Macalester Journal of Physics and Astronomy

Volume 6
Issue 1 *Spring 2018*

Article 2

Modeling Recombination in Solar Cells

Paul Chery
Macalester College, pchery@macalester.edu

Abstract

Solar cells are a competitive alternative to nonrenewable energy sources such as fossil fuels. However, the efficiency of these devices is limited by photogenerated carrier recombination. We use a finite difference numerical model to study recombination phenomena in the absorber layer of solar cells including alternate recombination models and the effects of spatial distribution of recombination centers. We compare the effect of using the constant lifetime approximation for recombination to the full Shockley-Read-Hall expression in Silicon solar cells and find that the constant lifetime approximation holds for high defect densities but not for high photon flux densities. Finally, we simulate a defect layer in a thin film solar cell such as CdTe by varying the spatial distribution of defects. We find that this additional complication to the model is equivalent to using an average, constant defect density across the cell.

Modeling Recombination in Solar Cells

Paul Chery
Advisor: Professor James R. Doyle

In partial fulfillment of the
requirements for the degree of
Bachelors of Arts
in
Physics

Macalester College Physics Department
St. Paul, MN
May 6, 2018

Abstract

Solar cells are a competitive alternative to nonrenewable energy sources such as fossil fuels. However, the efficiency of these devices is limited by photogenerated carrier recombination. We use a finite difference numerical model to study recombination phenomena in the absorber layer of solar cells including alternate recombination models and the effects of spatial distribution of recombination centers. We compare the effect of using the constant lifetime approximation for recombination to the full Shockley-Read-Hall expression in silicon solar cells and find that the constant lifetime approximation holds for high defect densities but not for high photon flux densities. Finally, we simulate a defect layer in a thin film solar cell such as CdTe by varying the spatial distribution of defects. We find that this additional complication to the model is equivalent to using an average, constant defect density across the cell.

Contents

1 Introduction

Solar cells are opto-electronic devices that convert the energy of photons emitted by the sun into usable power. These devices are often built from materials called semiconductors. Semiconductors are special materials due to their electronic structure which consists of an energy bandgap, a band of energy states forbidden to electrons. Most electrons in equilibrium sit in the valence band, which is the most energetic band of energy below the bandgap. Once the semiconductor absorbs a photon that has an energy higher than the bandgap, an electron in the valence band is promoted to the energy level above the energy bandgap, which we call the conduction band. Electrons at this energy are free to move and they contribute to the current within the material. The empty state in the valence left by the promoted electron is called a hole and is a quasi-particle which also contributes to the current in the semiconductor. This is the generation mechanism that contributes charges to the current output by the solar cell.

Silicon solar cells, as well as many other types of solar cells, have a p-n junction structure. This structure consists of two layers of doped semiconductor: an n-doped layer with excess electrons and a p-doped layer with excess holes. The excess charge is generated by doping, or introducing impurities whose electronic structure leads to their ionization within the semiconductor lattice. At the interface of the two n and p layers, the ionized donor and acceptor impurities create a potential difference between the two layers. The electric field that is created sweeps charge carriers away from this region, which is why it is referred to as the space-charge or depletion region. This built in electric field is what drives charge separation and the current extracted from the cell. The electrons generated in the p region and holes generated in the n region diffuse to the junction where they are swept to the other side of the junction where they can be collected by the external circuit.

2 Background

2.1 Recombination

Recombination is an important loss mechanism in semiconductors. It occurs when electrons and holes recombine, effectively decreasing the number of charge carriers available for extraction and use in a circuit. There exist various pathways and corresponding models of recombination including radiative, Auger, and defect assisted recombination. The recombination rate in a semiconductor can be expressed as the density of photogenerated carriers over the characteristic carrier lifetime:

$$R = \frac{n - n_0}{\tau_n} \tag{1}$$

for electrons where n is the electron density and n_0 is the equilibrium carrier density [1]. The equation for holes is identical.

The most important type of recombination in indirect bandgap semiconductors such as silicon and materials with high defect densities is recombination through defect states within the bandgap. A widely-known model of this type of recombination was developed by Shockley, Read, and Hall (SRH). The general expression for SRH recombination is given by

$$R = \frac{np - n_i^2}{\tau_n(p + p_1) + \tau_p(n + n_1)}. \tag{2}$$

where n_i is the intrinsic electron density when the semiconductor is not doped, and n_1 and p_1 are the electron and hole carrier densities when the trap energy is set to the Fermi level respectively. In addition,

$$\tau_n = \frac{1}{N_t \sigma_n v_n} \tag{3}$$

and

$$\tau_p = \frac{1}{N_t \sigma_p v_p} \tag{4}$$

are effective carrier recombination lifetimes and N_t is the defect density, v_n and v_p are the electron and hole thermal velocities respectively, and σ_n and σ_p are electron and hole capture cross sections respectively [2]. Equation 2 accounts for the emission and capture of charge carriers from defect states within the bandgap. It is derived by calculating the rate at which a carrier is emitted from the defect state and subtracting off the rate at which the carrier is captured by the defect state; this is accomplished using the probability that a carrier is in the defect state along with the carrier capture and emission cross sections. This expression is non-linear and coupled in the terms n and p. For this reason, many models use equation 1 with the constant lifetimes of equations 3 and 4 at the expense of losing the expression's dependence on carrier density.

2.2 Analytic Model

The movement of charged carriers within a semiconductor is modeled by the drift diffusion equations: two coupled, second-order partial differential equations for the electron and hole densities n and p respectively. For simplicity, we consider only one-dimensional models through a cross section of the solar cell. The carriers move due to diffusion caused by gradients in the concentration of particles at a certain point in space and due to electric fields within the material. The drift-diffusion equations are

$$\frac{\partial n}{\partial t} = D_n \frac{\partial^2 n}{\partial x^2} + \mu_n \mathcal{E} \frac{\partial n}{\partial x} - R + G \tag{5}$$

and

$$\frac{\partial p}{\partial t} = D_p \frac{\partial^2 p}{\partial x^2} + \mu_p \mathcal{E} \frac{\partial p}{\partial x} - R + G, \tag{6}$$

where D is the diffusion coefficient, μ is the mobility, $\mathcal{E}$ is the electric field [3]. The recombination rate R within the p layer is given by the general expression

$$R = \frac{n - n_0}{\tau_n}, \tag{7}$$

where n_0 is the equilibrium electron density in the p layer, and the generate rate G is given by the expression

$$\frac{\alpha\Gamma(\lambda)}{D_n} e^{-\alpha}, \tag{8}$$

where α is the absorption coefficient, Γ is the photon flux density. Once we solve for the carrier density as a function of position, we can obtain the current density with the equation

$$J_n = qD_n \frac{dn}{dx} \tag{9}$$

for electrons within the p layer. This equation is the same for holes within the n layer.

In order to solve this complicated set of equations, we must make some simplifying assumptions. Most models in the literature [3] [4] make the following key assumptions in order to solve the equations analytically:

Depletion Approximation:

- There is no electric field in the "neutral" n and p regions and all of the electric field is confined to the depletion region.
- The current only stems from diffusion of carriers in neutral regions.

Superposition Principle:

- The minority carrier density is much lower than the majority carrier density and thus the photocurrent and dark current can be treated independently.

After making the assumptions described above, we can neglect the electric field term $\mathcal{E}$ within the neutral n and p regions. In addition, we are only interested in the steady-state characteristics of the system so we set the time derivative equal to 0. We are left with the following equation:

$$\frac{d^2n}{dx^2} - \frac{n - n_0}{D_n\tau} = -\frac{\alpha\Gamma(\lambda)}{D_n} e^{-\alpha x}. \tag{10}$$

We can solve this second order differential equation analytically but the resulting equations are complicated. The analytic model referenced in this work and that we use to compare to our numerical results is derived by Fahrenbruch and Bube [3]. The equation for the short-current density they derive is

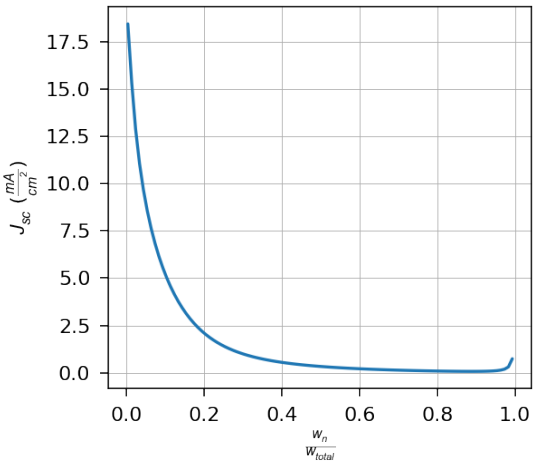

Figure 1: Short-circuit current vs ratio of width of the n layer to the width of the entire cell.

$$J_n(x) = \frac{q\Gamma}{1-(\alpha L_n)^{-2}}\left\{\frac{1}{\alpha L_n}\sinh(\frac{x}{L_n}+e^{-\alpha x}) - \frac{1}{\alpha L_n}\right. \\ \left.\left\{\frac{(\frac{S_n L_n}{D_n})[\cosh(\frac{x_0}{L_n}) - e^{-\alpha x_0}] + \sinh(\frac{x_0}{L_n}) + \alpha L_n e^{-\alpha x_0}}{(\frac{S_n L_n}{D_n})\sinh(\frac{x_0}{L_n}) + \cosh(\frac{x_0}{L_n})}\right\}\cosh(\frac{x}{L_n})\right\}, \tag{11}$$

where S_n is the electron surface recombination velocity, L_n is the electron diffusion length, and x_0 is the total width of the layer.

Using a full cell analytic model developed by Nelson [4][5], we can study how solar cells function by varying our simulation parameters. For example, we can vary the width of the n doped layer while keeping the cell width constant in order to see the effect of the relative widths of the n and p regions on the performance of the cell (see Figure 1). We find that as we increase the width of the n layer relative to the p layer, J_{sc} decreases significantly. The reason for this result is that typical values for the lifetime of electrons in silicon are several orders of magnitude larger than hole lifetimes. This motivates us to focus on the p absorbing layer of the cell.

3 Numerical Model

While it is possible to solve the drift diffusion equations analytically, it requires many assumptions and simplification as we have shown. Numerical models allow us to relax some of the assumptions made to solve the problem analytically and allow us to study inhomogeneities in some of the parameters used to model

the solar cell such as defect density. The model used in this work was built in Python using the finite difference methods. We assume that contributions to the current from the n layer are small for the parameters we use and thus we only focus on the absorbing p of the solar cell.

3.1 Solving the Boundary Value Problems

In order to solve second-order differential equations numerically, we use the central finite difference method. This method approximates derivatives by calculating the slope of the function over a discrete interval Δx. The finite difference expression for the second derivative is

$$\frac{d^2n}{dx^2} = \frac{n(x+\Delta x) - 2n(x) + n(x-\Delta x)}{\Delta x^2}. \tag{12}$$

Once we discretize all of the derivatives in the drift diffusion equation, we rearrange the equation such that all of the terms that depend on n are on the left side and all of the constant terms are on the right hand side. This linear system of equations can then be put into matrix form and formulated as an $Ax = b$ problem. Every equation depends on n_{i-1}, and n_i, and n_{i+1}, where n_i is the electron density at the i^{th} bin in our discrete space, so our matrix is a sparse tridiagonal matrix. If we put equation 10 into this form, we obtain the following matrix

$$\begin{bmatrix} -\frac{2L_n^2+\Delta x^2}{\Delta x^2 L_n^2} & \frac{1}{\Delta x^2} & 0 & \dots & 0 \\ \frac{1}{\Delta x^2} & -\frac{2L_n^2+\Delta x^2}{\Delta x^2 L_n^2} & \frac{1}{\Delta x^2} & \dots & 0 \\ \vdots & & \ddots & \ddots & \ddots \\ 0 & \dots & 0 & \frac{1}{\Delta x^2} & -\frac{2L_n^2+\Delta x^2}{\Delta x^2 L_n^2} \end{bmatrix} \begin{bmatrix} n(1) \\ n(2) \\ \vdots \\ n(N) \end{bmatrix} = \begin{bmatrix} -\frac{\alpha\Gamma}{D_n} e^{-\alpha x_1} \\ -\frac{\alpha\Gamma}{D_n} e^{-\alpha x_2} \\ \vdots \\ -\frac{\alpha\Gamma}{D_n} e^{-\alpha x_N} \end{bmatrix}.$$

3.2 Boundary Conditions

In order to solve this boundary value problem, we must choose appropriate boundary conditions. We use mixed boundary conditions, where

$$n - n_0 = n_0(e^{\frac{qV}{kT}} - 1) \tag{13}$$

for $V = 0$ at the right boundary with the junction and

$$q\frac{dn}{dx} = S_n(n - n_0) \tag{14}$$

at the left, external boundary. The right boundary condition is determined by the voltage across the depletion layer. The left boundary condition is set by the surface recombination velocity [3].

4 Results

We use our numerical model to study two different problems. The first problem is solving the diffusion equation with the full SRH expression and comparing this result to an analytic model with a constant lifetime for silicon solar cells. The second problem is studying the effect of a nonhomogeneous distribution of defect densities throughout the width of the layer to simulate a defect layer. This problem is especially relevant to thin films such as CdTe layers in thin film solar cells where there are high defect densities.

4.1 Validation of the Constant Lifetime Approximation

It is common practice to assume that the recombination rate modeled by Shockley-Read-Hall reduces to a rate with a constant lifetime across the cell when the minority carrier density is much smaller than the majority carrier density. We relax this assumption and keep the lifetime's carrier density dependence to calculate the characteristic carrier lifetime using the SRH recombination expression as follows:

$$\tau = \frac{n - n_0}{R} = \frac{\tau_p(n_1 + n) + \tau_n(p_1 + p_0)}{p_0} \tag{15}$$

We choose $p = p_0$, which decouples the equations. This assumption that the majority carrier density is much higher than the minority carrier density is similar to the assumption of the analytic model, but here we retain a first order dependence on carrier density. Thus, the majority carrier density is constant and independent of the photogenerated minority carriers within the p region so we only consider contributions from minority carriers.

This equation is nonlinear in the variable n which we are solving for. In order to solve the diffusion equation with this non-linear term, we linearize the expression with an iterative algorithm. We begin with an initial guess n^0 for the electron density which we plug into the equation for τ above. We solve the diffusion equation for n, which we call n^1, our first iteration. We use n^1 to solve for τ and calculate n^2. We continue this process until the norm of the difference between n^k and n^{k-1} is below a threshold that is sufficiently low to ensure the convergence of the algorithm (see Appendix B for the algorithm).

We solve the diffusion equation with the nonlinear expression derived above and compare it to the result of the analytic model previously introduced, which assumes a constant recombination lifetime. We vary the defect density N_t from 10^6 to $10^{16} \frac{1}{cm^3}$ to study the effect that high defect densities has on the constant lifetime approximations. The short-circuit current density J_{sc} as a function of defect density is shown in Figure 2. J_{sc} decreases and approaches 0 for high N_t, which agrees with our intuition that the greater the number of defects, the higher the recombination and thus the lower the current. We also compare the electron density across the cell for a range of defect densities and observe a significant decrease in carrier density beginning at $N_t = 10^{11} cm^{-3}$. We find

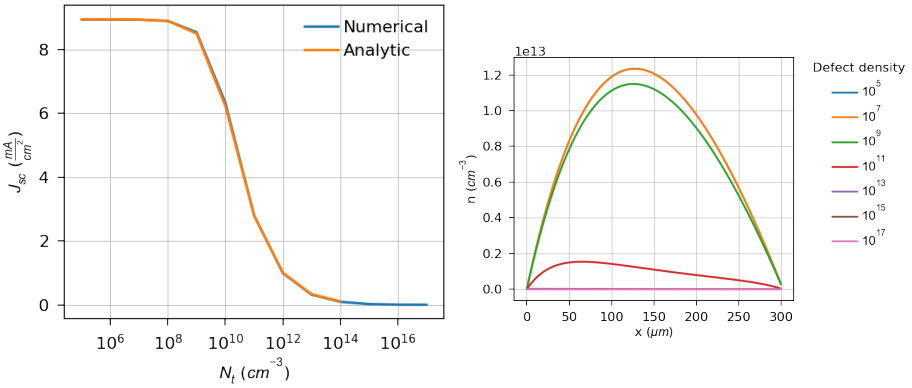

Figure 2: (Left) Short-circuit current density vs defect density for analytic model with constant τ and numerical model with τ dependent on carrier density. (Right) Carrier density across the layer for a range of defect densities.

that the approximation holds for high values of N_t because the models match closely.

Next, we study the effect of increasing the photon flux density Γ to determine when the approximation breaks down. We vary Γ across three orders of magnitude above typical values of $10^{15}\,\frac{photons}{cm^2}$. We see from Figure 3 that the constant lifetime model begins to diverge from the full SRH at $5 * 10^{17}\,\frac{photons}{cm^2}$. The total photon flux from the AM1.5 spectrum is on the order of $10^{17}\,\frac{photons}{cm^2}$, so our model begins to diverge when values of the photon flux become unrealistically high. We observe this same trend in the electron density for high Γ (see Figure 4). However, these values of Γ may be more reasonable for solar concentrators where the intensity of the incident light is significantly higher. At these values of Γ, the assumption that $n << p$ is no longer valid so we must use a different model that accounts for the contributions from majority carriers. This is not within the scope of this work.

From this investigation, we conclude that the constant lifetime approximation used in analytic models of solar cells are robust for high defect densities and over a reasonable range of photon flux densities, but begin to fail for unphysical values of photon flux densities.

4.2 Simulating a Defect Layer in Thin Film Solar Cells

The second problem we study with our numerical model is that of simulating a defect layer in a semiconductor. This is especially relevant to thin film solar cells where there are high defect densities. One example of a thin film solar cell we studied is CdTe. We want to know whether simulating a nonhomogeneous distribution of defects differs from using an average defect density across the layer.

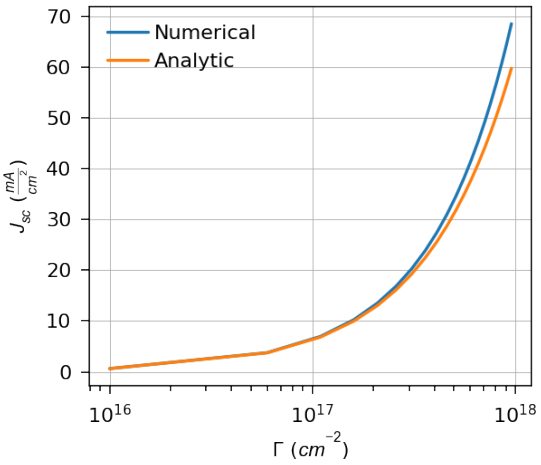

Figure 3: Short-circuit current density vs photon flux density for analytic model with constant τ and numerical model with τ dependent on carrier density.

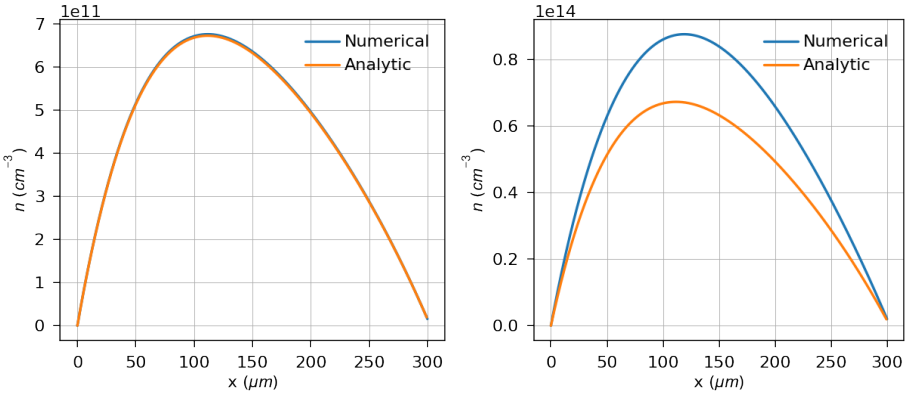

Figure 4: Electron density across p layer for $\Gamma = 10^{16} cm^{-2}$ (left) and $\Gamma = 10^{18} cm^{-2}$ (right).

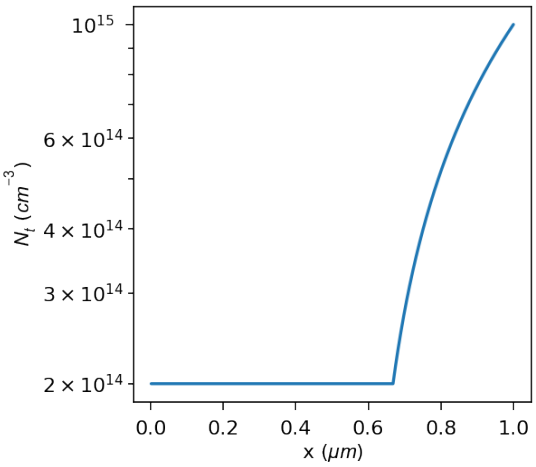

Figure 5: Nonhomogeneous defect density profile across CdTe layer.

We simulate a defect layer in the cell by imposing a nonhomogeneous defect density profile as shown in Figure 5. The defect density is constant at a value of $2 * 10^{14}$ for $\frac{2}{3}$ of the length of the layer. For the remaining third of layer, we increase the defect density linearly up to five times the constant value. In addition, we set the surface recombination velocity to a reasonable value of $10^5 \frac{cm}{s}$. We solve the diffusion equation numerically and calculate the total bulk recombination, ignoring the surface recombination, with the equation

$$R = \sum_{n=1}^{N} \frac{n_i - n_0}{\tau_i} \Delta x \tag{16}$$

where

$$\tau_i = \frac{1}{N_t \sigma_n v_n}, \tag{17}$$

the lifetime given by the N_t at a specific position within the layer. We repeat the same steps above but with a constant N_t across the entire layer. We fix $S_n = 10^5 \frac{cm}{s}$ and modify N_t until the total bulk recombination is approximately equal to that of the system described above.

We compare the electron density across the layer for the inhomogeneous defect density profile and the constant defect density profile (see Figure 6). We expect differences in the profile as we vary the wavelength of the incident photons. The carrier density profile for blue light differs less than for red light when comparing a constant and nonhomogeneous distribution of defects because the high energy photons get absorbed more quickly so the photogenerated electrons do not feel the effect of the defect layer. While we observe differences in the

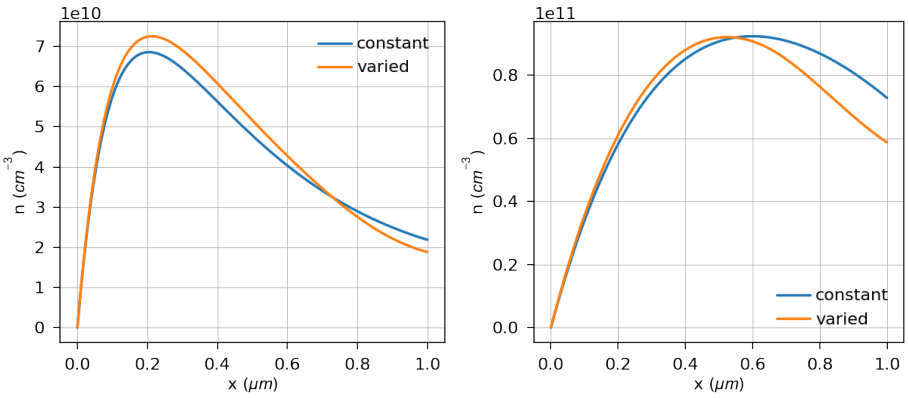

Figure 6: Comparison of nonhomogeneous defect density distribution and constant defect density distribution for blue light (left) and red light (right).

carrier density profiles for the two systems, the effect of simulating a defect layer does not change the short-circuit current density by more than 3% in our experiments. When using red light, we obtain a J_{sc} of $5.2\frac{mA}{cm^2}$ for the case of spatially constant defect density and $5.5\frac{mA}{cm^2}$ for the case of spatially inhomogeneous defect density. For red light, we obtain values for J_{sc} of $13.3\frac{mA}{cm^2}$ and $13.7\frac{mA}{cm^2}$ for the constant and varied cases respectively. Since our main concern is the performance of the absorbing layer, we conclude that adding a spatially nonhomogeneous distribution of defects does not differ significantly from using a constant, average defect density across the layer.

5 Conclusion

We began by describing how analytic models are built and derived. We used a full analytic model to study the performance of a cell as we vary the relative widths of the doped layers of the cell. Then, we described how to setup and solve a numerical model using the finite difference method. We studied two problems: the effect of the dependence of carrier lifetime on the electron density in order to validate the constant lifetime approximation, and the effect of simulating a defect layer instead of using an average defect density throughout a layer of a thin film solar cell. For the first problem, we found that for a wide range of defect densities, the constant lifetime approximation holds. However, as we increase the carrier generation rate, the analytic and numerical models begin to diverge. For the second problem, we found that using a nonhomogeneous, linearly increasing defect density could be simulated by using an average and constant defect density across the layer while keeping bulk recombination constant.

Bibliography

[1]S. J. Fonash, *Solar cell device physics*, English (Academic Press/Elsevier, Boston, MA, 1981).

[2]W. Shockley, and W. Read Jr, "Statistics of the recombinations of holes and electrons", Physical review **87**, 835 (1952).

[3]A. L. Fahrenbruch, and R. H. Bube, *Fundamentals of solar cells : photovoltaic solar energy conversion.* English (Academic Press, New York, 1983), p. 559.

[4]J. Nelson, *The physics of solar cells*, English (Imperial College Press, London : 2003).

[5]S. Singh, and J. Menart, "Mathematical modeling of a p-n junction solar cell using the transport equations", MA thesis (Wright State University, Dayton, Ohio, 2017).

[6]R. S. Sultana, A. N. Bahar, M. Asaduzzaman, M. M. R. Bhuiyan, and K. Ahmed, "Numerical dataset for analyzing the performance of a highly efficient ultrathin film CdTe solar cell", Data in Brief **12**, 336–340 (2017).

[7]A. Rakhshani, "Electrodeposited CdTe - optical properties", Journal of Applied Physics **81**, 7988–7993 (1997).

[8]M. Gloeckler, A. L. Fahrenbruch, and J. R. Sites, "Numerical modeling of cigs and cdte solar cells: setting the baseline", in 3rd world conference onphotovoltaic energy conversion, 2003. proceedings of, Vol. 1 (May 2003), 491–494 Vol.1.

[9]M. A. Green, "Self-consistent optical parameters of intrinsic silicon at 300k including temperature coefficients", Solar Energy Materials and Solar Cells **92**, 1305–1310 (2008).

Appendices

A Table of Simulation Parameters

A.1 Silicon

The parameters listed below were taken from [3] and [4] and were in our model of p-type silicon.

Cell Width	$300\mu m$
S_n	$100000\frac{cm}{s}$
D_n	$25\frac{cm^2}{s}$
σ_n	$10^{-12}cm^2$
σ_p	$10^{-15}cm^2$
v_{nt}	$2.3 * 10^5$
v_{pt}	$1.65 * 10^5$

A.2 CdTe

The parameters listed below were taken from [7][8] and were used in our model of CdTe.

Cell Width	$1\mu m$
S_n	$100000\frac{cm}{s}$
μ_e	$320\frac{cm^2}{Vs}$
σ_n	$10^{-12}cm^2$
σ_p	$10^{-15}cm^2$
v_{nt}	$10^5\frac{m}{s}$
v_{pt}	$10^5\frac{m}{s}$
N_c	$8 * 10^{17}\frac{1}{cm^3}$
N_v	$1.8 * 10^{19}\frac{1}{cm^3}$

B Code

Listing 1: Code written in Python to solve non-linear diffusion equation with full SRH expresion. All parameters are in SI units.

```
import math
import matplotlib
import matplotlib.pyplot as plt
import numpy as np
import scipy.sparse
import scipy.sparse.linalg
%matplotlib inline
font = {'family' : 'normal',
```

```
        'weight' : 'normal',
        'size'   : 20}
matplotlib.rc('font', **font)

a = 0 #start x
b = 300 * 10**(-6) #end x
N = 1000 #number of bins
dx = (b - a) / (N - 1) #bin spacing
x = np.linspace(a, b, num=N+1) #array of bins for the length of the layer

#FUNDAMENTAL CONSTANTS AND DEVICE PARAMETERS
V = 0.0 #applied voltage
T = 300 #room temperature
q = 1.602 * 10**(-19) #fundamental charge
k = 1.381 * 10**(-23) #Boltzmann constant
Sn = 1000 #electron surface recombination velocity
Dn = 25*10 **(-4) #diffusion coefficient
alpha = 1/(1.56*10**(-4)) #absorption coefficient
bs = (10**17) * (10**4) #photon flux density
p0 = 10.0**17 * (10**6) #equilibrium hole concentration
ni = 9.65 * 10**9 * 10**(6) #intrinsic electron density
n0 = (ni**2)/p0 #equilibrium electron concentration
Et = 0.6*1.602*10**(-19) #trap energy
Eg = 1.12*1.602*10**(-19) #bandgap energy
Ei = Eg/2 #intrinsic energy level
n1 = ni*math.exp((Et - Ei)/(k*T))
p1 = ni*math.exp(-(Et - Ei)/(k*T))
Nt = 10**11 #defect density
sp = 10**(-15) *(10**(-4)) #hole capture cross section
sn = 10**(-12) *(10**(-4)) #electron capture cross section
vnt = 2.3*(10**5) #electron thermal velocity
vpt = 1.65*(10**5) #hole thermal velocity
Cp = 1/(sp*Nt*vpt) #constant hole lifetime
Cn = 1/(sn*Nt*vnt) #constant electron lifetime


#####################Solving non-linear diffusion equation#########################

#initializing arrays for subsequent calculations
rhs = np.zeros(N); #right hand side of matrix equation
ns = np.zeros(N+1) #initial guess at electron density
ns[0] = n0
n = np.zeros(N+1) #electron density
n[0] = n0 #boundary condition

z = 0
```

```
chis = []
while (True):
  tau = np.zeros(N)
  A = scipy.sparse.lil_matrix((N, N))
  for i in range(0,N-1):
        #calculating SRH lifetime
        tau[i] = (Cp*(n1 + n[i + 1]) +  Cn*(p0 + p1))/p0
        #updating the matrix equation with nonhomogeneous lifetime
        rhs[i] = (-bs*alpha/Dn)*math.exp(-x[i+1]*alpha) - (n0/(Dn*tau[i]))
        A[i, [i - 1, i, i +1]] = np.array([1/dx**2, (-2/(dx**2))
                + (-1/(Dn*tau[i])) ,1/dx**2])
  #Setting boundary conditions
  tau[N-1] = (Cp*(n1 + n[N])/(p0)) +  (Cn*(p0 + p1)/(p0))
  rhs[N-1] = -(2*Sn*n0)/(Dn*dx) + (-bs*alpha/Dn)*math.exp(-x[N]*alpha)
                - (n0/(Dn*tau[N-1]))
  A[0, N-1] =0
  A[N-1, N - 2] = 2/(dx**2)
  A[N-1,N - 1] = (-2/(dx**2)) - (1/(Dn*tau[N-1])) - ((2*Sn)/(Dn*dx))

 #solving matrix equation
  A = A.tocsc()
  n[1:] = scipy.sparse.linalg.spsolve(A, rhs)

  #calculate the norm of the difference between the current
  #and previous iteration
  chi = np.linalg.norm(n - ns)
  chis.append(chi)
  np.copyto(ns, n)

  #if the norm of the norm does not change over
  #the previous iteration, the algorithm has converged
  if(z > 2 and abs(chi - chis[z-1]) < 0.001):
        break
  z += 1

  #if the algorithm has not converged in 100 iterations,
  #the system most likely will not converge or is oscillating
  if(z > 100):
    print("reached max number of iterations")
        break
```

Listing 2: Code written in Python to compare constant defect density profile to spatially inhomogeneous defect density profile. All parameters are in SI units.

```
import math
import matplotlib
import matplotlib.pyplot as plt
```

```
import numpy as np
import scipy.sparse
import scipy.sparse.linalg
%matplotlib inline
font = {'family' : 'normal',
        'weight' : 'normal',
        'size'   : 20}

matplotlib.rc('font', **font)

a = 0 #start x
b = 1 * 10**(-6) #end x
N = 1000 #number of bins
dx = (b - a) / (N - 1) #bin spacing
x = np.linspace(a, b, num=N+1) #array of bins for the length of the layer

#FUNDAMENTAL CONSTANTS AND DEVICE PARAMETERS
V = 0.0 #applied voltage
T = 300 #room temperature
q = 1.602 * 10**(-19) #fundamental charge
k = 1.381 * 10**(-23) #Boltzmann constant
Sn = 1000 #electron surface recombination velocity
mu_e = 320 * 10**(-4) #electron mobility
Dn = (mu_e*k*T)/q #diffusion coefficient
alpha = 10**4 * 10**2 #absorption coefficient
bs = (10**17) * (10**4) #photon flux density
p0 = 2*10**14 * 10**6 #equilibrium hole concentration
Et = 0.6*1.602*10**(-19) #trap energy
Eg = 1.5*1.602*10**(-19) #bandgap energy
Ei = Eg/2 #intrinsic energy level
Nc = 8 * 10**17 * 10**6
Nv = 1.8 * 10**19 * 10**6
n0 = Nc*math.exp((-0.5*Eg)/(k*T)) #equilibrium electron concentration
n1 = Nc*math.exp((-(2*Eg-Et)/(k*T)))
p1 = Nv*math.exp(-Et/(k*T))
Nt = 10**11 #defect density
sp = 10**(-15) * 10**(-4)  #hole capture cross section
sn = 10**(-12) * 10**(-4) #electron capture cross section
vnt = 10**5 #electron thermal velocity
vpt = 10**5 #hole thermal velocity
Cp = 1/(sp*Nt*vpt) #constant hole lifetime
Cn = 1/(sn*Nt*vnt) #constant electron lifetime


####################Constant average defect density#######################
Nt = 3.16195*10**(14) * 10**6 #defect density
```

```
tau = 1/(sn*vnt*Nt) #constant recombination lifetime
rhs = np.zeros(N); #initializing the rhs of matrix equation
n = np.zeros(N+1) #initializing electron density array
n[0] = n0 #setting boundary condition

#setting up sparse matrix equation
A = scipy.sparse.lil_matrix((N, N))
for i in range(0,N-1):
    rhs[i] = (-bs*alpha/Dn)*math.exp(-x[i+1]*alpha) - (n0/(Dn*tau))
    A[i, [i - 1, i, i +1]] = np.array([1/dx**2, (-2/(dx**2))
        + (-1/(Dn*tau)) ,1/dx**2])

#setting boundary conditions
rhs[N-1] += -(2*Sn*n0)/(Dn*dx)
A[0, N-1] =0
A[N-1, N - 2] = 2/(dx**2)
A[N-1,N - 1] = (-2/(dx**2)) - (1/(Dn*tau)) - ((2*Sn)/(Dn*dx))
A = A.tocsc()
n[1:] = scipy.sparse.linalg.spsolve(A, rhs)#solving matrix equation

#calculating the bulk recombination
R = 0
for i in range(0,N):
    R += (n[i+1] - n0)*dx/(tau)
print("R:", R)
#plotting electron density as a function of position
#to compare constant vs varied defect density profiles
plt.figure(1, figsize=(8, 7), dpi=80)
plt.plot(x*10**6, n*10**(-6))
plt.xlabel("x")
plt.ylabel("n")
plt.grid(True)

#calculating the current density
Jn = np.zeros(N)
for i in range(1, len(Jn)-1):
    Jn[i] = (n[i+1] - n[i-1])/(2*dx)
Jn *= q*Dn
Jn[0] = q*Dn*(n[1]-n[0])/dx #forward difference derivative
Jn *=0.1
print("Jsc (constant tau) ", Jn[0])

#plotting the short-circuit current as a function of position
#within the cell to compare the constant
#and varied defect densities across the layer
plt.figure(2, figsize=(8, 7), dpi=80)
```

```
plt.plot(x[0:N-1] *10**6, Jn[0:N-1])
plt.grid(True)
plt.xlabel(r"x ($\mu m$)")
plt.ylabel(r"$J$ ($\frac{mA}{{cm}^2}$)")
plt.legend(np.array(["constant", "varied"]))


##################Inhomogeneous defect density#####################
Nt = 2.0*10**(14) * 10**6 #constant defect density

#filling defect density array
Nts = np.zeros(N)
Nts.fill(Nt) #constant array

#calculating inhomogeneous lifetime
#based on spatially inhomogeneous defect density distribution
x1 = b - b/3 #defect layer is 1/3 of the length of the layer
s2 = int(x1/dx) + 1
Nts[s2:] = np.linspace(Nt,5*Nt, num=(N-s2))
tau = 1/(sn*vnt*Nts)


rhs = np.zeros(N); #right hand side of matrix equations
n1 = np.zeros(N+1)
n1[0] = n0

#setting up sparse matrix equation and boundary conditions
A = scipy.sparse.lil_matrix((N, N))
for i in range(0,N-1):
    rhs[i] = (-bs*alpha/Dn)*math.exp(-x[i+1]*alpha) - (n0/(Dn*tau[i]))
    A[i, [i - 1, i, i +1]] = np.array([1/dx**2, (-2/(dx**2)) + (-1/(Dn*tau[i]))
rhs[N-1] += -(2*Sn*n0)/(Dn*dx)
A[0, N-1] =0
A[N-1, N - 2] = 2/(dx**2)
A[N-1,N - 1] = (-2/(dx**2)) - (1/(Dn*tau[N-1])) - ((2*Sn)/(Dn*dx))
A = A.tocsc()

n1[1:] = scipy.sparse.linalg.spsolve(A, rhs) #solving matrix equation

#calculating current density
Jn1 = np.zeros(N)
for i in range(1, len(Jn1)-1):
    Jn1[i] = (n1[i+1] - n1[i-1])/(2*dx)
Jn1 *= q*Dn
Jn1[0] = q*Dn*(n1[1]-n1[0])/dx #forward difference derivative
Jn1 *=0.1
```

```
print("Jsc (varied tau) ", Jn1[0])

#plotting electron density as a function of position
#to compare constant vs varied defect density profiles
plt.figure(1, figsize=(8, 7), dpi=80)
plt.plot(x*10**6, n1*10**(-6))
plt.xlabel(r"x ($\mu m$)")
plt.ylabel(r"n ($cm^{-3}$)")
plt.legend(np.array(["constant", "varied"]))

#plotting the short-circuit current as a function of position
#within the cell to compare the constant
#and varied defect densities across the layer
plt.figure(2)
plt.plot(x[0:N-1] *10**6, Jn1[0:N-1])
plt.legend(np.array(["constant", "varied"]))
plt.grid(True)

#calculating the bulk recombination
R = 0
for i in range(0,N):
    R += (n1[i+1] - n0)*dx/(tau[i])
print("R: ",R)
```