of individual zones, the user-defined pitch distance in the tracker control website can be updated. For example, by inputting a pitch distances less than 3.35 meters in the tracker control website, the rows backtrack earlier than the preselected pitch distance. Therefore, selecting a pitch distance between 2.75 meters and 3.35 meters will be helpful for efficiently optimizing the backtracking algorithm.

7.2 PV_LIB Results

The backtracking algorithm used in the Gold Tree Solar Farm has to be identified and studied to decide on an optimum pitch distance and reduce inter-row shading. PV_LIB has a built-in MATLAB function called pvl_singleaxis, and one of its outputs is the singleaxis tracking angle with backtracking capabilities. The algorithm used in pvl_singleaxis is an algorithm from a research called "Tracking and Backtracking" [18]. The main input for the tracking algorithm is ground coverage ratio, which is the ratio of module length to pitch distance. With the module height of 1.956 meter and pitch distance of 3.35 m, Gold Tree Solar Farm's GCR is identified to be 58.5 percent.

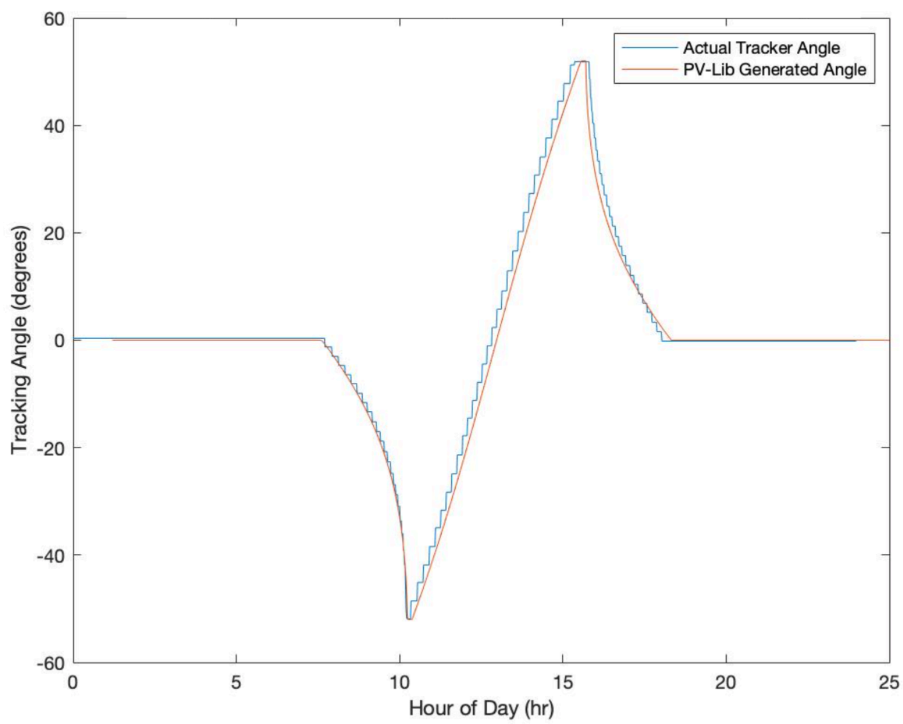


Figure 7.2: Actual Panel Tracker Angle Compared to PV_LIB generated tracker angle with backtracking.

As shown in figure 7.2, the actual tracker angle was plotted against PV_LIB generated tracker angle using GCR of 58.5 percent. The plot of the actual tracker angle is represented as a step function, because the tracker at the solar field maintains its position for five minutes. The backtracking algorithm used for the solar field appears to be similar after reviewing the actual and generated tracker angle.

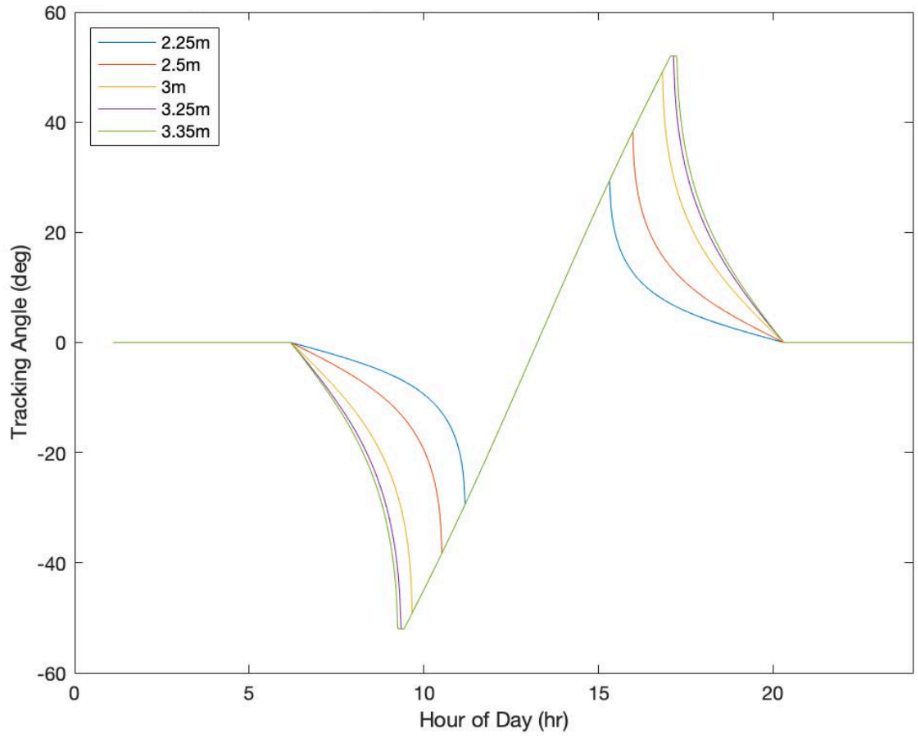


Figure 7.3: Varying Backtracking Angle with Respect to Pitch Distance

After identifying the backtracking algorithm used in the Gold Tree Solar Field, a relationship between different pitch distances and tracking angle was identified. As shown in figure 7.3, a pitch distance of 2.25 meters, 2.5 meters, 3 meters, 3.25 meters, and 3.35 meters were tested to verify the behavior of the tracking angle. Only pitch distances of 3.25 meters and 3.35 meters reached the full range of preset tracking angles, which is defined to be -52° to +52°. As pitch distance decreased from 3.25 meters, the tracking angles peaked at smaller angles, and the panels were backtracking earlier, allowing the solar panels to flatten out faster. Most inter-row shadings occur when the sun's elevation is low, and the tracking angle is steep. By allowing the solar panels to reduce its tracking angle during the times when the sun's elevation is low, backtracking can significantly reduce inter-row shading on the adjacent rows.

7.3 MATLAB System Model Results

The MATLAB system model outputs two figures for each row of solar panels; a visual representation of triangular shade on the panels and current-voltage (IV) and power-voltage (PV) curves for visualization of system performance. As shown in figure 7.4, each blue rectangle represents one string of PV modules, and it consists of 19 solar panels. There is a total of 4 strings, connected in parallel, in each row. The red triangle on the blue rectangles represents the measured shading created by the adjacent row of solar panels. The measured shading is assumed to be perfectly triangular for faster computation, but it is important to note that the slope of the ground can distort the shape of the shade.

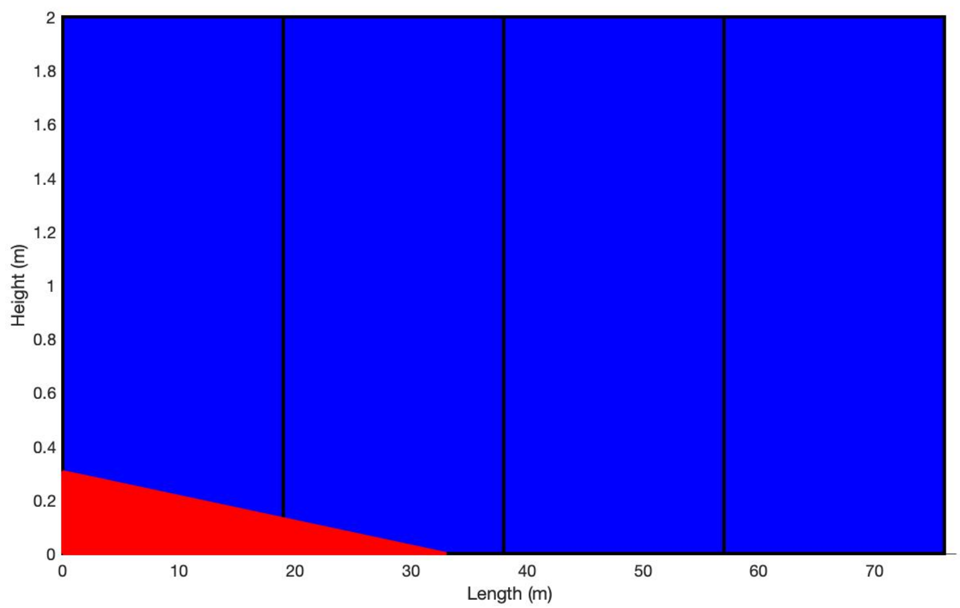


Figure 7.4: Visual Representation of Triangular Shade on Panels

The power output and the MPPT operating point is marked on the IV and PV curve for each row. IV curves for each string are compiled to form a complete IV curve of a row. Instead of using an MPPT tracking algorithm, the MPPT point is determined by finding the maximum power point on the PV plot. The inter-row shading creates a local maxima and a global maxima in the PV curve, and the model assumes that the inverter is operating at global maxima at all times. The amount of peak power generated by the defined row is labeled in the PV curve, and the MPPT operating point for the row is accordingly labeled in the IV curve, as shown in figure 7.5.

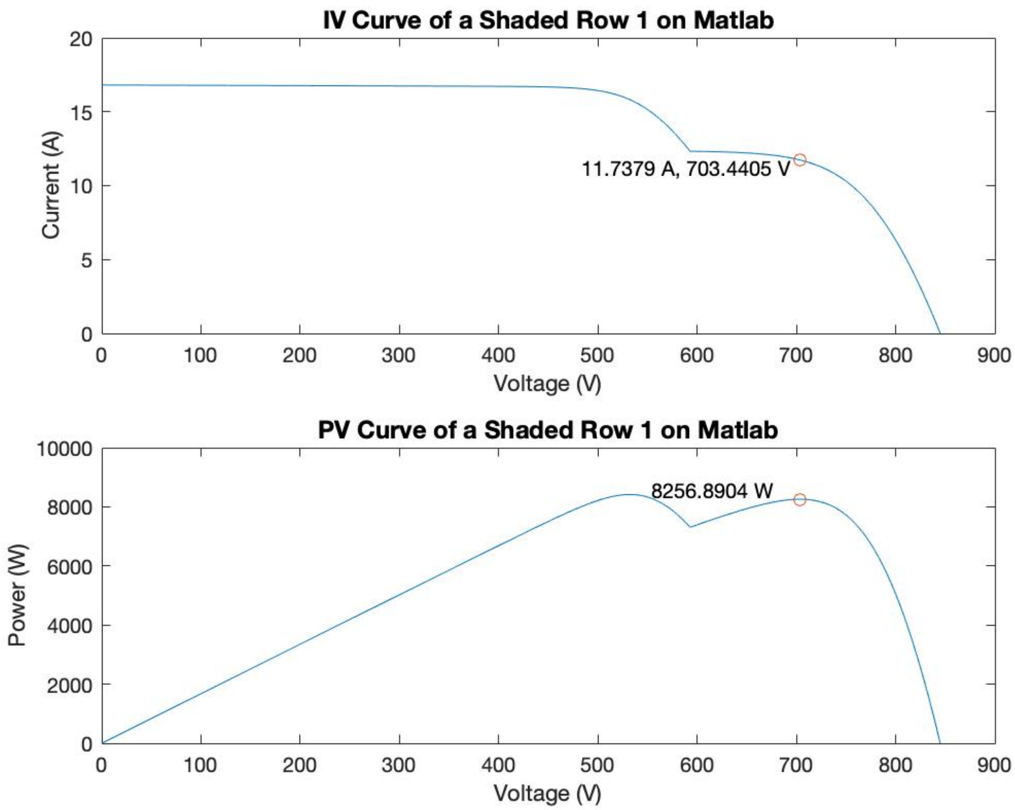


Figure 7.5: Example IV and PV Curve of a Row from the MATLAB model

Current, voltage, and power values calculated from MATLAB Model were compared against the actual data from the GPM portal. As mentioned in the previous chapter, both inverter 25 and inverter 55 had shade measurements taken at two separate times, and the model was validated using those data. The MATLAB model predicted the power output with a range of accuracy between 93.73 and 99.98 percent. Factors such as inaccurate estimations of irradiance and cell temperature, and assumptions mentioned in section 6.1 contribute to the error in the accuracy. Appendix G summarizes the percent error between calculated values and actual values of current, voltage, and power for both inverter 55 and inverter 25.

7.4 Case Study Results

The optimization method by decreasing pitch distance to retract the tracker angle earlier and reduce inter-row shading on the adjacent rows was validated using six different test cases. These cases have been studied between 2 pm and 7 pm to find the relationship between power output and the backtracking algorithm with different pitch distances as a parameter for the full-cut modules of inverter 55. Then the power outputs from the case study are compared with the actual power output from the Gold Tree Solar Farm with a pitch distance of 3.35 m with inter-row shading. The power outputs generated from the different pitch distances are assumed to be free of inter-row shading. The direct normal irradiance for each pitch distance is calculated using the same process as described in Section 6.4.

As shown in figure 7.6, the power output of an inverter with pitch distances of 2.25 meters, 2.5 meters, 2.75 meters, 3 meters, and 3.35 meters are compared against the actual power output of inverter 55 (Full-Cut Modules) with shading on April 23rd of 2020. All of the test cases were performed without taking consideration of inter-row shading because the PV_LIB Toolbox assumes that the PV modules are installed on a flat surface. With pitch distance of 2.25m and 2.5m, the modules backtracked too early, reducing the amount of power we can collect when the sun's elevation was still high.

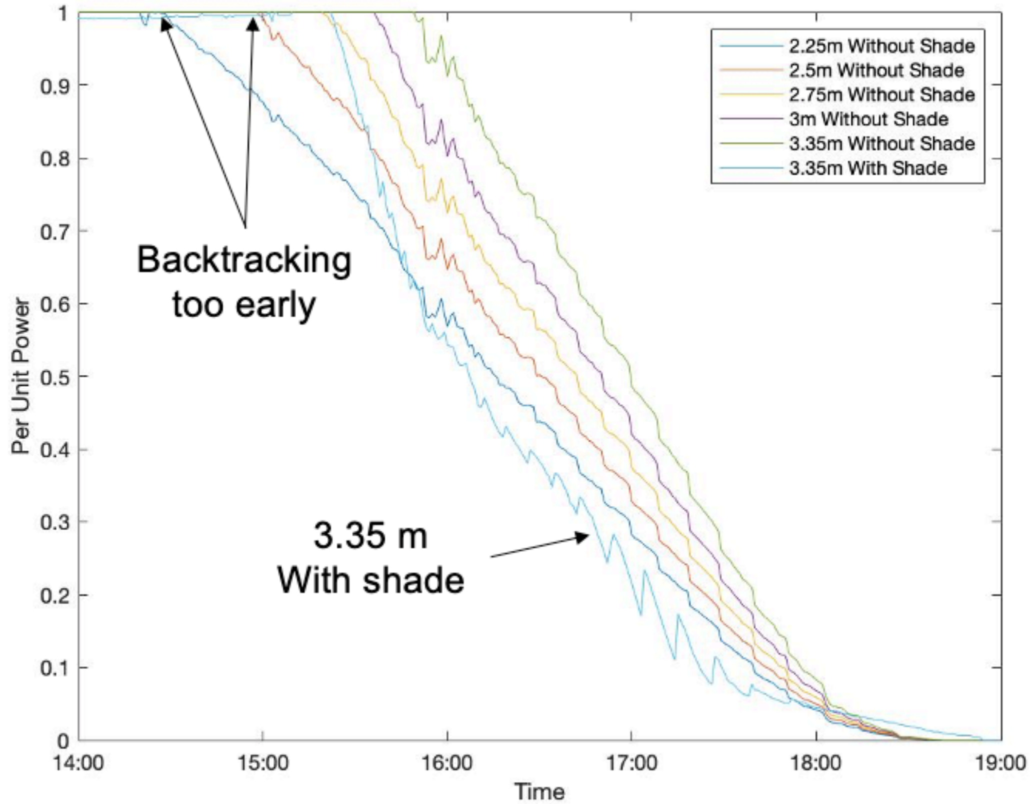


Figure 7.6: Per Unit Power of Inverter 55 (Full-Cut Cell) during sunset with different pitch distances for backtracking algorithm vs. actual output on 4/23/20

With these assumptions, the test cases showed -0.83 percent, 12.47 percent, 23.19 percent, 30.87 percent, and 39.27 percent difference in power production compared to the actual power output of the current PV system, respectively. The improvement in power production of pitch distances between 2.5 meters and 3 meters showed the validation of this optimization method. It is important to note that with inter-row shading, the power production of the inverter could potentially decrease depending on the size of the shade. But using shorter pitch distances in the backtracking algorithm will minimize the inter-row shading and provide improved power output overall.

Chapter 8

CONCLUSION AND FUTURE WORK

8.1 Conclusion

This thesis examined two PV system modeling software called PVsyst and PV_LIB Toolbox to identify the backtracking algorithm used in the Cal Poly Gold Tree Solar Farm and validated the approach of using optimum pitch distance to improve the system performance. This thesis also presented a PV system modeling tool for predicting system performance under inter-row shading in section 7.3. The performance modeling tool was validated against the real data collected from the Gold Tree Solar Farm with an accuracy between 93 to 99 percent. If cell temperature sensor and pyranometer was installed on the zones with inter-row shading, accuracy of the MATLAB PV performance model can be improved.

The case study of using different pitch distances as a parameter for the backtracking algorithm to retract the tracker angle earlier and reduce inter-row shading on the adjacent rows validated the proposed optimization method. This case study showed that the pitch distance of 2.25 m as a parameter is inadequate to use since the panels backtracked too early and reduced the power output. Also, pitch distances between 2.5 m and 3 m showed 12.47 to 30.86 percent improvements in the power generation in the afternoon, as explained in section 7.4. With the results from this thesis, the inter-row shading model can be incorporated to find the partial shading size related to the tracking angle and ground elevation to precisely model the performance of the PV system.

8.2 Next Step

The next step for this performance modeling tool is to improve the accuracy of the performance output generated. Furthermore, the accuracy of the model could greatly improve by acquiring an accurate direct irradiance and cell temperature readings by installing a pyranometer and a temperature sensor on zones that are heavily affected by inter-row shading. After some improvements in the accuracy of the model, it needs to be integrated with a 3D inter-row shading model of the PV modules. Due to the recent pandemic, acquiring the height of the tracker axis and the ground elevation from scanning the solar field with a lidar scanner has been delayed. After finding these exact positions of the tracker axis, the 3D shading model will need to be integrated with the system performance model to find the optimum pitch distance as a parameter for the backtracking algorithm. This pitch distance will be implemented into the actual backtracking algorithm in the Gold Tree Solar Field to reduce inter-row shading and improve the energy production.

REFERENCES

[1] A. Manton, “Solar Energy: A Renewable Resource with Global Importance”, *ESSAI*, 2015. [Online]. Available: https://dc.cod.edu/cgi/viewcontent.cgi?article=1541&context=essai. [Accessed: 30-Sep-2019].

[2] D. Chandler, “Explaining the Plummeting Cost of Solar Power”, *MIT News*, November 20, 2018. [Online]. Available: http://news.mit.edu/2018/explaining-dropping-solar-cost-1120 [Accessed: 24-Apr- 2020]

[3] R. Fu, D. Feldman, and R. Margolis, “U.S. Solar Photovoltaic System Cost Benchmark: Q1 2018,” *National Renewable Energy Laboratory,* October 2018.

[4] California Polytechnic State University San Luis Obispo, “Cal Poly Dedicates 4.5 Megawatt (AC) Solar Farm”, *Cal Poly News.* [Online]. Available: https://calpolynews.calpoly.edu/news_releases/2017/January/Solar. [Accessed 20-Jan-2020]

[5] REC Solar, “REC’s TwinPeak Technology: Raising the power of multicrystalline solar panels to levels only recently thought impossible!” [Online]. Available: https://www.recgroup.com/sites/default/files/documents/whitepaper_twinpeak_technology.pdf [Accessed 20-Jan- 2020]

[6] D. Albuquerque and M. Lima, “Backtracking Algorithm for Single-Axis Solar Trackers installed in a sloping field,” *Journal of Engineering Research and Application*, December 2015.

[7] J. Smalley, “Backtracking: Tracking Tool to Improve Project Performance,” *Solar Power World*, December 20, 2018. [Online]. Available: www.solarpowerworldonline.com/2015/07/how-does-solar-backtracking-make-projects-more-productive/. [Accessed 24-Jan- 2020]

[8] “What Is Maximum Power Point Tracking (MPPT),” *Northern Arizona Wind & Sun*. [Online]. Available: www.solar-electric.com/learning-center/mppt-solar-charge-controllers.html/. [Accessed 24-Jan- 2020]

[9] C. Deline, B. Sekulic, J. Stein, S. Barkaszi, J. Yang, and S. Kahn, “Evaluation of Maxim Module-Integrated Electronics at the DOE Regional Test Centers,” *National Renewable Energy Laboratory,* July 2014.

[10] D. Sharma, V. Verma, and A. Singh, “Review and Analysis of Solar Photovoltaic Softwares,” *International Journal of Current Engineering and Technology*, 2014. [Online]. Available: http://inpressco.com/wp-content/uploads/2014/04/ Paper48725 -731.pdf. [Accessed 1-Mar-2020]

[11] R. Perez, R. Seals, P. Ineichen, R. Stewart, D. Menicucci, "A new simplified version of the Perez diffuse irradiance model for tilted surfaces." *Solar Energy 39 (3)*, 1987, pp. 221–232

[12] J. Stein. "The Photovoltaic Performance Modeling Collaborative (PVPMC)", *2012 38th IEEE,* June 2012.

[13] K. Sauer, T. Roessler, C. W. Hansen, "Modeling the Irradiance and Temperature Dependence of Photovoltaic Modules in PVsyst", *IEEE Journal of Photovoltaics v5(1)*, January 2015.

[14] A. Mermoud, "PV modules modelling", *Presentation at the 2nd PV Performance Modeling Workshop*, May 2013.

[15] A. Mermoud, T. Lejeune, "Performance Assessment of a Simulation Model for PV modules of any available technology", *25th European Photovoltaic Solar Energy Conference*, September 2010.

[16] PV Performance Modeling Collaborative, "pvl_calcparams_PVsyst", *Sandia National Laboratories*. [Online]. Available: https://pvpmc.sandia.gov /PVLIB_Matlab_Help/ [Accessed 13-Feb- 2020]

[17] S.R. Wenham, M.A. Green, and M.E. Watt, "Applied Photovoltaics (Centre for Photovoltaic Devices and Systems, 1994)", (ISBN 0867589094)

[18] E. Lorenzo, L. Navarte, and J. Munoz, "Tracking and backtracking. Progress in Photovoltaics: Research and Applications", Vol. 19, 2012, pp.747-753.

[19] "System Advisor Model (SAM)," *National Renewable Energy Laboratory*. [Online]. Available: https://www.nrel.gov/about/ [Accessed 24-May-2020]

[20] "About NREL," *National Renewable Energy Laboratory*. [Online]. Available: https://sam.nrel.gov/about-sam.html [Accessed 24-May- 2020]

[21] "Solar Panel Shade: Does Shading Affect Solar?" *Wholesale Solar*, 3 Apr. 2020. [Online]. Available: www.wholesalesolar.com/blog/effect-shade-solar-panels. [Accessed 3-Mar- 2020]

[22] REC Solar "The REC TwinPeak Series: Innovative module design gives improved yield performance in shaded conditions" *REC Solar Twinpeak Shading Properties Whitepaper*, 2015. [Online]. Available: https://www.recgroup .com/sites/default/files/documents/whitepaper_twinpeak_shading_properti es_eng.pdf [Accessed 11-Apr-2020]

[23] D. Dolan, V. Prodanov, P. Salter, F. Cheein, J. Dolan, “Reducing Performance Loss Due to Backtracking Error Through Use of Half Cut Cell Modules”, *2019 9th International Conference on Power and Energy Systems (ICPES),* Perth, Australia, 2019. pp. 1-4, doi: 10.1109/ICPES47639.2019.9105393.

[24] C. Deline, “Characterizing Shading Losses on Partially Shaded PV System”, *National Renewable Energy Laboratory,* September 23, 2010.

[25] E. Skoplaki, A.G. Boudouvis, J.A. Palyvos, “A simple correlation for the operating temperature of photovoltaic modules of arbitrary mounting,” *Solar Engineering Unit, School of Chemical Engineering, NTUA, Athens GR-15780, Greece,* 2008.

[26] Amerisolar, “What is normal operating cell temperature?”, *Amerisolar News,* June 15, 2018.

[27] S. Wilcox and W. Marion, “Users Manual for TMY3 Data Sets”, *National Renewable Energy Laboratory*, May 2008. [Online]. Available: https://www.nrel.gov/docs/fy08osti/43156.pdf [Accessed 18-Mar-2020]

APPENDICIES

Appendix A: Cal Poly Solar Field Layout

Table A.1: Zone and Corresponding Tracker Name

Zone Number	Tracker Name	Number of Rows
1	X1	21
2	X2	19
3	X3	20
4	X4	13
5	Y1	18
6	Y2	18
7	Y3	19
8	Y4	20
9	Z1	26
10	Z2	14
11	Z3	21
12	Z4	15

Appendix B: System Component Data Sheet

YASKAWA

PVI 50TL & PVI 60TL

3-Ph Transformerless Commercial String Inverters

Features

- Integrated arc fault protection
- UL 1741SA listed
- 3 MPPTs with 5 inputs each
- Integrated DC and AC disconnects
- AC terminals compatible with copper and aluminum conductors
- Modbus communications
- Internal data logger
- 0 - 90° installation orientation
- Remote firmware upgrades
- Remote diagnostics
- Compatible with certain MLPE for module-level rapid shutdown*

Options

- Shade cover
- DC fuse bypass
- Web-based monitoring

Yaskawa Solectria Solar's PVI 50TL and PVI 60TL are grid-tied, transformerless three-phase inverters designed for ground mount, rooftop and carport arrays and can be installed from 0 - 90 degrees. The PVI 50/60TL inverters are the most reliable, efficient and cost effective in their class. They come standard with AC and DC disconnects, three MPPTs, a 15-position string combiner, remote diagnostics, remote firmware upgrades and various protection features. Options include shade cover, DC combiner fuse bypass, and web-based monitoring.

SOLECTRIA SOLAR

PVI 50TL & PVI 60TL

Specifications

	PVI 50TL	PVI 60TL
DC Input		
Absolute Maximum Input Voltage	1000 VDC	1000 VDC
Maximum Power Input Voltage Range (MPPT)	480-850 VDC	540-850 VDC
Operating Voltage Range (MPPT)	200-950 VDC	200-950 VDC
Maximum Operating Input Current	108 A (36 A per MPPT)	114 A (38 A per MPPT)
Number of MPP Trackers	3	3
Maximum Available PV Current (Isc x 1.25)	204 A (68 A per MPPT)	204 A (68 A per MPPT)
Maximum PV Power	75 kW (30 kW per MPPT)	90 kW (33 kW per MPPT)
Start Voltage	330 V	330 V
AC Output		
Nominal Output Voltage	480 VAC, 3-Ph/PE/N	480 VAC, 3-Ph/PE/N
AC Voltage Range (Standard)	-12/+10%	-12/+10%
PF=1.00 - Real/Apparent Power/Output Current	50 kW / 50 kVA / 60.2 A	60 kW / 60 kVA / 72.3 A
PF=+/-0.91 - Real/Apparent Power/Output Current	50 kW / 55 kVA / 66.2 A	60 kW / 66 kVA / 79.4 A
Nominal Output Frequency	60 Hz	60 Hz
Output Frequency Range	57-63 Hz	57-63 Hz
Power Factor	Unity, >0.99 (Adjustable 0.8 leading to 0.8 lagging)	Unity, >0.99 (Adjustable 0.8 leading to 0.8 lagging)
Fault Current Contribution (1 Cycle RMS)	55 A	55 A
Total Harmonic Distortion (THD) @ Rated Load	<3%	<3%
Recommended OCPD Device	90 A	100 A
AC Surge Protection	Type II MOV, 1240Vc, 15kA Itm (8/20μ)	
Efficiency		
Peak Efficiency	99.0%	99.0%
CEC Efficiency	98.5%	98.5%
Tare Loss	< 2 W	< 2 W
Integrated String Combiner		
Fused Inputs	15 Fused Positions (5 Positions per MPPT) 15 A Standard (20, 25, 30 A accepted)**	
Temperature		
Ambient Temperature Range	-22°F to +140°F (-30°C to +60°C); Derating occurs over +122°F (+50°C)	
Storage Temperature Range	No low temp minimum to +158°F (+70°C)	
Relative Humidity (non-condensing)	0-95%	
Operating Altitude	13,123 ft (4,000 m) Derating occurs from 9,842.5 ft (3,000 m)	
Communications		
Modbus Protocol	SunSpec (FW 11.0 +) / Proprietary	
SolrenView Web-Based Monitoring Service	Optional	
Revenue Grade Metering	Optional, External	
Communication Interface	RS-485 Modbus RTU and Ethernet TCP/IP	
Remote Firmware Upgrades	Standard	
Remote Diagnostics	Standard	
Features & Protections		
Arc-Fault	Standard	
Smart Grid Features	L/HVRT, L/HFRT, Volt-Var, Frequency-Watt and Volt-Watt, Soft-Start, Soft-Step	
Testing & Certifications		
Safety Listings & Certifications	UL 1741SA-2016, UL1699B, CSA-C22.2 #107.1, IEEE1547a-2014	
Advanced Grid Support Functionality	Rule 21, UL 1741SA	
Testing Agency	CSA	
FCC Compliance	FCC Part 15	
Warranty		
Standard Limited Warranty	10 Years	
Enclosure		
Acoustic Noise Rating	< 60 dBA @ 1 m at room temperature	
AC/DC Disconnect	Standard, fully-integrated	
Mounting Angle***	0-90° from horizontal (vertical, angled, flat)	
Dimensions (H x W x D)	39.4 in. x 23.6 in. x 10.2 in (1,000 mm x 600 mm x 260 mm)	
Weight	Inverter: 123.5 lbs (56 kg); Wiring Box: 33 lbs (15 kg)	
Enclosure Rating and Finish	Type 4X; Polyester Powder Coated Aluminum	

***Please inquire about compatible Module-Level Power Electronics (MLPE)**
****Yaskawa Solectria Solar does not supply optional fuses sizes**
*****Shade cover accessory required for installation of 75° or less**

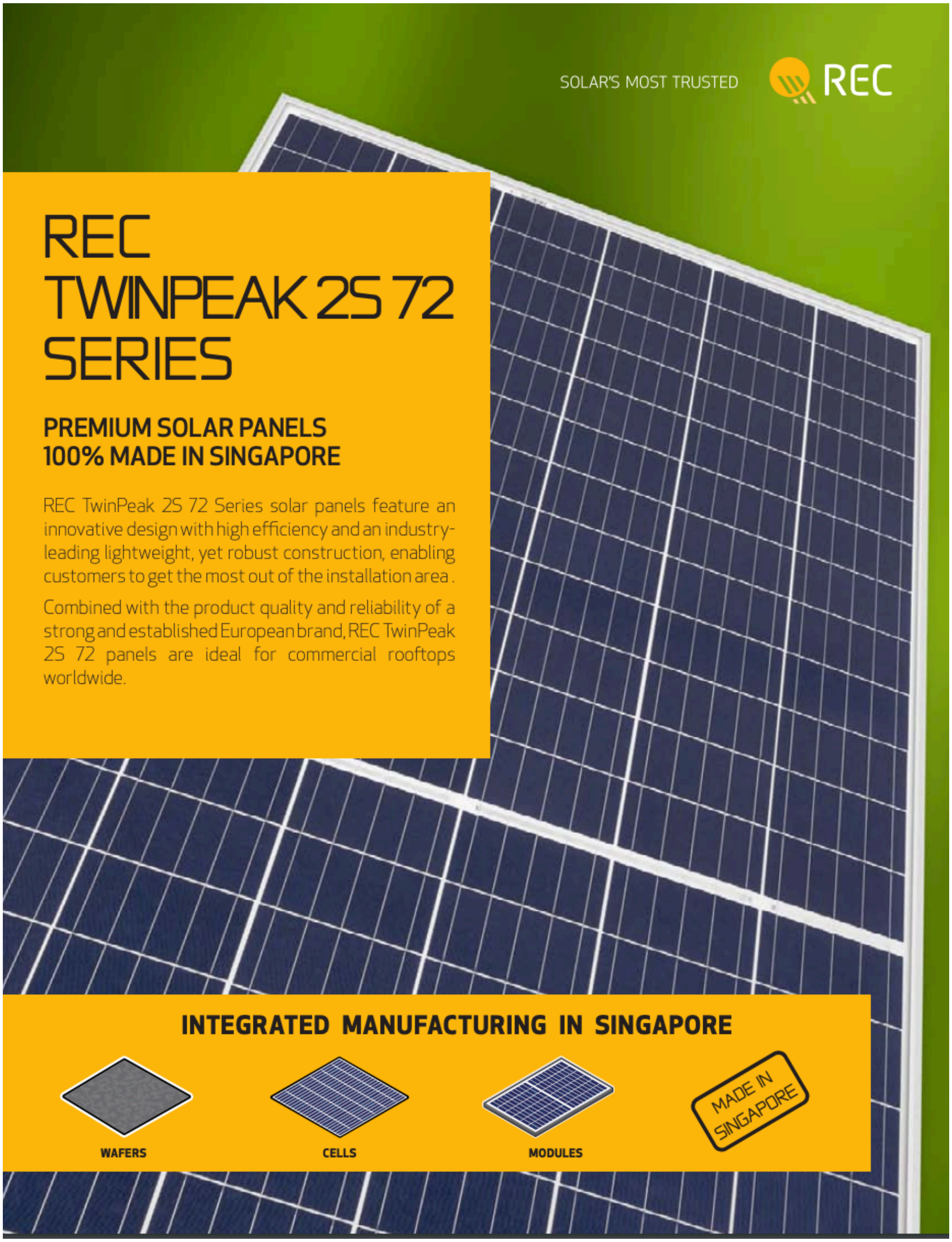
SOLECTRIA SOLAR

Yaskawa Solectria Solar
360 Merrimack Street
Lawrence, MA 01843
solectria.com

1-978-683-9700
inverters@solectria.com

DOCR-070642-S | November 2019

YASKAWA



SOLAR'S MOST TRUSTED REC

REC TWINPEAK 2S 72 SERIES

PREMIUM SOLAR PANELS 100% MADE IN SINGAPORE

REC TwinPeak 2S 72 Series solar panels feature an innovative design with high efficiency and an industry-leading lightweight, yet robust construction, enabling customers to get the most out of the installation area.

Combined with the product quality and reliability of a strong and established European brand, REC TwinPeak 2S 72 panels are ideal for commercial rooftops worldwide.

INTEGRATED MANUFACTURING IN SINGAPORE

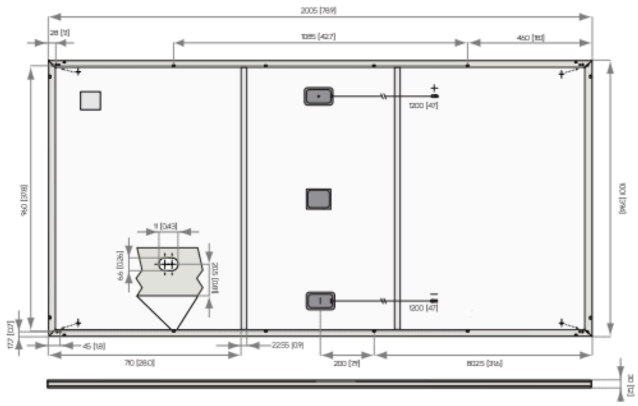
WAFERS

CELLS

MODULES

MADE IN SINGAPORE

REC TWINPEAK 2S 72 SERIES



All measurements in mm [in]

ELECTRICAL DATA @ STC	Product Code*: RECxxxTP2S 72					
Nominal Power - P_{MPP} (Wp)	330	335	340	345	350	355
Watt Class Sorting - (W)	-0/+5	-0/+5	-0/+5	-0/+5	-0/+5	-0/+5
Nominal Power Voltage - V_{MPP} (V)	38.1	38.3	38.5	38.7	38.9	39.1
Nominal Power Current - I_{MPP} (A)	8.67	8.75	8.84	8.92	9.00	9.09
Open Circuit Voltage - V_{OC} (V)	46.0	46.2	46.3	46.5	46.7	46.8
Short Circuit Current - I_{SC} (A)	9.44	9.52	9.58	9.64	9.72	9.78
Panel Efficiency (%)	16.5	16.7	16.9	17.2	17.4	17.7

Values at standard test conditions STC (airmass AM 1.5, irradiance 1000 W/m², cell temperature 77°F (25°C).
At low irradiance of 200 W/m² (AM 1.5 and cell temperature 77°F (25°C)) at least 95% of the STC module efficiency will be achieved.
*xxx indicates the nominal power class (P_{MPP}) at STC, and can be followed by the suffix XV for modules with a 1500 V maximum system rating.

ELECTRICAL DATA @ NOCT	Product Code*: RECxxxTP2S 72					
Nominal Power - P_{MPP} (Wp)	244	252	257	260	264	268
Nominal Power Voltage - V_{MPP} (V)	34.9	35.5	35.7	35.8	36.0	36.2
Nominal Power Current - I_{MPP} (A)	6.99	7.10	7.19	7.25	7.32	7.39
Open Circuit Voltage - V_{OC} (V)	42.3	42.8	42.9	43.1	43.2	43.3
Short Circuit Current - I_{SC} (A)	7.44	7.74	7.79	7.84	7.90	7.95

Nominal cell operating temperature NOCT (800 W/m², AM 1.5, windspeed 1 m/s, ambient temperature 68°F (20°C).
*xxx indicates the nominal power class (P_{MPP}) at STC, and can be followed by the suffix XV for modules with a 1500 V maximum system rating.

CERTIFICATION

UL 1703, Fire classification: Type 1 (1500 V XV); Type 2 (1000 V);
IEC 61215, IEC 61730, IEC 62804 (PID), IEC 62716 (Ammonia),
IEC 61701 (Salt Mist level 6), ISO 11925-2 (Class E)
ISO 9001: 2015, ISO 14001: 2004, OHSAS 18001: 2007

WARRANTY

10 year product warranty.
25 year linear power output warranty
(max. degression in performance of 0.7% p.a.).

17.7% EFFICIENCY

10 YEAR PRODUCT WARRANTY

25 YEAR LINEAR POWER OUTPUT WARRANTY

TEMPERATURE RATINGS	
Nominal Operating Cell Temperature (NOCT)	44.6°C (±2°C)
Temperature Coefficient of P_{MPP}	-0.36 %/°C
Temperature Coefficient of V_{OC}	-0.30 %/°C
Temperature Coefficient of I_{SC}	0.066 %/°C

GENERAL DATA	
Cell type:	6 strings of 24 REC HC multicrystalline PERC
Glass:	0.13" (3.2 mm) solar glass with anti-reflection surface treatment
Back Sheet:	Highly resistant polyester
Frame:	Anodized aluminum (silver)
Support bars:	Anodized aluminum (bonded to backsheet)
Junction Box:	IP67 rated with 3 bypass diodes 12 AWG (4 mm²) PV wire, 47" + 47" (1.2 m + 1.2 m)
Connectors:	Tonglin TL-Cable01S-F (4 mm²) (1500V) Tonglin TL-Cable01S-FR (4 mm²) (1000V)
Origins:	Silicon: Made in USA & Norway Wafer/Cell/Module: Made in Singapore

MAXIMUM RATINGS	
Operational Temperature:	-40 ... +185°F (-40 ... +85°C)
Maximum System Voltage:	1000 V / 1500 V* *Dependent on product type
Design Load:	(+) 75.2 lbs/ft² (3600 Pa)
Design Load:	(-) 33.4 lbs/ft² (1600 Pa) Refer to installation instructions
Max Series Fuse Rating:	20 A
Max Reverse Current:	20 A

MECHANICAL DATA	
Dimensions:	78.9" x 39.4" x 1.2" (2005 x 1001 x 30 mm)
Area:	21.6 ft² (2.01 m²)
Weight:	48.5 lbs (22 kg)

Note! Specifications subject to change without notice.

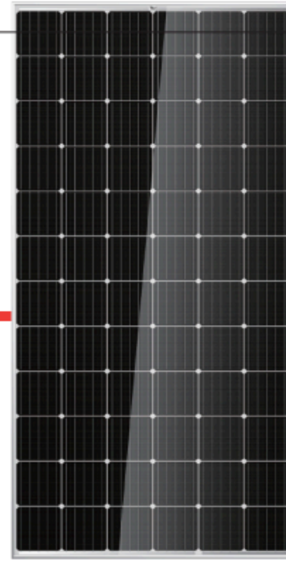
Ref: NE-05-07-13 Rev- C 2 06.18

REC

www.recgroup.com

THE TALLMAX M PLUS+

FRAMED 72-CELL MODULE (1500V)



72 CELL
MONOCRYSTALLINE MODULE

335-365W
POWER OUTPUT RANGE

18.8%
MAXIMUM EFFICIENCY

0~+5W
POSITIVE POWER TOLERANCE

Founded in 1997, Trina Solar is the world's leading comprehensive solutions provider for solar energy. we believe close cooperation with our partners is critical to success. Trina Solar now distributes its PV products to over 60 countries all over the world. Trina is able to provide exceptional service to each customer in each market and supplement our innovative, reliable products with the backing of Trina as a strong, bankable partner. We are committed to building strategic, mutually beneficial collaboration with installers, developers, distributors and other partners.

Comprehensive Products And System Certificates

IEC61215/IEC61730/UL1703/IEC61701/IEC62716
ISO 9001: Quality Management System
ISO 14001: Environmental Management System
ISO14064: Greenhouse gases Emissions Verification
OHSAS 18001: Occupation Health and Safety Management System

Trinasolar

Ideal for large scale installations

- Reduce BOS cost by connecting more modules in a string
- 1500V UL/1500V IEC certified

Maximize limited space with top-end efficiency

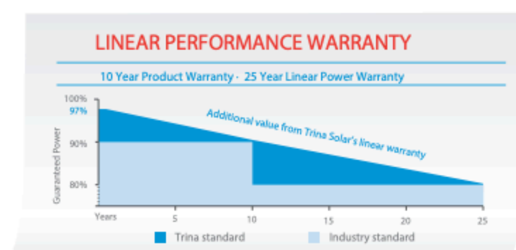
- Up to 188 W/m² power density
- Low thermal coefficients for greater energy production at high operating temperatures

Highly reliable due to stringent quality control

- Over 30 in-house tests (UV, TC, HF, and many more)
- In-house testing goes well beyond certification requirements
- 100% EL double inspection

Certified to withstand the most challenging environmental conditions

- 2400 Pa wind load
- 5400 Pa snow load

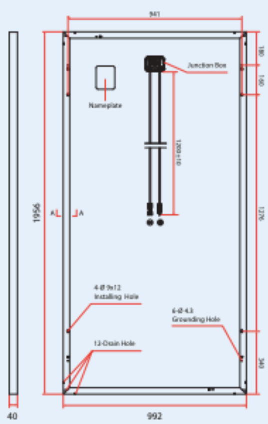


TALLMAX M PLUS

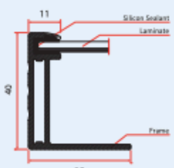
FRAMED 72-CELL MODULE (1500V)

PRODUCTS		POWER RANGE
TSM-DE14A(II)	|	335-365W

DIMENSIONS OF PV MODULE(mm)

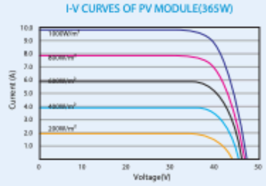


Back View

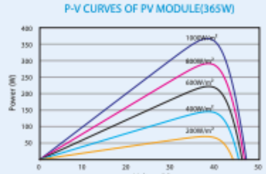


(A-A)

I-V CURVES OF PV MODULE(365W)



P-V CURVES OF PV MODULE(365W)



ELECTRICAL DATA (STC)

Peak Power Watts-P_{MAX} (Wp)*	335	340	345	350	355	360	365
Power Output Tolerance-P_{MAX} (W)				0 ~ +5			
Maximum Power Voltage-V_{MPP} (V)	37.9	38.2	38.4	38.5	38.7	38.9	39.1
Maximum Power Current-I_{MPP} (A)	8.84	8.90	9.00	9.09	9.17	9.26	9.35
Open Circuit Voltage-V_{OC} (V)	46.3	46.5	46.7	46.9	47.0	47.2	47.3
Short Circuit Current-I_{SC} (A)	9.36	9.45	9.50	9.60	9.69	9.79	9.88
Module Efficiency η_m (%)	17.3	17.5	17.8	18.0	18.3	18.5	18.8

STC: Irradiance 1000W/m², Cell Temperature 25°C, Air Mass AM1.5.
*Measuring tolerance: ±3%.

ELECTRICAL DATA (NOCT)

Maximum Power-P_{MAX} (Wp)	250	253	257	261	264	268	272
Maximum Power Voltage-V_{MPP} (V)	35.1	35.2	35.5	35.6	35.8	35.9	36.1
Maximum Power Current-I_{MPP} (A)	7.12	7.19	7.25	7.33	7.40	7.47	7.54
Open Circuit Voltage-V_{OC} (V)	43.1	43.2	43.4	43.5	43.7	43.8	43.9
Short Circuit Current-I_{SC} (A)	7.56	7.63	7.67	7.75	7.82	7.88	7.95

NOCT: Irradiance at 800W/m², Ambient Temperature 20°C, Wind Speed 1m/s.

MECHANICAL DATA

Solar Cells	Monocrystalline 156.75 × 156.75 mm (6 inches)
Cell Orientation	72 cells (6 × 12)
Module Dimensions	1956 × 992 × 40 mm (77.0 × 39.1 × 1.57 inches)
Weight	26.0 kg (57.3 lb) with 4.0 mm glass; 22.5 kg (49.6 lb) with 3.2 mm glass
Glass	4.0 mm (0.16 inches) for PERC Mono; 3.2 mm (0.13 inches) for Std Mono, High Transmission, AR Coated Tempered Glass
Backsheet	White
Frame	Silver Anodized Aluminium Alloy
J-Box	IP 67 or IP 68 rated
Cables	Photovoltaic Technology Cable 4.0mm² (0.006 inches²), 1200 mm (47.2 inches)
Connector	QC4 / TS4 (1500V)

TEMPERATURE RATINGS

NOCT (Nominal Operating Cell Temperature)	44°C (±2°C)
Temperature Coefficient of P_{MAX}	- 0.39%/°C
Temperature Coefficient of V_{OC}	- 0.29%/°C
Temperature Coefficient of I_{SC}	0.05%/°C

MAXIMUM RATINGS

Operational Temperature	-40~+85°C
Maximum System Voltage	1500V DC (IEC)
	1500V DC (UL)
Max Series Fuse Rating	15A

(DO NOT connect Fuse in Combiner Box with two or more strings in parallel connection)

WARRANTY

10 year Product Workmanship Warranty

25 year Linear Power Warranty

(Please refer to product warranty for details)

PACKAGING CONFIGURATION

Modules per box: 27 pieces

Modules per 40' container: 648 pieces

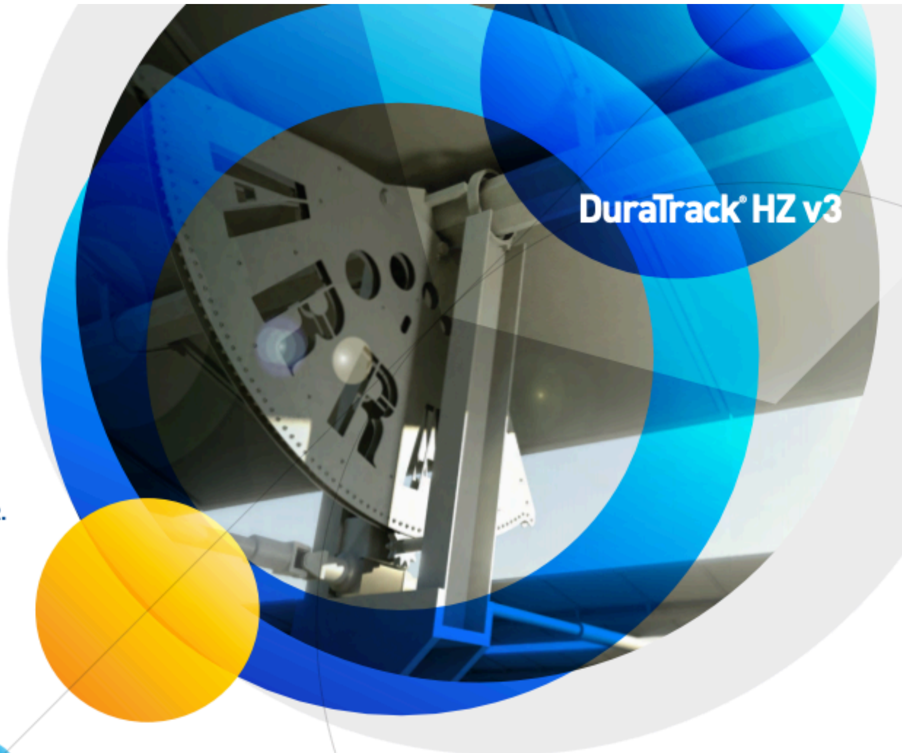
Trinasolar

CAUTION: READ SAFETY AND INSTALLATION INSTRUCTIONS BEFORE USING THE PRODUCT.

Version number: TSM_EN_2017_APAC_A

www.trinasolar.com

ARRAY TECHNOLOGIES



RELIABILITY IS POWER.

167 X

fewer components than competitive trackers

99.996%

Uptime Reliability

ARRAY TECHNOLOGIES, INC.

3901 Midway Place NE
Albuquerque, NM 87109 USA

+1 505.881.7567
+1 855.TRACKPV (872.2578)
+1 505.881.7572

sales@arraytechinc.com

arraytechinc.com

THE MOST RELIABLE TRACKER UNDER THE SUN

HIGHEST POWER DENSITY.

Higher density means more power and more profit. DuraTrack HZ v3 offers the unique ability to maximize the power density of each site, boasting 6% more density than our closest competitor.

LEADING TERRAIN ADAPTABILITY.

Uneven terrain? Hill yes! Our flexibly linked architecture, with articulating driveline joints and forgiving tolerances, create the most adaptable system in market for following natural land contours and creates the greatest power generation potential from every site.

FEWER COMPONENTS. GREATER RELIABILITY.

Less is more. Array was founded on a philosophy of engineered simplicity. Minimizing potential failure points (167 times fewer components than competitors), DuraTrack HZ v3 consistently delivers higher reliability and an unmatched uptime of 99.99%.

FAILURE-FREE WIND DESIGN.

DuraTrack HZ v3 was designed and field tested to withstand some of the harshest conditions on the planet. It is the only tracker on the market that reliably handles wind events with a fully integrated, fully automatic wind-load mitigation system.

ZERO SCHEDULED MAINTENANCE.

Three decades of solar tracker system design, engineering and testing has resulted in uncompromising reliability. Maintenance-free motors and gears, fewer moving parts, and industrial-grade components means maintenance-free energy generation.

DuraTrack® HZ v3

COST VERSUS VALUE

We believe value is more than the cost of a tracking system. It's about building with forgiving tolerances and fewer parts so construction crews can work efficiently. It means protecting your investment with a failure-free wind management system. It also includes increasing power density. But most of all, value is measured in operational uptime, or reliability. Ours is 99.996%... and we're still improving on it.

THE GLOBAL LEADER IN RELIABILITY

Array has spent decades designing and perfecting the most reliable tracker on the planet. Fewer moving parts, stronger components and intelligent design that protects your investment in the harshest weather are but a few of the innovative differences that keep your system running flawlessly all day and you resting easy at night.

STRUCTURAL & MECHANICAL FEATURES/SPECIFICATIONS	
Tracking Type	Horizontal single axis
kW per Drive Motor	Up to 907 kW DC using 360W crystalline
String Voltage	Up to 1,500V DC
Maximum Linked Rows	28
Maximum Row Size	80 modules (crystalline, 1,000V DC) & 90 modules (crystalline, 1,500V DC)
Drive Type	Rotating gear drive
Motor Type	2 HP, 3 PH, 480V AC
Motors per 1 MW AC	Less than 2
East-West / North-South Dimensions	Site / module specific
Array Height	54" standard, adjustable (46" min height above grade)
Ground Coverage Ratio (GCR)	Flexible, 28–45% typical, others supported on request
Terrain Flexibility	N-S tolerance: 0°–8.5° Standard, option to increase Driveline: 40° in all directions
Modules Supported	Most commercially available, including frameless crystalline and thin film
Tracking Range of Motion	± 52°
Operating Temperature Range	-30°F to 130°F (-34°C to 55°C)
Module Configuration	Single-in-portrait standard. Two-or-three in landscape (framed or frameless), four-in-landscape (thin film) also available.
Module Attachment	Single fastener, high-speed mounting clamps with integrated grounding. Traditional rails for crystalline in landscape, custom racking for thin film and frameless crystalline per manufacturer specs.
Materials	HDG steel and aluminum structural members
Allowable Wind Load (IBC 2012)	135 mph, 3-second gust exposure C
Wind Protection	Passive mechanical system relieves wind and obstruction damage — no power required
ELECTRONIC CONTROLLER FEATURES/SPECIFICATIONS	
Solar Tracking Method	Algorithm with GPS input
Control Electronics	MCU plus Central Controller
Data Feed	MODBUS over Ethernet to SCADA system
Night-time Stow	Yes
Tracking Accuracy	± 2° standard, field adjustable
Backtracking	Yes
INSTALLATION, OPERATION & MAINTENANCE	
PE Stamped Structural Calculations & Drawings	Yes
On-site Training & System Commissioning	Yes
Connection Type	Fully bolted connections, no welding
In-field Fabrication Required	No
Dry Slide Bearings & Articulating Driveline Connections	No lubrication required
Scheduled Maintenance	None required
GENERAL	
Annual Power Consumption (kWh per 1 MW)	400 kWh per MW per year, estimated
Land Area Required per 1 MW	Approx. 5 to 5.75 acres per MW @ 33% GCR (site and design specific)
Energy Gain vs. Fixed-Tilt	Up to 25%, site specific
Warranty	10 year structural, 5 year drive & control components
Patent Numbers	US patent 8,459,249. US patent 9,281,778. US patent 9,581,678 B2 and patents pending
Codes and Standards	UL Certified (3703 & 2703)

Appendix C: GPM Portal Instructions

Following figures explain how to obtain Cal Poly Solar Farm data from GPM Portal.

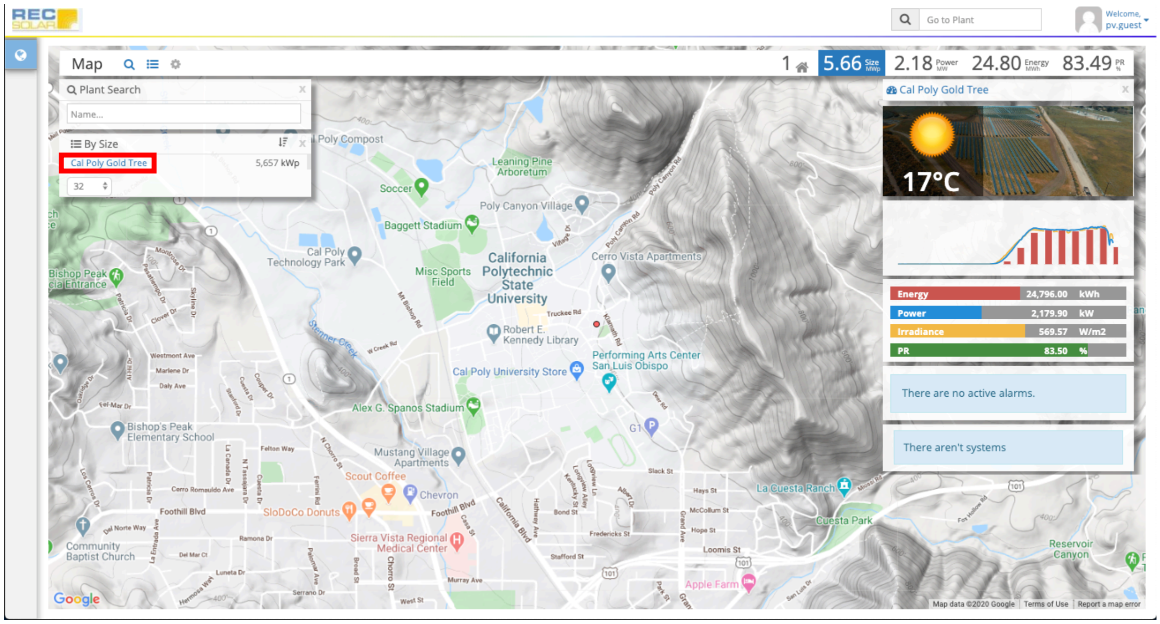


Figure C.1: GPM Portal Main Page

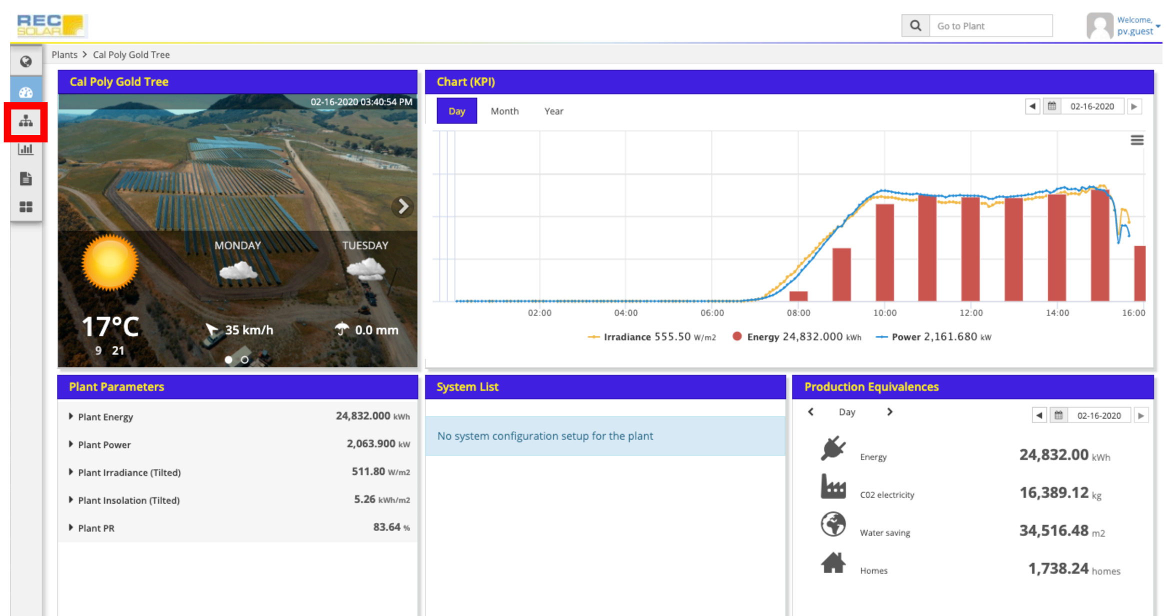


Figure C.2: Navigating to Quick Query

In the Quick Query page, user can choose which information to view in the element type dropdown menu box. Some of the available information include; inverter performance, plant parameter, tracking, and weather data. Within each element type, there is a selection of individual element and parameters to view. For example, to find POA and GHI of the plant, user must select Plant Parameters in the element dropdown menu and choose Plant Irradiance (Tilted) and Plant Irradiance (Horizontal) in parameter menu. Figure 2.7 displays an example plot of GHI and POA, but user can change element types by using the dropdown menu located in the orange box shown in the same figure. A table or a chart will be displayed according to the designated time chosen by the user. GPM Portal displays the real-time values if desired time is not defined. Previous data from different dates can be accessed by choosing desired date and time in the box labeled in red in figure 2.7.

Figure C.3: GHI and POA data from GPM Portal

List of data used in this thesis and chosen element type, element and plant parameters to find the data is given in the table below. For this study, inverter 50 was chosen as an ideal inverter, because strings of PV modules connected to inverter 50 are not shaded throughout the day, providing optimum power and energy generation on a clear day. Tracker Z1was chosen because angle of this tracker represents the actual position of PV modules in inverter 50. These data were gathered and compared with results from different simulation tools to validate the programs used in this thesis.

Table C.1: Quick Query Guidance to Useful Data

	POA / GHI	Power / Energy	Backtracking Angle
Element Type	Plant Parameters	Inverter	Tracker
Element	N/A	Inverter 50	Tracker Z1
Plant Parameters	Plant Irradiance (Titled)/ Plant Irradiance (Horizontal)	Active power/ Daily Active Energy	Tracker – Actual Position

To download data from GPM Portal, user must select an option to view the data in a tabulated form and click on the download button to begin the download. It is important to note that data are recorded per minute, so user must expect some lag while loading the data. Both options to view the data in tabulated form and the download button is highlighted with a red box shown in a figure below. GHI data was downloaded from GPM Portal to be used in the simulation tool to mimic similar irradiance condition.

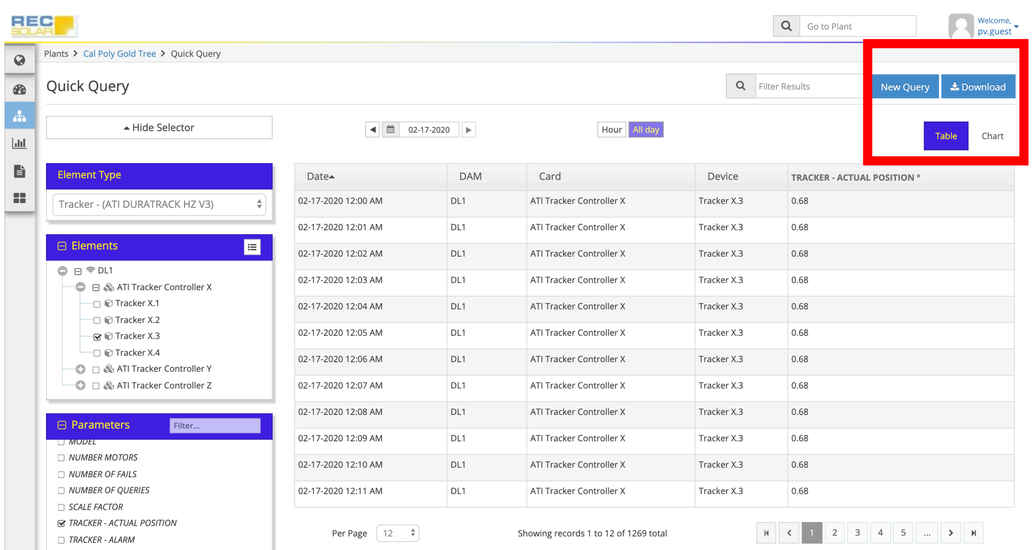


Figure C.4: Downloading Data from GPM Portal

Appendix D: PVsyst Instructions

Following figures and explanation demonstrate how to run simulations and view data in PVsyst.

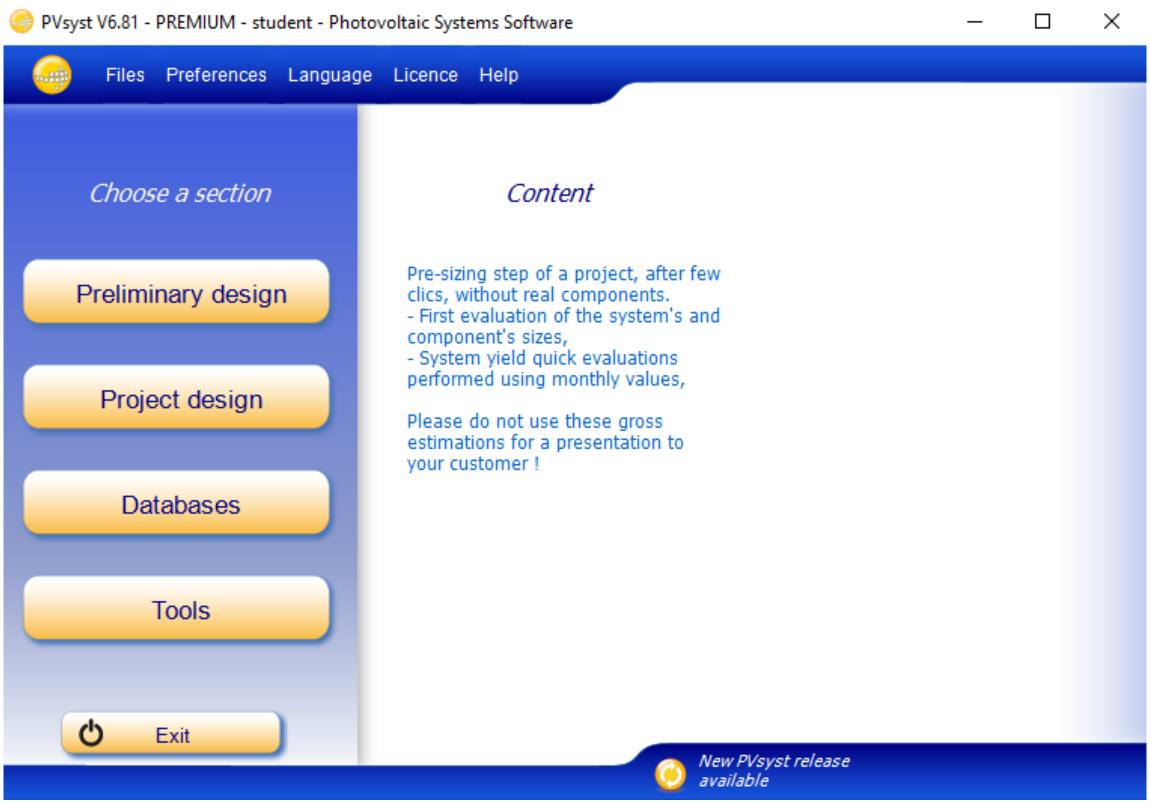


Figure D.1: PVsyst Start Menu

After selecting project design from the start menu, user can choose from four different systems as seen in figure D.1. All of the analysis and study is done on grid-connected option, because Cal Poly Solar Farm is a grid-connected system.

Figure D.2: Project Design Menu

In order to obtain accurate result from simulation, user needs to define both type and amount of PV module, Inverter, and tracking system, and select meteorological data of the system location in Typical Meteorological Year 3 format. TMY 3 are data sets containing solar radiation and meteorological information for a 1-year period given in hourly values. Main purpose of TMY 3 is to be used as a solar radiation resource to perform computer simulations of PV systems, and compare the simulated energy generation to the actual system energy generation [28].

PVsyst database includes a wide compilation of photovoltaic system components including modules, inverters and batteries, but users can also import or create their own technical components to be used in the simulation. Built-in meteorological data, such as global irradiation, diffuse irradiation, and average external temperature is also provided by Meteonorm in PVsyst. User can also look up details of existing components or import meteorological data file when selecting Databases option in figure D.2. Figure below represents the Databases menu with meteorological data on the left side and system components on the right side.

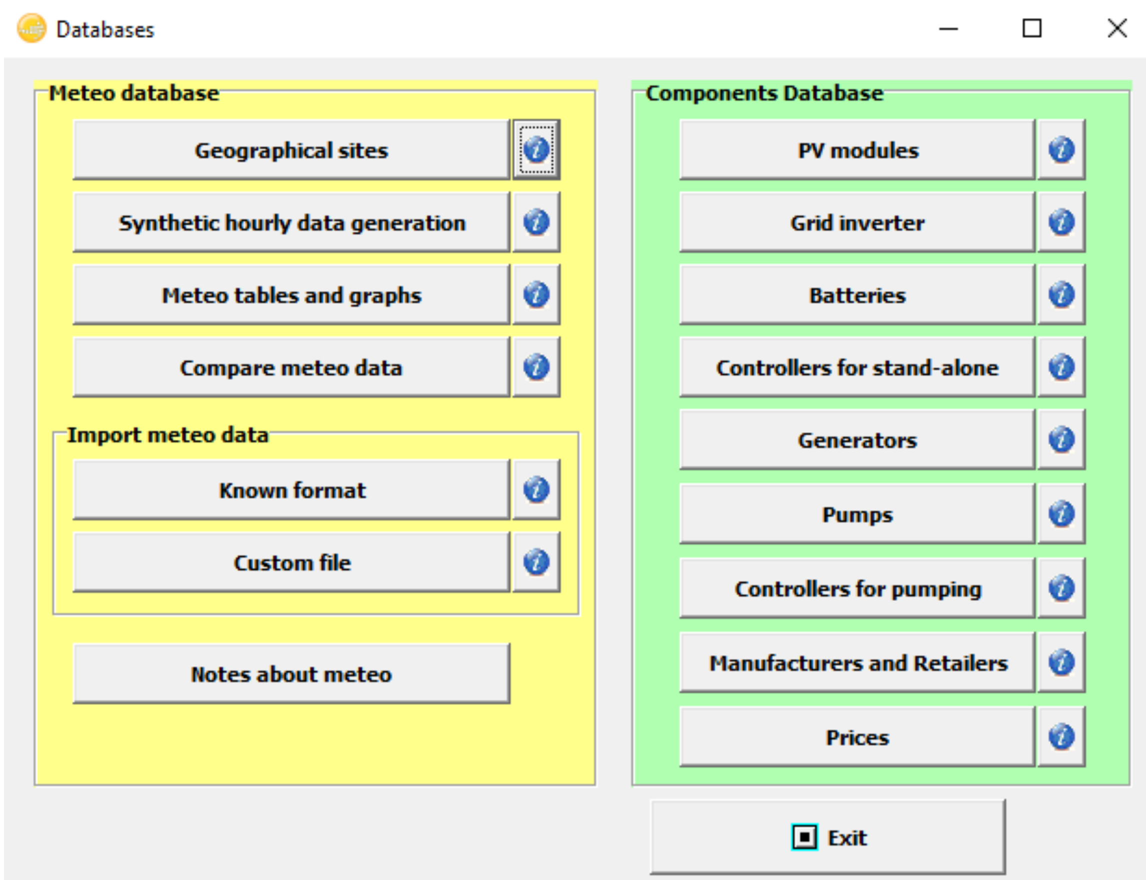


Figure D.3: System Database

Although satellite data from the NASA-SSE project and data from NSRDB PSM are available in the PV Syst database, those two options doesn't provide a meteorological data for a specific location. Also, data provided by PV Syst range from 1998 to 2014, so it will be inaccurate to compare the current power and energy generation with a computer simulation using data from 2014. So, choosing to import current meteorological data corresponding to the location of solar farm helps with accuracy of power and energy generation simulation. Figure below shows the importing weather data window from PV Syst, and it is important to note that data from Solar Anywhere is used for this thesis as shown in a dropdown menu labeled with a red box.

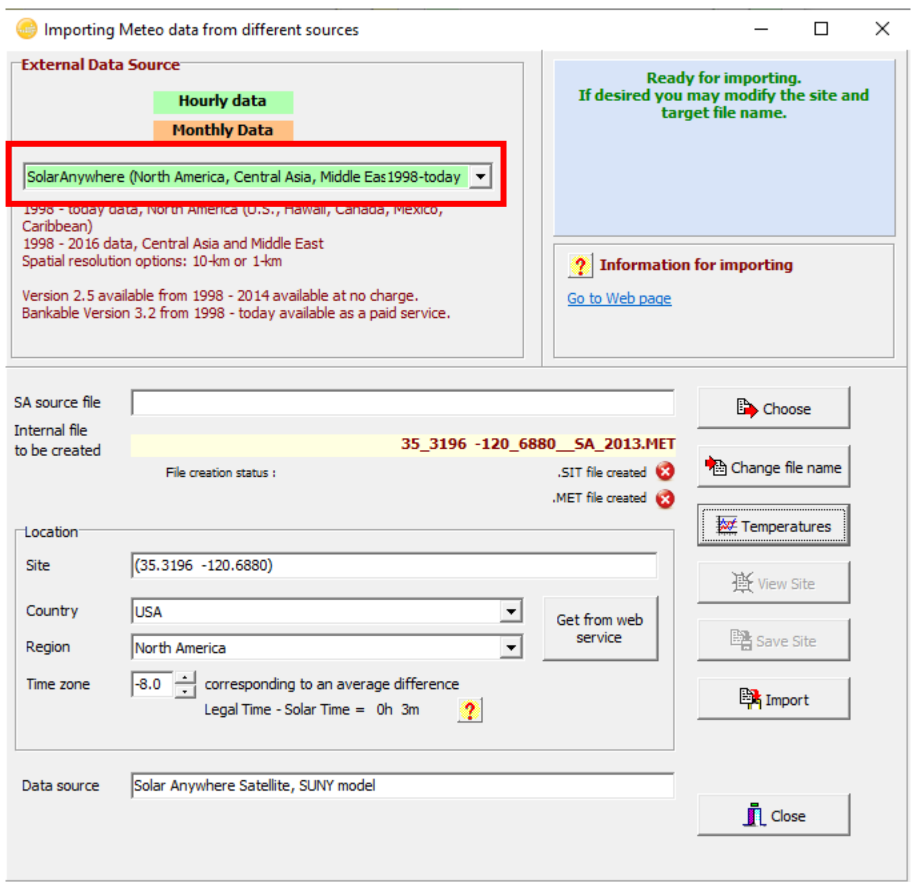


Figure D.4: Importing Meteorological Data Window

Simulations of energy generated in PVsyst are calculated in hourly intervals over a year. PV Syst takes the meteorological data system parameters provided by a user and performs the simulation. Before running the simulation, it is important to check to see if site file and meteorological file corresponds to site of interest. Those informations can be verified by looking in the Project's description box shaded in blue shown in figure D.5.

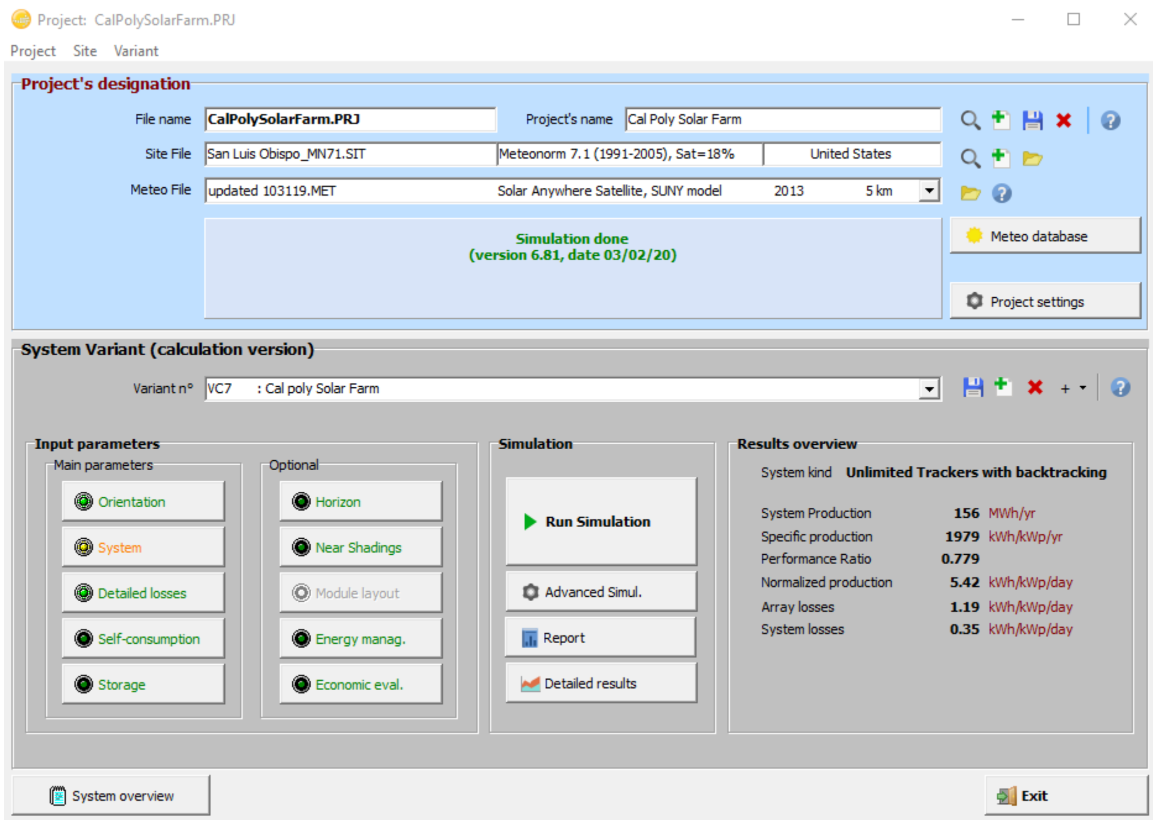


Figure D.5: Project Description Window

After verifying the description of the project, main parameters of the simulation need to reflect the parameters of the actual system being modeled. For this study, PV modules are ground mounted with a single axis tracking capability. Trackers are equipped with backtracking, with tracking angles ranging from -52° to +52°. Pitch distance, the distance between each pole supporting the PV modules, is set to 3.35 meters. There are total of 12 trackers in Cal Poly Solar Field that are set to the same setting. These informations are reflected in the Orientation window from simulation input parameters shown in figure D.6.

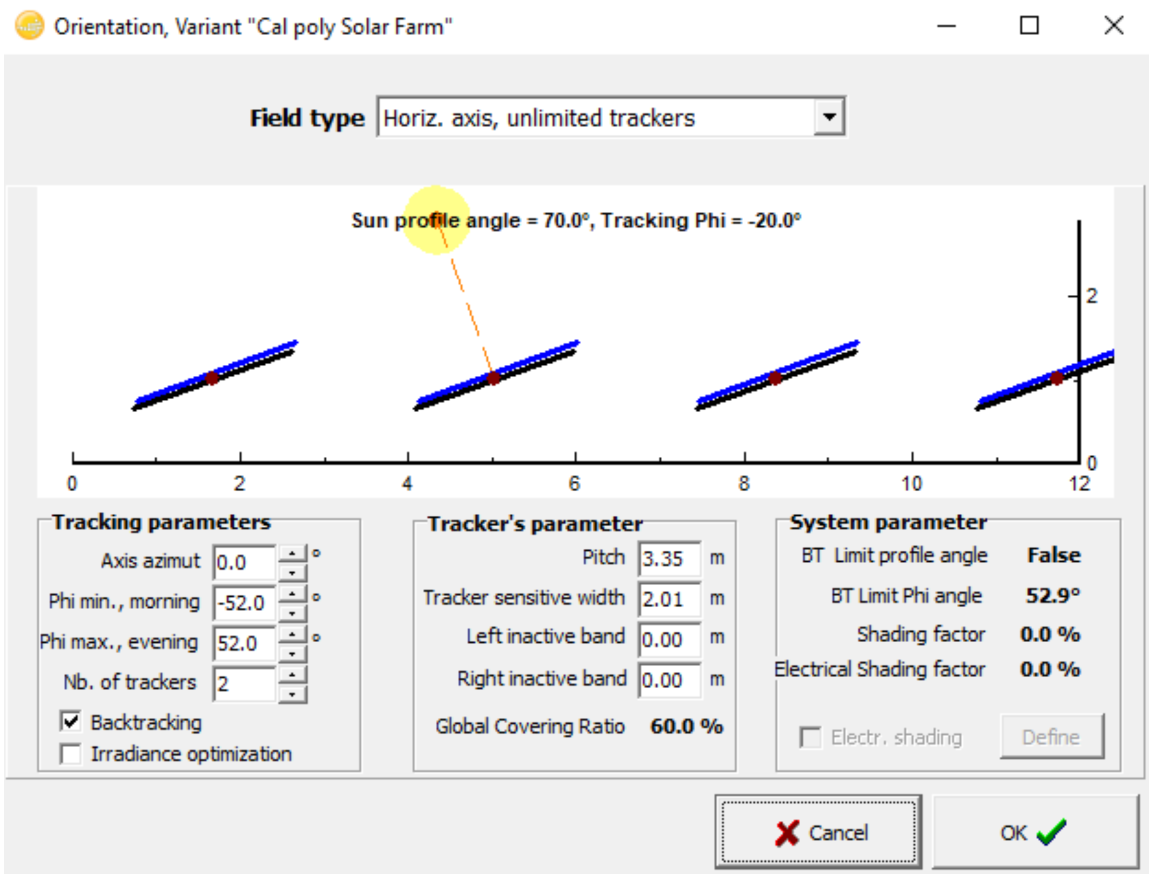


Figure D.6: Orientation Setting Window

Next input parameter is defining system configuration such as PV module and inverter. PV Syst database contains large quantity of different PV modules and inverter, where most of the existing products can be found in the database. For this study, Trina Solar 345-Watt monocrystalline module and Yaskawa Solectria Solar 60TL inverter were selected in the dropdown menu provided in the corresponding section shown in figure 3.9. Each inverter has three mppt input and it is connected to 12 strings of PV modules, where each string has 19 PV modules connected in series. Amount of string and PV module can be inputted in the red box labeled on figure D.7. Figure D.7 represents a simulation model of a single inverter connected to 12 strings of modules to compare the performance of inverter #50 from Cal Poly Solar Field.

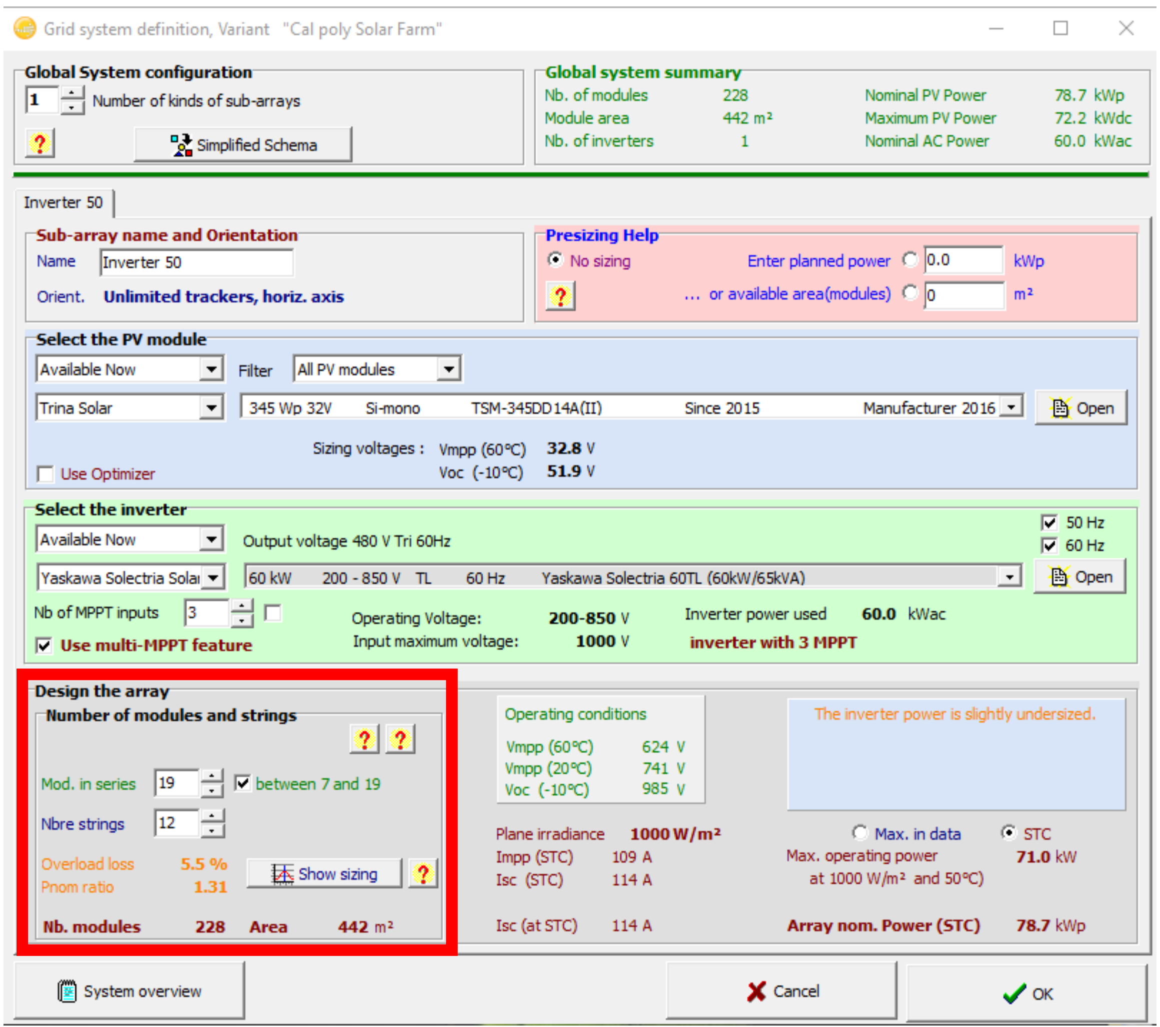


Figure D.7: Grid System Definition Window

Appendix E: MATLAB Model of Inter-row Shaded Panels (Full-Cut Cell)

Table of Contents

Performance Modeling of Inter-row Shaded Panels (Full-Cut Cells)

Andy Kim -- %

Description: The purpose of this modeling tool is to model the performance of solar panels in a row that are affected by inter-row shading.

Input Parameters:

T_amb = Ambient Temperature of the field in degrees celcius Effect_Ir = Effective Irradiance in w/m^2 h_sh = height of triangular shade in m l_sh = length of shade across the row in m shft_sh = length of shade shifted from the end of the panel in m

Output Parameters:

I_mppt = total current output from the panels in amp V_mppt = total voltage output from the panels in volts pmppt = total power output from the panels in watts -- %

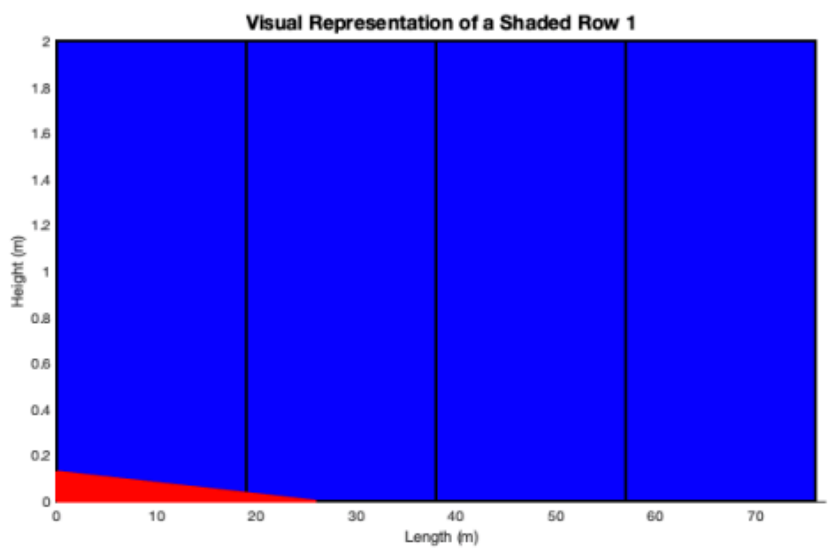
```
clear all;
close all;
```

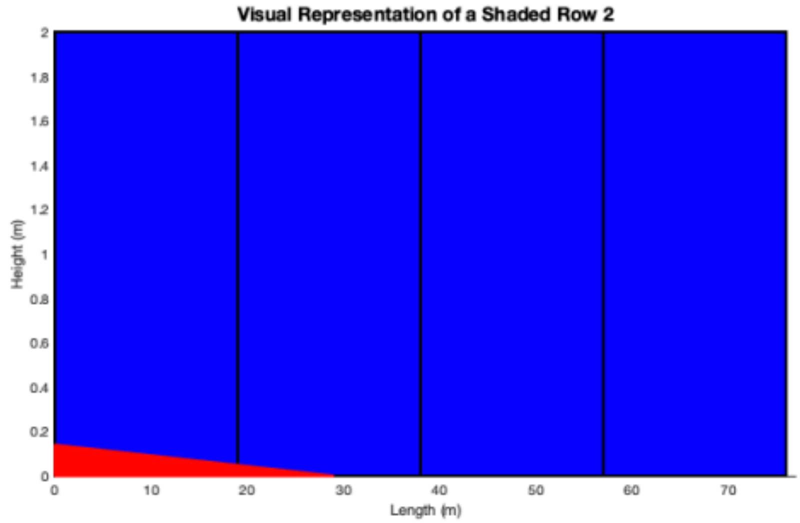
Visual Representation of Shade

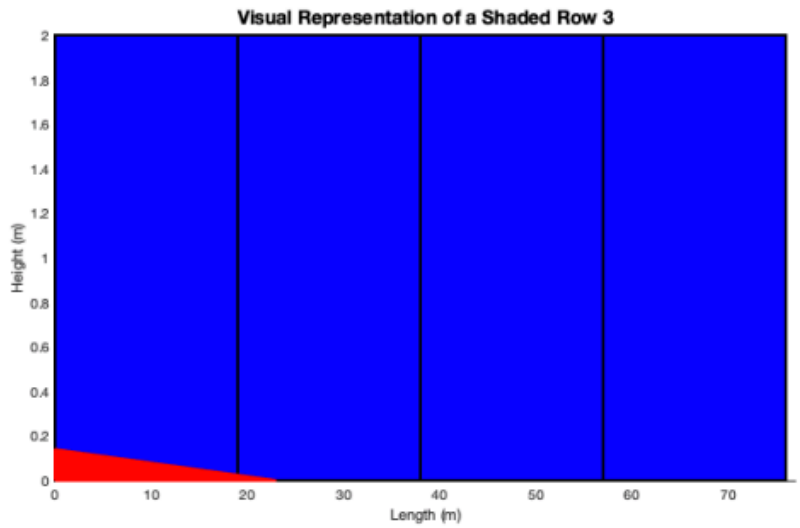
```
for g = 1:4

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%
```

```
%%%%% USER INPUT - TRIANGULAR SHADING %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%
% height of shade from the bottom of panels (m) at 4:37 pm on 4/23/20
h1 = [0.127 0.1397 0.1397 0];
% length of shade from the bottom of panels (m) at 4:37 pm on 4/23/20
l1 = [26 29 23 0];
% distance of shade shifted from left side of panels(m) at 4:37 pm on
 4/23/20
sh1 = [0 0 0 0];

% height of shade from the bottom of panels (m) at 5:17 pm on 4/23/20
h2 = [0.3048 0.3048 0.2921 0];
% length of shade from the bottom of panels (m) at 5:17 pm on 4/23/20
l2 = [33 35 34 0];
% distance of shade shifted from left side of panels(m) at 5:17 pm on
 4/23/20
sh2 = [0 0 0 0];

%%%%% USER INPUT - EFFECTIVE IRRADIANCE AND AMBIENT TEMPERATURE %%%%%
%%%%%%
% effective irradiance in w/m^2 at 5:17 pm on 4/23/20
Effect_Ir = 650;
% ambient temperature in deg c at 5:17 pm on 4/23/20
T_amb= 18;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%

if g ~= 4
    % creating a rectanglualr shapes to represent a row of solar
 panels
    % Each rectangle represents string of 19 modules. total 4 strings
 per
    % mppt input.

    figure((g*2)-1)
    rectangle('Position',[0 1 19 2],'FaceColor',[0 0
 1],'EdgeColor','k',...
        'LineWidth',2);
    hold on
    rectangle('Position',[19 1 19 2],'FaceColor',[0 0
 1],'EdgeColor','k',...
        'LineWidth',2);
    rectangle('Position',[38 1 19 2],'FaceColor',[0 0
 1],'EdgeColor','k',...
        'LineWidth',2);
    rectangle('Position',[57 1 19 2],'FaceColor',[0 0
 1],'EdgeColor','k',...
        'LineWidth',2);
```

```
end

% taking shade measurements one row at a time
h_sh = h1(g); % height of the shade from the bottom of panels (m)
l_sh = l1(g); % length of the shade from the bottom of panels (m)
shft_sh = sh1(g); % distance of shade shifted from left side of
 panels(m)

% plotting outline of triangular shade with solar panels
h_0 = h_sh;   % adjusted height of shade (m) for plotting
ang = atand(h_sh/l_sh);    % angle of triangular shade

h = linspace(0,h_0,l_sh); % height of shadow for animation
l = linspace(shft_sh,l_sh,l_sh); % length of shadow for animation

% figure and annotation for the plot
if g ~= 4
    for i=1:length(h)
        x = [shft_sh shft_sh l(i) shft_sh];
        y = [0 h(i) 0 0];
        plot(x,y,'LineWidth',2 ,'Color', 'r')
        axis([0 77 0 2])
    end

xlabel('Length (m)')
ylabel('Height (m)')
title(strcat('Visual Representation of a Shaded Row',{' '},
 num2str(g)),'FontSize',14)
set(gcf, 'Position',  [360, 300, 700, 400])
end
```







Define Full Cut Cell Module

Propertis of Trina TSM-DE14A(II) solar panel (visit PV_LIB explanantion for each variable listed below)

```
Module.gamma_ref=0.941;
Module.mugamma=-0.00039;
Module.IL_ref=9.51;
Module.I0_ref=0.22e-10;
Module.Rsh_ref=750.3;
Module.Rsh0=2000;
Module.Rshexp=20;
Module.Rs_ref=0.339;
```

```
Module.eG=0.95;
Module.Ns=72;
aIsc=0.0048;
```

Calculate Module IV Performance

Estimating cell temperature using ambient temperature

```
Tcell = T_amb + (30*Effect_Ir/1000); % temperature of the cell in deg
 c

% built in functino from PV_LIB to calculate I and V for IV curve
[IL, I0, Rs, Rsh, a] = pvl_calcparams_PVsyst(Effect_Ir, Tcell, aIsc,
 Module);
NumPoints = 2850; % generates 1000 values for current and voltage
[IVResult] = pvl_singlediode(IL, I0, Rs, Rsh, a, NumPoints);
```

Decimal place adjustment for Shifting Shade

this for loop below outputs exact decimal places of input shift_sh to change shift_sh to a whle number depending on the length of the shift and converage of the cells. function zeros only accepts whole numbers

```
place = zeros(1,3);

for i = 1:3
    d = [0.1 0.01 0.001];
    place(i) = round( (mod(shft_sh,10*d(i))-mod(shft_sh,d(i)))/d(i) );
end
dec = place(1)/10 + place(2)/100 + place(3)/1000; % decimal of
 shift_sh up to thousands place

% this if statements determine the amount of spaces to shift the shade
 when
% shift_sh is not a whole number. If a shade is covering more than 70
 % of
% a substring, then if statement includes that substring to be shaded.
 each
% subtring is considered to be 1/3 of the panel (width wise).

if dec == 0 % if shift_sh is a whole numer
    shft_sh = shft_sh*3; % outputs amount to shift the shade
elseif     (0 < dec) && (dec < 1/3) % detects if the shade is within
 the first substring
    if dec < 0.7 /3 % filters shade covering more than 30%
        if round(shft_sh) == 0 % if the shade is close to the end of
 the entire string (19 panels)
            shft_sh = 1; % indicates that shade is covering the very
 first substring
        else
            shft_sh = round(shft_sh)*3+1; % shifts the shade according
 to the amount of panels
        end
    else
```

```
        shft_sh = round(shft_sh); % if the shade is not covering 70%
 of the substring
                                    % then current can flow through
 this string
    end
elseif (1/3 < dec) && (dec < 2/3) % detects shade within second
 substring
    if (1/3 < dec) && (dec < 2/3 * 0.7) % detects if shade is covering
 more than 30% of second substring
        shft_sh = round(shft_sh)*3+1; % blocks out second substring
 and the substring previous to this stirng
    else
        shft_sh = (round(shft_sh)-1)*3 + 2; % opens this substring
 since its not covering 30% or more
    end
elseif (2/3 < dec) && (dec < 0.99999) % detects shade within thrid
 substring
     if (2/3 < dec) && (dec < 0.7) % detects if shade is covering more
 than 30% of third substring
        shft_sh = round(shft_sh)*3; % blocks out third substring and
 the substrings previous to this stirng
     else
        shft_sh = round(shft_sh)*3 - 1; % opens this substring since
 its not covering 30% or more
     end
end

% defines where the shade is according to the shift of the shade
z = [zeros(1,shft_sh),linspace(l_sh,1,l_sh*3-shft_sh),zeros(1,228-
l_sh*3)];
h = tand(ang)*z; % finds the height of the shade at each substring

% this if statement detects when there is no shading, thus setting
 shade
% height to 0 for all substring.
if isnan(h)
    h = zeros(1,228); % set all height of shade to zero on every
 substring
end
```

Potential Current Putput of Shaded Panels

this for loop determines the current for each substring depending on the height of the shade.

```
string =zeros(1,228);
a_mpp = IVResult.Imp; % mppt current at user input irradiance

for i= 1:length(z)
    if h(i) >= 0.083425 % if height of the shade is covering more than
 30% of the cell
       string(i) = 0; % that substring will produce zero amps
    elseif (0.0364 < h(i)) && (h(i) < 0.083425)  % for shade covering
 less than 30 % of the solar cell
```

```
      string(i) = a_mpp*(1-((h(i)-0.0364)/0.1524)); % current reduces
 accroding the amount of cell coverage
    elseif h(i) < 0.0364 % if the shade is less than 0.022 m (right
 below the solar cell's loaction)
      string(i) = a_mpp; % that substring will produce maximum
 current
    end
end
```

rows to panel

this for loop reshapes the string into 76 by 3 array to represent each panel per row. According to the REC twinpeak series, if two substring are shaded on the bottom half of the panel, it blocks out the last substring so that the panel only proudces energy from the top half at max power.

```
% sizing the array before the for loop
panel = zeros(76,3); % each row represents one panel and each column
 represents one substring
stringx = zeros(228,1);

for i = 1:76
    % turns 1 by 228 array to 76 by 3 array to represent individual
 panel with each subtring current
    panel(i,:) =
 [string(1,1+(3*i-3)),string(1,2+(3*i-3)),string(1,3+(3*i-3))];

    % converts the 76 by 3 array to 228 by 1 array in a correct order
    stringx(1+(3*i-3):1+(3*i-3)+2,1) = panel(i,:).';
end
string = stringx.'; % transposed the 228 by 1 array to 1 by 228, back
 to the orginal form to represent current of each substring
string1 = string(1,1:57); % current output of substrings for first
 string
string2 = string(1,58:114); % current output of substrings for second
 string
string3 = string(1,115:171); % current output of substrings for third
 string
string4 = string(1,172:228); % current output of substrings for fourth
 string

% combines above strings into one 4 by 57 array for better visual
 understanding
stringz = [string1; string2; string3; string4];
```

Assigning minimum amp for each row

these series of if statements determines the minimum current of each string and since matlab outputs empty value for a false statement, when the whole string is shaded, minimum amp is zero, and if its not shaded, the code finds the lowest current for the string.

```
min_amp = []; % empty array set for minimum amp
```

```
if isempty(min(string1(string1>0))) % if the maximum current of the
 first string is 0
    min_amp(1) = 0; % assigns zero current to the minimum current for
 first string
  else
    % if the minimum current is not zero, code finds the minimum
 current at assigns it to min_amp
    min_amp(1) = min(string1(string1>0));
end

if isempty(min(string2(string2>0)))
    min_amp(2) = 0;
  else
    min_amp(2) = min(string2(string2>0));
end

if isempty(min(string3(string3>0)))
    min_amp(3) = 0;
  else
    min_amp(3) = min(string3(string3>0));
end

if isempty(min(string4(string4>0)))
    min_amp(4) = 0;
  else
    min_amp(4) = min(string4(string4>0));
end
```

Finding amount of active substring for each string

initializing array size for each row

```
r1 = zeros(1,19);
r2 = zeros(1,19);
r3 = zeros(1,19);
r4 = zeros(1,19);

% this for loop runs through all eevery substring in all 76 panels and
% finds amount of substring not operating due to shade. Then amount of
 non
% working subtring is subtracted by 57 to get the amount of working
% subtring for each row.

for i = 1:76
    if 1 <= i && i <= 19
        r1(i) = sum(panel(i,:)==0);
        r1 = sum(r1);
        row(1) = 57 - r1;
    end
    if 20 <= i && i <= 38
        r2(i) = sum(panel(i,:)==0);
```

```
        r2 = sum(r2);
        row(2) = 57 - r2;
    end
    if 39 <= i && i <= 57
        r3(i) = sum(panel(i,:)==0);
        r3 = sum(r3);
        row(3) = 57 - r3;
    end
    if 58 <= i && i <= 76
        r4(i) = sum(panel(i,:)==0);
        r4 = sum(r4);
        row(4) = 57 - r4;
    end
end
```

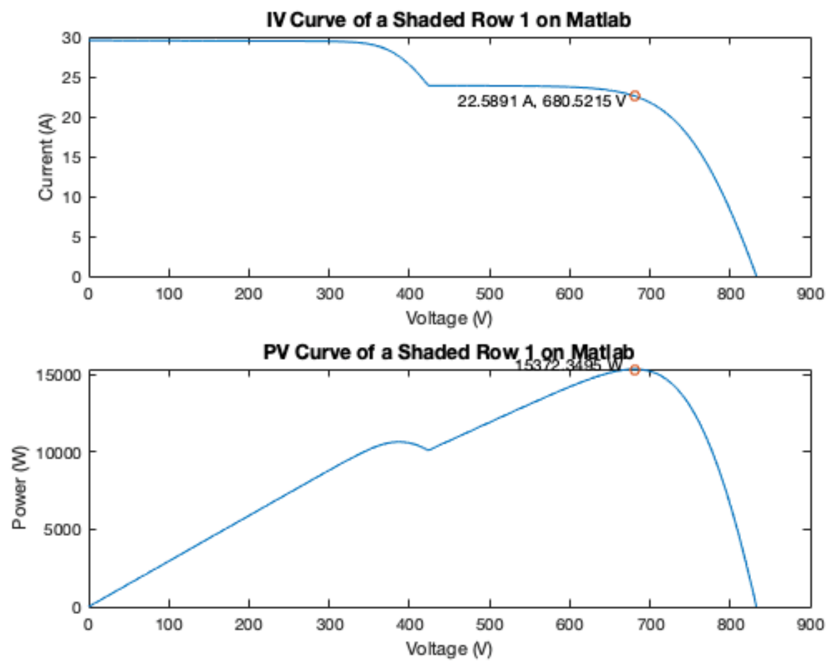
Finding single MPPT point for all strings

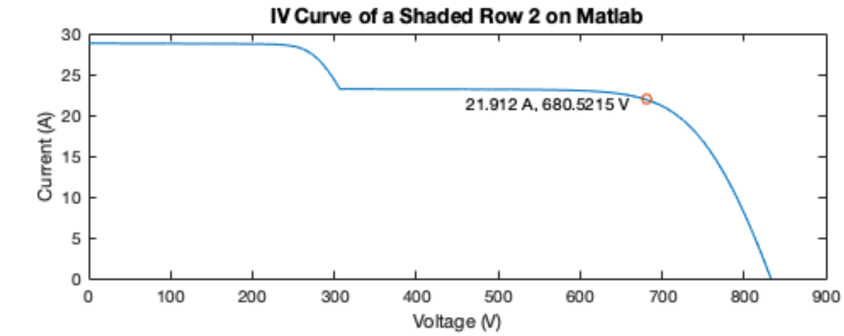
finding the maximum data points for IV curve

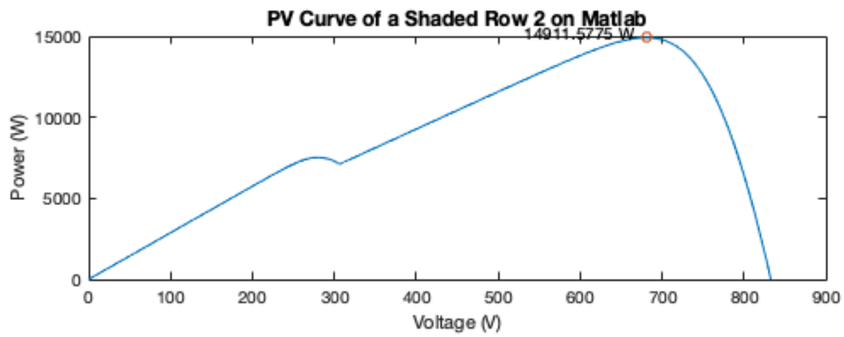
```
pts = row * 50;
numpt = max(pts);

% initializing variables
I_str = zeros(4,row(4)*50);

% this for loop finds the IV curve for each row. Because strings are
% connected in parallel and have different shading conditions. It is
% necessary to find IV curve for each string. Each row is multiplied
 by the
% ratio of miniumum amp and mppt amp, since entie string is limited by
 the
% smallest current.

for i = 1:4
    % built in functino from PV_LIB to calculate current and voltage
 for IV curve
    [IL, I0, Rs, Rsh, a] = pvl_calcparams_PVsyst(Effect_Ir, Tcell,
 aIsc, Module);
    NumPoints = pts(i);
    [IVResult] = pvl_singlediode(IL, I0, Rs, Rsh, a, NumPoints);
    I_str(i,:) = [IVResult.I(1,:), zeros(1,numpt-
NumPoints)]*(min_amp(i)/a_mpp);
    V_str = IVResult.V(1,:)* row(i)/3;
end

% calculating toal current, voltage and power
I = sum(I_str); % current in amps
V = V_str; % voltage in volts
P = I .* V; % power in watts

% finding mppt power, current, and voltage
% maximum power point is assumed to be global maxima
pks = findpeaks(P);
```

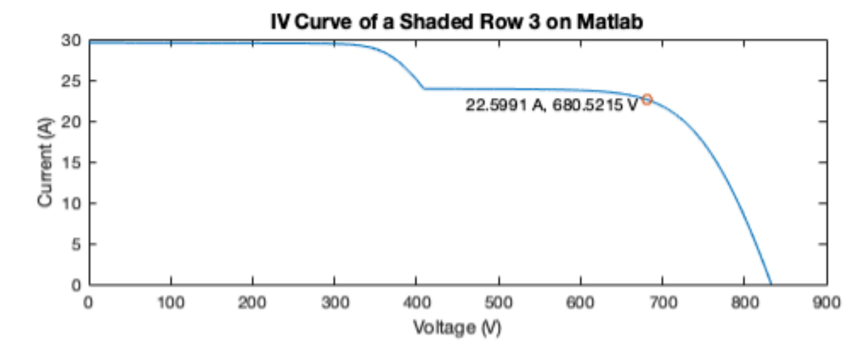
```
if length(pks) == 2
    pks = pks(2);
end

pmppt(g) = max(pks); % mppt power at global maxima
V_mppt(g) = V(P == max(pks)); % mppt voltage related to global maxima
I_mppt(g) = I(V==V_mppt(g)); % mppt current at the selected mppt
 voltage

if g ~=4

    % plotting IV curve
    figure(g*2)
    subplot(2,1,1);
    plot(V,I);
    hold on
    plot(V_mppt(g),I_mppt(g),'o')
    text(V_mppt(g)-220,I_mppt(g)-0.5,[num2str(I_mppt(g)) ' A, '
 num2str(V_mppt(g)), ' V'])
    xlabel('Voltage (V)')
    ylabel('Current (A)')
    title(strcat('IV Curve of a Shaded Row',{' '}, num2str(g),' on
 Matlab') ,'FontSize',12)

    % plotting PV curve
    subplot(2,1,2);
    plot(V,P);
    hold on
    plot(V_mppt(g),max(pks),'o')
    text(V_mppt(g)-150,max(pks)+300,[num2str(max(pks)) ' W'])
    xlabel('Voltage (V)')
    ylabel('Power (W)')
    title(strcat('PV Curve of a Shaded Row',{' '}, num2str(g),' on
 Matlab') ,'FontSize',12)
end

% calculates the difference between actual values and theoratical
 value
a_actual = [21.9   22.1   23.3   30.1]; % actual current output at
 4/23/20 16:37
v_actual = [685.1  683.3  683.1  670.7]; % actual voltage output at
 4/23/20 16:37

a_actual2 = [11.8   12   12.6  23.5]; % actual current output at
 4/23/20 17:17
v_actual2 = [702.6  703  680   677.7]; % actual voltage output at
 4/23/20 17:17

% initializing variables for power
p_act = zeros(1,4);
p_act2 = zeros(1,4);

% this for loop calculates power from measured current and voltage
for b = 1:4
```

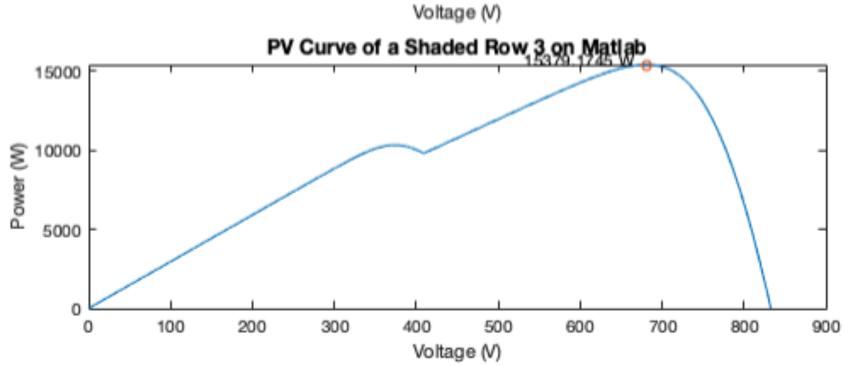
```
    p_act(b) = a_actual(b)*v_actual(b); % actual power output at
 4/23/20 16:37
    p_act2(b) = a_actual2(b)*v_actual2(b); % actual power output at
 4/23/20 17:17
end

% calculates the percent error of current, voltage, and power
if g ~= 4
    Diff.current = (a_actual2(g) - I_mppt(g))/a_actual2(g) *100; %
 current percent error
    Diff.voltage = (v_actual2(g) - V_mppt(g))/v_actual2(g) *100; %
 voltage percent error
    Diff.power = (p_act2(g)- max(pks)) / p_act2(g) * 100; % power
 percent error
    %disp(Diff)
end
```




IV Curve of a Shaded Row 2 on Matlab
Current (A)
Voltage (V)
21.912 A, 680.5215 V
0
5
10
15
20
25
30
0
100
200
300
400
500
600
700
800
900


PV Curve of a Shaded Row 2 on Matlab
Power (W)
Voltage (V)
14911.5775 W
0
5000
10000
15000
0
100
200
300
400
500
600
700
800
900


IV Curve of a Shaded Row 3 on Matlab
Current (A)
Voltage (V)
22.5991 A, 680.5215 V
0
5
10
15
20
25
30
0
100
200
300
400
500
600
700
800
900


PV Curve of a Shaded Row 3 on Matlab
Power (W)
Voltage (V)
0
5000
10000
15000
0
100
200
300
400
500
600
700
800
900

12

```
end
```

Power Loss calculation based on unshaded row

```
ploss =(3*p_act2(4) - sum(pmppt(1:3)))/ (3*p_act2(4))*100;
fprintf('power loss in percentange')
disp(ploss)
```

```
power loss in percentange    4.4262
```

power difference

```
pdiff = 100 - abs((45.5*1000 - sum(pmppt(1:3))))/(45.5*1000)*100 ;
disp(pdiff)
```

```
   99.6415
```

Published with MATLAB® R2020a

Appendix F: MATLAB Model of Inter-row Shaded Panels (Half-Cut Cell)

Table of Contents

Performance Modeling of Inter-row Shaded Panels (Half-Cut Cells)

Andy Kim --

Description: The purpose of this modeling tool is to model the performance of solar panels in a row that are affected by inter-row shading.

Input Parameters:

T_amb = Ambient Temperature of the field in degrees celcius Effect_Ir = Effective Irradiance in w/m^2 h_sh = height of triangular shade in m l_sh = length of shade across the row in m shft_sh = length of shade shifted from the end of the panel in m

Output Parameters:

I_mppt = total current output from the panels in amp V_mppt = total voltage output from the panels in volts pmppt = total power output from the panels in watts

NOTE: BECUASE THRID ROW HAS 11 STRINGS, MODELING OF THRID ROW IS DIFFERENT THAN THE FIRST TWO ROWS.

-- %

```
clear all;
close all;
```

Visual Representation of Shade

```
for g = 1:4

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%

% USER INPUT - TRIANGULAR SHADING
```

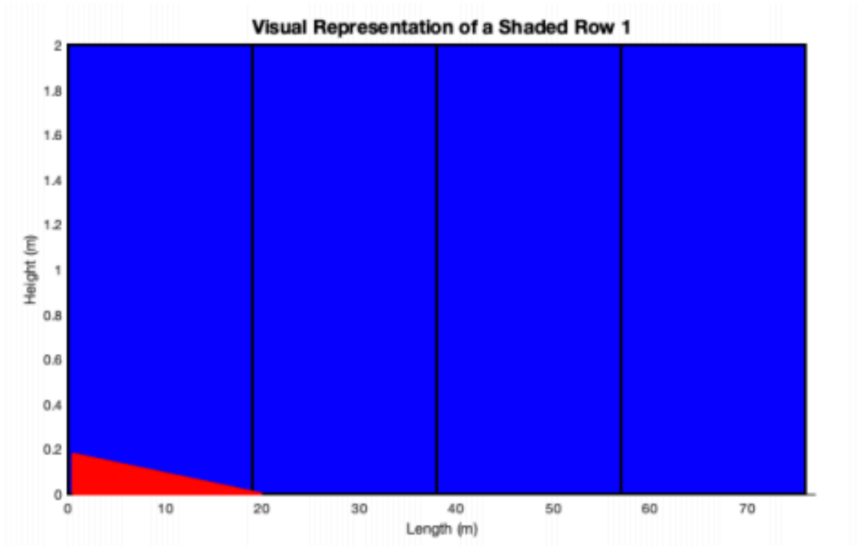
```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%

% USER INPUT - TRIANGULAR SHADING
% height of the shade from the bottom of panels (m)
h1 = [0.1778 0.1524 0.1016 0];
% length of the shade from the bottom of panels (m)
l1 = [20 29 9 0];
% shift of the shade from the edge of panels (m)
sh1 = [0.4953 0.5334 0.4572 0];

% height of the shade from the bottom of panels (m)
h2 = [0.1905 0.1778 0.127 0];
% length of the shade from the bottom of panels (m)
l2 = [20 22 10 0];
% shift of the shade from the edge of panels (m)
sh2 = [0.4445 0.508 0.4318 0];
```

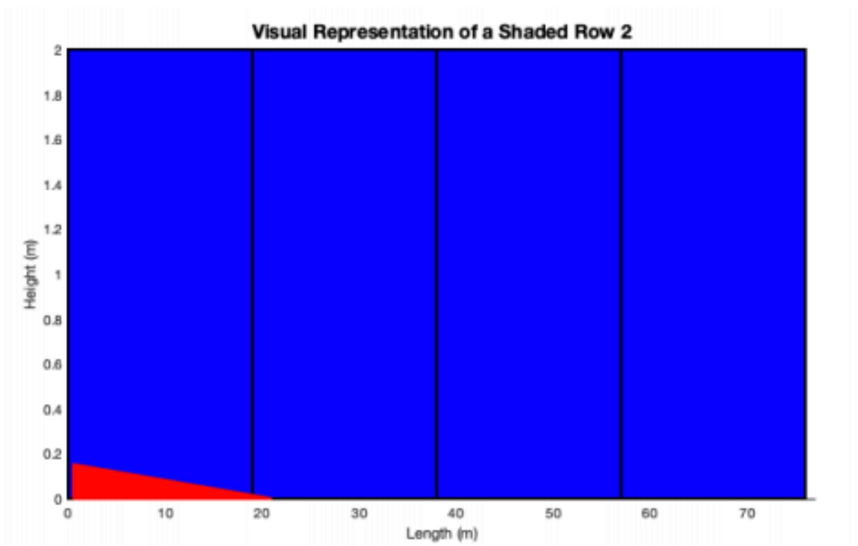
USER INPUT - EFFECTIVE IRRADIANCE AND AMBIENT TEMPERATURE

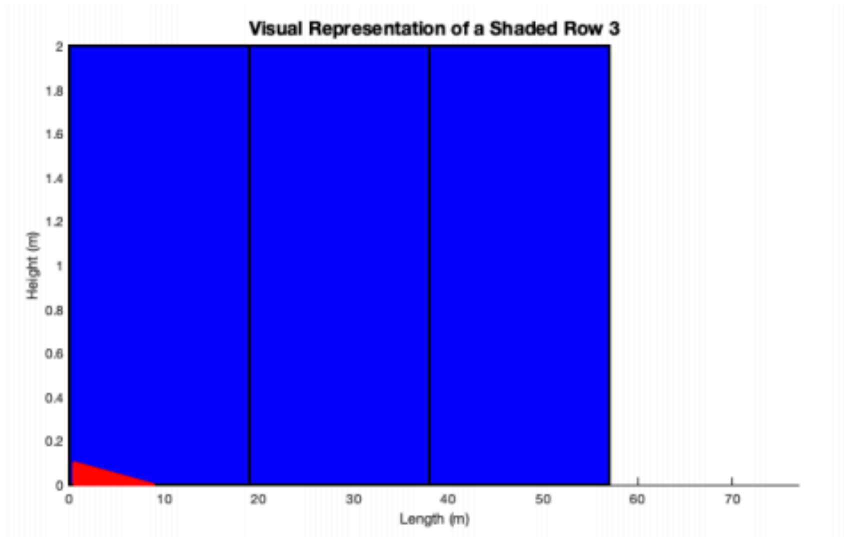
effective irradiance (w/m^2)

```
Effect_Ir = 600;
% ambient temperature (deg c)
T_amb= 18.1;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%

% Each rectangle represents string of 19 modules. total 4 strings per
 mppt
% input.
% creates figure of solar panels for first two rows
if g ~= 3 && g ~= 4
figure((g*2)-1)
rectangle('Position',[0 1 19 2],'FaceColor',[0 0
 1],'EdgeColor','k',...
    'LineWidth',2);
hold all
rectangle('Position',[19 1 19 2],'FaceColor',[0 0
 1],'EdgeColor','k',...
    'LineWidth',2);
rectangle('Position',[38 1 19 2],'FaceColor',[0 0
 1],'EdgeColor','k',...
    'LineWidth',2);
rectangle('Position',[57 1 19 2],'FaceColor',[0 0
 1],'EdgeColor','k',...
```

```
    'LineWidth',2);
end

% creates figure of solar panels for the thrid row
if g == 3
    figure((g*2)-1)
    rectangle('Position',[0 1 19 2],'FaceColor',[0 0
 1],'EdgeColor','k',...
    'LineWidth',2);
    hold all
    rectangle('Position',[19 1 19 2],'FaceColor',[0 0
 1],'EdgeColor','k',...
        'LineWidth',2);
    rectangle('Position',[38 1 19 2],'FaceColor',[0 0
 1],'EdgeColor','k',...
        'LineWidth',2);
end

% translating the user input to the model
h_sh = h2(g); % height of the shade from the bottom of panels (m)
l_sh = l2(g); % length of the shade from the bottom of panels (m)
shft_sh = sh2(g); % distance of shade shifted from left side of
 panels(m)

% plotting outline of triangular shade with solar panels
h_0 = 1 + h_sh;   % adjusted height of shade (m) for plotting
ang = atand(h_sh/l_sh);   % angle of triangular shade

h = linspace(1,h_0,l_sh); % height of shadow for animation
l = linspace(shft_sh,l_sh,l_sh); % length of shadow for animation

% figure and annotation for the plot
if g ~= 4
    for i=1:length(h)
        x = [shft_sh shft_sh l(i) shft_sh];
        y = [1 h(i) 1 1];
        plot(x,y,'LineWidth',2 ,'Color', 'r')
        axis([0 77 0 4])
    end

    xlabel('Length (m)')
    ylabel('Height (m)')
    title(strcat('Visual Representation of a Shaded Row',{' '},
 num2str(g)),'FontSize',14)
    set(gcf, 'Position',  [360, 300, 700, 400])
end
```


Visual Representation of a Shaded Row 1
Height (m)
Length (m)


Visual Representation of a Shaded Row 2
Height (m)
Length (m)



Define Half-Cut Cell Module

Propertis of REC Solar 345 TP 2S 72 solar panel (visit PV_LIB explanantion for each variable listed below)

```
Module.gamma_ref=1.027;
Module.mugamma=-0.00023;
Module.IL_ref=9.48;
Module.I0_ref=2.21e-10;
Module.Rsh_ref=500;
Module.Rsh0=750;
Module.Rshexp=2;
Module.Rs_ref=0.30;
Module.eG=0.95;
Module.Ns=72;
aIsc=0.0064;
```

Calculate Module IV Performance

Estimating cell temperature using ambient temepratture

```
Tcell = T_amb + 28 * Effect_Ir / 800; % temperature of the cell in deg
 C

% built in functino from PV_LIB to calculate I and V for IV curve
[IL, I0, Rs, Rsh, a] = pvl_calcparams_PVsyst(Effect_Ir, Tcell, aIsc,
 Module);
NumPoints = 2850; % generates 1000 values for current and voltage
[IVResult] = pvl_singlediode(IL, I0, Rs, Rsh, a, NumPoints);
```

Decimal place adjustment for Shifting Shade

this for loop below outputs exact decimal places of input shift_sh to change shift_sh to a whle number depending on the length of the shift and converage of the cells. function zeros only accepts whole numbers

```
place = zeros(1,3);

for i = 1:3
    d = [0.1 0.01 0.001];
    place(i) = round( (mod(shft_sh,10*d(i))-mod(shft_sh,d(i)))/d(i) );
end
dec = place(1)/10 + place(2)/100 + place(3)/1000; % decimal of
 shift_sh up to thousands place

% this if statements determine the amount of spaces to shift the shade
 when
% shift_sh is not a whole number. If a shade is covering more than 70
 % of
% a substring, then if statement includes that substring to be shaded.
 each
% subtring is considered to be 1/3 of the panel (width wise).

if dec == 0 % if shift_sh is a whole numer
    shft_sh = shft_sh*3; % outputs amount to shift the shade
elseif    (0 < dec) && (dec < 1/3) % detects if the shade is within
 the first substring
    if dec < 0.7 /3 % filters shade covering more than 30%
        if round(shft_sh) == 0 % if the shade is close to the end of
 the entire string (19 panels)
           shft_sh = 1; % indicates that shade is covering the very
 first substring
        else
           shft_sh = round(shft_sh)*3+1; % shifts the shade according
 to the amount of panels
        end
    else
        shft_sh = round(shft_sh); % if the shade is not covering 70%
 of the substring
                                  % then current can flow through
 this string
    end
elseif (1/3 < dec) && (dec < 2/3) % detects shade within second
 substring
    if (1/3 < dec) && (dec < 2/3 * 0.7) % detects if shade is covering
 more than 30% of second substring
        shft_sh = round(shft_sh)*3+1; % blocks out second substring
 and the substring previous to this stirng
    else
        shft_sh = (round(shft_sh)-1)*3 + 2; % opens this substring
 since its not covering 30% or more
    end
elseif (2/3 < dec) && (dec < 0.99999) % detects shade within thrid
 substring
```

```
    if (2/3 < dec) && (dec < 0.7) % detects if shade is covering more
 than 30% of third substring
       shft_sh = round(shft_sh)*3; % blocks out third substring and
 the substrings previous to this stirng
    else
       shft_sh = round(shft_sh)*3 - 1; % opens this substring since
 its not covering 30% or more
    end
end
```

Total Current Output Calculation

```
% Total Current Output Calculation for First and Second row
if g ~= 3
    % defines where the shade is according to the shift of the shade
    z = [zeros(1,shft_sh),linspace(l_sh,1,l_sh*3-shft_sh),zeros(1,228-
l_sh*3)];
    h = tand(ang)*z; % finds the height of the shade at each substring

    % this if statement detects when there is no shading, thus setting
 shade
    % height to 0 for all substring.
    if isnan(h)
        h = zeros(1,228); % set all height of shade to zero on every
 substring
    end

    string = zeros(1,228); % sizing the string before for loop
    a_mpp = IVResult.Imp; % mppt current at user input irradiance

    % this for loop determines the current for each substring
 depending on the
    % height of the shade.

    for i= 1:length(z)
        if h(i) > 0.04826 % if height of the shade is bigger than
 0.07886 m
            string(i) = 0; % that substring will produce zero amps
        elseif h(i) < 0.0254 % if the shade is less than 0.056 m
 (right below the solar cell's loaction)
            string(i) = a_mpp/2; % that substring will produce maximum
 current (half of mppt current for half cut cells)
        elseif (0.0254 < h(i)) && (h(i) < 0.04826)  % for shade
 covering less than 30 % of the solar cell
            string(i) = a_mpp/2*(1-((h(i)-0.0254)/0.0762)); % current
 reduces accroding the amount of cell coverage
        end
    end

    % this for loop reshapes the string into 76 by 3 array to
 represent each
    % panel per row. According to the REC twinpeak series, if two
 substring are
```

```
    % shaded on the bottom half of the panel, it blocks out the last
substring
    % so that the panel only proudces energy from the top half at max
power.


    panel = zeros(76,3); % each row represents one panel with each
column represents each substring (sizing the array before for loop)
    stringx = zeros(228,1); % sizing the array befroe for loop
    for i = 1:76
        panel(i,:) =
[string(1,1+(3*i-3)),string(1,2+(3*i-3)),string(1,3+(3*i-3))]; %
turns 1 by 228 array to 76 by 3 array to represent individual panel
with each subtring current
            if (panel(i,2) == 0) && (panel(i,3) == 0) % if substring 2
and 3 is shaded
                panel(i,1) = 0; % the first subtring will not produce
any current
            elseif (panel(i,2) == 0) && (panel(i,1) == 0) % if
substring 2 and 1 is shaded
                panel(i,3) = 0; % the thrid subtring will not produce
any current
            end
        stringx(1+(3*i-3):1+(3*i-3)+2,1) = panel(i,:).'; % converts
the 76 by 3 array to 228 by 1 array in a correct order
    end
    string = stringx.'; % transposed the 228 by 1 array to 1 by 228,
back to the orginal form to represent current of each substring
    string1 = string(1,1:57); % current output of substrings for first
string out of 4 strings
    string2 = string(1,58:114); % current output ofsubstrings for
second string out of 4 strings
    string3 = string(1,115:171); % current output ofsubstrings for
third string out of 4 strings
    string4 = string(1,172:228); % current output ofsubstrings for
fourth string out of 4 strings
    stringz = [string1; string2; string3; string4]; % combines above
strings into one 4 by 57 array for better visual understanding

    % these series of if statements determines the minimum current of
each
    % string and since matlab outputs empty value for a false
statement, when
    % the whole string is shaded, minimum amp is zero, and if its not
shaded,
    % the code finds the lowest current for the string.
    min_amp = []; % empty array set for minimum amp

    if isempty(min(string1(string1>0))) % if the maximum current of
the first string is 0
        min_amp(1) = 0; % assigns zero current to the minimum current
for first string
      else
```

```
        min_amp(1) = min(string1(string1>0)); % if the minimum current
 is not zero, code finds the minimum current at assigns it to min_amp
    end

    if isempty(min(string2(string2>0)))
        min_amp(2) = 0;
      else
        min_amp(2) = min(string2(string2>0));
    end

    if isempty(min(string3(string3>0)))
        min_amp(3) = 0;
      else
        min_amp(3) = min(string3(string3>0));
    end

    if isempty(min(string4(string4>0)))
        min_amp(4) = 0;
      else
        min_amp(4) = min(string4(string4>0));
    end
end


% Total Current Output Calculation of Thrid Row
if g == 3
    % defines where the shade is according to the shift of the shade
    z = [zeros(1,shft_sh),linspace(l_sh,1,l_sh*3-shft_sh),zeros(1,171-
l_sh*3)];
    h = tand(ang)*z; % finds the height of the shade at each substring

    % this if statement detects when there is no shading, thus setting
 shade
    % height to 0 for all substring.
    if isnan(h)
        h = zeros(1,171); % set all height of shade to zero on every
 substring
    end

    string = zeros(1,171); % sizing the string before for loop
    a_mpp = IVResult.Imp; % mppt current at user input irradiance

    % this for loop determines the current for each substring
 depending on the
    % height of the shade.

    for i= 1:length(z)
        if h(i) > 0.04826 % if height of the shade is bigger than
 0.07886 m
           string(i) = 0; % that substring will produce zero amps
        elseif h(i) < 0.0254 % if the shade is less than 0.056 m
 (right below the solar cell's loaction)
           string(i) = a_mpp/2; % that substring will produce maximum
 current (half of mppt current for half cut cells)
```

```
        elseif (0.0254 < h(i)) && (h(i) < 0.04826)  % for shade
covering less than 30 % of the solar cell
            string(i) = a_mpp/2*(1-((h(i)-0.0254)/0.0762)); % current
reduces accroding the amount of cell coverage
        end
    end

    panel = zeros(57,3); % each row represents one panel with each
column represents each substring (sizing the array before for loop)
    stringx = zeros(171,1); % sizing the array befroe for loop
    for i = 1:57
        panel(i,:) =
[string(1,1+(3*i-3)),string(1,2+(3*i-3)),string(1,3+(3*i-3))]; %
turns 1 by 228 array to 76 by 3 array to represent individual panel
with each subtring current
            if (panel(i,2) == 0) && (panel(i,3) == 0) % if substring 2
and 3 is shaded
                panel(i,1) = 0; % the first subtring will not produce
any current
            elseif (panel(i,2) == 0) && (panel(i,1) == 0) % if
substring 2 and 1 is shaded
                panel(i,3) = 0; % the thrid subtring will not produce
any current
            end
        stringx(1+(3*i-3):1+(3*i-3)+2,1) = panel(i,:).'; % converts
the 76 by 3 array to 228 by 1 array in a correct order
    end
    string = stringx.'; % transposed the 228 by 1 array to 1 by 228,
back to the orginal form to represent current of each substring
    string1 = string(1,1:57); % current output of substrings for first
string out of 4 strings
    string2 = string(1,58:114); % current output ofsubstrings for
second string out of 4 strings
    string3 = string(1,115:171); % current output ofsubstrings for
third string out of 4 strings
    stringz = [string1; string2; string3]; % combines above strings
into one 4 by 57 array for better visual understanding

    % these series of if statements determines the minimum current of
each
    % string and since matlab outputs empty value for a false
statement, when
    % the whole string is shaded, minimum amp is zero, and if its not
shaded,
    % the code finds the lowest current for the string.
    min_amp = []; % empty array set for minimum amp

    if isempty(min(string1(string1>0))) % if the maximum current of
the first string is 0
        min_amp(1) = 0; % assigns zero current to the minimum current
for first string
      else
        min_amp(1) = min(string1(string1>0)); % if the minimum current
is not zero, code finds the minimum current at assigns it to min_amp
```

```
    end

    if isempty(min(string2(string2>0)))
        min_amp(2) = 0;
      else
        min_amp(2) = min(string2(string2>0));
    end

    if isempty(min(string3(string3>0)))
        min_amp(3) = 0;
      else
        min_amp(3) = min(string3(string3>0));
    end
end

% we assume there will no shading on top of the panels, therefore it
 is
% outputing half of mppt current
a_top = a_mpp/2;
```

MPPT Tracking

calculating amount of active panels for each row

```
r1 = zeros(1,19);
r2 = zeros(1,19);
r3 = zeros(1,19);
r4 = zeros(1,19);


if g ~= 3 % mppt calculation for Frist and Second row

    % this for loop checks every moudle and its substring to see if it
    % shaded or not. Then outputs number of working substrings in the
    % entire row.

    for i = 1:76
        if 1 <= i && i <= 19
            r1(i) = sum(panel(i,:)==0);
            r1 = sum(r1);
            row(1) = 57 - r1;
        end
        if 20 <= i && i <= 38
            r2(i) = sum(panel(i,:)==0);
            r2 = sum(r2);
            row(2) = 57 - r2;
        end
        if 39 <= i && i <= 57
            r3(i) = sum(panel(i,:)==0);
            r3 = sum(r3);
            row(3) = 57 - r3;
        end
        if 58 <= i && i <= 76
```

```
            r4(i) = sum(panel(i,:)==0);
            r4 = sum(r4);
            row(4) = 57 - r4;
            if row(4) == 57
                top = row(4);
            else
                top = 57;
            end
        end
    end

    % finding the maximum data points for IV curve
    pts = row * 50;
    numpt = max(pts);

    % number of working substring is multiplied by 50 to add more data
 points
    % on the IV curve. number of data point is based on the maximum
 number out
    % of the pts array (usually row 4 has the most working substring).
 For loop
    % below calculates current and voltage for each row. In order to
 have equal
    % amount of data points for current, array of zero is added
 accordingly to
    % match the longest string.

    % initizlaizing variables
    I_str = zeros(4,row(4)*50);

    for i = 1:5
        [IL, I0, Rs, Rsh, a] = pvl_calcparams_PVsyst(Effect_Ir, Tcell,
 aIsc, Module);
        if i == 5
            NumPoints = top*50;
            [IVResult] = pvl_singlediode(IL, I0, Rs, Rsh, a,
 NumPoints);
            I_top = IVResult.I(1,:)/2;
            V_top = IVResult.V(1,:)* top/3;
        else
            % built in functino from PV_LIB to calculate I and V for
 IV curve
            NumPoints = pts(i);
            [IVResult] = pvl_singlediode(IL, I0, Rs, Rsh, a,
 NumPoints);
            I_str(i,:) = [IVResult.I(1,:), zeros(1,numpt-
NumPoints)]/2*(min_amp(i)/a_top);
            V_str = IVResult.V(1,:)* row(i)/3;
        end
    end

    % Curernt, voltage and power calculation. Current from each row is
 added
    % plus the top half of the panels working at full potential
```

```
    I = sum(I_str)+ I_top*4;
    V = V_top;
    P = I .* V;

end


if g == 3 % mppt calculation for thrid row

    % this for loop checks every moudle and its substring to see if it
    % shaded or not. Then outputs number of working substrings in the
    % entire row.

    for i = 1:57
        if 1 <= i && i <= 19
            r1(i) = sum(panel(i,:)==0);
            r1 = sum(r1);
            row(1) = 57 - r1;
        end
        if 20 <= i && i <= 38
            r2(i) = sum(panel(i,:)==0);
            r2 = sum(r2);
            row(2) = 57 - r2;
        end
        if 39 <= i && i <= 57
            r3(i) = sum(panel(i,:)==0);
            r3 = sum(r3);
            row(3) = 57 - r3;
            if row(3) == 57
                top = row(3);
            else
                top = 57;
            end
        end
    end

    % finding the maximum data points for IV curve
    pts = row * 50;
    numpt = max(pts);

    % number of working substring is multiplied by 50 to add more data
 points
    % on the IV curve. number of data point is based on the maximum
 number out
    % of the pts array (usually row 4 has the most working substring).
 For loop
    % below calculates current and voltage for each row. In order to
 have equal
    % amount of data points for current, array of zero is added
 accordingly to
    % match the longest string.

    % initizlaizing variables
    I_str = zeros(3,row(3)*50);
```

```
    for i = 1:4
        [IL, I0, Rs, Rsh, a] = pvl_calcparams_PVsyst(Effect_Ir, Tcell,
 aIsc, Module);
        if i == 4
            NumPoints = top*50;
            [IVResult] = pvl_singlediode(IL, I0, Rs, Rsh, a,
 NumPoints);
            I_top = IVResult.I(1,:)/2;
            V_top = IVResult.V(1,:)* top/3;
        else
            % built in functino from PV_LIB to calculate I and V for
 IV curve
            NumPoints = pts(i);
            [IVResult] = pvl_singlediode(IL, I0, Rs, Rsh, a,
 NumPoints);
            I_str(i,:) = [IVResult.I(1,:), zeros(1,numpt-
NumPoints)]/2*(min_amp(i)/a_top);
            V_str = IVResult.V(1,:)* row(i)/3;
        end
    end

    % Curernt, voltage and power calculation. Current from each row is
 added
    % plus the top half of the panels working at full potential

    I = sum(I_str)+ I_top*3;
    V = V_top;
    P = I .* V;

end
```

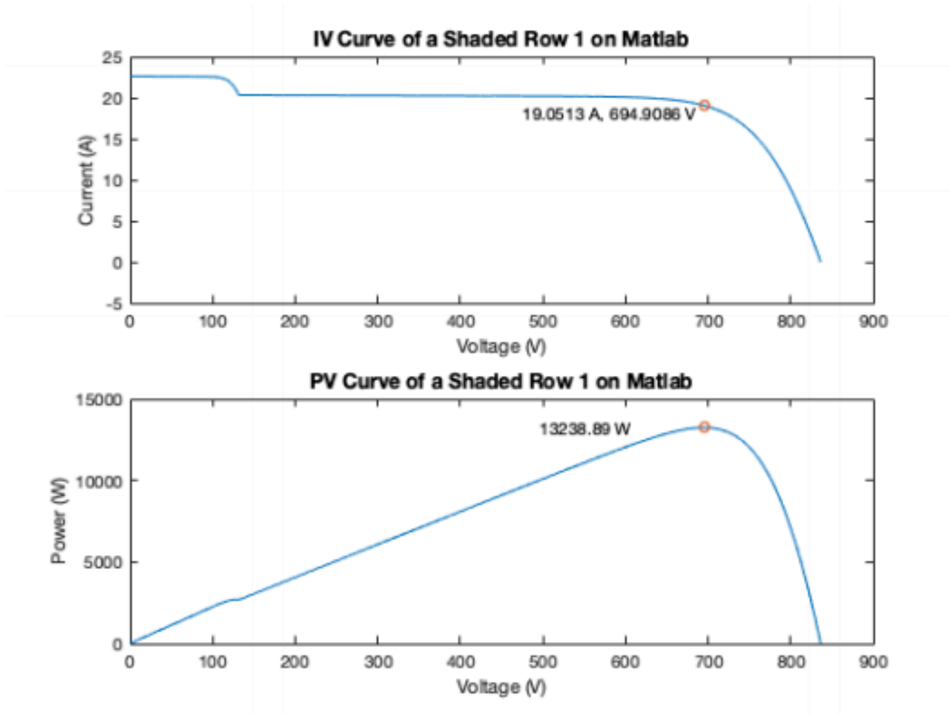
finding mppt power, current, and voltage

maximum power point is assumed to be global maxima

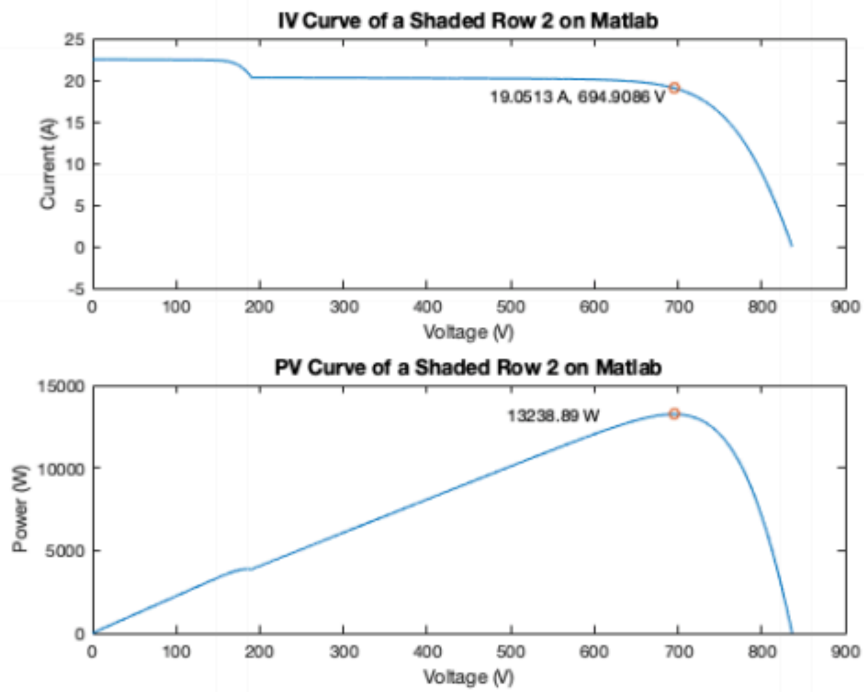
```
pks = findpeaks(P);
if length(pks) == 2
    pks = pks(2);
end

pmppt(g) = max(pks); % mppt power at global maxima
V_mppt(g) = V(P == max(pks)); % mppt voltage related to global maxima
I_mppt(g) = I(V==V_mppt(g)); % mppt current at the selected mppt
 voltage


if g ~=4
    % plotting IV curve
    figure(g*2)
    subplot(2,1,1);
    plot(V,I);
    hold on
```

```
    plot(V_mppt(g),I_mppt(g),'o')
    text(V_mppt(g)-220,I_mppt(g)-1,[num2str(I_mppt(g)) ' A, '
 num2str(V_mppt(g)), ' V'])
    xlabel('Voltage (V)')
    ylabel('Current (A)')
    title(strcat('IV Curve of a Shaded Row',{' '}, num2str(g),' on
 Matlab') ,'FontSize',12)

    % plotting PV curve
    subplot(2,1,2);
    plot(V,P);
    hold on
    plot(V_mppt(g),max(pks),'o')
    text(V_mppt(g)-200,max(pks),[num2str(max(pks)) ' W'])
    xlabel('Voltage (V)')
    ylabel('Power (W)')
    title(strcat('PV Curve of a Shaded Row',{' '}, num2str(g),' on
 Matlab') ,'FontSize',12)
end


% difference to actual value
a_actual = [20.6 19.6 14.7    23.2]; % actual current output at
 4/23/20 17:31
v_actual = [682.2 683.6 686.6   679.4]; % actual current output at
 4/23/20 17:31

a_actual2 = [18.8 17.9   13.5   21.8]; % actual current output at
 4/23/20 17:38
v_actual2 = [687.6   687.2   686.1   678.4]; % actual current output at
 4/23/20 17:38

% initializing variables for power
p_act = zeros(1,4);
p_act2 = zeros(1,4);

% this for loop calculates power from measured current and voltage
for b = 1:4
    p_act(b)  = a_actual(b)*v_actual(b);
    p_act2(b)  = a_actual2(b)*v_actual2(b);
end

% calculates the percent error of current, voltage, and power
if g ~= 4
    Diff.current = (a_actual2(g) - I_mppt(g))/a_actual2(g) *100; %
 current percent error
    Diff.voltage = (v_actual2(g) - V_mppt(g))/v_actual2(g) *100; %
 voltage percent error
    Diff.power = (p_act2(g)- pmppt(g)) / p_act2(g) * 100; % power
 percent error
    disp(Diff)
end

    current: -1.3365
```

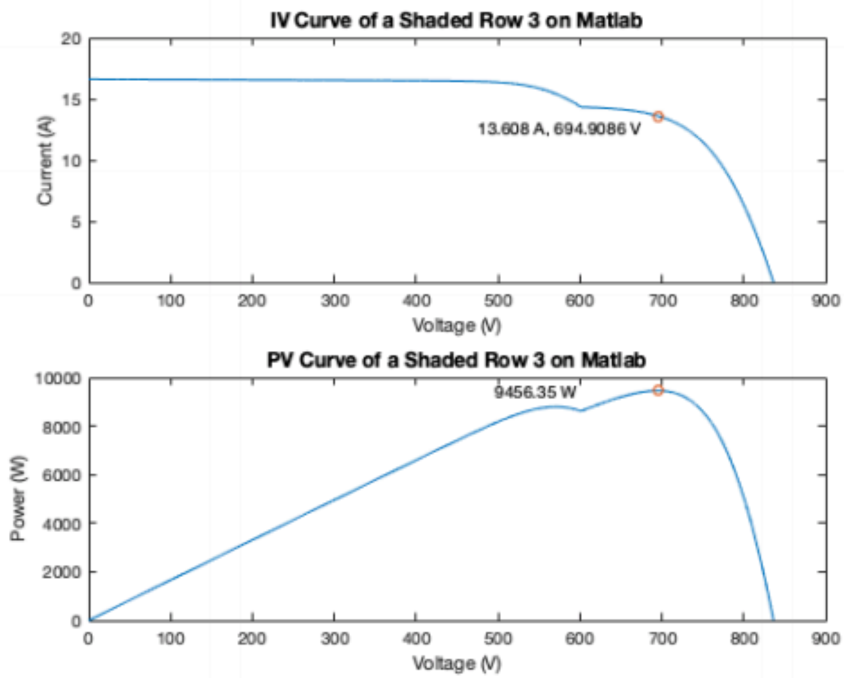
```
voltage: -1.0629
  power: -2.4137
```




```
current: -6.4317
voltage: -1.1217
  power: -7.6256
```




```
current: -0.8004
voltage: -1.2839
  power: -2.0945
```




```
end
```

Power Loss calculation based on unshaded row

```
ploss =(3*p_act2(4) - sum(pmppt(1:3)))/ (3*p_act2(4))*100;
fprintf('power loss in percentange')
disp(ploss)
```

```
power loss in percentange   19.0077
```

power difference

```
pdiff = 100 - abs((37.1*1000 - sum(pmppt(1:3))))/(37.1*1000)*100 ;
disp(pdiff)
```

```
   96.8575
```

Published with MATLAB® R2020a

Appendix G: Percent Error of Current, Voltage, and Power

Tables shown below summarizes the percent error of modeled values and actual values of current, voltage, and power of inverter 55 and inverter 25.

Table G.1: Percent Error of Modeled Values and Actual Values of Current, Voltage, and Power of Inverter 55

Inverter 55 on 4/23/20						
Time	**4:37 pm**			**5:17 pm**		
Row	**1**	**2**	**3**	**1**	**2**	**3**
Current	-3.0491	0.9443	3.0997	0.5266	2.1844	6.8423
Voltage	-0.7009	-0.9662	-0.9957	-0.1196	-0.0627	-3.4471
Power	3.7921	0.0127	2.1348	0.4076	2.1301	3.6311

Table G.2: Percent Error of Modeled Values and Actual Values of Current, Voltage, and Power of Inverter 25

Inverter 25 on 4/23/20						
Time	**5:31 pm**			**5:38 pm**		
Row	**1**	**2**	**3**	**1**	**2**	**3**
Current	1.1414	0.1632	1.0453	0.1256	-4.0635	0.6540
Voltage	-2.4785	0.4568	-1.8218	-2.0594	-2.1188	-2.2825
Power	1.3096	0.6192	-0.7575	-1.9312	-6.2684	-1.6136