Arizona State University

School of Electrical, Computer and Energy Engineering

EEE 465 591: Photovoltaic Energy Conversion

Mini Project #2: Solar Cell Simulator

Overview of how this mini project fits into the goal of PV system design

The goal of the series of mini-projects is to build a model to allow us to calculate and optimize a PV system. In the first project, we re-acquainted everyone with programming, and calculated solar power density and energy on a surface with arbitrary tilt and azimuth. In this project, we will calculate and optimize the efficiency of a solar cell. Mini Project 3 will calculate the power from a module; Project 4 calculates the PV system parameters.

Overview

A solar cell is characterized by the equation:

$$J = J_0\left(e^{\frac{(V-JR_{series})}{kT}} - 1\right) - J_L$$

In order to calculate its performance, we need to (1) Calculate J_0; (2) Calculate J_L; (3) Calculate R_{series}. Then from these, we can calculate the parameters that define the efficiency. The equations for all of these parameters are in the lecture notes, and also in the more detailed section descriptions below.

$$\text{efficiency} = \frac{V_{oc}J_{sc}FF}{P_{in}}$$

We divide the assignment into five parts:

Part 1: Read in material and solar cell device parameters.

Part 2: Calculate the IV curve parameters: J_0, J_L and R_{series}. As part of this we also calculate the quantum efficiency.

Part 3: Calculate the efficiency parameters V_{OC}, J_{SC}, FF and P_{in} and calculate efficiency.

Part 4: Vary parameters to optimize efficiency.

Part 5: Use your Solar Cell Simulator to Analyze a solar cell.

Part 1: Read in or specify input data

Set up your program to read in or acquire the material and device parameters. "Hardwiring" them in the code is OK. The default values for Silicon (Si) are in the table below.

	Name in code	Emitter Value		Base Value		Units
MATERIAL PARAMETERS ASSUMED INDEPENDENT OF DOPING						
N-p or p-n junction		n-type				
Intrinsic carrier concentration (n_i)	ni	8.6×10^9 cm^{-3} or 1×10^{10} (first is more accurate, second is more commonly used.)				cm^{-3}
Minority carrier lifetime τ_0		τ_0 = 1; calculate τ_{eff} based off of this value and parameters given below in table				msec
Absorption coefficient $\alpha(\lambda)$	ab	Absorption coefficient and AM1.5G spectrum as a function of λ in "10_nm_AM15G_absorption.txt"				cm^{-1}
EMITTER AND BASE PARANETERS						
Diffusivity (D)	D_E D_B	15		30		cm^2/s
Doping (N)	N_E N_B	1×10^{18}		5×10^{16}		cm^{-3}
Surface Recombination (S)	S_E, S_B	500		1000		cm/sec
Thickness (x_j, H)	W_E, W_B	1		300		Microns or cm
CALCULATED PARAMETERS						
Diffusion Length (L)	L_E or L_B	$L = \sqrt{D\tau_{eff}}$				Microns or cm
Mobility (μ)		$\frac{D}{\mu} = \frac{kT}{q}$				$cm^2/V.s$
Minority carrier lifetime		$\frac{1}{\tau_{eff}} = C_{Auger}n^2 + \frac{1}{\tau_0}$		$C_{Auger} = 1.5 \times 10^{-30}$		cm^6/s

****** WATCH OUT FOR THE UNITS ******

The biggest mistake (by far) I see in debugging is unit errors, which admittedly is not helped by the semiconductor convention of specifying thickness in microns, and all the other parameters in units that use "cm". If you get strange numbers, first check unit consistency.

Output:

(1) Plot the minority carrier lifetime as a function of doping for the n-type emitter – use a log plot for the x-axis doping and y-axis lifetime. LABEL UNITS!!

(2) Plot the absorption coefficient as a function of wavelength, labelling units and using log plot for α.

(3) Print the emitter, base and material parameters in a readable format and compare to tabulated values.

Part 2: Calculations IV curve parameters: J_0, J_L and Rseries.

1. Calculate J_0 for a device with the test solar cell parameters and compare with the values in Appendix A: Output of the Python script for the test case. The variation generally comes from two sources: accuracy (significant figures) in physical constants; and variation in material parameters. For example, the most commonly used value of n_i in Si is 1×10^{10} cm^{-3}, but an updated value is 8.6×10^9 cm^{-3}. You can use either.

   For reference, the J_0 equation for electrons in p-type material is repeated from the notes below; n-type is the same, expect all hole carrier parameters are used and N_A is replaced by N_D.

$$J_{0n} = q\frac{D_n}{L_n}\frac{n_i^2}{N_A}\left[\frac{\frac{S_n L_n}{D_n}\cosh\left(\frac{H}{L_n}\right)+\sinh\left(\frac{H}{L_n}\right)}{\frac{S_n L_n}{D_n}\sinh\left(\frac{H}{L_n}\right)+\cosh\left(\frac{H}{L_n}\right)}\right]$$

   Output: **Print the emitter, base and total J_0 for the test solar cell and compare to the given value in Appendix A.**

2. Calculate J_L.

   Calculating J_L is conceptually straight forward (although computationally long). It is done by:

   a. Calculating the number of photons/m² in each wavelength band in the solar spectrum – this is called photon flux.
   b. Calculating the quantum efficiency at each wavelength, which physically corresponds to the fraction of incident photons converted to current at each wavelength.
   c. Multiplying to two above and adding them all up over all the wavelengths.

   More details on each of the a,b,c parts from above:

   a. Calculate the photon flux at each wavelength.

      For this, we need to read in the standard solar spectrum (which is the spectral irradiance or the POWER DENSITY per wavelength range). Then we convert it to the number of photons at each wavelength (photon flux) by

$$\Phi_i = Spectral\ Irradiance_i \times [\Delta\lambda]\ \times \frac{\lambda_i}{hc}$$

      Where "i" denotes the wavelength at which the spectral irradiance is given. Please, make sure that you calculate the photon energy $\frac{hc}{\lambda_i}$ in Joules. If you use the expression $\frac{1240}{\lambda_i\,(nm)}$, you need to multiply the energy in eV by the electron charge q. $[\Delta\lambda]$ is the wavelength interval the spectral irradiance corresponds to. However, you need to be careful if the spectral irradiance has already been normalized to 1 nm bin width or already been multiplied by $[\Delta\lambda]$. You need to look at the file header or other information on the units of the spectral irradiance. If you are uncertain, sum the spectral irradiance and compare it to the value of approximately 840 W/m^2, which is the integral of the spectral irradiance over the wavelength range.

      In Mini Project 1, you read in a file, did a calculation and printed it out, so this is the same except the file you read is the AM1.5G solar spectrum and the calculation is the equation above.

   b. Calculate and plot the quantum efficiency as a function of wavelength in both the emitter and the base of the solar cell as a function of wavelength. The quantum efficiency equations are calculated as functions in the python library called IQE_emitter(ab, We, Le, De, Se)

and IQE_base(ab, We_Wd, Wb, Lb, Db, Sb). The equations are given in the lecture notes and also repeated below for convenience.

The parameter "α" is the absorption coefficient and is dependent on wavelength (we omitted the (λ) after the α to keep the equations to one line. In the function call in the code it is an array called "ab". α(λ) is in a file called "10_nm_AM15G_absorption.txt", which you read into your program in Part 1.

*** We do not use Wd, the width of the depletion region, so use Wd = 0.

Output: The two plots of QE_emitter and QE_base are typically combined into one plot with two lines, so please plot it this way.

$$QE_{\_emit}(\lambda) = \frac{\alpha L_{em}}{(\alpha^2 L_{em}^2 - 1)}\left[\frac{\left(\frac{S_{em}L_{em}}{D_{em}} + \alpha L_{em}\right) - e^{-\alpha x_j}\left(\frac{S_{em}L_{em}}{D_{em}}\cosh\left(\frac{x_j}{L_{em}}\right) + \sinh\left(\frac{x_j}{L_{em}}\right)\right)}{\frac{S_{em}L_{em}}{D_{em}}\sinh\left(\frac{x_j}{L_{em}}\right) + \cosh\left(\frac{x_j}{L_{em}}\right)} - \alpha L_{em} e^{-\alpha x_j}\right]$$

$$QE_{\_base}(\lambda) = \frac{\alpha L_b e^{-\alpha x_j'}}{(\alpha^2 L_b^2 - 1)}\left[\alpha L_b - \frac{\frac{S_b L_b}{D_b}\left(\cosh\left(\frac{H}{L_b}\right) - e^{-\alpha H}\right) + \sinh\left(\frac{H}{L_b}\right) + \alpha L_b e^{-\alpha H}}{\frac{S_b L_b}{D_b}\sinh\left(\frac{H}{L_b}\right) + \cosh\left(\frac{H}{L_b}\right)}\right]$$

For Matlab users, copy the equation for these in the definitions into Matlab and define a function around it. One thing you have to change in the equation for Matlab (or other programming languages) is how you call the hyperbolic functions; in python it is np.cosh, so change this for another programming language.

c. Calculate J_L. To find the J_L, we multiply the photon flux (part a) by the quantum efficiency (part b), and then sum over all the wavelengths.

$$J_{Li} = Spectral\ Irradiance_i \times [\Delta\lambda] \times \frac{\lambda_i}{hc} \times q \times QE_i = \Phi_i \times q \times QE_i$$

Output: Plot J_{Li} as a function of wavelength and in the graph title give the total J_L in mA/cm^2.

Note 1: In some textbooks or calculations, you may see a similar equation as above in terms of a parameter called spectral response; this is the same basic equation, just calling the last two terms in the equation above the spectral response:

$$Spectral\ response_i = q\ \frac{\lambda_i}{hc} \times QE_i$$

To get:

$$J_{sc_i} = J_{L_i} = Spectral\ Irradiance_i \times Spectral\ response_i$$

Note 2: if you want to use other absorption coefficients (e.g, you choose to do a different solar cell material), the wavelength at which the absorption coefficient is defined must be

the same wavelength at which the spectral irradiance is defined. Since the absorption coefficient and spectral irradiance don't in general have the same wavelength spacing, we have to convert either the absorption or the spectral irradiance to the same wavelength interval. We did this already for the files we handed out.

Output: **Print the emitter, base and total J_L for the test case.**

3. **Calculate the emitter series resistance**. In the notes, we calculate multiple terms of the series resistance; for simplicity, here we only calculate the emitter series resistance. The area-normalized series resistance is given by the equation:

$$R_{ser} \approx R_{emitter} = \rho_{\square}\frac{S^2}{12}$$

where S is the finger spacing (**NOT** the surface recombination velocity – semiconductor notation is terrible!) and $\rho_{\square}$ is the sheet resistance of the emitter calculated by the equation:

$$\rho_{\square} = \frac{\rho}{W_e} = \frac{1}{q\mu N_e}\frac{1}{W_e}$$

where We is the emitter thickness, N is the emitter doping, and μ is the mobility in the emitter.

We have set S = 0.2 cm corresponding to a typical value for a commercial silicon grid. If you want to see the optimization of the top metallization from which we get this value, it is at: http://www.pveducation.org/pvcdrom/5-design-of-silicon-cells/optimization-of-finger-spacing.

The python library has two function definitions which calculate the sheet resistivity and emitter series resistance for you. They are: *def emitter_resistance(Sf, Rsheet):* and *def sheet_resistivity(doping, thickness):*

Output:

(1) Print out the series resistance (including units) and compare to test case.
(2) Plot the IV curve, labelling the graph.

Part 3: Calculate the efficiency parameters V_{OC}, J_{SC}, FF and P_{in} and calculate efficiency

(1) Calculate J_{SC} using $J_{SC} = J_L$. Unless the series resistance is very high (in which case we aren't using the solar cell anyway), the current at V=0 (which is how J_{SC} is defined) is the same as J_L.

Note: If you are bored or writing the code in a more general fashion, you can find J_{SC} from:

$$J_{SC} = J_0\left(e^{\frac{(J_{SC}R_{series})}{kT}} - 1\right) - J_L$$

This equation is solved recursively (you can't solve exactly in closed form), which is the reason we typically ignore it.

(2) Calculate Voc using:

$$Voc = \frac{kT}{q}ln\left(\frac{J_{SC}}{J_0}\right)$$

(3) Calculate FF of the IV curve.

The FF is defined as the voltage and current where the power (P=V x I) is the largest. We can find this value one of two ways:

- Trace out the IV curve and find at what values of V and I the power is the largest. Advantage: you need the full IV curve in another mini-project, so might as well do it

now. Disadvantage: Longer to compute (not important now but depending on what you are doing in your final project, it might be an issue).

- Use the closed form for FF from the two equations below:

$FF_0 = \frac{v_{oc} - \ln(v_{oc}+0.72)}{v_{oc}+1}$ where $v_{oc} = \frac{q}{nkT} V_{oc}$, n being the ideality factor and

$FF = FF_0\left(1 - \frac{R_{ser}}{R_{CH}}\right)$ where $R_{CH} = \frac{V_{MP}}{I_{MP}} \approx \frac{V_{OC}}{I_{SC}}$

(4) Calculate the efficiency. and the impact of the maximum power point & FF.

$$efficiency = \eta = \frac{J_{SC} V_{OC} FF}{P_{incident}}$$

Where P is the incident power density. Since we are using the AM1.5G spectrum, it is 1000 W/m² (or equivalently 0.1 W/cm²), but if you are generalizing the program, you should calculate it from the input spectrum.

Output: **Print Voc, Fill Factor and Efficiency.**

Part 4: Optimize the solar cell

The goal of this part of the problem is to optimize the solar cell, **changing only the emitter doping and emitter thickness.**

IMPORATANT: The **maximum emitter doping** you should use is $N_{emitter} = 1 \times 10^{20}$ cm^{-3}. Values above this will give higher efficiency in the calculations, but the equations for minority carrier lifetime do not adequately work for such high doping – in practice the minority carrier lifetime is so low at higher doping that the material is not particularly electrically active. The **minimum thickness of the emitter** should be 100 nm – below this in practice the solar cell is subject to shunting.

Output:

(1) Emitter doping and thickness, J_{SC}, V_{OC}, FF, and efficiency for the maximum efficiency.

(2) Justify that your answer is the optimum. You can do a plot of efficiency vs doping and thickness; you can describe how you optimized it and what changed and why it makes sense; or you can prove it's the optimum in other ways. **You cannot say "this is what my program spit out**". Basically, any time you give an optimum result for an engineering problem, you have to also show that your answer is reasonable and correct.

Note: your optimum may not be the same as someone else's given different values of n_i or number of data points, etc. Not that you should be comparing your answer to other people's, but we are not expecting a single answer (design problems rarely have a single correct answer).

Part 5: Analyze the losses in the solar cell

The goal of this part of the problem is to analyze what the losses are in the solar cell, and how we might improve the solar cell. Parameters you can look at are the quantum efficiency, compare J_0 from the emitter and from the base. You can plot different parameters (J_{SC}, V_{OC}) as function of doping and thickness. You can change a material parameter and see what its effect is.

Output:

Give the material and device parameters which are key in limiting the device efficiency. From this, suggest some approaches to increasing cell efficiency.

Appendix A: Output of the Python script for the test case.

This is just for you to check if your Python installation works or if your code in Excel or Matlab produces realistic values for the test case.

emitter diffusion length (cm) 0.00387104830569

base diffusion length (cm) 0.0925820099773

emitter resistivity (ohm/sqr) 208.0503042999924

series resistance (ohm.cm²) 0.6935010143333081

light generated current density in emitter (A/cm²) 0.012054716461

light generated current density in base (A/cm²) 0.0268994750224

total light generated current density (A/cm²) 0.0389541914834

J0 emitter (A/cm²) 7.08712167027e-15

J0 base (A/cm²) 1.32750778369e-13

J0 total (A/cm²) 1.39837900039e-13

Voc 0.677074345174

FF without series resistance 0.842761558955

FF with series resistance 0.806066612651

Efficiency (%) 21.2599131551