SAM Photovoltaic Model Technical Reference

P. Gilman
National Renewable Energy Laboratory

SAM Photovoltaic Model Technical Reference

P. Gilman
National Renewable Energy Laboratory

Prepared under Task No. SS13.5020

This report is available at no cost from the National Renewable Energy Laboratory (NREL) at www.nrel.gov/publications.

Available electronically at SciTech Connect http://www.osti.gov/scitech

Available for a processing fee to U.S. Department of Energy and its contractors, in paper, from:

> U.S. Department of Energy
> Office of Scientific and Technical Information
> P.O. Box 62
> Oak Ridge, TN 37831-0062
> OSTI http://www.osti.gov
> Phone: 865.576.8401
> Fax: 865.576.5728
> Email: reports@osti.gov

Available for sale to the public, in paper, from:

> U.S. Department of Commerce
> National Technical Information Service
> 5301 Shawnee Road
> Alexandra, VA 22312
> NTIS http://www.ntis.gov
> Phone: 800.553.6847 or 703.605.6000
> Fax: 703.605.6900
> Email: orders@ntis.gov

Cover Photos by Dennis Schroeder: (left to right) NREL 26173, NREL 18302, NREL 19758, NREL 29642, NREL 19795.

NREL prints on paper that contains recycled content.

Executive Summary

This manual describes the photovoltaic performance model in the System Advisor Model (SAM). The U.S. Department of Energy's National Renewable Energy Laboratory maintains and distributes SAM, which is available as a free download from `https://sam.nrel.gov`. These descriptions are based on SAM 2015.1.30 (SSC 41).

SAM is a techno-economic feasibility model for renewable energy projects. It is designed for a range of different users, including project developers, system designers, policy makers, financial planners, and academic researchers.

SAM's photovoltaic performance model is available both as part of the SAM desktop application, and in the SAM software development kit (SDK). This manual is intended for people who want to understand SAM's photovoltaic model, or for people who are using the SDK to develop their own applications.

SAM runs on Windows and OS X operating systems, and is a user interface that performs the following functions:

- Organizes and displays the performance and financial model inputs in a user-friendly interface.

- Manages tasks associated with running model simulations.

- Provides options for "advanced" simulations that involve multiple simulation runs for parametric and sensitivity studies.

- Stores arrays of model results.

- Calculates secondary results such as monthly and annual totals, capacity factor, system performance factor, and system losses.

- Displays tables and graphs of results.

- Allows for exporting data in different formats, CSV, graph images, Microsoft Excel, and a PDF report.

The SAM SDK is a package containing the SAM Simulation Core (SSC) libraries and a set of software development tools that allow model developers to create their own interfaces to the simulation modules as either web or desktop applications.

Table of Contents

List of Figures

List of Tables

1 Nomenclature

The organization of this manual is based roughly on the Sandia National Laboratories PV Performance Modeling Collaborative (PVPMC) website "Modeling Steps" (PVPMC Modeling Steps 2014). The nomenclature and general descriptions also draw from the "PVCDROM" electronic document on the pveducation.org website hosted by the Arizona State University Solar Power Labs (PVCDROM 2014).

The variable names in this manual are listed in tables at the beginning of each section. In some cases, the same variable name might be used in two different sections of the manual to represent two different quantities. That is because we have tried to preserve the variable names used in the original sources. For example, E is used to represent solar irradiance in most equations, and for the band-gap energy in two equations in the Sandia module model section.

For solar irradiance values, the letter E indicates data from the weather file, I indicates irradiance incident on the photovoltaic array before soiling and shading, and G indicates effective irradiance incident on the array (at the top of the module cover) after soiling and shading. The subscripts b, d, and g indicate beam, diffuse, and global irradiance values.

The variable P indicates an electrical power value in Watts or kilowatts. Because SAM is an hourly simulation model, these power values are average hourly values, equivalent to Watt-hour (or kilowatt-hour) per hour.

2 Photovoltaic Performance Model Overview

SAM's photovoltaic performance model combines module and inverter submodels (see Table 1) with supplementary code to calculate a photovoltaic power system's hourly AC output given a weather file and data describing the physical characteristics of the module, inverter, and array. The main submodels are listed in Table 1 with a citation of the publication that originally described the modeling approach or an indication that the submodel uses standard equations, or was developed by the National Renewable Energy Laboratory (NREL), in which case the description in this manual is the only documentation of the modeling approach.

The photovoltaic performance model can simulate any size of system, from a small rooftop array and a single inverter to a large system with multiple subarrays and banks of inverters.

The model calculates the system's AC electrical output over one year as an array of 8,760 hourly AC power values. It reads hourly solar resource and temperature data from a weather file describing the resource at the system's location for the year, and uses them with inputs describing the system's design in equations to calculate module and inverter conversion efficiencies and energy losses.

The modeled system must consist of a single type of photovoltaic module and a single type of inverter – it cannot combine different sizes or brands of modules and inverters. The array may consist of up to four subarrays, each with its own set of parameters for tracking, surface angles, shading and soiling, and DC losses. Each subarray can have a different number of modules, but all subarrays must have the same number of modules per string so that all subarrays have the same nominal DC voltage, which serves as the inverter's nominal input voltage.

The array must be connected either to a single inverter or to a bank of inverters connected to each other in parallel. It is not possible to model a system with subarrays connected to different inverters.

The module model (Section 9) and inverter model (Section 11) calculate solar-energy-to-DC-electricity and DC-to-AC electricity conversion efficiencies, respectively, and account for losses associated with each component. The self-shading models (Section 7.3) calculate losses caused by shading of modules in the array by neighboring modules. The photovoltaic performance model does not explicitly calculate the remaining system losses. They are represented by user-specified inputs:

- Beam and diffuse shading losses for near-object shading of the array (Section 7.2). These can be specified as hourly (8,760 values), month-by-hour (288 values), or sun azimuth angle by elevation (number of values varies), and may be generated by shading analysis equipment and software.

- Monthly soiling losses for dust and other accumulation on the array (Section 7.5).

- DC losses for module mismatch, DC wiring and connections, tracking, and other losses associated with the array (Section 10.3).

- AC losses for AC wiring and transformer losses (Section 12.2).

The photovoltaic model uses the same algorithm to calculate self-shading losses for fixed arrays and one-axis trackers (Section 7.3). It does not calculate self-shading losses for two-axis or azimuth-axis trackers. The self-shading algorithm calculates a reduction in diffuse POA irradiance and a DC loss factor to account for the performance impact of the reduction in beam POA irradiance.

The model does not calculate module mismatch losses within a subarray. For systems with more than one subarray, an optional algorithm can estimate mismatch losses between the subarrays (Section 10.6).

2.1 Model Algorithm

This section describes the basic algorithm of SAM's photovoltaic performance model. The details of each step listed below are described in the sections that follow. See Figure 1 for a basic block diagram of the model. Note that the

Table 1. The Primary SAM Photovoltaic Performance Model Submodels

Submodel	Reference
Weather file reader	NREL
Sun position	Michalsky (1988), Iqbal (1983), NREL
Surface angles	standard geometry
Backtracking for one-axis trackers	NREL
Isotropic incident irradiance model	Liu (1963)
HDKR incident irradiance model	Duffie and Beckman (2013), Reindl (1988)
Perez 1990 incident irradiance model	Perez (1988), Perez (1990)
Self shading model for fixed arrays	Deline (2013)
Self shading model for one-axis trackers	NREL
Sandia module model	King (2004)
CEC module model	De Soto (2004a)
Simple efficiency module model	NREL
Subarray mismatch calculator	NREL
Sandia inverter model	King (2007)
Part load inverter model	NREL

block diagram does not include the subarray mismatch and string voltage calculations described in the steps below to make the diagram easier to follow.

The hourly simulation model performs the following calculations for each of the 8,760 hours in a year:

1. For each of up to four subarrays:

    A. Calculate sun angles from date, time, and geographic position data from the weather file. (Section 4.2)

    B. Calculate the nominal beam and diffuse irradiance incident on the plane of array (POA irradiance). This depends on the solar irradiance data in the weather file, sun angle calculations, user-specified subarray parameters such as tracking and orientation parameters, and backtracking option for one-axis trackers. (Section 7.1)

    C. Apply the user-specified beam and diffuse near-object shading factors to the nominal POA irradiance. (Section 7.2)

    D. For subarrays with one-axis tracking and self-shading enabled, calculate and apply the self-shading loss factors to the nominal POA beam and diffuse irradiance. (Section 7.3)

    E. Apply user-specified monthly soiling factors to calculate the effective POA irradiance on the subarray. (Section 7.5)

2. If there is a single subarray (Subarray 1) with no tracking (fixed) and self-shading is enabled, calculate the reduced diffuse POA irradiance and self-shading DC loss factor. (Section 8)

3. Determine subarray string voltage calculation method (Section 10.1).

4. For each of up to four subarrays, run the module model with the effective beam and diffuse POA irradiance and module parameters as input to calculate the DC output power, module efficiency, DC voltage, and cell temperature of a single module in the subarray.

5. Calculate the subarray string voltage using the method determined in Step 3.

6. Loop through the subarrays to calculate the array DC power (Section 10):

    A. For Subarray 1, apply the fixed self-shading DC loss to the module DC power if it applies.
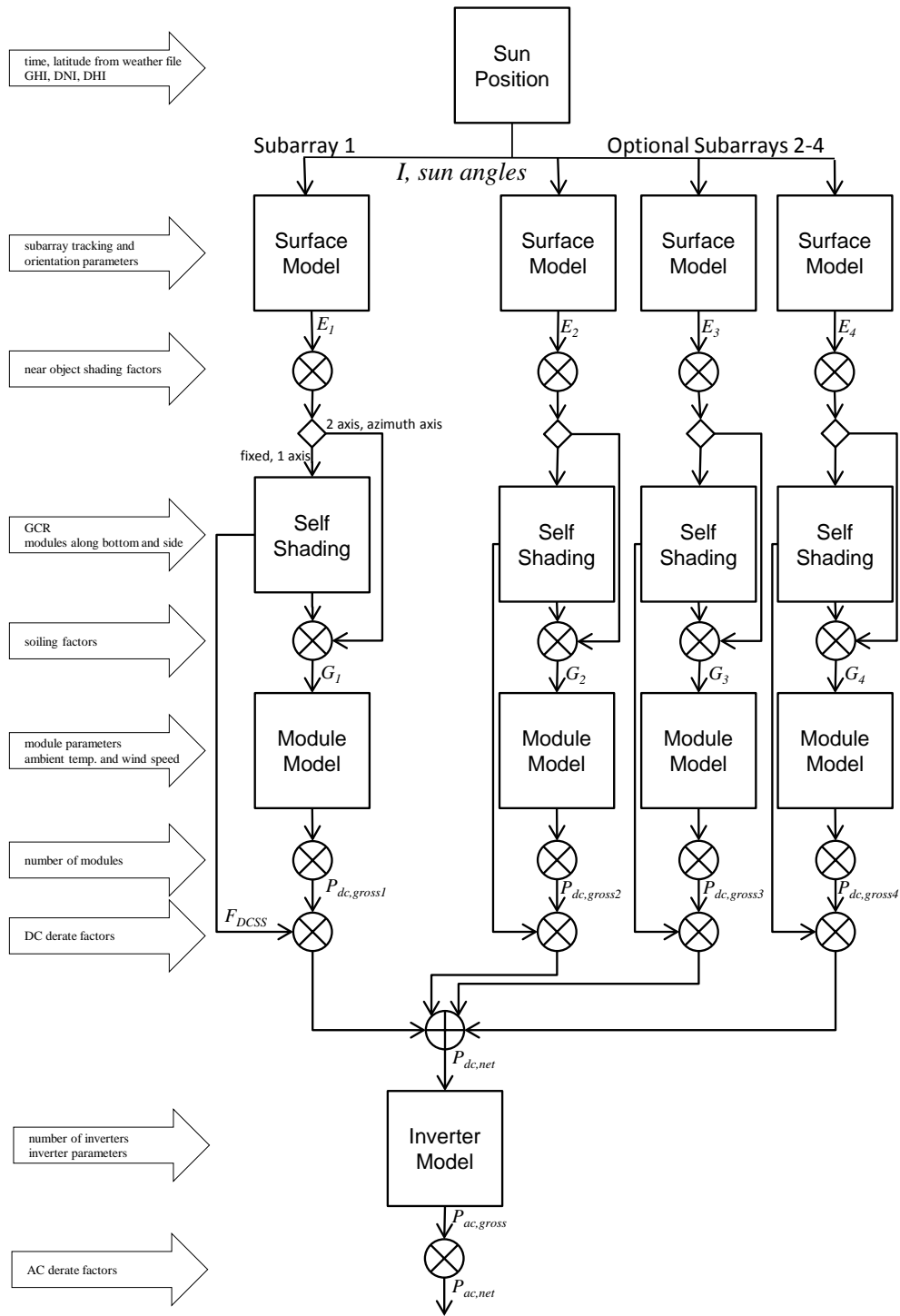
Figure 1. Photovoltaic Performance Model Simplified Block Diagram

B. For each subarray, calculate the subarray gross DC power by multiplying the module DC power by the number of modules in the subarray.

C. For each subarray, calculate subarray net DC power by multiplying the gross subarray power by the DC loss.

D. For each subarray, calculate the subarray string voltage by multiplying the module voltage by the number of modules per string.

E. Calculate the array net and gross DC power by adding up the subarray values.

7. Run the inverter submodel to calculate the gross AC power and inverter conversion efficiency (Section 11).

8. Calculate the net AC power by applying the AC loss to the gross AC power. (Section 12.3)

2.2   Component Libraries and Weather Files

SAM includes a set of component libraries and weather files that store copies of data managed by different organizations. The component libraries store module and inverter parameters that represent the components' physical properties. The weather files store data representing the solar resource at different locations.

The following is a list of the databases used by SAM's photovoltaic performance model:

- California Energy Commission Eligible Photovoltaic Modules (Go Solar California 2014b)
- California Energy Commission Eligible Inverters (Go Solar California 2014c)
- Sandia National Laboratories Module Database (Sandia 2014)
- NREL National Solar Radiation Database 1961-1990 (TMY2) (NSRDB 2014)
- NREL National Solar Radiation Database 1991-2010 Update (TMY3) (NSRDB 2014)
- U.S. DOE EnergyPlus Weather Data (EnergyPlus Weather 2014)
- NREL Solar Prospector (Solar Prospector 2015)

Note that these data collections are not part of SSC. If you are developing a model using the SAM software development kit and want to use the component libraries, you will have to write your own code to read the data. (You can use SAM's library editor to export data from the component libraries to a text file as described in SAM Help - Libraries (2014).)

2.3   System Sizing

SAM's System Design input page has an option that automatically calculates the number of modules per string, number of parallel strings in the array, and number of inverters given a desired array size in DC kW and a DC-to-AC nameplate capacity ratio. The sizing algorithm is described in SAM Help - Sizing the Flate Plate PV System (2014) and uses the module and inverter reference parameters for the capacity calculations.

The system sizing algorithm is not available in SSC.

3   Irradiance and Weather Data

The photovoltaic performance model requires a weather file with data describing the solar radiation and weather observations at the system location. It uses location information and hourly time stamps to calculate hourly sun position angles (Section 4), and irradiance data to calculate hourly plane-of-array (POA) irradiance values (Section 6). The cell temperature submodel (Section 9.2) uses ambient temperature and wind speed data to calculate the hourly photovoltaic cell temperature values.

SAM can read files in the formats listed below. For general descriptions of the formats, and references to more detailed documentation see (SAM Help - Weather File Formats 2014).

- SAM CSV: Comma-separated text format developed by NREL for use with SAM.

- TMY3: Comma-separated text format developed by NREL for the National Solar Radiation Database 1991-2010 update.

- TMY2: Text format developed by NREL for the National Solar Radiation Database 1961 - 1990 data set.

- EPW: Text format derived from the TMY2 format for the EnergyPlus building modeling software.

SAM reads and stores the weather file data shown in Table 2. It ignores any additional data elements that may be included in the different weather file formats. For example, it ignores the source and uncertainty flags in the TMY3 file format.

3.1   Time Convention and Resolution

The weather file contains one year of data. The number of data rows in the file determines the simulation time step. For example, a file with 8,760 rows of weather data (not including header rows) would result in an hourly simulation. A weather file for a 15-minute simulation time step should have 35,040 rows in addition to the header rows.

The first row of data in the weather file is for the first time step after midnight on January 1, local standard time. For hourly data, the first row is for the hour ending at 1 a.m. on January 1. For 15-minute data, the first row is for the quarter hour ending at 1:15 a.m. on January 1.

By default, SAM calculates the sun position for the mid-point of the time step. For example, for hourly data, SAM calculates the sun position for each hour at 30 minutes after the hour so that the sun position for the 10:00 a.m. hour would be its position at 10:30 a.m. However, if the weather file includes a Minute column, SAM will instead use the minute indicated to calculate each time step's solar position.

3.2   Irradiance and Insolation

Solar irradiance is a measure of the instantaneous power from the sun on a surface over an area, typically given in the SI units of watts per square meter (W/m^2). In the weather files from the National Solar Radiation Database (NSRDB 2014), each irradiance value is the total solar radiation in the 60 minutes ending at a given time step. These values represent the average solar power over a given hour in watt-hours per hour per square meter (Wh/h·m^2). In SAM these values are expressed in the mathematically equivalent W/m^2 (or converted to kW/m^2).

The weather file stores hourly values for the three components of solar irradiance:

- The total solar irradiance on a surface parallel to the ground (horizontal), called global horizontal irradiance.

- The portion of the solar irradiance that reaches a surface normal to the sun in a direct line from the solar disk (typically assuming a measurement device with a 5° field of view), called beam normal or direct normal irradiance.

9

Table 2. Weather Data

Symbol	Field	Description
	Metadata Location Description	
-	Location ID*	numerical identifier [*722780*]
-	City*	location name [*"Phoenix Sky Harbor Intl Ap"*]
-	State*	two-letter state abbreviation [*AZ*]
tz	Time zone	hours W of GMT [*-7.0*]
lat	Latitude	decimal degree N of 0 [*33.450*]
lon	Longitude	decimal degree W of 0 [*-11.983*]
elv	Elevation	meters above sea level [*337*]
	Hourly Data Records	
yr	Year	typical year [*1988*]
mo	Month	typical month (1-12)
day	Day	day of year (1-365)
hr	Hour	hour of day (0-23) in local time
min	Minute	minute past the hour (0-59)
E_g	Global horizontal irradiance (Wh/m^2)	total radiation on a horizontal surface
E_b	Direct normal irradiance (W/m^2)	direct radiation on a surface normal to the sun
E_d	Diffuse horizontal irradiance (Wh/m^2)	radiation on a horizontal surface from the sky
v_{wind}	Wind speed (m/s)	wind speed
-	Wind direction* (°E of N)	wind direction
T_a	Dry bulb temperature (°C)	ambient dry-bulb temperature
-	Wet bulb temp* (°C)	ambient wet-bulb temperature
-	Dew point temp* (°C)	ambient dew-point temperature
-	Relative humidity* (%)	relative humidity
-	Pressure* (mbar)	atmospheric pressure
D_{snow}	Snow depth (cm)	depth of snow
ρ	Ground reflectance	ground reflectance factor or albedo

Table Notes

The photovoltaic model does not use fields marked (*), but they are required by the weather file reader. The italicized values in brackets are examples from a TMY3 file's header.

- The solar irradiance on a horizontal surface from the sky excluding the solar disc, or diffuse horizontal irradiance.

Insolation is a measure of the energy from the sun over a given time on a surface given in watt-hours per square meter (Wh/m^2). SAM calculates the monthly and annual insolation incident on the plane of the photovoltaic array (Section 6).

3.3 Weather Observations

SAM uses ambient temperature and wind speed data from the weather file to estimate the effect of photovoltaic cell temperature on the array's performance. Although wind direction does have an effect on cell temperature, the photovoltaic performance model assumes that its cumulative effect over the year is negligible (King 2004).

The air temperature or dry-bulb temperature is the temperature in degrees Celsius (°C) of ambient air measured by a thermometer exposed to the air but shielded from the sun and rain.

SAM assumes that the hourly wind speed measurements from the weather file are in m/s and measured at a height of 10 m above the ground.

4 Sun Position

SAM calculates the sun altitude, zenith, and declination angles that define its position for each time step in the weather file. SAM's sun position algorithm is based on the method described in Michalsky (1988), which in turn is based on the Astronomical Almanac's algorithm for the period 1950-2050. NREL modified the Michalsky algorithm to calculate sun azimuth angles for locations south of the equator using the approach described in (Iqbal 1983). The algorithm is further discussed in O'Brien (2012).

The overall sun position algorithm steps are:

1. Calculate the effective time in hours for the current time step.

2. Calculate the sun angle for the current hour.

3. Determine the current day's sunrise and sunset time.

4. Determine the sunup flag for the current hour.

5. Calculate the extraterrestrial radiation for the current hour.

Although all of the sun angle input and results variables in SAM are in degrees, most of the internal equations use angle values in radians.

Table 3 lists the sun position algorithm's inputs and outputs. Figure 2, adapted from Dunlap (2007), shows the sun angles. The inputs are for each time step, and come from the weather file.

The sun position algorithm does not include an air mass equation. Instead, each submodel or algorithm that needs it uses its own equation to calculate the value: The Perez incident sky diffuse model uses Equation 6.16, and the Sandia and CEC module models use the same equation, shown in Equations 9.1 and 9.22, respectively.

4.1 Effective Time

The first step in the sun position algorithm is to determine the effective time of the current time step. SAM reads the time stamp data for the current time step in the weather file (represented by a data row) as a year, month, day, hour, and minute value.

The standard weather files from the National Solar Radiation Database (NSRDB 2014) start at $hr = 1$. The algorithm converts the first hour number $hr = 0$:

$$hr = hr - 1 \tag{4.1}$$

The algorithm uses the midpoint of the time step for sun position calculations (except for the hours containing sunrise and sunset, see Section 4.3), and assumes an hourly time step, so it replaces the minute value from the weather file:

$$min = 30 \tag{4.2}$$

The Julian day of year *jdoy* is the number of days since Noon on January 1 of the current year (strictly speaking, it should be called the *ordinal date*). Because some weather files are typical year files, the time stamps in a given file may not all use the same year. For example, all January time stamps may be 1988, while all February time stamps may be 2005 (NSRDB 2014a).

To account for leap years:

$$k = \begin{cases} 1 & \text{if } year \mod 4 = 0 \\ 0 & \text{if } year \mod 4 \neq 0 \end{cases} \tag{4.3}$$

Note that this accounts for leap years to correctly calculate effective time, but is separate from the energy simulation, which does not account for leap years.

Table 3. Sun Position Variable Definitions

Symbol	Description / **Name in SAM**	Name in SSC
	Inputs	
tz	time zone (hrs W of GMT)	`tz`
lat	latitude (decimal °N of equator)	`lat`
lon	longitude (decimal °W of GMT)	`lon`
yr	year (e.g., 1988)	`year`
mo	month year (1-12)	`month`
day	day of year (1-365)	`day`
hr	hour of day local time (0-23)	`hour`
min	minute of hour (0-59)	`minute`
	Outputs	
Z	**Solar zenith angle (deg)**	`sun_zen`
α	**Solar altitude angle (deg)**	`sun_elv`
δ	solar declination angle	`sun_dec`
γ	**Solar azimuth angle (deg)**	`sun_azm`
sunup	**Sun up over horizon (0/1)**	-
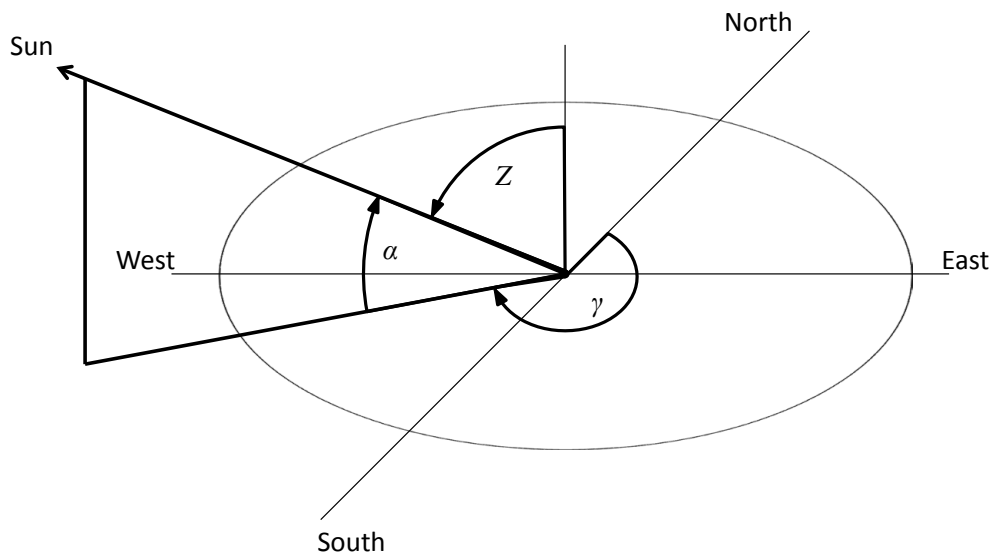
Figure 2. Sun Angles

SAM calculates the number of days since January 1 a from the number of days in each of the months (January = 31, February = 28, March = 31, etc.) before the current month, and the number of days since the first of the current month.

The Julian day of year is then:

$$jdoy = \begin{array}{ll} day + a & \text{for January and February} \\ day + a + k & \text{for March through December} \end{array} \tag{4.4}$$

The current decimal time of day expressed as an offset from UTC depends on the hour and minute of the current time stamp, and the location's time zone:

$$t_{utc} = hr + \frac{min}{60} - tz \tag{4.5}$$

For some combinations of time stamp and time zone, Equation 4.5 may yield a value less than zero or greater than 24 hours, in which case the following correction applies:

$$\begin{array}{ll} t_{utc} = t_{utc} + 24, jdoy = jdoy - 1 & \text{if } t_{utc} < 0 \\ t_{utc} = t_{utc} - 24, jdoy = jdoy + 1 & \text{if } t_{utc} > 24 \end{array} \tag{4.6}$$

The Julian date *julian* of the current hour is the Julian day of the preceding noon plus the number of hours since then. The Julian day is defined as the number of days since Noon on January 1, 2000:

$$julian = 32916.5 + 365(yr - 1949) + \frac{yr - 1949}{4} + jdoy + \frac{t_{utc}}{24} - 51545 \tag{4.7}$$

4.2 Sun Angles

The sun angle equations are from Michalsky (1988). The sun angles (Figure 2) are the altitude angle α, declination angle δ, and zenith angle Z. SAM also calculates the sun azimuth angle γ for use in the incident irradiance calculations. The solar declination angle is not used in the incident irradiance calculations, but is required to calculate the sun azimuth angle. The bold font in Table 3 indicates that SAM reports the sun zenith, altitude, and azimuth angles in the hourly results.

The first step in the sun angle calculation for a given time step is to determine the ecliptic coordinates of the location, which define the photovoltaic array's position on the earth relative to the sun. The ecliptic coordinate variables are the mean longitude, mean anomaly, ecliptic longitude, and obliquity of the ecliptic. The algorithm uses ecliptic coordinates instead of equatorial coordinates to include the effect of the earth's inclination in the sun angle calculations.

Where limits are indicated for the equations below, if the variable's value falls outside of the limits, SAM adjusts the value. For example, for a value x with the limits $0 \leq x < 360°$, SAM divides x by $360°$, and checks to see whether the remainder is less than zero, and if it is, adds $360°$ to the remainder:

$$a = x - 360° \text{ trunc}\left(\frac{x}{360°}\right)$$
$$x = \begin{array}{ll} a & \text{if } a \geq 0 \\ a + 360° & \text{if } a < 0 \end{array} \tag{4.8}$$

Mean longitude in degrees ($0 \leq mnlong < 360°$). Note that the mean longitude is the only value not converted to radians:

$$mnlong = 280.46 + 0.9856474 \, julian \tag{4.9}$$

Mean anomaly in radians ($0 \leq mnanom < 2\pi$):

$$mnanom = \frac{\pi}{180}(357.528 + 0.9856003 \, julian) \tag{4.10}$$

14

Ecliptic longitude in radians ($0 \le eclong < 2\pi$):

$$eclong = \frac{\pi}{180}\left[mnlong + 1.915\sin mnanom + 0.02\sin(2mnanom)\right] \tag{4.11}$$

Obliquity of ecliptic in radians:

$$obleq = \frac{\pi}{180}\left(23.439 - 0.0000004\,julian\right) \tag{4.12}$$

The next step is to calculate the celestial coordinates, which are the right ascension and declination.

The right ascension in radians:

$$ra = \begin{cases} \arctan\left(\frac{\cos obleq\sin eclong}{\cos eclong}\right) + \pi & \text{if } \cos eclong < 0 \\ \arctan\left(\frac{\cos obleq\sin eclong}{\cos eclong}\right) + 2\pi & \text{if } \cos obleq\sin eclong < 0 \end{cases} \tag{4.13}$$

The solar declination angle in radians:

$$\delta = \arcsin\left(\sin obleq\sin eclong\right) \tag{4.14}$$

Next are the local coordinates, which require calculating the hour angle.

The Greenwich mean siderial time in hours ($0 \le gmst < 24$) with limits applied as shown in Equation 4.8 depends on the current time at Greenwich t_{utc} from Equation 4.5, and the Julian day from Equation 4.7:

$$gmst = 6.697375 + 0.0657098242\,julian + t_{utc} \tag{4.15}$$

Local mean siderial time in hours ($0 \le lmst < 24$):

$$lmst = gmst + \frac{lon}{15} \tag{4.16}$$

The hour angle in radians ($-\pi < HA < \pi$):

$$b = 15\frac{\pi}{180}lmst - ra$$
$$HA = \begin{cases} b + 2\pi & \text{if } b < -\pi \\ b - 2\pi & \text{if } b > \pi \end{cases} \tag{4.17}$$

The sun altitude angle in radians, not corrected for refraction:

$$a = \sin\delta\sin\left(\frac{\pi}{180}lat\right) + \cos\delta\cos\left(\frac{\pi}{180}lat\right)\cos HA$$
$$\alpha_0 = \begin{cases} \arcsin a & \text{if } -1 \le a \le 1 \\ \frac{\pi}{2} & \text{if } a > 1 \\ -\frac{\pi}{2} & \text{if } a < 1 \end{cases} \tag{4.18}$$

The sun altitude angle α corrected for refraction is:

$$\alpha_{0d} = \frac{180}{\pi}\alpha_0$$
$$r = \begin{cases} \alpha_{0d} + 3.51561\left(\frac{0.1594 + 0.0196\alpha_{0d} + 0.00002\alpha_{0d}^2}{1 + 0.505\alpha_{0d} + 0.0845\alpha_{0d}^2}\right) & \text{if } \alpha_{0d} > -0.56 \\ 0.56 & \text{if } \alpha_{0d} \le -0.56 \end{cases}$$
$$\alpha = \begin{cases} \frac{\pi}{2} & \text{if } \alpha_{0d} + r > 90 \\ \frac{\pi}{180}(\alpha_{0d} + r) & \text{if } \alpha_{0d} + r \le 90 \end{cases} \tag{4.19}$$

The sun azimuth angle γ in radians is from (Iqbal 1983) rather than (Michalsky 1988) because the latter is only for northern hemisphere locations:

$$a = \frac{\sin \alpha_0 \sin \left(\frac{\pi}{180} lat \right) - \sin \delta}{\cos \alpha_0 \cos \left(\frac{\pi}{180} lat \right)}$$

$$b = \begin{cases} \arccos a & \text{if } -1 \leq a \leq 1 \\ \pi & \text{if } \cos \alpha_0 = 0, \text{ or if } a < -1 \\ 0 & \text{if } a > 1 \end{cases}$$

$$\gamma = \begin{cases} b & \text{if } HA < -\pi \\ \pi - b & \text{if } -\pi \leq HA \leq 0, \text{ or if } HA \geq \pi \\ \pi + b & \text{if } 0 < HA < \pi \end{cases} \tag{4.20}$$

The sun zenith angle Z in radians:

$$Z = \frac{\pi}{2} - \alpha \tag{4.21}$$

4.3   Sunrise and Sunset Hours

For the hours of each day that contain the time of sunrise and sunset, it is not appropriate to use the midpoint of the hour (see Equation 4.2) to calculate that hour's solar position because the midpoint may be before sunrise or after sunset. For the sunrise hour, the solar position angle is for the minute at the midpoint between the minute of sunrise and the end of the hour. For the sunset hour, the angle is for the midpoint between the beginning of the hour and sunset.

To determine whether the current time stamp is for an hour that contains a sunrise, or is a nighttime or daytime hour, the sunrise hour angle in radians is:

$$a = -\tan lat \tan \delta$$

$$HAR = \begin{cases} 0 & \text{if } a \geq 1 & \text{sun is down} \\ \pi & \text{if } a \leq -1 & \text{sun is up} \\ \arccos a & \text{if } -1 < a < 1 & \text{sunrise hour} \end{cases} \tag{4.22}$$

The equation of time in hours:

$$a = \frac{1}{15} \, mnlong - \frac{\pi}{180} ra$$

$$EOT = \begin{cases} a & \text{if } -0.33 \leq a \leq 0.33 \\ a + 24 & \text{if } a < -0.33 \\ a - 24 & \text{if } a > 0.33 \end{cases} \tag{4.23}$$

The sunrise time in local standard decimal time:

$$t_{sunrise} = 12 - \frac{1}{15} \frac{180}{\pi} HAR - \frac{\lambda}{15} - tz - EOT \tag{4.24}$$

And, the sunset in local standard time:

$$t_{sunrise} = 12 + \frac{1}{15} \frac{180}{\pi} HAR - \frac{\lambda}{15} - tz - EOT \tag{4.25}$$

The solar position minute for the sunrise hour is at the midpoint between the minute during which the sun rose and the end of the current hour. Assuming a 60 minute time step and that that the sunup time is in decimal minutes:

$$min_{sunrise} = 60 \left[1 - (t_{sunrise} - hr) + \frac{(t_{sunrise} - hr)}{2} \right]$$ (4.26)

Similarly, the solar position minute for the sunset hour is at the midpoint between the beginning of the current hour and the minute during which the sun set:

$$min_{sunset} = \frac{60(t_{sunset} - hr)}{2}$$ (4.27)

4.4 Sunup Flag

The sunup flag indicates whether the sun is above or below the horizon in the current hour. It is determined from the sunrise hour angle (Equation 4.22):

- If the sun is down in the current hour, $sunup = 0$.

- If the sun is up in the current hour, $sunup = 1$.

- If a sunrise occurs in the current hour, $sunup = 2$.

- If a sunset occurs in the current hour, $sunup = 3$.

4.5 Extraterrestrial Radiation

Extra terrestrial radiation H is the solar radiation at the top of the earth's atmosphere in W/m^2. SAM uses the value in the incident irradiance calculations described in Section 6.

The extraterrestrial radiation equation is from Chapter 1.10 of Duffie and Beckman (2013):

$$G_{on} = 1367 \left[1 + 0.033 \cos \left(\frac{\pi}{180} \frac{360doy}{365} \right) \right]$$

$$H = \begin{cases} G_{on} \cos Z & \text{if } 0 < Z < \frac{\pi}{2} \text{ (sun is up)} \\ G_{on} & \text{if } Z = 0 \\ 0 & \text{if } Z < 0, \text{ or if } Z > \frac{\pi}{2} \end{cases}$$ (4.28)

4.6 True Solar Time and Eccentricity Correction Factor

The sun position algorithm calculates two values that are not used by the photovoltaic performance model.

True solar or apparent time in decimal hours:

$$t_{truesolar} = hr + \frac{min}{60} + \frac{lon}{15} - tz + EOT$$ (4.29)

The eccentricity correction factor:

$$E_0 = [1.00014 - 0.01671 \cos mnanom - 0.00014 \cos(2mnanom)]^{-2}$$ (4.30)

5 Surface Angles

SAM considers each subarray in the system to be a flat surface with one tilt angle β_s and one azimuth angle γ_s that define the surface orientation. The surface angles depend on whether the subarray is fixed, or mounted on one-axis, two-axis, or azimuth-axis trackers. Surfaces with one-axis trackers have a third surface angle σ defining its rotation around the tracking axis.

The surface angle equations are based on standard geometric relationships defined by the surface orientation and sun angles (Section 4.2).

SAM calculates each subarray's surface angles for each of the 8,760 hours of the simulation. For systems with trackers, the surface angles in a given hour are fixed over the hour. Because the sun position angles are at the midpoint of the hour (except for the hours containing the sunrise and sunset (Section 4.3), the surface angles are also at the midpoint of the hour.

Table 4 defines the variables used for the equations in this section. Figure 3, adapted from Dunlap (2007), shows how the surface angles are defined.

5.1 Angle of Incidence

The angle of incidence AOI is the sun incidence angle defined as the angle between beam irradiance and a line normal to the subarray surface (See Figure 3). It is a function of the sun azimuth angle γ, sun zenith angle Z, surface azimuth angle γ_s, and the surface tilt angle β_s:

$$a = \sin Z \cos(\gamma - \gamma_s) \sin \beta_s + \cos Z \cos \beta_s$$

$$AOI = \begin{cases} \pi & \text{if } a < -1 \\ 0 & \text{if } a > 1, \text{ or for two-axis tracking} \\ \arccos a & \text{if } -1 \leq a \leq 1 \end{cases} \tag{5.1}$$

5.2 Fixed, Azimuth and Two-axis Tracking

The surface azimuth and tilt angle values depend on the tracking option as described below.

For a fixed surface (no tracking):

$$\beta_s = \beta_0$$
$$\gamma_s = \gamma_0 \tag{5.2}$$

For azimuth tracking, the surface tilt angle is fixed, and the surface azimuth angle follows the sun azimuth angle:

$$\beta_s = \beta_0$$
$$\gamma_s = \gamma \tag{5.3}$$

For two-axis tracking, the surface tilt and azimuth angles follow the sun zenith and azimuth angles, respectively:

$$\beta_s = Z$$
$$\gamma_s = \gamma \tag{5.4}$$

Table 4. Surface Angle Variable Definitions

Symbol	Description / **Name in SAM**	Name in SSC
	Inputs	
-	**Fixed**, **1 Axis**, **2 Axis**, **Azimuth Axis**	`track_mode`
β_0	**Tilt (deg)**	`tilt`
γ_0	**Azimuth (deg)**	`azimuth`
θ_{lim}	**Tracker rotation limit (deg)**	`rotlim`
Z	sun zenith angle	`sun_zen`
γ	sun azimuth angle	`sun_azm`
-	**Backtracking**	`backtrack`
GCR	**Ground coverage ratio (GCR)**	`GCR`
-	**Beam and diffuse**, **Total and beam**	`irrad_mode`
-	**Isotropic**, **HDKR**, **Perez**	`sky_model`
	Outputs	
AOI	incidence angle	`incidence`
β_s	**Subarray [n] Surface tilt (deg)**	`surf_tilt`
γ_s	**Subarray [n] Surface azimuth (deg)**	`surf_azm`
θ	**Subarray [n] Axis rotation for 1 axis trackers (deg)**	`axis_rotation`
θ_0	**Subarray [n] Ideal axis rotation for 1 axis trackers (deg)**	-
$\Delta\theta$	backtracking difference from ideal rotation	`bt_diff`
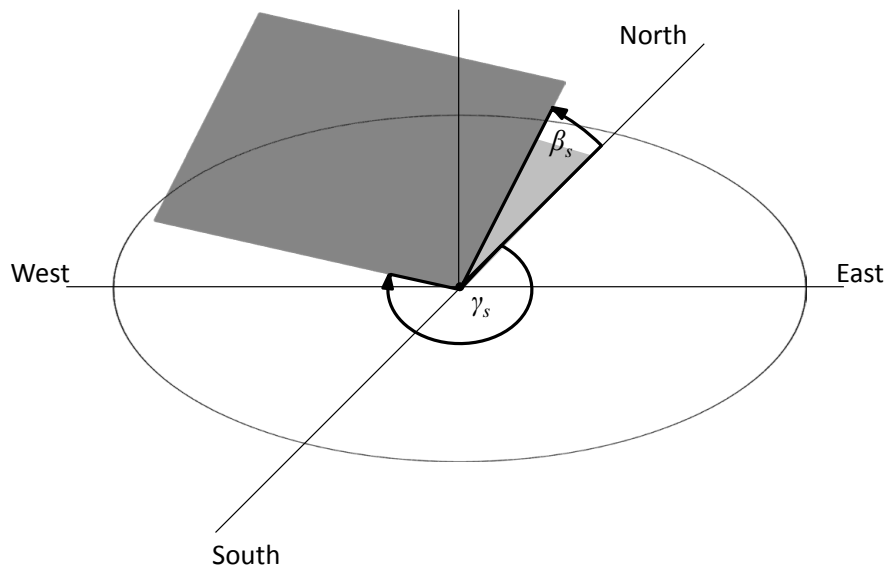
Figure 3. Surface Angles

5.3 One-axis Tracking

One-axis tracking involves a third surface angle θ describing the surface's rotation about the tracking axis and a rotation angle limit θ_{lim}.

SAM offers three shade modes for the one-axis tracking option on the PV Subarrays input page:

- **Self-shaded** accounts for shading of modules in one row by those in neighboring rows (see Section 7.3), but does not model backtracking.

- **Backtracking** adjusts the tracking angle to avoid shading of modules in one row by those in neighboring rows (see Section 5.3.2).

- **None** does not account for self-shading or backtracking.

All one-axis surface angle equations use angle values converted from degrees to radians.

The one-axis tracking tilt angle in radians (See Sections 5.3.1 and 5.3.2 for rotation angle equations):

$$a = \cos\beta_0 \cos\theta$$

$$\beta_s = \begin{cases} \pi & \text{if } a < -1 \\ 0 & \text{if } a > 1 \\ \arccos a & \text{if } -1 \leq a \leq 1 \end{cases} \tag{5.5}$$

The one-axis tracking surface azimuth angle in radians:

$$a = \frac{\sin\theta}{\sin\beta}$$

$$\gamma_s = \begin{cases} \pi & \text{if } \beta_s = 0 \\ \frac{3\pi}{2} + \gamma_0 & \text{if } a < -1 \\ \frac{\pi}{2} + \gamma_0 & \text{if } a > 1 \\ \gamma_0 - \pi - \arcsin a & \text{if } \theta < -\frac{\pi}{2} \\ \gamma_0 + \pi - \arcsin a & \text{if } \theta > \frac{\pi}{2} \\ \gamma_0 + \arcsin a & \text{if } -1 \leq a \leq 1 \text{ or } -\frac{\pi}{2} \leq \theta \leq \frac{\pi}{2} \end{cases} \tag{5.6}$$

The surface azimuth angle in radians must be between zero and 2π:

$$\gamma_s = \begin{matrix} \gamma_s - 2\pi & \text{if } \gamma_s > 2\pi \\ \gamma_s + 2\pi & \text{if } \gamma_s < 0 \end{matrix} \tag{5.7}$$

5.3.1 Rotation Angle for One-axis Trackers

For a subarray with one-axis tracking, SAM assumes that the tracker rotates about an axis tilted from the horizontal at the surface tilt angle along the line that defines the surface azimuth angle shown in Figure 3.

The **tracker rotation angle** θ is the angle of the subarray surface from the horizontal about the tracking axis. For a surface tilted as shown Figure 3, the rotation angle is the angle as viewed from the raised end of the surface, with a negative rotation angle indicating counter-clockwise rotation from the horizontal. For a surface in the northern hemisphere with a surface azimuth angle of $180°$, a negative rotation angle is for a surface facing east toward the morning sun with $-90°$ for a vertical, east-facing surface. A positive rotation angle is for an array facing the afternoon sun, with $90°$ for a vertical, west-facing surface. In the southern hemisphere, for a surface with an azimuth angle of $0°$, a negative rotation limit indicates a west-facing surface.

The **tracker rotation limit** θ_{lim} is the physical limit of the tracker's motion about the tracking axis in degrees with valid values between 0 and $\pm90°$. (Note that SAM's user interface restricts the rotation limit values to $\pm85°$.) For

example, a tracker in the northern hemisphere with a rotation limit of $45°$ would start tracking at a rotation angle of $-45°$ in the sunrise hour, and stop tracking at a rotation angle of $45°$ in the sunset hour.

SAM adjusts the surface rotation angle as follows:

- If the tracker rotation angle exceeds the user-specified tracker rotation limit, then the tracker rotation is set to the limit.

- If backtracking is enabled and the array is self-shaded, then the surface tilt angle is adjusted to minimize self-shading (See Section 5.3.2).

The one-axis tracking equations described below use angle values converted from degrees to radians. SAM displays surface angle values in the results in degrees.

The "ideal rotation angle" θ_0 for one-axis tracking is the rotation angle without application of the rotation limit θ_{lim} or backtracking:

$$a = \frac{\sin Z \sin(\gamma - \gamma_s)}{\sin Z \cos(\gamma - \gamma_s) \sin \beta + \cos Z \cos \beta}$$

$$\theta_{01} = \begin{cases} -\frac{\pi}{2} & \text{if } a < -99,999.9 \\ \frac{\pi}{2} & \text{if } a > 99,999.9 \\ \arctan a & \text{if } -99,999.9 \leq a \leq 99,999.9 \end{cases} \tag{5.8}$$

The following corrections ensure that the ideal rotation angle is in the correct quadrant (II or III) when the surface azimuth angle γ_s is less than or greater than π:

$$\theta_0 = \begin{cases} \theta_{01} + \pi & \text{if } \gamma_s < \pi \text{ and } \gamma < \gamma_s \leq \gamma_s + \pi \text{ and } \theta_0 < 0 \text{ (quadrant II positive rotation)} \\ \theta_{01} - \pi & \text{if } \gamma_s < \pi \text{ and } \gamma < \gamma_s \leq \gamma_s + \pi \text{ and } \theta_0 > 0 \text{ (quadrant III negative rotation)} \\ \theta_{01} + \pi & \text{if } \gamma_s > \pi \text{ and } \gamma < \gamma_s \leq \gamma_s - \pi \text{ and } \theta_0 < 0 \text{ (quadrant II positive rotation)} \\ \theta_{01} - \pi & \text{if } \gamma_s > \pi \text{ and } \gamma < \gamma_s \leq \gamma_s - \pi \text{ and } \theta_0 > 0 \text{ (quadrant III negative rotation} \end{cases}$$

The rotation angle in radians is the ideal rotation angle limited by the rotation limit:

$$\theta = \begin{cases} \theta_0 & \text{if } -\theta_{\text{lim}} \leq \theta_0 \leq \theta_{\text{lim}} \\ -\theta_{\text{lim}} & \text{if } \theta_0 < -\theta_{\text{lim}} \\ \theta_{\text{lim}} & \text{if } \theta_0 > \theta_{\text{lim}} \end{cases} \tag{5.9}$$

5.3.2 Backtracking for One-axis Trackers

Backtracking is a technique used with some one-axis trackers to minimize self-shading of neighboring rows of photovoltaic modules during times that the sun is low in the sky. When neighboring rows shade each other, the tracker rotates toward the horizontal to reduce the size of shadows on the array.

SAM's backtracking for one-axis tracking algorithm was developed by NREL for SAM. It involves the following steps:

1. Run the self-shading algorithm (Section 7.3). It determines whether a portion of the subarray is self-shaded, given the ground coverage ratio GCR, sun angles (Section 4.2), surface angles (Section 5), and ideal surface rotation angle θ_0 (Equation 5.9).

2. If a portion of the subarray is self-shaded, adjust the tracker rotation angle θ toward $0°$ by $1°$.

3. Repeat Steps 1-2 until $F_{\text{shad1x}} = 0$, or one hundred times, whichever comes first (to avoid an infinite loop).

The backtracking rotation difference $\Delta\theta$ reported by SSC as `bt_diff`:

$$\Delta\theta = \theta - \theta_0 \tag{5.10}$$

6 Incident Irradiance

The incident irradiance, also called plane-of-array irradiance or POA irradiance, is the solar irradiance incident on the plane of the photovoltaic array in a given time step. SAM calculates the incident irradiance for the sunrise hour, sunup hours, and sunset hour. The incident angle algorithm calculates the hourly beam and diffuse irradiance incident on the photovoltaic subarray surface for a given sun position, latitude, and surface orientation. For each time step in the simulation, the incident irradiance algorithm steps are:

1. Calculate the angle of incidence (Section 5.1).

2. Calculate the beam irradiance on a horizontal surface.

3. Check to see if the beam irradiance on a horizontal surface exceeds the extraterrestrial radiation.

4. Calculate the incident beam irradiance.

5. Calculate the sky diffuse horizontal irradiance using one of the three sky diffuse irradiance methods.

6. Calculate the ground-reflected irradiance.

6.1 Incident Beam Irradiance

The incident beam irradiance is solar energy that reaches the surface in a straight line from the sun:

$$I_b = E_b \cos AOI \tag{6.1}$$

The beam irradiance on a horizontal surface:

$$I_{bh} = E_b \cos Z \tag{6.2}$$

SAM compares I_{bh} to the extraterrestrial radiation H (Equation 4.28). If $I_{bh} > H$, it generates an error flag that causes the calculations to stop and SAM to display the error message "failed to compute irradiation on surface."

6.2 Incident Sky Diffuse Irradiance

Incident sky diffuse irradiance I_d is solar energy that has been scattered by molecules and particles in the earth's atmosphere before reaching the surface of the subarray.

The sky diffuse model calculates values of the components of the sky diffuse irradiance (isotropic, circumsolar, and horizon brightening), but those components are not used by the photovoltaic model. The equations are provided in the sections below for reference.

The **Tilted Surface Radiation Model** option on the Location and Resource input page in SAM (sky_model in SSC) allows you to choose from three different sky diffuse irradiance models:

- The **Isotropic** model is the simplest of the three models, and assumes that diffuse radiation is uniformly distributed across the sky, called isotropic diffuse radiation (sky_model = 0).

- **HDKR**, named for the algorithm developed by Hay, Davies, Klucher and Reindl in 1979, also assumes isotropic diffuse radiation, but accounts for the higher intensity of circumsolar diffuse radiation, which is the diffuse radiation in the area around the sun (sky_model = 1).

- The **Perez** model uses a more complex computational method than the other two methods that accounts for both isotropic and circumsolar diffuse radiation, as well as horizon brightening (sky_model = 2).

Table 5. Incident Irradiance Variable Definitions

Symbol	Description / **Name in SAM**	Name in SSC
	Inputs	
E_b	**Beam irradiance**, beam normal from weather file	`beam`
E_d	**Diffuse irradiance**, diffuse horiz. from weather file, or calculated	`diffuse`
E_g	**Global horizontal irradiance**, global horizontal from weather file	`global`
H	extraterrestrial irradiance	-
ρ	**Albedo** (ground reflectance)	`albedo`
AOI	angle of incidence	`incidence`
β_s	**Subarray [n] Surface tilt**	`tilt`
Z	**Solar zenith angle**	`sun_zen`
-	**Beam and diffuse**, **Total and beam**	`irrad_mode`
-	**Isotropic**, **HDKR**, **Perez**	`sky_model`
-	**Use albedo in weather file if it is specified**	`use_wf_albedo`
	Outputs	
I_b	incident beam irradiance	`poa_beam`
I_d	incident sky diffuse irradiance	`poa_skydiff`
I_r	incident ground-reflected irradiance	`poa_gnddiff`
D_i	isotropic component of incident diffuse irradiance	`poa_skydiff_iso`
D_c	circumsolar component of incident diffuse irradiance	`poa_skydiff_cir`
D_h	horizon brightening component of incident diffuse irradiance	`poa_skydiff_hor`

The **Radiation Components** option on the Location and Resource input page (`irrad_mode` in SSC) allows you to choose whether to use the diffuse horizontal irradiance from the weather file, or whether to calculate it from the global horizontal and beam normal irradiance values:

$$E_d = \begin{array}{ll} E_d & \text{Use weather file value if option is \textbf{Beam and diffuse} (\texttt{irrad\_mode} = 0)} \\ E_g - E_b \cos Z & \text{Calculate for \textbf{Total and beam} (\texttt{irrad\_mode} = 1)} \end{array} \tag{6.3}$$

6.2.1 Isotropic Model

The isotropic diffuse sky irradiance (Liu 1963):

$$I_d = E_d \frac{1 + \cos \beta}{2} \tag{6.4}$$

The isotropic model does not account for circumsolar diffuse irradiance or horizon brightening:

$$D_i = I_d$$
$$D_c = 0$$
$$D_h = 0 \tag{6.5}$$

6.2.2 HDKR Model

The Hay, Davies, Klucher, Reindl (HDKR) sky diffuse model is from Chapter 2.16 of Duffie and Beckman (2013) and Reindl (1988).

The total irradiance incident on a horizontal surface:

$$I_{gh} = I_{bh} + E_d \tag{6.6}$$

The ratio of incident beam to horizontal beam:

$$R_b = \frac{\cos AOI}{\cos Z} \tag{6.7}$$

The anisotropy index for forward scattering circumsolar diffuse irradiance depends on the beam horizontal irradiance I_{bh} and extraterrestrial irradiance H (Equation 4.28):

$$A_i = \frac{I_{bh}}{H} \tag{6.8}$$

The modulating factor for horizontal brightening correction:

$$f = \sqrt{\frac{I_{bh}}{I_{gh}}} \tag{6.9}$$

The horizon brightening correction factor:

$$s = \sin^3 \frac{\beta_s}{2} \tag{6.10}$$

The circumsolar, isotropic, and horizon brightening components of the diffuse sky radiation are then:

$$cir = E_d A_i R_b$$
$$iso = E_d (1 - A_i) \frac{1 + \cos AOI}{2}$$
$$isohor = iso (1 + fs) \tag{6.11}$$

The isotropic, circumsolar, and horizon brightening components of the incident diffuse irradiance:

$$D_i = iso$$
$$D_c = cir$$
$$D_h = isohor - iso \tag{6.12}$$

The HDKR sky diffuse irradiance:

$$I_d = isohor + cir \tag{6.13}$$

6.2.3 Perez 1990 Model

SAM's Perez sky diffuse irradiance model was adapted from PVWatts Version 1 (Dobos 2013a) and is described in Perez (1988) and (Perez 1990). The SAM (and PVWatts) implementation includes a modification of the Perez model that treats diffuse radiation as isotropic for $87.5° \leq Z \leq 90°$. See also (Perez Sky Diffuse Model 2014) for a general description of the model.

The Perez model differs from the isotropic and HDKR models in that it uses the empirical coefficients in Table 6 derived from measurements over a range of sky conditions and locations instead of mathematical representations of the sky diffuse components.

Table 6. Perez Sky Diffuse Irradiance Model Coefficients

	f_{11}	f_{12}	f_{13}	f_{21}	f_{22}	f_{23}
$\varepsilon \leq 1.065$	−0.0083117	0.5877285	−0.0620636	−0.0596012	0.0721249	−0.0220216
$\varepsilon \leq 1.23$	0.1299457	0.6825954	−0.1513752	−0.0189325	0.065965	−0.0288748
$\varepsilon \leq 1.5$	0.3296958	0.4868735	−0.2210958	0.055414	−0.0639588	−0.0260542
$\varepsilon \leq 1.95$	0.5682053	0.1874525	−0.295129	0.1088631	−0.1519229	−0.0139754
$\varepsilon \leq 2.8$	0.873028	−0.3920403	−0.3616149	0.2255647	−0.4620442	0.0012448
$\varepsilon \leq 4.5$	1.1326077	−1.2367284	−0.4118494	0.2877813	−0.8230357	0.0558651
$\varepsilon \leq 6.2$	1.0601591	−1.5999137	−0.3589221	0.2642124	−1.127234	0.1310694
$\varepsilon > 6.2$	0.677747	−0.3272588	−0.2504286	0.1561313	−1.3765031	0.2506212

The parameters a and b describe the view of the sky from the perspective of the surface:

$$a = \max(0, \cos AOI)$$
$$b = \max(\cos 85°, \cos AOI) \tag{6.14}$$

The sky clearness ε with $\kappa = 5.534 \times 10^{-6}$ and the sun zenith angle Z in degrees:

$$\varepsilon = \frac{(E_d + E_b)/E_d + \kappa Z^3}{1 + \kappa Z^3} \tag{6.15}$$

The absolute optical air mass with the incidence angle b and sun zenith angle Z in degrees:

$$AM_0 = \left[\cos b + 0.15 \, (93.9° - Z)^{-1.253}\right]^{-1} \tag{6.16}$$

The sky clearness Δ assumes an extraterrestrial irradiance value of $1,367.0 \, \text{W/m}^2$:

$$\Delta = E_d \frac{AM_0}{1367} \tag{6.17}$$

The coefficients F_1 and F_2 are empirical functions of the sky clearness ε and describe circumsolar and horizon brightness, respectively. The sun zenith angle Z is in radians:

$$F_1 = \max\left[0, (f_{11}(\varepsilon) + \Delta f_{12}(\varepsilon) + Z f_{13}(\varepsilon))\right]$$
$$F_2 = f_{21}(\varepsilon) + \Delta f_{22}(\varepsilon) + Z f_{23}(\varepsilon) \tag{6.18}$$

SAM uses a lookup table with empirical values shown in Table 6 to determine the value of the f coefficients in Equation 6.18 for a given sky clearness coefficient ε from Equation 6.15.

The isotropic, circumsolar, and horizon brightening components of the sky diffuse irradiance:

$$D_i = E_d(1 - F_1)\frac{1+\cos \beta}{2} \quad \text{if } 0° \leq Z \leq 87.5° \qquad D_i = \frac{1+\cos \beta}{2} \quad \text{if } 87.5° < Z < 90° \text{ (isotropic only)}$$
$$D_c = E_d F_1 \frac{a}{b} \qquad\qquad\qquad\qquad\qquad\qquad\quad D_c = 0$$
$$D_h = E_d F_2 \sin \beta \qquad\qquad\qquad\qquad\qquad\qquad D_h = 0$$

The Perez incident diffuse irradiance:

$$I_d = D_i + D_c + D_h \tag{6.19}$$

6.3   Incident Ground-reflected Irradiance

The incident ground-reflected irradiance is solar energy that reaches the array surface after reflecting from the ground. The ground reflects light diffusely, so the ground-reflected irradiance is diffuse irradiance. It is a function of the beam normal irradiance and sun zenith angle, sky diffuse irradiance, and ground reflectance (albedo) (Liu 1963):

$$I_r = \rho \left(E_b \cos Z + E_d \right) \frac{(1 - \cos \beta)}{2} \tag{6.20}$$

The albedo is either the set of twelve user-specified values from the Location and Resource input page, or the value from the weather file, depending on whether the option on the Location and Resource input page **Use albedo in weather file if it is specified** is checked (`use_wf_albedo=1` in SSC) and there is albedo data in the weather file. NSRDB TMY3 weather files typically include albedo values, while the other formats do not. Otherwise (`use_wf_albedo=1` in SSC), SAM uses the twelve monthly albedo values from the Location and Resource input page (`albedo` in SSC), assuming that the albedo is constant over a single month.

7 Effective POA Irradiance

The effective, or plane-of-array (POA) irradiance is the incident irradiance less losses due to near-object shading, self shading, and soiling. A set of user-specified adjustment factors represents near-object shading and soiling, and must be generated outside of SAM, either by separate computer models, or by measuring equipment and its associated software. SAM calculates the self-shading factors (Section 7.3) for fixed arrays and for subarrays with one-axis tracking.

The effective irradiance is the solar energy that reaches the top of the module cover. The module submodel (Section 9) accounts for the effect of the module cover on the energy that reaches the photovoltaic cell including angle-of-incidence and reflection losses. Each module submodel option uses a different approach to calculating these losses.

7.1 Nominal Global Incident Irradiance

The nominal global incident irradiance I in kW/m^2 is the sum of the incident beam irradiance, incident sky diffuse irradiance, and incident ground-reflected irradiance:

$$I = I_b + I_d + I_r \qquad (7.1)$$

7.2 Near Object Shading

Near-object shading is a reduction in the POA incident irradiance by objects near the array, such as buildings, poles, trees, hills, etc. Near-object shading reduces both the beam and diffuse POA irradiance.

SAM represents the reduction in beam POA irradiance using a set of beam shading losses. A beam shading loss is a percentage that represents a reduction in the POA beam irradiance for a given hour. A shading loss of zero represents no shading. A shading loss of 100% represents complete shading.

SAM stores a set of shading losses for each subarray. The losses are specified in the Edit Shading Data window, which opens from the Shading input page when you click the **Edit Shading** button for a subarray.

SAM reads beam shading losses from three different lookup tables that make it possible to use shading data from different sources, such as shade analysis tools and modeling software:

- **Hourly** is a column of 8,760 shading losses, one for each hour of the year.

- **Hour by month** is a 24-by-12 table of 288 shading losses with one value for each hour of the day by month. For each month, SAM applies the same shading loss to a given hour of the day.

- **Solar azimuth by altitude** is a table of shading losses for a range of sun azimuth and altitude angles. SAM uses bilinear interpolation to calculate the shading loss for a given hour based on that hour's solar position angles and four shading losses from the table.

SAM's user interface provides access to conversion functions that allow for importing shading data from files created by either the PVsyst, Solmetric SunEye, or SolarPathfinder^{TM} software as described in SAM Help - Shading (2014). The conversion functions translate the data from the files into the appropriate table listed above. These functions are not available in SSC.

The shading loss inputs are expressed as percentages. SAM converts each percentage to a factor using the equation:

$$S = 1 - \frac{L}{100\%} \qquad (7.2)$$

Table 7. Effective POA Irradiance Variable Definitions

Symbol	Description / **Name in SAM**	Name in SSC
	Inputs	
I_b	incident beam irradiance	`poa_beam`
I_d	incident sky diffuse irradiance	`poa_skydiff`
I_r	incident ground-reflected irradiance	`poa_gnddiff`
L_h	8,760 hourly beam shading loss percentages	`poa_beam`
L_{mh}	hour-by-month beam shading loss percentages	`subarray[n]_shading_mxh`
L_{azal}	sun azimuth-by-altitude beam shading loss percentages	`subarray[n]_shading_azal`
L_{dns}	near object sky diffuse shading loss percentage	`subarray[n]_shading_diff`
L_{soilng}	monthly soiling loss percentages	`subarray[n]_soiling`
	Outputs	
I	**...Nominal POA total irradiance**	`...poa_nom`
$G_{g,shad}$	**...POA total irradiance after shading only**	`...poa_shaded`
G_b	**...POA beam irradiance after shading and soiling**	`...poa_eff_beam`
G_d	**...POA diffuse irradiance after shading and soiling**	`...poa_eff_skydiff`
G	**...POA total irradiance after shading and soiling**	`...poa_eff`
S_{bns}	**...Beam irradiance shading factor**	`...beam_shading_factor`
O	**...Soiling loss**	`...soiling_derate`

Ellipses (...) indicate **Subarray [n]** in SAM, and `hourly_subarray[n]_` in SSC

For self-shading variable definitions, see Section 7.3

The beam near-object shading factor for a given hour is the product of the values in the three beam shading loss factors:

$$S_{bns} = S_{hr}\, S_{mh}\, S_{azalt} \tag{7.3}$$

The sky diffuse near-object shading factor S_{dns} is a single value that represents a reduction in the POA diffuse irradiance for the entire year.

7.3 Self Shading

Self shading is shading of photovoltaic modules in one row by modules in a neighboring rows. SAM uses the same self-shading model for fixed subarrays (no tracking) and subarrays with one-axis tracking. SAM does not model self shading for subarrays with two-axis or azimuth-axis tracking. Because the model assumes that modules consist of photovoltaic cells with three bypass diodes, it is suitable for modules mono- or poly-crystalline silicon photovoltaic cells, but not for thin film modules. The self-shading algorithm is described in detail in Section 8.

The self-shading algorithm calculates three factors:

- Sky diffuse shading factor F_{dss} that reduces the POA incident diffuse irradiance

- Ground-reflected diffuse shading factor F_{gss} that also reduces the POA incident diffuse irradiance

- Beam DC loss factor F_{dcss} that applies to the DC output of the array

7.4 Effective Irradiance after Shading Only

SAM reports the effective irradiance after shading only (and before soiling) for reference. It is an intermediate value that is not used in any calculations.

The beam, sky diffuse, and ground-reflected components of the effective irradiance after shading only are:

$$G_{bshad} = I_b \, S_{bns}$$
$$G_{dshad} = I_d \, S_{dns} \, S_{dss} \qquad\qquad S_{dss} = 1 \text{ if no self shading}$$
$$G_{rshad} = I_r \, S_{rss} \qquad\qquad\qquad S_{rss} = 1 \text{ if no self shading} \qquad (7.4)$$

The global effective irradiance after shading only:

$$G_{shad} = G_{bshad} + G_{dshad} + G_{rshad} \qquad (7.5)$$

7.5 Soiling

Soiling losses are caused by a reduction in the incident irradiance due to dust and dirt on the module surface. SAM accounts for soiling losses using a set of monthly soiling loss percentages that are user inputs on the Losses input page (`subarray[n]_soiling` in SSC). The soiling loss applies to both the beam and diffuse components of the incident irradiance.

SAM converts each soiling loss percentage to a factor:

$$O = 1 - \frac{L}{100\%} \qquad (7.6)$$

For each hour of each month, SAM applies that month's soiling factor O to all components of the incident irradiance.

7.6 Effective Irradiance after Shading and Soiling

The effective irradiance after shading and soiling is the irradiance value that the module models use as input (see Section 9).

The components of the effective irradiance after shading and soiling are, where O is the soiling factor:

$$G_b = G_{bshad} \, O$$
$$G_d = G_{dshad} \, O$$
$$G_r = G_{rshad} \, O \qquad (7.7)$$

SAM reports the global effective irradiance G and the total diffuse effective irradiance $G_{d,t}$ in the hourly results. (Note that the module submodels use the component values in Equation 7.7 as input rather than the global value.):

$$G = G_b + G_d + G_r$$
$$G_{d,t} = G_d + G_r \qquad (7.8)$$

SAM also reports the effective beam shading factor S_b in the hourly results (S_{bns}) is the near-object shading factor from Equation 7.3:

$$S_b = \begin{array}{ll} S_{bns} \, S_{ss} \, O & \text{for fixed self shading} \\ S_{bns} \, O & \text{for one-axis self shading} \end{array} \qquad (7.9)$$

8   Self Shading Algorithm

This section describes SAM's self-shading model (Deline 2013). It supplements Section 7.3 and is presented here for clarity because of the length of the algorithm.

To model self shading for a subarray with no tracking (fixed) or one-axis tracking in SAM, after specifying the tracking and orientation options on the System Design page, on the Shading page, for the subarray's **Shading mode** input, choose **self-shaded**. (For a description of the backtracking algorithm for one-axis tracking, see Section 5.3.2.) In SSC, to enable self shading, set the `pvsamv1` input variable `subarray[n]_shade_mode` to zero.

The self-shading model consists of two sets of equations, one to calculate the reduction in diffuse POA irradiance, and a separate one to calculate a DC loss factor to account for the non-linear performance impact of the beam POA irradiance reduction:

- Reduction in diffuse POA irradiance:
    1. Calculate the shade mask angle ψ and ground coverage ratio *GCR*.
    2. Calculate the reduction in sky diffuse radiation.
    3. Calculate the reduction in ground-reflected diffuse radiation.
- Reduction in beam POA irradiance:
    1. Calculate the shadow dimensions *X* and *S*.
    2. Calculate the fixed self-shading DC loss factor to account for the beam POA reduction's impact on the array's output.

Some diffuse radiation is always blocked by neighboring rows of modules, regardless of whether the rows block beam radiation (Goswami 1989). The diffuse radiation consists of a sky diffuse component and a ground-reflected component, both of which are affected in this way.

Table 8 and Figure 4 show the variables used to calculate the fixed self-shading factors. The equations and algorithm are described in (Appelbaum 1979) and (Deline 2013).

8.1   Self Shading Assumptions

The self-shading model makes a set of simplifying assumptions to minimize the number of inputs:
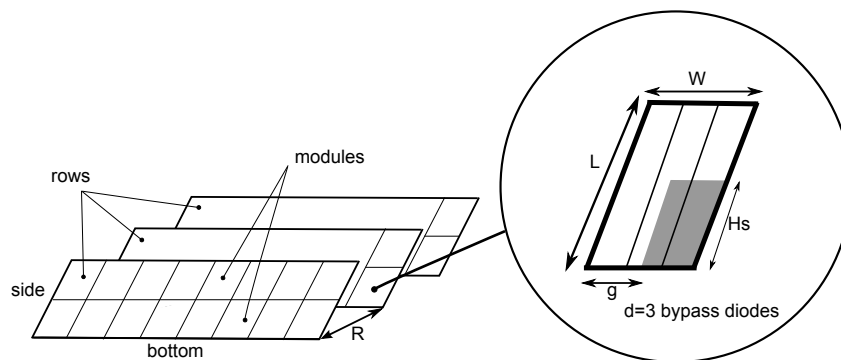
Figure 4. Shadow Dimensions for Portrait Module Orientation

This report is available at no cost from the National Renewable Energy Laboratory at www.nrel.gov/publications

Table 8. Self Shading Variable Definitions

Symbol	Description / **Name in SAM**	Name in SSC
	Inputs	
M_{total}	**Modules in subarray**	-
M_{string}	Number of modules per string	-
M_{side}	**Number of modules along side of row**	`subarray[n]_nmody`
M_{bottom}	**Number of modules along bottom of row**	`subarray[n]_nmodx`
$N_{strings}$	Number of strings in parallel	`strings_in_parallel`
GCR	**GCR**, ground coverage ratio	`subarray[n]_gcr`
A_m	module area	`[spe,cec,6par,snl]_area`
β_s	surface tilt angle	`subarray[n]_tilt`
γ_s	surface azimuth angle	`subarray[n]_azimuth`
Z	sun zenith angle	-
γ	sun azimuth angle	-
E_b	direct normal irradiance (from weather file)	-
G_b	POA beam irradiance	-
G_d	POA diffuse irradiance (sky and ground)	-
ρ	albedo (from weather file, or set to 0.2)	`albedo`
	Outputs	
S_{dss}	sky diffuse factor / **Self-shading diffuse derate**	`hourly_ss_diffuse_derate`
S_{rss}	ground diffuse factor / **Self-shading reflected derate**	`hourly_ss_reflected_derate`
F_{dcss}	DC loss factor / **Self-shading derate**	`hourly_ss_derate`

- Modules have an aspect ratio of $R_{aspect} = 1.7$, so that the length of the long side is 1.7 times the length of the short side.

- The module area A is the product of the module's total length and width. The value may be from one of the module libraries or a user input.

- Each module has three bypass diodes $d = 3$ so that the module is made up of three "submodules." A submodule is a string of photovoltaic cells in the module protected by a single bypass diode. For example, a 60-cell module would consist of three submodules of 20 cells each. This assumption makes the algorithm unsuitable for modeling self shading of thin film modules.

- A partially shaded submodule behaves in the same way as it would if the entire submodule were uniformly shaded.

- Modules are wired together in strings parallel to the ground (horizontal strings) so that the number of modules along the bottom of a subarray row is a multiple of the number of modules per string.

- Parallel strings in the subarray are uniformly shaded.

Based on the simplifying assumptions listed above, the number of rows N_{rows} in a subarray is:

$$N_{rows} = \left\lfloor \frac{M_{total}}{M_{side}M_{bottom}} \right\rfloor \tag{8.1}$$

where M_{total} is the number of modules in the subarray $M_{total} = M_{string}N_{strings}$. Note that if M_{total} is not an even multiple of the product $M_{string}N_{strings}$, SAM generates a simulation message indicating that the self-shading configuration does not match the subarray configuration, but still calculates the DC loss factor as described below.
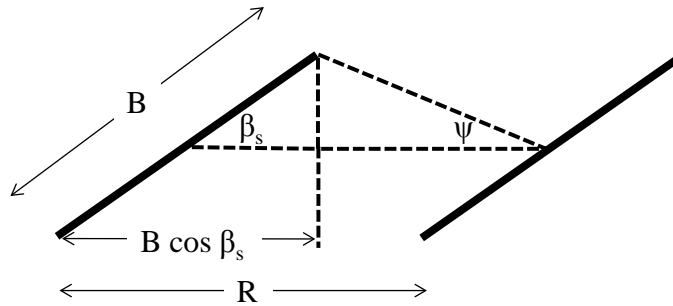
Figure 5. Side View of Two Rows with Self-shading Mask Angle Variables

SAM calculates the module length and width from the module area A in m^2 and the fixed aspect ratio assumption $R_{aspect} = 1.7$:

$$W = \sqrt{\frac{A_m}{R_{aspect}}} \tag{8.2}$$

$$L = W R_{aspect} \tag{8.3}$$

The module area is either from the module library, or an input, depending on the module submodel (see Section 9.1).

The length of the side of a row (distance from the bottom edge to the top edge of a row):

$$B = \begin{array}{ll} M_{side}L & \text{portrait module orientation} \\ M_{side}W & \text{landscape module orientation} \end{array}$$

The distance between bottom edges of neighboring rows:

$$R = \frac{B}{GCR} \tag{8.4}$$

where GCR is the ground coverage ratio.

The sun altitude angle:

$$\alpha = 90° - Z \tag{8.5}$$

8.2 Shade Mask Angle

The equations for the reduction in sky diffuse irradiance require values for the shade mask angle and ground coverage ratio.

The mask angle is the minimum array tilt angle at which the view of the sky at a given point along the side of the row is obstructed by a neighboring row as shown in Figure 5 from Passias (1984):

$$\psi = \arctan\frac{(B)\sin\beta_s}{R - B\cos\beta_s} \tag{8.6}$$

SAM uses the "worst case" approach described in (Deline 2013a), where the diffuse irradiance mask angle is calculated for the bottom of the array rather than averaged across the entire array.

8.3 Sky Diffuse POA Irradiance Reduction

For the purposes of self-shading calculations, SAM assumes isotropic sky diffuse irradiance, so that the total horizontal diffuse irradiance G_{dh} is:

$$G_{dh} = G_d \left[\frac{2}{1 + \cos \beta_s} \right] \tag{8.7}$$

The beam horizontal irradiance is:

$$G_{bh} = E_b \cos Z \tag{8.8}$$

The reduction in sky diffuse POA irradiance incident on the portion of the subarray shaded by neighboring rows is a function of the surface tilt angle β_s and mask angle ψ with a derating term $\frac{N_{rows}-1}{N_{rows}}$ for the number of interior rows in the array:

$$G_{sky,\text{red}} = G_d - G_{dh} \left[1 - \cos^2 \frac{\psi}{2} \right] \left[\frac{N_{rows} - 1}{N_{rows}} \right] \tag{8.9}$$

The sky diffuse shading factor:

$$S_{dss} = \frac{G_{sky,\text{red}}}{G_d} \tag{8.10}$$

8.4 Ground Diffuse POA Irradiance Reduction

The first row in the subarray is not affected by self shading. The length of ground in front of each shaded row that reflects beam radiation onto the row, with Y constrained to a minimum value of 0.00001:

$$Y = R - B \left[\frac{\sin(180° - \alpha - \beta_s)}{\sin \alpha} \right] \tag{8.11}$$

The view factor on the first row F_1, and the beam and diffuse reflected component factors F_2 and F_3, respectively:

$$
\begin{aligned}
F_1 &= \rho \sin^2 \frac{\beta_s}{2} \\
F_2 &= \frac{\rho}{2} \left(1 + \frac{Y}{B} - \sqrt{\frac{Y^2}{B^2} - \frac{2Y}{B} \cos(180° - \beta_s) + 1} \right) \\
F_3 &= \frac{\rho}{2} \left(1 + \frac{R}{B} - \sqrt{\frac{R^2}{B^2} - \frac{2R}{B} \cos(180° - \beta_s) + 1} \right)
\end{aligned}
\tag{8.12}
$$

The reduced ground-reflected diffuse POA irradiance on the entire subarray:

$$G_{gnd,red} = \left[\frac{F_1 + F_2(N_{rows} - 1)}{N_{rows}} \right] G_{bh} + \left[\frac{F_1 + F_3(N_{rows} - 1)}{N_{rows}} \right] G_{dh} \tag{8.13}$$

The ground diffuse shading factor:

$$
S_{rss} = \begin{cases} S_{rss} = \dfrac{G_{gnd,red}}{F_1(G_{bh} + G_{dh})} & \text{if } F_1(G_{bh} + G_{dh}) > 0 \\ 1 & \text{if } F_1(G_{bh} + G_{dh}) \leq 0 \end{cases}
$$

8.5 Shadow Dimensions

The shadow dimensions are defined by a shadow height H_S and shadow displacement g as shown in Figure 4. The equations are based on Appelbaum (1979).

The shadow dimensions equations assume that a surface azimuth angle γ_s of zero is for a south-facing surface in the northern hemisphere, and for a north-facing surface in the southern hemisphere. Because this differs from SAM's convention where $\gamma_s = 0$ is for a north-facing surface and $\gamma_s = 180°$ is for a south-facing surface, the equations use the effective surface azimuth $\gamma_{s,eff}$:

$$\gamma_{s,eff} = \gamma - \gamma_s \tag{8.14}$$

For an array that is not horizontal and for hours when the sun is up, the shaded portion of the array is:

$$P_y = B \left[\cos\beta_s + \cos\gamma_{s,eff} \frac{\sin\beta_s}{\tan(90-Z)} \right]$$

$$P_x = B \left[\sin\beta_s \frac{\sin\gamma_{s,eff}}{\tan(90-Z)} \right] \tag{8.15}$$

The shadow displacement g:

$$g = R\frac{P_x}{P_y} \tag{8.16}$$

with the following constraints:

$$g = \begin{cases} |g| & \text{when } g < 0 \\ 0 & \text{when } P_y = 0 \\ 0 & \text{when } M_{bottom} > M_{string} \\ B & \text{when } g > B \end{cases}$$

The shadow height H_s in Figure 4 is:

$$H_S = B \left[1 - \frac{R}{P_y} \right] \tag{8.17}$$

with the following constraints:

$$H_S = \begin{cases} 0 & \text{when } P_y = 0 \\ |H_S| & \text{when } g < 0 \\ B & \text{when } H_s > B \end{cases}$$

When there is self shading, some fraction X of each row in the subarray is shaded. In each row, a fraction S of the number of modules along the side is of the row is shaded (Deline 2013). The equations for S and X are from Deline (2013a).

For landscape module orientation, the case where $H_S > W$ assumes complete shading even though in a given string, some modules will be completely shaded while others are only partly shaded ($\lceil ... \rceil$ and $\lfloor ... \rfloor$ indicate the ceiling and floor functions, respectively):

$$X = \left\lceil \frac{H_S}{W} \right\rceil \frac{R-1}{RM_{side}} \tag{8.18}$$

$$S = \begin{cases} \left\lceil \frac{H_S d}{W} \right\rceil d^{-1} \left[1 - \left\lfloor \frac{g}{L} \right\rfloor \right] M_{bottom}^{-1} & \text{when } H_S \leq W \text{ for a fixed subarray} \\ 1 & \text{when } H_S > W, \text{ or for one-axis tracking} \end{cases}$$

For portrait module orientation, the configuration shown in Figure 4:

$$X = \left\lceil \frac{H_S}{L} \right\rceil \frac{R-1}{RM_{side}} \tag{8.19}$$

$$S = \begin{cases} 1 - \dfrac{gd}{W} \ (dM_{bottom})^{-1} & \text{for a fixed subarray} \\ 1 & \text{for one-axis tracking} \end{cases}$$

SAM assumes that each module has three bypass diodes so that $d = 3$ (See Section 8.1).

8.6 Beam Self-shading DC Loss Factor

The self-shading DC loss factor accounts for the reduction in beam POA irradiance caused by self shading. Its calculation is based on the proposed simplified analytical model from (Deline 2013) and requires the following variables:

- The shading fractions S and X described in Section 8.5.

- Fraction of solar irradiance reaching the shaded submodule E_e, which is a function of the reduction in diffuse POA irradiance from the sky and ground described in Section 8.3.

- Submodule fill factor FF_0 determined from the module nameplate parameters.

The ratio of incident diffuse irradiance to total incident irradiance is a measure of the solar irradiance reaching the shaded submodule:

$$R_{dt} = \frac{G_{sky,red} + G_{gnd,red}}{G_b + G_{sky,red} + G_{gnd,red}} \tag{8.20}$$

The fill factor depends on the photovoltaic module's rated maximum power, open-circuit voltage, and short-circuit current:

$$F_{fill} = \frac{P_{mp0}}{V_{oc0} I_{sc0}} \tag{8.21}$$

The DC loss factor equation requires three coefficients, which were determined using the empirical analysis described in Deline (2013):

$$C_1 = (109 F_{fill} - 54.3) e^{-4.5X}$$
$$C_2 = -6X^2 + 5X + 0.28 \quad \text{(if } X > 0.65, \text{ set } X = 0.65\text{)}$$
$$C_3 = \max \left[(-0.05 R_{dt} - 0.01) X + (0.85 F_{fill} - 0.7) R_{dt} - 0.0855 F_{fill} + 0.05, \ R_{dt} - 1 \right] \tag{8.22}$$

The DC loss equations for each shade factor condition are:

$$F_{dc1} = 1 - C_1 S^2 - C_2 S$$
$$F_{dc2} = \begin{cases} \dfrac{X - S \ 1 + 0.5 m_d V_{mp}^{-1}}{X} & \text{if } X > 0 \\ 0 & \text{if } X = 0 \end{cases}$$
$$F_{dc3} = C_3 (S - 1) + R_{dt} \tag{8.23}$$

SAM chooses the maximum of the three factors to calculate the system DC loss factor accounting for the reduction in beam POA irradiance due to self shading, and constrains the factor to between 0 and 1:

$$F_{dcss} = X \max (F_{dc1}, F_{dc2}, F_{dc3}) + (1 - X) \tag{8.24}$$

9 Module DC Output

SAM uses one of four module submodels and one of three cell temperature submodels to calculate the DC power output of a single module in each subarray. SAM's implementations of the submodels are effectively point-value efficiency models because SAM assumes that each subarray operates at its maximum power point (except with the mismatch option described in the next paragraph):

- The subarray maximum power point is determined by the maximum power point of a single module.

- All modules in each subarray operate uniformly. SAM does not calculate module mismatch losses.

For an array with multiple subarrays and the CEC module submodel, SAM's optional PV subarray mismatch option adjusts the maximum power voltage to account for mismatch *between subarrays* as described in Section 10.6.

Calculating the module's DC output in a given hourly time step when the sun is up (see Section 4.4) is a two step process:

1. Calculate the photovoltaic cell temperature. SAM assumes that the temperature of cells in all of the modules in each subarray is uniform.

2. Calculate the module's DC power output based on its physical characteristics, the effective irradiance, and cell temperature.

Table 9 lists the input and output variables used by each of the module and cell temperature submodels described. Each submodel uses an additional set of input parameters as described in the submodel descriptions.

9.1 Module Submodel Options

SAM calculates the a single photovoltaic module's DC output with one of the module submodels listed below, with descriptions from SAM Help - Module (2014) and Blair (2013):

- **Simple Efficiency Module Model** (Section 9.6) is a simple representation of module performance that requires you to provide the module area, a set of conversion efficiency values, and temperature correction parameters. The simple efficiency model is the least accurate of the three models for predicting the performance of specific modules. It is useful for preliminary performance predictions before you have selected a specific module, and allows you to specify a module efficiency and temperature performance parameters, which is useful for analyses involving sensitivity or parametric analysis.

- **California Energy Commission (CEC) Performance Model with Module Database** (Section 9.5) is an implementation of the six-parameter single-diode equivalent circuit model used in the CEC New Solar Homes Partnership Calculator (Go Solar California 2014a), and is an extension of the five-parameter model described in De Soto (2004a). The model calculates the photovoltaic module hourly DC output using equations with parameters stored in SAM's CEC module library. The library contains data provided by the California Energy Commission (Go Solar California 2014a) (Go Solar California 2014b).

- **CEC Performance Model with User Entered Specifications** (Section 9.5) is the same implementation as the CEC Performance Model with Module Database, but with a coefficient calculator (Dobos 2012a) to calculate the model parameters from the standard module specifications provided on manufacturer data sheets. This makes it possible to use the six-parameter model for modules not included in the CEC module library.

- **Sandia PV Array Performance Model with Module Database** (Section 9.4) is an implementation of the Sandia National Laboratories photovoltaic module and array performance model (King 2004). The model calculates hourly efficiency values based on data measured from modules and arrays in realistic outdoor operating conditions. The database includes modules with different cell types, including crystalline silicon, and various thin film technologies.

Table 9. Module Submodel Variable Definitions

Symbol	Description / **Name in SAM**
	Inputs
G_b	effective beam irradiance
G_d	effective sky diffuse irradiance
G_r	effective ground-reflected irradiance
T_{dry}	ambient dry bulb temperature
v_w	wind speed from weather file
Z	sun zenith angle
h	elevation above sea level in meters from weather file
β_s	subarray **tilt** angle
γ_s	subarray **azimuth** angle
AOI	incidence angle
hr	hour of day local time
	Outputs
P_{dc}	module power
V_{dc}	module voltage
I_{dc}	module current
η_m	module efficiency
V_{oc}	operating open circuit voltage
I_{sc}	operating closed circuit current
T_c	cell temperature

Table 10. Module Submodels in SSC

Name in SAM	`module_model`
Simple Efficiency Module Model	0
CEC Performance Model with Module Database	1
CEC Performance Model with User Entered Specifications	2
Sandia PV Array Performance Model with Module Database	3

In SSC, the `pvsamv1` input variable `module_model` determines the module submodel as shown in Table 10.

9.2 Cell Temperature Submodel Options

Each module submodel uses one of the cell temperature submodels shown in Table 11 to calculate the photovoltaic cell temperature in a given hour. Each module submodel uses a single cell temperature submodel, except for the CEC module model with database parameters, which offers two cell temperature submodel options.

As described above, in each hourly time step, SAM calls the appropriate cell temperature submodel once before calling the module submodel. When the subarray mismatch option (Section 10.6) is enabled, it also calls the cell temperature submodel as part of that set of calculations.

9.3 Module Degradation

SAM's module submodels calculate the DC power output of a single module in a single hourly time step. For a given simulation, SAM runs the module submodel 8,760 times to calculate the output over a single year. Because

Table 11. Cell Temperature Submodels

Module Model	Temperature Model	Section
Simple efficiency	Sandia	9.9
Sandia	Sandia	9.9
CEC with database parameters	NOCT or Heat Transfer	9.7 or 9.8
CEC with user-specified parameters	NOCT	9.7

the model calculates the output in a single year, it does not have the information required to model degradation of module output over a number of years of operation.

SAM's photovoltaic performance model calculates the hourly AC output of the photovoltaic system. Its financial models add up these 8,760 hourly values to calculate the system's total AC output in one year, and treat this value as the system's total output in the first year of the system's operation.

The financial models use a lifetime in years and annual degradation factor to calculate the system's output in the second year and later for cash flow calculations. This degradation factor can approximate the impact of module degradation on the performance of a photovoltaic system. However, because the degradation factor applies to the total annual *AC output* of the system, it only very roughly represents module degradation. In an actual system, module degradation reduces the array's *DC output*, and this reduction can affect inverter sizing decisions in a way that SAM cannot model.

If the weather file for a given analysis contains typical-year data (rather than data for a specific year), using the AC degradation factor to represent module degradation is further complicated by the fact that the solar resource input to the module model represents the resource over a multi-year period.

9.4   Sandia Module Model

SAM's Sandia Module Model (see Section 9.1) is an implementation of the empirical model of a photovoltaic module described in King (2004).

For a description of the Sandia cell temperature model, see Section 9.9.

The Sandia module model uses the module parameters in Table 12 along with the general module input variables in Table 9. The parameters in Table 12 are from the Sandia module database maintained by Sandia National Laboratories (Sandia 2014). In SAM, the parameters are stored in the Sandia module library (SAM Help - Libraries 2014). In SSC, the implementation of the Sandia module model described here is part of the `pvsamv1` module and does not include a parameter library (See Section 2.2).

The reference conditions for the Sandia module model parameters are total incident irradiance of 1,000 W/m^2 and reference cell temperature of 25°C.

When you select a module on SAM's Module input page for the **Sandia PV Array Performance Model with Module Database**, SAM displays some of the module's reference parameters in Table 12 from the Sandia Module library as uneditable values. You can use SAM's library editor to see a complete list of the module parameters or to modify parameter values (SAM Help - Libraries 2014).

The absolute air mass is a relative measure of the optical path length that sunlight travels through the atmosphere and depends on the sun zenith angle in degrees (Section 4.2) and the elevation above sea level ($AM = 1$ at sea level with the sun directly overhead):

$$AM = \left[\cos(\frac{\pi}{180}Z) + 0.5057(96.08 - Z)^{-1.634} \right]^{-1} e^{-0.0001184h} \qquad (9.1)$$

Table 12. Sandia Module Model Variable Definitions

Symbol	Description / **Name in SAM**	Name in SSC
	Inputs	
$V_{mp,ref}$	reference **Max Power Voltage**	`snl_vmpo`
$I_{mp,ref}$	reference **Max Power Current**	`snl_i_mpo`
$V_{oc,ref}$	reference **Open Circuit Voltage**	`snl_voco`
$I_{sc,ref}$	reference **Short Circuit Current**	`snl_isco`
$\alpha_{sc,ref}$	short circuit temperature coefficient	`snl_aisc`
$\beta_{oc,ref}$	open circuit temperature coefficient	`snl_bvoco`
$\beta_{mp,ref}$	maximum power voltage temperature coefficient	`snl_bvmpo`
$\gamma_{mp,ref}$	maximum power temperature coefficient	`snl_aimp`
$M_{\beta oc}$	relates $\beta_{oc,ref}$ to effective irradiance	`snl_mbvoc`
$M_{\beta mp}$	relates $\beta_{mp,ref}$ to effective irradiance	`snl_mbvmp`
s	number of cells in series	`snl_series_cells`
C_0, C_1	coefficients relating I_{mp} to G	`snl_c0`, `snl_c1`
C_2, C_3	coefficients relating V_{mp} to G	`snl_c2`, `snl_c3`
C_4, C_5	coefficients relating I_x to G	`snl_c4`, `snl_c5`
C_6, C_7	coefficients relating I_{xx} to G	`snl_c6`, `snl_c7`
n	diode factor	`snl_n`
f_d	fraction of diffuse irradiance used by module	`snl_fd`
$a_{0...4}$	air mass coefficients 0...4	`snl_a[0...4]`
$b_{0...5}$	incidence angle modifier coefficients 0...5	`snl_b[0...5]`
	Output	
$P_{dc,m}$	module DC output power	

The F_1 polynomial relates the spectral effects on I_{sc} to the variation of air mass over the day:

$$F_1 = a_0 + a_1 AM + a_2 AM^2 + a_3 AM^3 + a_4 AM^4 \qquad (9.2)$$

The F_2 polynomial relates the optical effects on I_{sc} to the angle of incidence AOI (Section 5.1):

$$F_2 = b_0 + b_1 AOI + b_2 AOI^2 + b_3 AOI^3 + b_4 AOI^4 + b_5 AOI^5 \qquad (9.3)$$

The short circuit current is a function of the cell temperature (Section 9.9), effective irradiance (Section 7) and the short circuit temperature coefficient in A/°C. The diffuse utilization factor F_d is one of the parameters from the Sandia module library, and is equal to one for flat plate modules:

$$I_{sc} = I_{sc,ref} F_1 \; \frac{G_b F_2 + f_d(G_d + G_r)}{1000} \; 1 + \alpha_{sc,ref}(T_c - 25) \qquad (9.4)$$

The effective irradiance on the module surface to which the cells respond:

$$E_e = \frac{I_{sc}}{I_{sc,ref} \; 1 + \alpha_{sc,ref}(T_c - 25)} \qquad (9.5)$$

The current at the maximum power point:

$$I_{mp} = I_{mp,ref}(C_0 E_e + C_1 E_e^2) \; 1 + \alpha_{sc,ref}(T_c - 25) \qquad (9.6)$$

The following two intermediate values are required for the maximum power voltage equations:

$$\Delta T_c = n \; \frac{1.38066 \times 10^{-23}(T_c + 273.15)}{1.60218 \times 10^{-19}}$$
$$\beta_{mp} = \beta_{mp,ref} + M_{\beta mp}(1 - E_e) \qquad (9.7)$$

The voltage at the maximum power point is zero if $E_e \leq 0$. Otherwise:

$$V_{mp} = V_{mp,ref} + C_2 \; s \; \Delta T_c \; \log(E_e) + C_3 \; s \; [\Delta T_c \log(E_e)]^2 + \beta_{mp}(T_c - 25) \qquad (9.8)$$

The module's DC power output is at the maximum power point:

$$P_{dc,m} = V_{mp} \; I_{mp} \qquad (9.9)$$

9.5 CEC Module Model

SAM's CEC Module Model (see Section 9.1) is an implementation of the single-diode equivalent circuit model of a photovoltaic module described in De Soto (2004a), and with more detail in De Soto (2004b). It is also described in Section 23.2 of Duffie and Beckman (2013).

SAM includes two options for the model (see Section 9): It may use a set of reference parameters from the CEC module library with data provided by the California Energy Commission (Go Solar California 2014a), or a set of reference parameters generated by SAM's coefficient generator from module specifications that you provide as input to the model. The equations described in this section are for the module's DC output, not for the module coefficients. For a description of the coefficient generator equations, see Dobos (2012a).

Table 13 shows the five module specifications and six parameters that are input variables specific to the CEC module model. The model also uses the general module input variables in Table 9.

Table 13. CEC Module Model Variable Definitions

Symbol	Description / **Name in SAM**	Name in SSC
	Inputs	
$I_{mp,ref}$	reference **Max Power Current**	`cec_i_mp_ref`
$V_{oc,ref}$	reference **Open Circuit Voltage**	`cec_v_oc_ref`
$I_{sc,ref}$	reference **Short Circuit Current**	`cec_i_sc_ref`
$\alpha_{sc,ref}$	short circuit current temperature coefficient (A/°C)	`cec_alpha_sc`
$\beta_{oc,ref}$	open circuit voltage temperature coefficient (V/°C)	`cec_beta_oc`
$I_{L,ref}$	reference light current, **I_L_ref**	`cec_i_l_ref`
$I_{o,ref}$	reference diode saturation current, **I_o_ref**	`cec_i_o_ref`
$R_{s,ref}$	reference series resistance, **R_s**	`cec_r_s_ref`
a_{ref}	reference ideality factor, **A_ref**	`cec_a_ref`
$R_{sh,ref}$	reference shunt resistance **R_sh_ref**	`cec_r_sh_ref`
$adjust$	temperature coefficient adjustment factor	`cec_adjust`
	Output	
$P_{dc,m}$	module DC output power	

In SSC, the implementation of the CEC module model described here is part of the `pvsamv1` module. The coefficient generator is available in SSC as a separate compute module called `6parsolve`.

The reference conditions for the CEC module model parameters are total incident irradiance of 1,000 W/m^2 and reference cell temperature of 25°C.

When you select a module on SAM's Module input page for the **CEC Performance Model with Module Database**, SAM displays the module's reference parameters shown in Table 13 from the CEC Module library as uneditable values. You can use SAM's library editor to modify reference parameter values for a module (SAM Help - Libraries 2014).

The five-parameter single-diode equivalent circuit equation for the module current I at a given voltage V is:

$$I = I_L - I_o \left[\exp\left(\frac{V + IR_S}{a} \right) - 1 \right] - \frac{V + IR_S}{R_{sh}} \tag{9.10}$$

The temperature coefficient of short circuit current μ_{isc} and open circuit voltage β_{voc} in the model are adjusted from the reference coefficients using a sixth parameter, $adjust$, which may be either from the CEC module library, or calculated by SAM's coefficient generator:

$$\mu_{isc} = \alpha_{sc,ref} \left(1 - \frac{adjust}{100} \right)$$

$$\beta_{voc} = \beta_{oc,ref} \left(1 + \frac{adjust}{100} \right) \tag{9.11}$$

The global effective irradiance at the top of the module cover:

$$G = G_b + G_d + G_r \tag{9.12}$$

Equations for the transmittance through the module cover use three constants:

$$n = 1.526 \quad \text{(refractive index of glass)}$$
$$L = 0.002 \quad \text{(thickness of glass cover in meters)}$$
$$K = 4 \quad \text{(proportionality constant in meters}^{-1}) \tag{9.13}$$

The angle of refraction, assuming that the refractive index of air is one:

$$\theta_r = \arcsin\left(\frac{1}{n}\sin AOI\right) \tag{9.14}$$

The transmittance as a function of incidence angle θ_i in radians:

$$\tau(\theta) = e^{-KL/\cos\theta_r}\left[1 - \frac{1}{2}\left(\frac{\sin^2(\theta_r - \theta_i)}{\sin^2(\theta_r + \theta_i)} + \frac{\tan^2(\theta_r - \theta_i)}{\tan^2(\theta_r + \theta_i)}\right)\right] \tag{9.15}$$

The incidence angle for the sky diffuse and ground-reflected components of the effective irradiance:

$$\theta_d = 59.7 - 0.1388\beta + 0.001497\beta^2 \quad \text{(sky diffuse angle)}$$
$$\theta_g = 90 - 0.5788\beta + 0.002693\beta^2 \quad \text{(ground-reflected diffuse angle)} \tag{9.16}$$

The transmittance for each component of the effective irradiance is calculated using Equation 9.15 with the module cover properties in Equation 9.13 and the following incidence angle values:

$$(\tau\alpha)_n : \theta_i = 1 \quad \text{(surface normal)}$$
$$(\tau\alpha)_b : \theta_i = \theta \quad \text{(beam)}$$
$$(\tau\alpha)_d : \theta_i = \theta_d \quad \text{(sky diffuse)}$$
$$(\tau\alpha)_g : \theta_i = \theta_g \quad \text{(ground diffuse)} \tag{9.17}$$

Note that $(\tau\alpha)_n$ is set to 1 degree (a very small number) instead of zero to avoid potential divide-by-zero errors.

The incidence angle modifier for each component of the effective irradiance is then:

$$K_{\tau\alpha,b} = \frac{(\tau\alpha)_b}{(\tau\alpha)_n}$$
$$K_{\tau\alpha,d} = \frac{(\tau\alpha)_d}{(\tau\alpha)_n}$$
$$K_{\tau\alpha,g} = \frac{(\tau\alpha)_g}{(\tau\alpha)_n} \tag{9.18}$$

The irradiance absorbed by the photovoltaic cell is:

$$G_0 = G_b K_{\tau\alpha,b} + G_d K_{\tau\alpha,d} + G_r K_{\tau\alpha,g} \tag{9.19}$$

The model limits sun zenith angle, setting its value to the minimum or maximum when it is exceeded:

$$0 < Z < 86° \tag{9.20}$$

The transmittance-absorptance product:

$$\tau\alpha = 0.9\frac{G_0}{G} \tag{9.21}$$

The air mass, with an exponential correction factor adapted from the Sandia model for elevation above sea level:

$$AM = \left[\cos\left(\frac{\pi}{180}Z\right) + 0.5057(96.080 - Z)^{-1.634}\right]^{-1} e^{-0.0001184h} \tag{9.22}$$

The air mass modifier M accounts for the effect of air mass on spectral distribution, and is from King (2004) with coefficients for polycrystalline cells:

$$a_0 = 0.918093$$
$$a_1 = 0.086257$$
$$a_2 = -0.024459$$
$$a_3 = 0.002816$$
$$a_4 = -0.000126$$
$$M = a_0 + a_1 AM + a_2 AM^2 + a_3 AM^3 + a_4 AM^4 \tag{9.23}$$

The adjusted effective transmitted irradiance:
$$G = MG_0 \tag{9.24}$$

The module DC output is only calculated when $G > 1$.

The cell temperature T_c for the CEC module model is calculated using either the NOCT cell temperature model (Section 9.7) or the heat transfer cell temperature model (Section 9.8). For the next set of equations, the cell temperature is in Kelvin, and the reference cell temperature of $25°C$ in Kelvin is $25°C + 273.15K = 298.15K$.

The light current I_L:
$$I_L = \frac{G}{1000}\left[I_{L,ref} + \mu_{I,sc}(T_c - 298.15)\right] \tag{9.25}$$

The diode reverse saturation current I_o, assuming a reference band-gap energy of silicon of 1.12 eV:

$$k = 8.618 \times 10^{-5} \qquad\qquad \text{Boltzmann constant in eV/K}$$
$$E_{bg} = 1.12\left[1 - 0.0002677(T_c - T_{c,ref})\right] \qquad\qquad \text{cell material band-gap energy in eV}$$
$$I_o = I_{o,ref}\left(\frac{T_c}{298.15}\right)^3 \exp\left[\frac{1}{k}\left(\frac{1.12}{T_{c,ref}} - \frac{E_{bg}}{T_c}\right)\right] \tag{9.26}$$

At the open circuit voltage V_{oc}, $I = 0$, and Equation 9.10 reduces to:
$$I = I_L - I_o\left[\exp\left(\frac{V_{oc} + IR_S}{a}\right) - 1\right] - \frac{V_{oc}}{R_{sh}} \tag{9.27}$$

SAM uses the bisection method to solve Equation 9.27 for V_{oc} with the reference open circuit voltage $V_{oc,ref}$ as an initial guess and the following parameter values:

$$a = a_{ref}\frac{T_c}{T_{c,ref}}$$
$$R_{sh} = R_{sh,ref}\frac{1000}{G} \tag{9.28}$$

The short circuit current:
$$I_{sc} = \frac{I_L}{1 + \frac{R_{s,ref}}{R_{sh}}} \tag{9.29}$$

For the CEC module model, SAM can calculate the module operating voltage either using the string mismatch equations described in Section 10.6, or the equations described below.

Table 14. Simple Efficiency Module Model Variable Definitions

Symbol	Description / **Name in SAM**	Name in SSC
	Inputs	
$V_{mp,ref}$	reference **Maximum Power Voltage**	-
$V_{oc,ref}$	reference **Open Circuit Voltage**	-
A_m	module **Area**	`spe_area`
$\gamma_{mp,ref}$	maximum power **Temperature Coefficient**	`spe_temp_coeff`
a	Sandia temperature parameter **a**	`spe_a`
b	Sandia temperature parameter **b**	`spe_b`
ΔT	Sandia temperature parameter **dT**	`spe_dT`
$G_{0...4}$	radiation values in efficiency table	`spe_rad[0...4]`
G_{ref}	reference radiation value from efficiency table	`spe_reference`
$\eta_{0...4}$	module efficiency values in efficiency table	`spe_eff[0...4]`
f_d	**Diffuse utilization factor**	`spe_fd`
	Output	
$P_{dc,m}$	module DC output power	

For the current at maximum power I_{mp}, SAM uses Newton's method to find a solution of Equation 9.10 using the calculated values for I_L, I_o, a, and R_{sh} at the maximum power point voltage. Because the effect of series resistance on the maximum power point is relatively small, for R_S, the model uses the reference value so that $R_S = R_{S,ref}$. Also, I_L from Equation 9.25 is multiplied by a factor of 0.9.

For systems with one subarray, or for systems with more than one subarray with the subarray mismatch option disabled, SAM calculates the maximum power point voltage using a golden section search to determine the voltage between 0 and V_{oc} that results in the maximum power given the current at maximum power.

For systems with more than one subarray and the subarray mismatch option enabled (Section 10.6, SAM sets the maximum power voltage to the value calculated by the mismatch algorithm.

The module DC output is the power at maximum power:

$$P_{dc,m} = V_{mp} \, I_{mp} \tag{9.30}$$

9.6 Simple Efficiency Module Model

SAM's Simple Efficiency Module Model (see Section 9.1) is a basic single point efficiency representation of a photovoltaic module that uses an efficiency table and the Sandia cell temperature model (Section 9.9) to calculate the module's DC power output.

The simple efficiency module model uses the module parameters shown in Table 14 along with the general module input variables in Table 9. Note that the model does not use the reference parameters $V_{mp,ref}$ and $V_{oc,ref}$. Those values are required by the sizing algorithm that is part of SAM's user interface (Section 2.3).

The cell temperature parameters a, b, and ΔT depend on the **Module Structure and Mounting** option (`spe_-module_structure` in SSC) as listed in Table 18. Note that the "use database values" option is not available for the simple efficiency module model, so that `spe_module_structure = 0` is for the "glass/cell/polymer sheet - open rack" option instead of the"use database option" as shown in the table for `snl_module_structure`.

The reference irradiance condition for the simple efficiency module model is the value that you specify, typically 1,000 W/m^2. The reference cell temperature is 25°C.

Table 15. NOCT Cell Temperature Model Variable Definitions

Symbol	Description / **Name in SAM**	Name in SSC
	Inputs	
G	effective irradiance transmitted to the photovoltaic cell	-
$\tau\alpha$	effective transmittance-absorbtance product	-
v_w	wind speed from weather file	-
-	**Array height**	`cec_height`
-	**Mounting standoff**	`cec_standoff`
T_{noct}	module NOCT temperature rating	`6par_tnoct`
I_{mp}	module maximum power current rating	`6par_imp`
V_{mp}	module maximum power voltage rating	`6par_vmp`
A_m	module area in meters	`6par_area`
	Output	
T_c	cell temperature	`hourly_subarray[n]_celltemp`

The effective total irradiance (the diffuse utilization factor F_d is adapted from the Sandia module model, and is equal to one for flat plate modules):

$$G = G_b + f_d(G_d + G_r) \tag{9.31}$$

SAM determines the module's conversion efficiency η_m by using linear interpolation on the radiation and efficiency values in the efficiency table to estimate the conversion efficiency at the effective total irradiance value G.

The module's DC power output:

$$P_{dc,m} = G\eta_m A_m \frac{\gamma_{mp,ref}}{100}(T_c - 25) \tag{9.32}$$

9.7 NOCT Cell Temperature Model

The NOCT photovoltaic cell temperature model is from Chapter 23.3 of Duffie and Beckman (2013), also described in De Soto (2004b). In SAM's implementation of the model, it is available with the **CEC Performance Model with Module Database** module model on the Module input page as the **NOCT cell temp model** option (`cec_temp_corr_mode = 0` in SSC `pvsamv1`).

The cell temperature is only calculated when the effective transmitted irradiance from Equation 9.24 is greater than zero, $G > 0$.

The reference module efficiency is based on the module specifications at 1,000 W/m^2 incident irradiance:

$$\eta_{ref} = \frac{I_{mp}V_{mp}}{1000A_m} \tag{9.33}$$

The wind speed adjusted for height above the ground depends on the **Array height** option on the Module page (`cec_height` in SSC):

$$v_{w,adj} = \begin{array}{ll} 0.51v_w & \text{one story or lower} \\ 0.61v_w & \text{two stories or higher} \end{array} \tag{9.34}$$

The NOCT adjusted for mounting stand-off type depends on the **Mounting standoff** option on the Module page

Table 16. Heat Transfer Cell Temperature Model Variable Definitions

Description / **Name in SAM**	Name in SSC
Inputs	
Mounting Configuration	`cec_mounting_config`
Heat Transfer Dimensions	`cec_heat_transfer`
Mounting Structure Orientation	`cec_mounting_orientation`
Module Width	`cec_module_width`
Module Length	`cec_module_length`
Rows of modules in array	`cec_array_rows`
Columns of modules in array	`cec_array_cols`
Temperature behind the module	`cec_backside_temp`
Gap Spacing	`cec_gap_spacing`
Output	
cell temperature	`hourly_subarray[n]_celltemp`

(`cec_standoff` in SSC):

$$T_{noct,adj} = \begin{cases} T_{noct} + 2 & \text{building integrated, greater than 3.5 in, or ground/rack mounted} \\ T_{noct} + 2 & \text{2.5 to 3.5 in} \\ T_{noct} + 6 & \text{1.5 to 2.5 in} \\ T_{noct} + 11 & \text{0.5 to 1.5 in} \\ T_{noct} + 18 & \text{less than 0.5 in} \end{cases} \tag{9.35}$$

The cell temperature in degrees Celsius (°C):

$$T_c = T_a + \frac{G}{800}\left(T_{noct,adj} - 20\right)\left(1 - \frac{\eta_{ref}}{\tau\alpha}\right)\left(\frac{9.5}{5.7 + 3.8 v_{w,adj}}\right) \tag{9.36}$$

9.8   Heat Transfer Cell Temperature Model

The heat transfer cell temperature model is from the work described in Neises (2011), with model descriptions in Chapters 4 and 5, and an implementation in FORTRAN in Appendix H. SAM's model is adapted from the FORTRAN implementation, and its description is omitted from this section because of its length.

In SAM, the heat transfer cell temperature model is available with the **CEC Performance Model with Module Database** module model on the Module input page as the **Mounting specific cell temp model** option (`cec_temp_corr_mode` = 1 in SSC `pvsamv1`).

9.9   Sandia Cell Temperature Model

The Sandia photovoltaic cell temperature model is the cell temperature model described in King (2004) as part of the Sandia photovoltaic array performance model. In SAM's implementation of the model, it is available with the **Sandia PV Array Performance Model with Module Database** module model on the Module input page.

The Sandia cell temperature model requires the effective incident irradiance and three empirical parameters from the Sandia module library, which stores data provided by the Sandia National Laboratories Photovoltaic Test and Evaluation program (Sandia 2014).

Table 17. Sandia Cell Temperature Model Variable Definitions

Symbol	Description / **Name in SAM**	Name in SSC
	Inputs	
G_b	effective beam irradiance	-
G_d	effective sky diffuse irradiance	-
G_r	effective ground-reflected irradiance	-
v_w	wind speed from weather file	-
T_a	ambient temperature from weather file	-
a	Temperature coefficient **a**	`snl_a` or `snl_ref_a`
b	Temperature coefficient **b**	`snl_b` or `snl_ref_b`
ΔT	Temperature coefficent **dT**	`snl_dtc` or `snl_ref_dtc`
	Output	
T_c	cell temperature	`hourly_subarray[n]_celltemp`

Table 18. Sandia Module Structure Options

Structure and Mounting	`snl_module_structure`	a	b	ΔT
Use Database Values	0	-	-	-
Glass/Cell Polymer Sheet - Open Rack	1	-3.56	-0.075	3
Glass/Cell/Glass - Open Rack	2	-3.47	-0.0594	3
Polymer/Thin Film/Steel - Open Rack	3	-3.58	-0.113	3
Glass/Cell/Polymer Sheet - Insulated Rack	4	-2.81	-0.0455	0
Glass/Cell/Glass - Close Roof Mount	5	-2.98	-0.0471	1
User Defined	6	-	-	-

SAM's Module input page provides the **Module Structure and Mounting** option, which allows you to choose to use a, b, and ΔT values from the Sandia module library, choose from a set of pre-determined values for different module types and air circulation options, or provide your own "user-defined" values. In SSC, the variable `snl_module_structure` determines the option, and the variables shown in Table 17 with '_ref' are for user-specified values. Table 18 shows the a, b, and ΔT values associated with each option in both SAM and SSC.

The effective irradiance:

$$G = G_b + G_d + G_r \tag{9.37}$$

The module back temperature in degrees Celsius (°C):

$$T_m = Ge^{a+bv_w} + T_a \tag{9.38}$$

The cell temperature in degrees Celsius (°C):

$$T_c = T_m + \frac{G}{1000}\Delta T \tag{9.39}$$

10 Array DC Output

SAM calculates the photovoltaic array's DC output by multiplying a single module's DC output (Section 9) by the number of modules in the array. This assumes that all of the modules in the array operate uniformly at the maximum power point of a single module.

SAM does not explicitly calculate electrical losses due to maximum power point mismatches between modules that would occur in a real system. You can account for those losses using the DC loss factor (Section 10.3).

10.1 DC String Voltage

SAM calculates the array's DC string voltage in each hourly time step to determine the inverter's input voltage.

For a system with a single subarray, the array's string DC voltage in a given hourly time step is the module voltage times the number of modules per string.

$$V_{dc} = V_{dc,m} N_{modules} \tag{10.1}$$

For systems with more than one subarray, when the subarrays have different orientations, tracking, shading factors, or DC loss factors, the maximum power point of each subarray is different. In such cases, SAM calculates the array's DC output voltage in one of two ways:

- By averaging the values of the subarray string voltages (available with all module models):

$$V_{dc} = \frac{1}{N_{sub}} \sum_{n=1}^{N_{sub}} V_{dc,n} \tag{10.2}$$

- By running the module model iteratively for each subarray to determine the array's maximum string voltage (available with the CEC module model only) as described in Section 10.6.

When you choose the CEC module model, the **Calculate maximum power voltage for array and associated losses due to subarray mismatch** check box (`enable_mismatch_vmax_calc = 1` in the SSC `pvsamv1` module) determines what method SAM uses to calculate the array's maximum power point DC voltage.

10.2 Gross DC Power Output

For systems with a single subarray, the array's gross DC power output is the output of a single module multiplied by the number of modules in the array, and, for arrays with no tracking, by the optional fixed self shading DC loss factor (Equation 8.24):

$$P_{dc,gross} = N_{modules}\, N_{parstrings}\, P_{dc,m}\, F_{dcss} \tag{10.3}$$

For systems with more than one subarray, the subarray's DC power output is the product of the number of modules in each subarray and the output of a single module. The array's gross DC output is the sum of the subarray values (the fixed self shading DC loss factor does not apply to arrays with more than one subarray):

$$P_{dc,gross,n} = N_{modules}\, N_{strings,n}\, P_{dc,m}$$

$$P_{dc,gross} = \sum_{n=1}^{N_{sub}} P_{dc,gross,n} \tag{10.4}$$

Table 19. Array DC Output Variable Definitions

Symbol	Description / **Name in SAM**	Name in SSC
	Inputs	
$V_{dc,m}$	module dc voltage	-
$P_{dc,m}$	module dc power	-
$N_{modules}$	**Modules per string**	`modules_per_string`
$N_{parstrings}$	**Strings in parallel**	`strings_in_parallel`
$N_{strings,n}$	**Strings allocated to subarray** n	`subarray[n]_nstrings`
N_{sub}	number of enabled subarrays	-
$L_{dc,n}$	**Total DC power loss** for subarray n	`subarray[n]_dcloss`
	Outputs	
$V_{dc,n}$	**Subarray [n] DC string voltage**	`hourly_subarray[n]_dc_voltage`
V_{dc}	**Inverter dc input voltage**	`hourly_inverter_dc_voltage`
$P_{dc,gross,n}$	**Subarray [n] Gross DC output**	`hourly_subarray[n]_dc_gross`
$P_{dc,net,n}$	**Subarray [n] Net DC output**	`hourly_subarray[n]_dc_net`
$P_{dc,gross}$	**Gross dc array output**	`hourly_dc_gross`
$P_{dc,net}$	**Net dc array output**	`hourly_dc_net`

10.3 DC Losses

SAM models electrical losses on the DC side of the system using a single DC loss factor for each subarray in the system. In SAM, the DC loss factor for each subarray is calculated from the five loss percentages on the Losses input page:

$$L_{dc} = L_1\ L_2\ L_3\ L_4\ L_5$$

$$L_1 = 100 \quad 1 - \frac{L_{mismatch}}{100}$$

$$L_2 = 100 \quad 1 - \frac{L_{diodesconnections}}{100}$$

$$L_3 = 100 \quad 1 - \frac{L_{dcwiring}}{100}$$

$$L_4 = 100 \quad 1 - \frac{L_{trackingerror}}{100}$$

$$L_5 = 100 \quad 1 - \frac{L_{nameplate}}{100} \tag{10.5}$$

The DC loss factor for each subarray n:

$$F_{dc,n} = 1 - \frac{L_{dc,n}}{100} \tag{10.6}$$

In the SSC `pvsamv1` module, each subarray's DC loss is stored as a single value `subarray[n]_dcloss`. There are no loss categories in SSC: The categories shown in Equation 10.5 are part of SAM's user interface.

10.4 Net DC Power Output

The net DC power output of each subarray is each subarray's gross DC power output multiplied by the subarray's DC loss factor:

$$P_{dc,net,n} = P_{dc,gross,n}\ F_{dc,n} \tag{10.7}$$

The array's net DC power output is sum of the subarray gross DC power output values :

$$P_{dc,net} = \sum_{n=1}^{N_{sub}} P_{dc,net,n} \tag{10.8}$$

The array's net DC power output is the inverter's DC input power.

10.5 Maximum Power Point Tracking

SAM does not explicitly model a maximum power point tracker or other power conditioning equipment. However, SAM does assume that the array operates at its maximum power point, so it implicitly assumes that photovoltaic system is equipped with a maximum power point tracking system.

You can account for electricity consumption of maximum power point tracking equipment and other related losses using the DC losses (see Section 10.3).

10.6 Subarray Mismatch Losses

For a system with more than one subarray, SAM can estimate losses due to maximum power point mismatches between subarrays, but only with the CEC module submodel (either with database parameters or user-defined inputs, see Section 9.5). For the Sandia module model and simple efficiency module model, SAM does not estimate subarray mismatch losses as explained in Section 10.1.

The subarray mismatch loss algorithm (Dobos 2012b) is intended to model situations that may occur in residential rooftop systems when the array is installed on different parts of the roof with different orientations and connected to a single inverter. The group of modules on each roof surface is a subarray, and may have its own set of parameters defining its tilt and azimuth angles, tracking, shading factors, and DC losses.

In larger systems, modules are typically oriented uniformly across the entire array. SAM's subarray mismatch algorithm is not suitable for such systems. SAM does not calculate mismatch losses due to maximum power point losses between individual modules in the array or in each subarray.

The algorithm determines the system's maximum power point in a given hourly time step by running the CEC module model over a range of maximum power point voltage values to find the voltage that results in the complete array's highest maximum power given each of its subarray's effective incident irradiance, module performance and temperature parameters.

The subarray mismatch algorithm runs the cell temperature submodel (Section 9.2) and CEC module submodel separately from the submodel runs for the main system energy calculations. Therefore, in addition to the input variables listed in Table 19, the algorithm also uses the inputs listed in Tables 9 and 13.

11 Inverter AC Output

SAM uses one of two inverter submodels (Section 11.1) to calculate the total AC power output of all of the inverters in the system. For systems with more than one inverter, SAM models the inverters as a single large inverter that operates with the array DC string voltage as the inverter's input voltage.

The inverter submodels calculate the inverter's DC-to-AC power conversion efficiency, and account for inverter saturation and clipping. They do not explicitly account for the effect of temperature on inverter performance or for the impact on inverter performance of power factor control or grid outages.

Regardless of the inverter submodel, SAM calculates the inverter input power for each hourly time step by dividing the array's total DC power output by the number of inverters in the system:

$$P_{dc} = \frac{P_{dc,net}}{N} \tag{11.1}$$

11.1 Inverter Submodel Options

SAM calculates the a single photovoltaic module's DC output with one of the module submodels listed below, with descriptions from SAM Help - Module (2014) and Blair (2013):

- **Inverter CEC Database** (Section 11.2), also called the Sandia inverter model, is an empirical model that uses manufacturer specifications with four empirically derived coefficients C_0, C_1, C_2, C_3 (King 2007). It uses parameters from a database maintained by the California Energy Commission (Go Solar California 2014a).

- **Inverter Datasheet** (Section 11.2) is an implementation of the Sandia inverter model that sets the values of the empirical coefficients to zero so that the inverter can be modeled with only manufacturer specifications.

- **Inverter Part Load Curve**) (Section 11.3) is a model developed by NREL for SAM that uses a table of efficiency values at different inverter load levels to represent the inverter's performance.

In the SSC `pvsamv1` module, the input variable `inverter_model` determines the module submodel as shown in Table 21.

11.2 Sandia Inverter Submodel

SAM's Sandia Inverter model is an implementation of the empirical model described in King (2007). As explained in Section 11.1, SAM has two implementations of the Sandia inverter model: One uses a set of reference parameters from the Sandia Inverters library with data provided by the California Energy Commission (Go Solar California 2014a), and the other uses manufacturer specifications without the additional C coefficients included in the library. Both versions are described in this section.

In SSC, the implementation of the Sandia inverter model described here is part of the `pvsamv1` module. The SSC module `pvsandiainv` is a standalone implementation of the Sandia inverter model that is not used by SAM, but is available as part of the SSC software development kit.

For the **Inverter Datasheet** option (`inverter_model = 1` in SSC), there is an additional input for the inverter's nominal DC-to-AC power conversion efficiency $\eta_{inv,0}$. SAM calculates the maximum DC input power from the rated efficiency and rated maximum AC output power values:

$$P_{dc,0} = \frac{P_{ac,0}}{\eta_{inv,0}} \tag{11.2}$$

Table 20. Inverter Submodel Variable Definitions

Symbol	Description / **Name in SAM**	Name in the SSC Module `pvsamv1`
	Inputs	
$P_{dc,net}$	net DC power output of the array	
N	number of inverters	-
V_{dc}	DC string voltage	-
	Output	
P_{ac}	**Gross ac output**	`hourly_ac_net`
η_{inv}	**Inverter efficiency**	`hourly_inv_eff`
P_{clip}	**Inverter clipping loss**	`hourly_inv_cliploss`
P_{so}	**Inverter power consumption loss**	`hourly_inv_psoloss`
P_{nt}	**Inverter night time loss**	`hourly_inv_ntloss`
P_{par}	AC parasitic power consumption (W)	-
$R_{partload}$	part load ratio	-

Table 21. Inverter Submodels in SSC

Name in SAM	`inverter_model`
Inverter CEC Database (Sandia)	0
Inverter Datasheet	1
Inverter Part Load Curve	2

Table 22. Sandia Inverter Submodel Variable Definitions

Symbol	Description / **Name in SAM**	Name in SSC
	Inputs	
P_{dc}	inverter DC input power (Equation 11.1)	-
V_{dc}	inverter DC input voltage (DC string voltage)	-
$V_{dc,max}$	**Maximum DC power**	`inv_snl_vdcmax`
$I_{dc,0}$	**Nominal DC voltage**	`inv_snl_vdco`
$P_{s,0}$	**Power consumption during operation**	`inv_snl_pso`
$P_{nt,0}$	**Power consumption at night**	`inv_snl_pnt`
$P_{ac,0}$	**Maximum AC power**	`inv_snl_paco`
$P_{dc,0}$	**Maximum DC power**	`inv_snl_pdco`
c_0	Curvature between AC power and DC power W^{-1}	`inv_snl_c0`
c_1	Coefficient of $P_dc,0$ variation with DC input voltage V^{-1}	`inv_snl_c1`
c_2	Coefficient of P_{so} variation with DC input voltage V^{-1}	`inv_snl_c2`
c_3	Coefficient of C_0 variation with DC input voltage V^{-1}	`inv_snl_c3`
	Output	
P_{ac}	**Gross ac output**	`hourly_ac_net`
η_{inv}	**Inverter efficiency**	`hourly_inv_eff`
P_{clip}	**Inverter clipping loss**	`hourly_inv_cliploss`
P_s	**Inverter power consumption loss**	`hourly_inv_psoloss`
P_{nt}	**Inverter night time loss**	`hourly_inv_ntloss`

The Sandia inverter model parameters:

$$A = P_{dc,0} \; 1 + C_1(V_{dc} - V_{dc,0}$$
$$B = P_{s,0} \; 1 + C_2(V_{dc} - V_{dc,0}$$
$$C = C_0 \; 1 + C_3(V_{dc} - V_{dc,0} \tag{11.3}$$

The Sandia inverter model equation:

$$P_{ac} = \frac{P_{ac,0}}{A - B} - C(A - B) \; (P_{dc} - B) + C(P_{dc} - B)^2 \tag{11.4}$$

For the special case of the model that is the **Inverter Datasheet** option in SAM (`inverter_model = 1` in SSC), the Sandia model C coefficients are all set to zero so that Equation 11.4 reduces to:

$$P_{ac} = \frac{P_{ac,0}}{P_{dc,0} - P_{s,0}} \; (P_{dc} - P_{so,0})^2 \tag{11.5}$$

The coefficient B accounts for operating power loss. By setting $B = 0$ in Equation 11.4, the operating power less operating power losses is:

$$P_{ac,s=0} = \frac{P_{ac,0}}{A} - CA \; P_{dc} + CP_{dc}^2 \tag{11.6}$$

The operating power losses (inverter self-consumption):

$$P_s = \begin{cases} P_{ac,s=0} - P_{ac} & \textbf{(Inverter CEC Database} \text{ option, and} \\ & \textbf{Inverter Datasheet} \text{ option with manufacturer efficiency)} \\ 0 & \textbf{(Inverter Datasheet} \text{ option with weighted efficiency)} \end{cases} \tag{11.7}$$

For the **Inverter Datasheet** option, the operating losses are zero:

SAM considers the inverter to be operating at night when the DC input power is less than the operating power losses $P_{dc} < P_{so}$. At night:

$$P_{ac} = -P_{nt,0}$$
$$P_{nt} = P_{nt,0} \tag{11.8}$$

When the inverter's output power exceeds the inverter's rated capacity (maximum AC power) $P_{ac} > P_{ac,0}$, SAM clips the inverter's output to the rated capacity and records the remaining power as the clipping loss:

$$P_{ac,noclip} = P_{ac}$$
$$P_{ac} = P_{ac,0}$$
$$P_{clip} = P_{ac,noclip} - P_{ac} \tag{11.9}$$

The inverter's DC-to-AC power conversion efficiency:

$$\eta_{inv} = \frac{P_{ac}}{P_{dc}} \tag{11.10}$$

Table 23. Inverter Part Load Curve Submodel Variable Definitions

Symbol	Description / **Name in SAM**	Name in SSC
	Inputs	
P_{dc}	inverter DC input power (Equation 11.1)	-
V_{dc}	inverter DC input voltage (DC string voltage)	-
$V_{dc,max}$	**Maximum DC power**	`inv_ds_vdcmax`
$I_{dc,0}$	**Nominal DC voltage**	`inv_ds_vdco`
$P_{s,0}$	**Power consumption during operation**	`inv_ds_pso`
$P_{nt,0}$	**Power consumption at night**	`inv_ds_pnt`
$P_{ac,0}$	**Maximum AC power**	`inv_ds_paco`
$P_{dc,0}$	**Maximum DC power**	`inv_ds_pdco`
$\eta_{inv,0-N}$	column of N efficiency values	`inv_ds_efficiency`
$P_{ac,0-N}$	column of N AC power output values	`inv_ds_partload`
	Output	
P_{ac}	**Gross ac output**	`hourly_ac_net`
η_{inv}	**Inverter efficiency**	`hourly_inv_eff`
P_{clip}	**Inverter clipping loss**	`hourly_inv_cliploss`
P_s	**Inverter power consumption loss**	`hourly_inv_psoloss`
P_{nt}	**Inverter night time loss**	`hourly_inv_ntloss`

11.3 Inverter Part Load Curve Submodel

NREL developed the Inverter Part Load Curve model specifically for SAM. The model's input and output variables are shown in Table 23.

The part-load ratio for the current hour depends on the inverter's DC input power and its maximum rated DC power:

$$x = \frac{P_{dc}}{P_{dc,0}} \tag{11.11}$$

The model then uses linear interpolation to calculate the inverter's DC-to-AC conversion efficiency η_{inv} at the DC input power.

The inverter's AC power output for hours when $P_{dc} > 0$:

$$P_{ac} = \begin{cases} \eta_{inv}P_{dc} & \text{if } \eta_{inv}P_{dc} \leq P_{ac,0} \\ P_{ac,0} & \text{if } \eta_{inv}P_{dc} > P_{ac,0} \end{cases} \tag{11.12}$$

For hours when the AC power is greater than the maximum rated value, the clipping loss value is:

$$P_{clip} = P_{ac} - P_{ac,0} \tag{11.13}$$

At night when $P_{dc} \leq 0$:

$$P_{ac} = -P_{nt,0}$$
$$P_{nt} = P_{nt,0} \tag{11.14}$$

12 System AC Output

The system AC output is the electricity generated by the photovoltaic system and delivered to either the grid, a building load, or both.

12.1 Gross AC Power Output

The gross AC power output is the inverter output before the AC and other losses:

$$P_{ac,gross} = N_{inv} \, P_{ac} \tag{12.1}$$

12.2 AC Losses

SAM models electrical losses on the AC side of the system using a single AC loss factor. In SAM, the AC loss factor is calculated from the two loss percentages on the Losses input page:

$$L_{ac} = L_1 \, L_2$$

$$L_1 = 100 \quad 1 - \frac{L_{acwiring}}{100}$$

$$L_2 = 100 \quad 1 - \frac{L_{transformer}}{100} \tag{12.2}$$

The AC loss factor:

$$F_{ac} = 1 - \frac{L_{ac}}{100} \tag{12.3}$$

In the SSC `pvsamv1` module, the AC loss is the single value `ac_derate`. There are no loss categories in SSC: The categories shown in Equation 12.2 are part of SAM's user interface.

12.3 Net AC Power Output

The net AC power output is the gross AC output multiplied by the AC loss factor:

$$P_{ac,net} = P_{ac,gross} \, F_{ac} \tag{12.4}$$

Table 24. System AC Output Variable Definitions

Symbol	Description / **Name in SAM**	Name in SSC
	Inputs	
N_{inv}	**Number of inverters**	`inverter_count`
P_{ac}	AC output of a single inverter	-
L_{ac}	**Total AC power loss**	`ac_loss`
	Outputs	
$P_{ac,gross}$	**Gross ac output**	`hourly_ac_gross`
$P_{ac,net}$	**Net ac output**	`hourly_ac_net`

12.4 Hourly Energy

SAM uses the **Hourly Energy** result variable for all of the performance models to represent the electricity generated by the renewable energy system (or, in the case of a solar hot water system, the energy saved by the system) after all losses and adjustments (SAM Help - Hourly Energy 2014). For the photovoltaic performance model, **Hourly Energy** (`hourly_e_net_delivered` in SSC `annualoutput`) is equal to **Net ac output** less any adjustments specified on the Performance Adjustment input page.

For a photovoltaic project that uses one of the Utility IPP financial models, hourly energy is the electricity delivered to the grid and sold at a negotiated power price. For projects with either the residential or commercial financing options, hourly energy is the electricity generated by the photovoltaic system that displaces purchases of electricity at a retail rate.

References

Appelbaum, J.; Bany, J. "Shadow effect of adjacent solar collectors in large scale systems." *Solar Energy* (23) 1979; pp. 497-507.

Blair, N.; Dobos, A.; Gilman, P. (April 2013). "Comparison of Photovoltaic Models in the System Advisor Model." Preprint. Prepared for American Solar Energy Society National Solar Conference Solar 2013, April 16-20, 2013. NREL/CP-6A20-58057. Golden, CO: National Renewable Energy Laboratory, 6 pp. Accessed February 27, 2014: `http://www.nrel.gov/docs/fy13osti/58057.pdf`

De Soto, W.; Klein, S.; Beckman, W. (2004) "Improvement and Validation of a Model for Photovoltaic Array Performance." *Solar Energy* (80:1); pp. 78-88.

De Soto, W. (2004). "Improvement and Validation of a Model for Photovoltaic Array Performance." University of Wisconsin-Madison. Accessed February 26, 2014: `http://sel.me.wisc.edu/publications/theses/desoto04.zip`

Deline, C.; Dobos, A.; Janzou, S.; Meydbrey, J.; Donoval, M. (2013). "A Simplified Model of Uniform Shading in Large Photovoltaic Arrays." *Solar Energy* (96); pp. 274-282. `http://www.sciencedirect.com/science/article/pii/S0038092X13002739`

Deline, C. (2013). "SAM Shade Geometry V3." Unpublished. Golden, CO: National Renewable Energy Laboratory.

Dobos, A. (2012). "An Improved Coefficient Calculator for the California Energy Commission 6 Parameter Photovoltaic Module Model." *Journal of Solar Energy Engineering* (134:2).

Dobos, A. (June 2012). "Modeling of Annual DC Energy Losses due to Off Maximum Power Point Operation in PV Arrays." Prepared for 38th IEEE Photovoltaic Specialists Conference (PVSC '12). 3 pp. Accessed March 17, 2014: `http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6318207`

Dobos, A. (2013). "PVWatts Version 1 Technical Reference." TP-6A20-60272. Golden, CO: National Renewable Energy Laboratory. Accessed February 20, 2014. `http://www.nrel.gov/docs/fy14osti/60272.pdf`

Dobos, A. (2013). "5-Parameter PV Module Model." Presented at the 2013 Sandia PV Performance Modeling Workshop on May 1, 2013. Accessed March 4, 2014: `http://pvpmc.org/home/2013-pv-performance-modeling-workshop/`

Duffie, J.; Beckman, W. (2013). *Solar Engineering of Thermal Processes, 4th ed*. New York, NY: Wiley.

Dunlap, J. (2007). *Photovoltaic Systems*. Homewood, IL: American Technical Publishers.

"EnergyPlus Energy Simulation Software: Weather Data." U.S. Department of Energy. Accessed March 19, 2014: `http://apps1.eere.energy.gov/buildings/energyplus/weatherdata_about.cfm`

Goswami, D.; Stefanakos, E.; Hassan A.; Collis,W. (1989). "Effect of Row-to-Row Shading on the Output of Flat-Plate South-Facing Photovoltaic Arrays." *Journal of Solar Energy Engineering* (111:3); pp. 257-259.

"Incentive Eligible Photovoltaic Modules in Compliance with SB1 Guidelines." (2014). Go Solar California. Accessed March 4, 2014: `http://www.gosolarcalifornia.ca.gov/equipment/pv_modules.php`

Iqbal, M. (1983) *An Introduction to Solar Radiation*. New York, NY: Academic Press.

King, D.; Boyson, W.; and Kratochvil, J. (2004). "Photovoltaic Array Performance Model." 41 pp.; Albuquerque, NM: Sandia National Laboratories. SAND2004-3535. Accessed February 22, 2014: `http://prod.sandia.gov/techlib/access-control.cgi/2004/043535.pdf`

King, D.; Gonzalez, S.; Galbraith, G.; Boyson, W. (2007). "Performance Model for Grid-Connected Photovoltaic Inverters." 47 pp.; Albuquerque, NM: Sandia National Laboratories. SAND2007-5036. Accessed March 7, 2014: `http://prod.sandia.gov/techlib/access-control.cgi/2007/075036.pdf`

Liu, B.; Jordan, R. (1963). "A Rational Procedure for Predicting The Long-term Average Performance of Flat-plate Solar-energy Collectors." *Solar Energy* (7:2); pp. 53-74.

Michalsky, J. (1988). "The Astronomical Almanac's Algorithm for Approximate Solar Position (1950-2050)." *Solar Energy* (40:3); pp. 227-235.

Neises, T. (2011). "Development and Validation of a Model to Predict the Temperature of a Photovoltaic Cell." University of Wisconsin-Madison. Accessed March 3, 2014: `http://sel.me.wisc.edu/publications/theses/neises11.zip`

"New Solar Homes Partnership Calculator." (2014). Go Solar California. Accessed March 4, 2014: `http://www.gosolarcalifornia.ca.gov/tools/nshpcalculator/index.php`

"Incentive Eligible Photovoltaic Modules in Compliance with SB1 Guidelines." (2014). Go Solar California. Accessed March 19, 2014: `http://www.gosolarcalifornia.ca.gov/equipment/pv_modules.php`

"List of Eligible Inverters per SB1 Guidelines." (2014). Go Solar California. Accessed March 19, 2014: `http://www.gosolarcalifornia.ca.gov/equipment/pv_inverters.php`

"National Solar Radiation Database (NSRDB)." (2014). National Renewable Energy Laboratory. Accessed November 13, 2013. `http://rredc.nrel.gov/solar/old_data/nsrdb/`.

"National Solar Radiation Database (NSRDB) 1991 - 2005 Update: Typical Meteorological Year 3." (2014). National Renewable Energy Laboratory. Accessed November 13, 2013. `http://rredc.nrel.gov/solar/old_data/nsrdb/1991-2005/tmy3/`.

O'Brien, J. (2012)."Position of the sun given time of day, latitude and longitude." *stackoverflow.com*. Accessed February 13, 2014. `http://stackoverflow.com/questions/8708048/position-of-the-sun-given-time-of-day-latitude-and-longitude`

"PVCDROM." (2014) pveducation.org. Accessed February 26, 2014. `http://www.pveducation.org/pvcdrom`

"PVPMC: Modeling Steps." (2014) PV Performance Modeling Collaborative. Accessed November 13, 2013. `http://pvpmc.org/modeling-steps/`.

"PVPMC: Irradiance and Insolation." (2014). PV Performance Modeling Collaborative. Accessed November 14, 2013. `http://pvpmc.org/modeling-steps/irradiance-and-weather-2/irradiance-and-insolation/`

Passias, D.; Kallback, B. (1984). "Shading effects in rows of solar cell panels", Solar Cells (11); pp. 281-291.

Perez, R.; Stewart, R.; Seals, R.; Guertin, T. (1988) "The Development and Verification of the Perez Diffuse Radiation Model." SAN88-7030. Albuquerque, NM: Sandia National Laboratories. Accessed February 20, 2014: `http://prod.sandia.gov/techlib/access-control.cgi/1988/887030.pdf`

Perez, R.; Ineichen, P.; Seals, R.; Michalsky, J.; Stewart, R. (1990) "Modeling Daylight Availability and Irradiance Components from Direct and Global Irradiance." *Solar Energy* (44:5); pp. 271-289.

"Perez Sky Diffuse Model." *Modeling Steps*. PV Performance Modeling Collaborative. Albuquerque, NM: Sandia National Laboratories. Accessed February 20, 2014: `http://pvpmc.org/modeling-steps/incident-irradiance/plane-of-array-poa-irradiance/calculating-poa-irradiance/poa-sky-diffuse/perez-sky-diffuse-model/`

Reindl, D. (1988). "Estimating Diffuse Radiation on Horizontal Surfaces and Total Radiation on Tilted Surfaces." Masters Thesis. University of Wisconsin Madison, Solar Energy Laboratory. `http://sel.me.wisc.edu/publications/theses/reindl88.zip`

"SAM Help: Hourly Energy." (2014) SAM 2014.1.14 Help System. National Renewable Energy Laboratory. Available on line at `https://www.nrel.gov/analysis/sam/help/html-php/index.html?mtp_annual_energy.htm` (accessed March 18, 2014).

"SAM Help: Libraries." (2014) SAM 2014.1.14 Help System. National Renewable Energy Laboratory. Available on line at `https://www.nrel.gov/analysis/sam/help/html-php/index.html?libraries.htm` (accessed February 26, 2014).

"SAM Help: Module." (2014). SAM 2014.1.14 Help System. National Renewable Energy Laboratory. Available on line at `https://www.nrel.gov/analysis/sam/help/html-php/index.html?pv_module.htm` (accessed February 26, 2014).

"SAM Help: Performance Adjustment." (2014). SAM 2014.1.14 Help System. National Renewable Energy Laboratory. Available on line at `https://www.nrel.gov/analysis/sam/help/html-php/index.html?fin_annual_performance.htm` (accessed March 18, 2014).

"SAM Help: Shading." (2014). SAM 2014.1.14 Help System. National Renewable Energy Laboratory. Available on line at `https://www.nrel.gov/analysis/sam/help/html-php/index.html?pv_shading.htm` (accessed February 26, 2014).

"SAM Help: Sizing the Flat Plate PV System." (2014). SAM 2014.1.14 Help System. National Renewable Energy Laboratory. Available on line at `https://www.nrel.gov/analysis/sam/help/html-php/index.html?pvfp_sizing_the_pv_system.htm` (accessed March 20, 2014).

"SAM Simulation Core SDK." (2014). National Renewable Energy Laboratory. Accessed March 19, 2014: `https://sam.nrel.gov/content/sam-simulation-core-sdk`

"SAM Help: Weather File Formats." (2014) SAM 2014.1.14 Help System. National Renewable Energy Laboratory. Available on line at `https://www.nrel.gov/analysis/sam/help/html-php/index.html?weather_format.htm` (accessed November 13, 2013).

"Sandia National Laboratories Photovoltaic Test and Evaluation Program." Sandia National Laboratories. Accessed March 7, 2014: `http://energy.sandia.gov/?page_id=279`

"Solar Prospector." National Renewable Energy Laboratory. Accessed March 3, 2015: `http://maps.nrel.gov/prospector`

Van Schalkwijk, M.; Kil, T.; Van der Weiden, J. (1991). "Dependence of diffuse light blocking on the ground cover ratio for stationary PV arrays", Solar Energy (61:1); pp. 381-387.

Wilcox, S.; Marion, W. (2008). "Users Manual for TMY3 Datasets." TP-581-43156. Golden, CO: National Renewable Energy Laboratory. Accessed November 13, 2013. `http://www.nrel.gov/docs/fy08osti/43156.pdf`.