Overview of the model

The overall modeling flow of this model is illustrated in the following chart:

Select a solar cell model

detailed balance

analytical drift-diffusion diffusion

drift diffusion (partial differential equation)

Extract from experimental data

calculate diode parameters and resistances

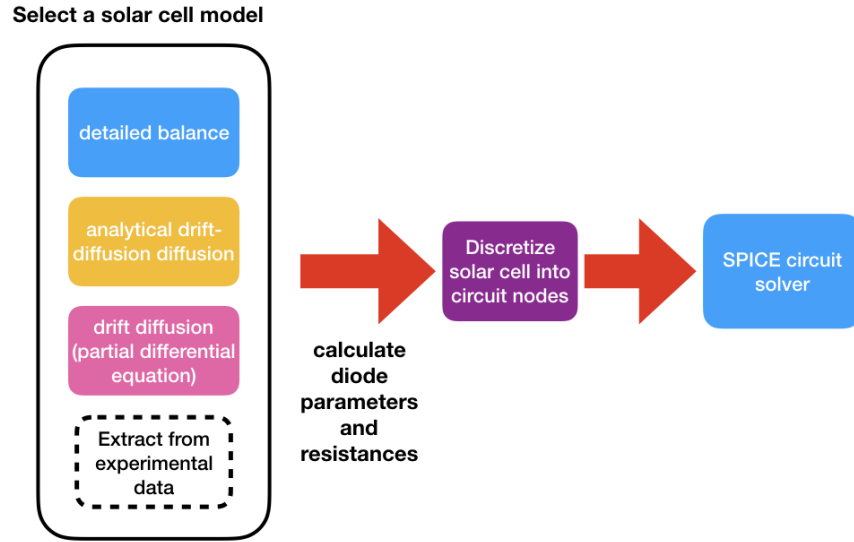Discretize solar cell into circuit nodes

SPICE circuit solver

Figure 1: The flow chart of the SPICE-based solar cell simulation model.

To start with, the user should select a solar cell model of their choice. The default one in the demonstration code is **detailed balance model**. Details of the suppporte solar cell models can be found in the white paper of solcore5 or its officical documentation on the web. A solar cell model should then give the diode parameters, resisatnce values and the injection current for feeding into the equivalent circuit.

After that, the program discretizes the solar cell into small pixels, and each pixel is represented by a small circuit unit that contains two diodes, one current source and a number of resistors. Detailed procedure of the discretizaion is dscribed in solcore5's ofiicial webpage.

After the circuit network is built, the program feeds the generated circuitry file into the SPICE solver, and then calculates the I-V characteristics.

Generalized discretization

A circuit solver essentially does $N \times N$ matrix inversion, where N is the number of nodes in the circutry. The time complexity of solving matrix inversion is typically between $O(N^{2.3})$ to $O(N^3)$*. We therefore implemented an approximation algorithm to reduce the number of nodes but retain reasonable accuracy.
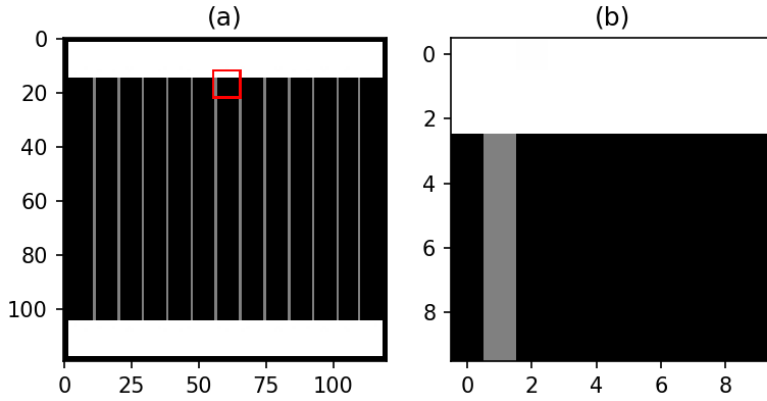

Figure 2: (a) An example of the full image of the front grid. (b) An enlarged image within the red square in (a).

The above figure (a) shows an example of the front grid, whereas (b) is the enlarged image within the red square in (a). Our aim is to merge all the pixels in (b) into a single pixel. We then have to calculate the following parameters of this merged pixel:

- contact resisatnce
- metal grid resisatnce in x- and y-direction
- metal coverage (for calculating photon injection density)

In the meantime, the calculation the following parameter remain the same. Just be sure that the physical widths are that of merged pixel widths.

- sheet resistance
- diode parameters
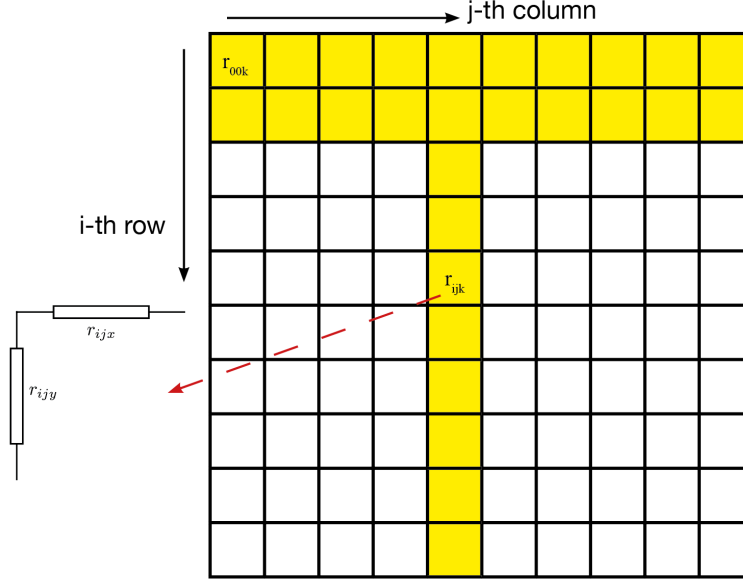- series resistance

Metal grid resistance

Figure 3: Discretization of the metal grid resistance.

Assume that the resistance in x- and y-direction of each pixel is R_{ijx} and R_{ijy}. Our goal is to calculate the overall grid resisatnce R_X and R_Y, as shown in the above figure. The lumped resistance of the bus bar or the fingers in x-direction is calculated by simply applying the resistance superposition law:

$$R_{jx} = \sum_{i=0}^{N_I - 1} R_{ijx}$$

$$R_x = \frac{1}{\sum_{j=0}^{N_J - 1} \frac{1}{R_{jx}}}$$

Contact resistance

Contact resisatnce of the merged pixel is

$$R_c = \rho_c / \gamma_m$$

where ρ_c is contact resistivity (Ωm^2) and γ_m is the number of metal-grid pixels divided by the total number of pixels.

Photon injection

The photon injection simply is

$$\phi_m = \phi_0 \cdot \gamma_m$$

where ϕ_0 is the original photon injection density.