

Simiconductor – RegionEditor: manual

Jori Liesenborgs
`jori.liesenborgs@uhasselt.be`

December 16, 2014

Chapter 1

Introduction

Setting up a simulation in `simiconductor` allows for a great deal of flexibility, the side effect being that it can easily become quite cumbersome and care must be taken to avoid errors. As will be explained in the next chapter, in many cases several relationships need to be fulfilled between variables in the simulation. Because of this, in practice less parameters will need to be specified than one might think based on the bare drift-diffusion equations.

To make it easier to exploit these relationships, a graphical region editor tool was created in which one can define several properties in a more intuitive manner. A screen shot of this program can be seen in Fig. 1.1: it allows you to define regions (indicated by different colors) in either a 1D or 2D setting, and assign properties (relative permittivity, LUMO/HOMO levels, ...) to them. Once the necessary input parameters have been assigned, the program can generate the set of commands that can be read by `simiconductor` to recreate this simulation scenario. More detailed information about how to use the editor itself can be found in chapter 3.

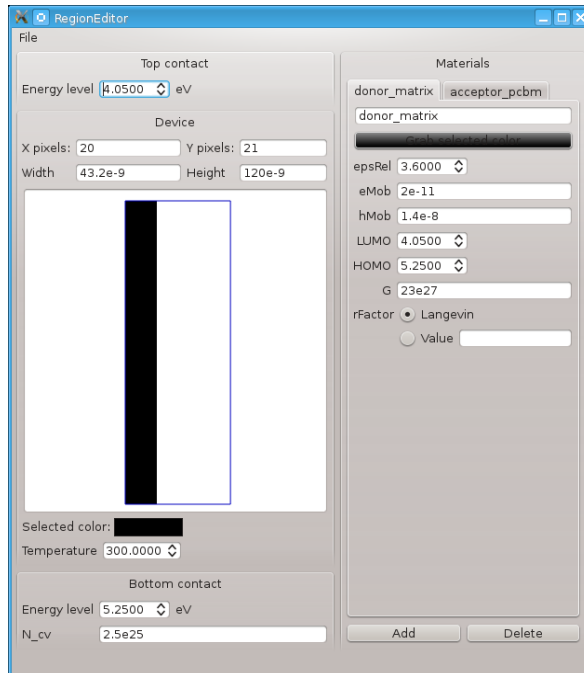


Figure 1.1: An example of the user interface developed to generate all properties of the drift-diffusion equations in a consistent way from a limited number of input parameters.

Chapter 2

Consistent device description

2.1 Drift-diffusion model

Similar to the works of e.g. [1] and [2], simulations of organic solar cells in `simiconductor` are based on the drift-diffusion equations shown below:

$$\begin{aligned}\vec{\nabla} \cdot (\varepsilon_{rel} \vec{\nabla} \phi) &= -\frac{e}{\varepsilon_0} (p - n), \\ \vec{J}_n &= -D_n \vec{\nabla} n + \mu_n n \vec{\nabla} \phi, \\ \vec{J}_p &= -D_p \vec{\nabla} p - \mu_p p \vec{\nabla} \phi, \\ \frac{\partial n}{\partial t} &= G - R - \vec{\nabla} \cdot \vec{J}_n, \\ \frac{\partial p}{\partial t} &= G - R - \vec{\nabla} \cdot \vec{J}_p.\end{aligned}$$

In these equations, ϕ , n and p are the variables to be solved for, representing the electrostatic potential, the electron number density and hole number density respectively. The potential and concentration differences cause drift and diffusion currents, yielding total number currents \vec{J}_n and \vec{J}_p . Together with generation and recombination these currents cause changes in the carrier densities. Eventually, the situation evolves towards a steady-state situation in which n , p and ϕ no longer change.

As explained in the `simiconductor` manual, these drift-diffusion equations can be solved in either a 1D or 2D situation (in the latter case with periodic boundary conditions) by discretizing them on a grid. The charge carrier densities typically vary on an exponential scale, and to avoid needing a very large number of grid points for accurate results, the technique by Scharfetter and Gummel [4] is used to calculate the currents. This allows very accurate calculations when using even a small amount of grid points in each direction.

Apart from values such as the relative permittivity or diffusion constants which have to be defined on all grid points, the boundary values of the potential and charge densities must be supplied as input as well. These values will remain fixed while looking for the steady state solution. In a one dimensional simulation the boundaries consist of a single grid point at each side of the simulation. In two dimensions, periodic boundary conditions are used in one direction (similar to [2]), so the boundary points are two strips of grid points at each side of the simulated device.

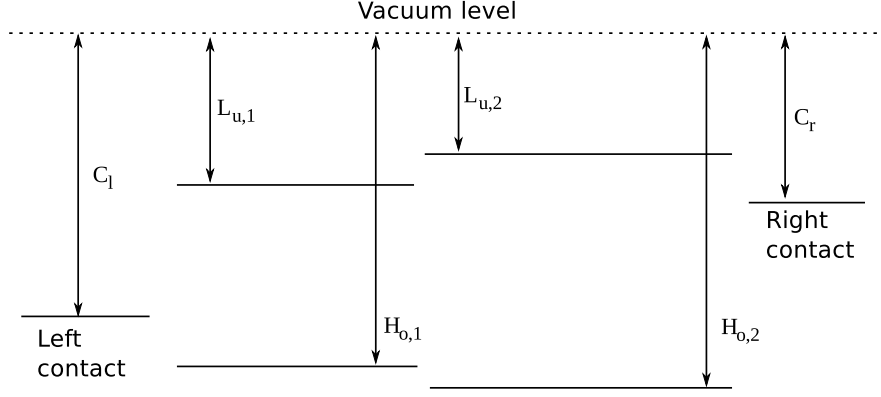


Figure 2.1: In a simulated device with work functions and energy levels as indicated in this figure, one can formulate several constraints that should be fulfilled to obtain a consistent description (see text).

2.2 Device description

In a general one- or two-dimensional simulation using materials with different HOMO and LUMO levels, as shown in Fig. 2.1 for example, one would like concentration differences across the boundaries between the two materials, corresponding to the change in energy level. As described in the supplementary material of [3], this can be accomplished by introducing additional electric fields, or equivalently additional potentials, to which only specific charge carrier types respond. Calling V_n the extra potential to which the electrons respond, and V_p the extra potential to which the holes respond, the equations for the currents become

$$\begin{aligned}\vec{J}_n &= -D_n \vec{\nabla} n + \mu_n n (\vec{\nabla} \phi + \vec{\nabla} V_n), \\ \vec{J}_p &= -D_p \vec{\nabla} p - \mu_p p (\vec{\nabla} \phi + \vec{\nabla} V_p).\end{aligned}$$

As shown in appendix A, the required changes in charge carriers can be obtained by setting V_n and V_p to the difference in LUMO and HOMO levels respectively in the appropriate region.

Employing the Metal-Insulator-Metal (MIM) model, in thermodynamic equilibrium the Fermi levels of the contacts align, and injection barriers for both carrier types arise at the contacts. Using N_{cv} as the charge concentration at the contacts, the boundary condition for n at the left side of the device is then

$$n(0) = N_{cv} \exp\left(-\frac{C_l - L_{u,1}}{k_B T}\right),$$

with similar boundary conditions on the other side and for the holes. Also shown in appendix A is that for a consistent description one should then use an internal device voltage of

$$eV_{int} = C_l - C_r.$$

If a straightforward bimolecular recombination model is used,

$$R = r(np - n_i^2),$$

then one has the additional conditions that, in the case of Fig. 2.1, the values for n_i in each side of the simulated device must be

$$n_{i,left} = N_{cv} \exp\left(-\frac{H_{o,1} - L_{u,1}}{2k_B T}\right), \quad n_{i,right} = N_{cv} \exp\left(-\frac{H_{o,2} - L_{u,2}}{2k_B T}\right).$$

In two dimensions, similar conditions hold.

It is clear that employing such energy level definitions to base the simulation on, automatically fixes several properties of the device. While `simiconductor` itself merely solves the drift-diffusion equations once all properties have been defined, defining them consistently can be quite a burden. For this reason, a graphical interface was developed in which the independent properties can be specified and from which the settings for the drift-diffusion solver can be generated. An example of the interface is shown in Fig 1.1; instructions for using the program will be given in the next chapter.

Chapter 3

The region editor

3.1 Introduction

The main idea is to allow you to define the layout of a 1D or 2D device by using an image editor, like Photoshop or Gimp. Such an image can then be imported by the region editor, where you'll be able to assign different material properties to pixels with different colors. The resolution of the image you have created, will determine the resolution of the simulation grid.

Note that it is important to save the image you have created in a so-called lossless format. A good way to store an image is as a PNG image. Do *not* use a JPG format to store your image in: JPG is a 'lossy' format, meaning that the pixel values that are read from it may be different from the pixel values you set in your image manipulation program.

Also note that in simulations in which there are distinct regions with different values for specific properties, it is important to consider the resolution with which the equations are solved numerically. Because abrupt changes in properties can only be represented using an infinite number of grid points, simulations of such situations can be highly resolution dependent.

3.2 Starting a new device model

When the region editor program has been started, you should see something similar as shown in Fig. 3.1. The top and bottom contacts are at the top and bottom of the window, and their energy levels can be set. For a 1D simulation, the top contact corresponds to the right contact from Fig. 2.1 and the bottom contact corresponds to the left contact from the same figure. *This also means that the value of the top contact energy level is assumed to be lower than that of the bottom contact, to have a positive internal voltage.*

At this point, you can already set a few device properties which do not depend on the grid size or on the materials in the simulation: the energy levels of the contacts, the temperature and N_{cv} , the electron density at the contacts. In the file menu, you can already save these settings to a file, or load a previously created file.

Suppose we've created a 64x64 image with two colors, as shown in Fig. 3.2. To use this in the region editor, the option 'Import image' must be selected from the 'File' menu. When this has been loaded, you'll see the X pixels and Y pixels fields being set to 64, i.e. the dimensions of the image. You can zoom in and out by using the mouse wheel, but note that

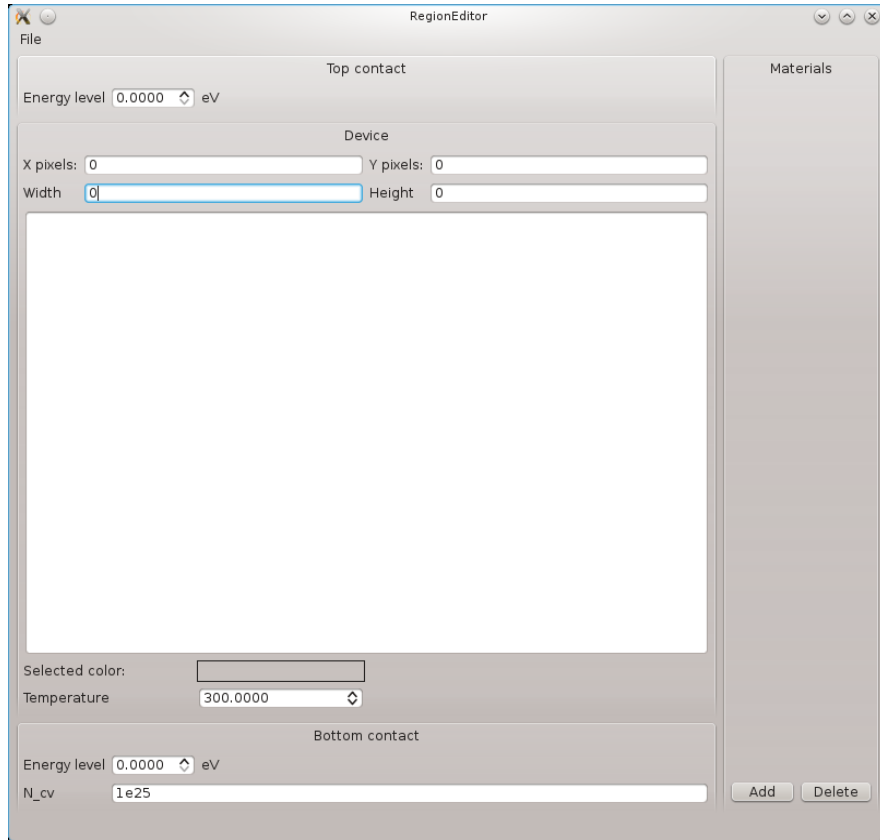


Figure 3.1: This is what you should see when you have just started the region editor application.

it is possible you have to click on the image first.

By default each pixel is assumed to be 1 nm by 1 nm, and the corresponding **Width** and **Height** values will be filled in. These can be changed according to your own needs, and the physical width is allowed to differ from the height. Although it won't make a difference in this case, since periodic boundary conditions are used in the X-direction and each column of the image is the same, let's change the width to $16\text{e-}9$, which means 16nm or $16 \times 10^{-9}\text{m}$. You'll see that the scale of the image is adjusted according to these physical dimensions, even though the number of points stays the same.



Figure 3.2: A 64x64 image which can be used as a 2D model for a bilayer device. The image contains a top region in blue (27 pixels high) and a bottom region in yellow (37 pixels high).

We're going to simulate something similar to the situation from Fig. 2.1, so let's set the energy

level of the top contact to $4eV$, and that of the bottom contact to $5eV$. The temperature and the N_{cv} value are going to be left at their default values. Afterwards, the situation corresponds to the one shown in Fig. 3.3.

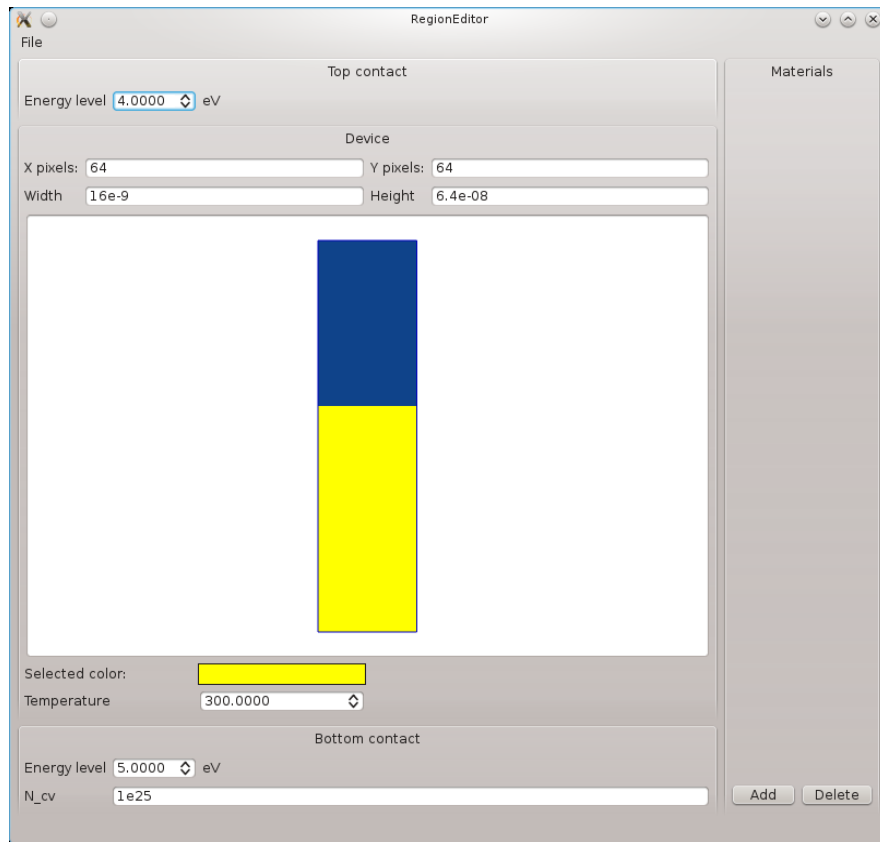


Figure 3.3: After loading an image which defines the regions and setting some device properties, the region editor looks like this.

3.3 Defining materials

To define properties for the two colors in the image, we must first add a material by pressing **Add** in the ‘Materials’ part of the window (on the right). You’ll notice that if you click on different colors in the image itself, the **Selected color** indicator below the image, will change color as well. In the materials section you can click a button called **Grab selected color** to indicate that the properties specified in this part are for the currently selected color. This color is probably yellow by default, but if not, we can click the yellow part of the image to make sure the yellow color is selected, and press the ‘grab’ button afterwards. The color of the button will also be yellow to indicate that the properties refer to that part of the image.

In the field where it says ‘no_name’, we’re going to just enter the text ‘Yellow’. The ϵ_{psRel} value refers to the relative permittivity, and we’re going to set this to 3.5. We’re not going to change the mobilities, they can keep their default values of $10^{-8}m^2V^{-1}s^{-1}$ (note that we’re using SI units). The LUMO level will be $4.1eV$, while the HOMO level will be $5.1eV$. The generation rate is left at zero, we’re going to calculate the dark current, and the prefactor for bimolecular recombination is set to be the Langevin value. When all this has been done, the region editor will look like Fig. 3.4.

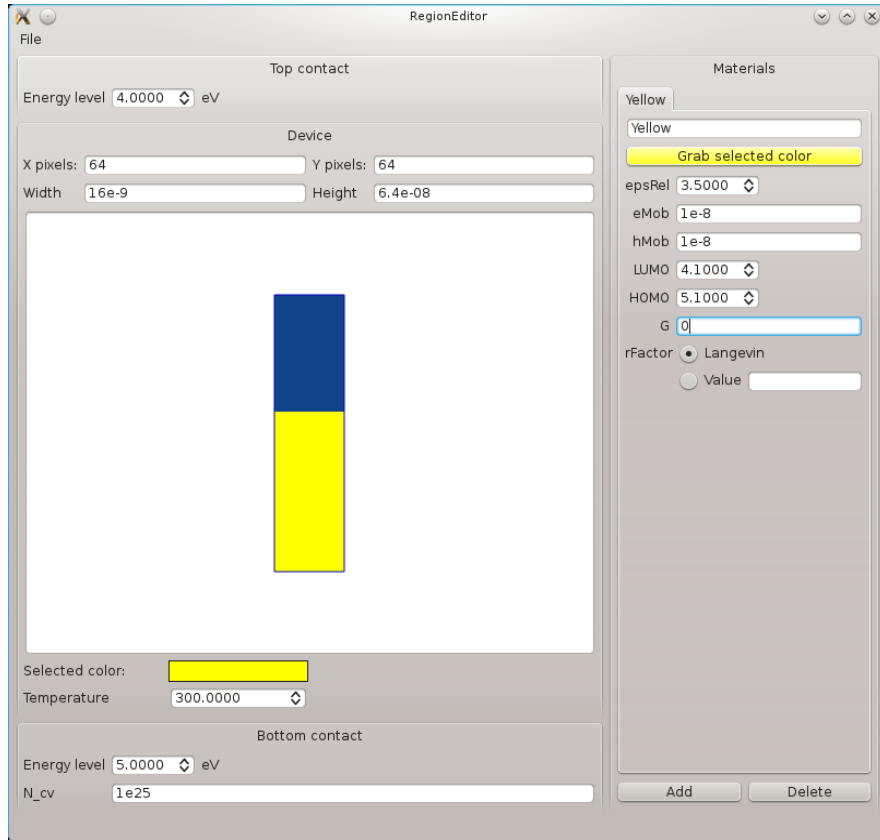


Figure 3.4: This is what the region editor will look like after adding the first material.

Then, we're going to press **Add** again, and follow the same procedure to define the blue region, which we'll call 'Blue'. The only properties that we're going to give a different value are the LUMO and HOMO levels, which we'll set to $4.2eV$ and $5.2eV$ respectively. This will result in the situation shown in Fig. 3.5.

3.4 Generating semiconductor commands

At this point, we've prepared the simulation grid from within the region editor, and we'd like to actually use this in `simiconductor`. To do so, the current setup needs to be exported to a text file which will contain a series of commands for `simiconductor`, and this can be done by selecting 'Generate 2D' from the 'File' menu. Let's assume that the file these commands are exported to, is named `bilayer2d.grid.txt`, a file which will contain something like this:

```
math/def T 300.0
math/def kT_ev T*kB/e
math/def topContact 4.0
math/def botContact 5.0
math/def VBI (botContact-topContact)
...
reg2/new REG_Yellow
reg2/append/rect REG_Yellow 1 1 64 37
reg2/new REG_Blue
reg2/append/rect REG_Blue 1 38 64 64
```

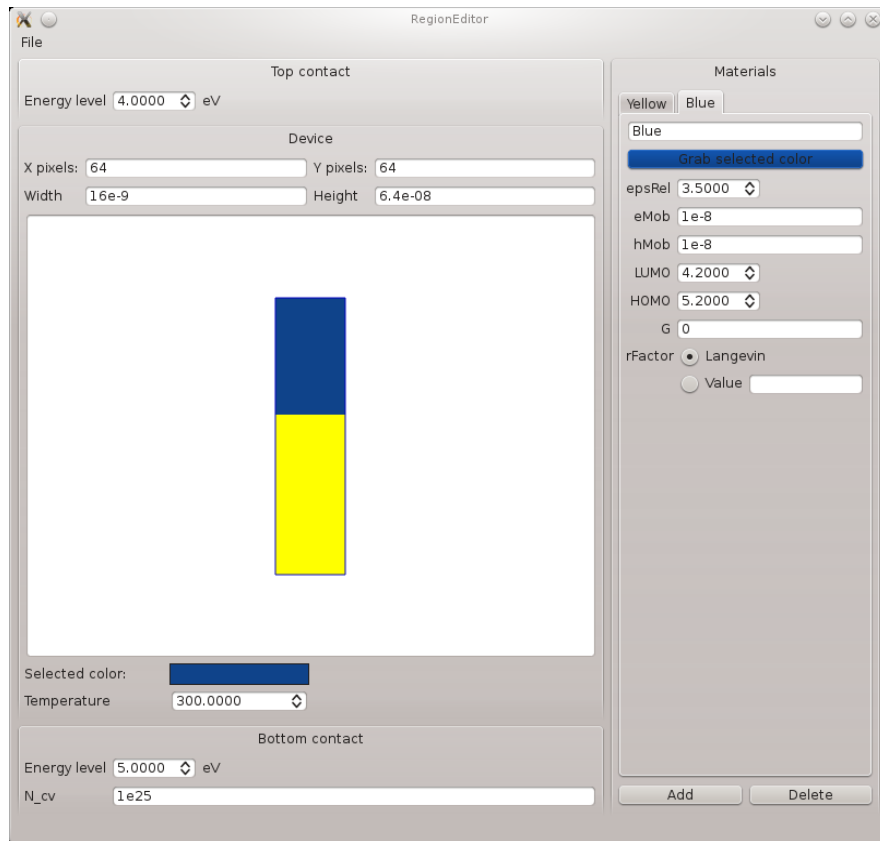


Figure 3.5: This is what the region editor will look like after adding the second material in the same way.

```
reg2/new BOTCONTACT_Yellow
reg2/append/rect BOTCONTACT_Yellow 1 1 64 1
reg2/new TOPCONTACT_Blue
reg2/append/rect TOPCONTACT_Blue 1 64 64 64
...
sim2/reg/set n BOTCONTACT_Yellow n_bot_Yellow
sim2/reg/set p BOTCONTACT_Yellow p_bot_Yellow
sim2/reg/set n TOPCONTACT_Blue n_top_Blue
sim2/reg/set p TOPCONTACT_Blue p_top_Blue
```

The easiest way to use such a file in `simiconductor` is to use the `import` command. This command will read the specified file and execute the commands it contains. Alternatively, the generated file could be edited and then executed, but this has the disadvantage that you will not be able to write to this file again (using ‘Generate 2D’) without losing the commands you’ve added.

Here, we’ll use the ‘import’ approach and execute the following commands, either by typing them inside an interactive version of the program or by storing them in a file and reading this as input:

```
import bilayer2d_grid.txt
sim2/grid/init n
sim2/grid/init p
```

```
sim2/rundirect
sim2/iv VBI -0.2 0.2 * iv_2d.txt no
```

First, we'll load the `bilayer2d_grid.txt` file so that the situation from the region editor is loaded inside `simiconductor`. Next, we'll initialize the electrons and holes throughout the device so that the boundary values are interpolated on a logarithmic scale. If these commands are not used, the values of n and p inside the device are not changed and will be zero. Usually, initializing them is a good idea to help the search for equilibrium. Then, the `sim2/rundirect` command is used to search for the equilibrium situation using the Newton-Raphson method. Finally, the current-voltage curve is calculated for the applied voltage range $[-0.2 : 0.2]$ and stored in a file called `iv_2d.txt`. In this last command we're using the variable called `VBI` for the built-in voltage, which was actually defined in the imported file `bilayer2d_grid.txt`.

The resulting current-voltage curve can be seen in the left panel of Fig. 3.6, which looks like the expected exponential-like shape. When looking at the part below the X-axis (shown in the right panel), it is clear that the curve nicely passes through the origin, an illustration that the constraints placed on the variables described in the previous chapter (and derived in the appendix) make sense.

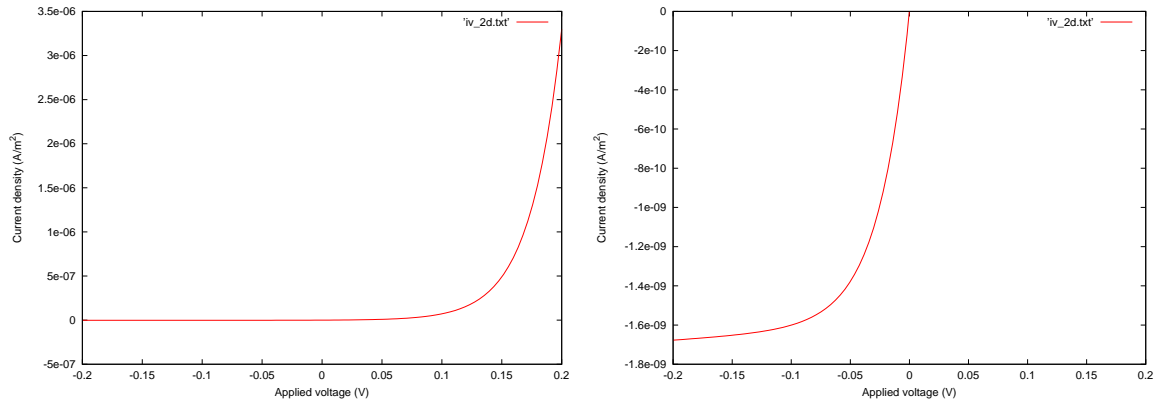


Figure 3.6: The JV curve for the bilayer described in the text. The left panel shows the overall curve, while the right panel shows the part below the X-axis. From this last figure it is clear that this dark current (the generation rate was set to zero) nicely passes through the origin.

3.5 1D simulations

Due to the periodic boundary conditions in the X-direction and the fact that each column of the image was the same, we could also have created a 1D simulation and obtained the same results. For a 1D simulation, the procedure is basically the same, but the image used is only allowed to have one column, so the number of pixels in the X-direction must be one. Importing such a figure will by default show something very narrow, but we can adjust the physical width to something else, e.g. to $32nm$, to make it easier to see the layout. This setting will be ignored for calculations though. Creating the exact same situation as in the previous example, will then yield a view like in Fig. 3.7. This looks the same as Fig. 3.5, but the number of pixels in the X-direction is one, and we've set the width to $32e-9$, so to $32nm$. When comparing such a situation to the one in Fig. 2.1, remember that the situation

is rotated, and the right contact from Fig. 2.1 now corresponds to the top one in the region editor, while the left contact from Fig. 2.1 corresponds to the bottom one in the region editor.

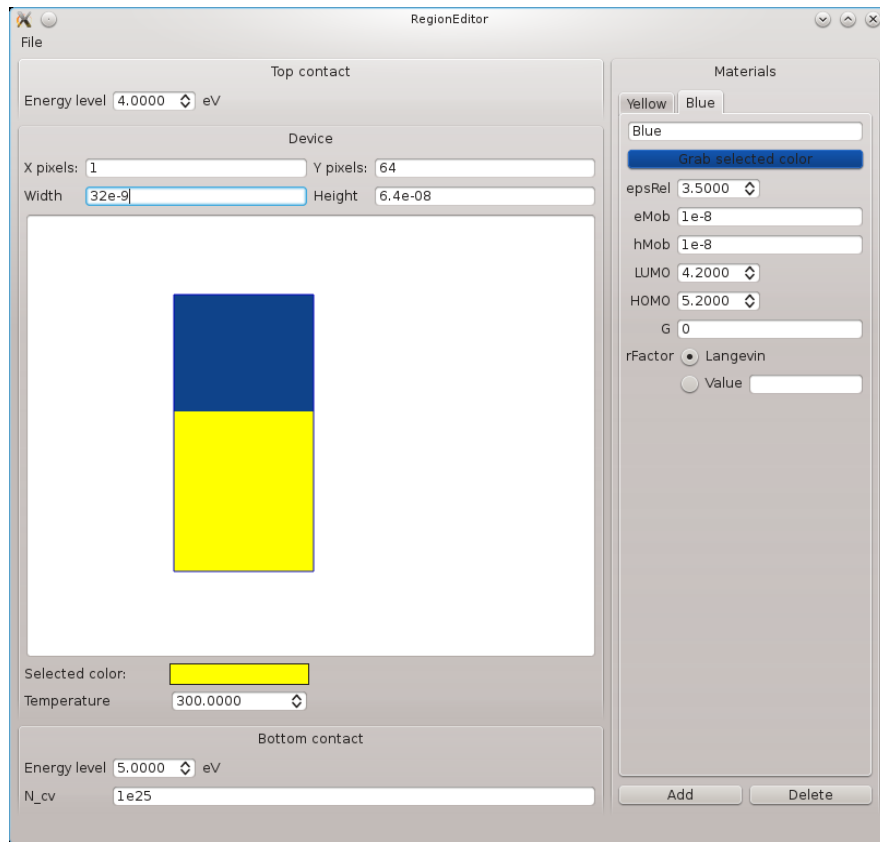


Figure 3.7: The same settings as in the 2D simulation, but now for a 1D one. Notice that the number of X pixels is one now.

From the 'File' menu, we can then export the necessary commands for the `simiconductor` program using the 'Generate 1D' option. Suppose we've written these commands to a file called `bilayer1d_grid.txt`, then it's possible to use a similar series of commands as before to produce a current-voltage curve:

```
import bilayer1d_grid.txt
sim1/grid/init n
sim1/grid/init p
sim1/rundirect
sim1/iv VBI -0.2 0.2 * iv_1d.txt no
```

When plotted, the resulting file `iv_1d.txt` will show exactly the same graphs as in Fig. 3.6.

Bibliography

- [1] LJA Koster, ECP Smits, VD Mihailetschi, and PWM Blom. Device model for the operation of polymer/fullerene bulk heterojunction solar cells. *Physical Review B*, 72(8):085205, 2005.
- [2] K Maturova, SS Van Bavel, MM Wienk, RAJ Janssen, and Martijn Kemerink. Morphological device model for organic bulk heterojunction solar cells. *Nano letters*, 9(8):3032–3037, 2009.
- [3] Klára Maturová, Svetlana S Van Bavel, Martijn M Wienk, René AJ Janssen, and Martijn Kemerink. Description of the morphology dependent charge transport and performance of polymer: Fullerene bulk heterojunction solar cells. *Advanced Functional Materials*, 21(2):261–269, 2011.
- [4] D.L. Scharfetter and H.K. Gummel. Large-signal analysis of a silicon read diode oscillator. *IEEE Transactions on Electron Devices*, 16(1):64–77, 1969.

Appendix A

Analytical description

A.1 Basic case

To obtain some insight into how the boundary conditions, internal voltage and the n_i value should be set in a consistent way, let's investigate a 1D case in which only relatively low carrier densities are present. Due to the low density, the carriers themselves do not influence the electrostatic potential which is then only determined by the boundary values and varies in a linear way:

$$\phi(x) = \frac{x}{L}(V(L) - V(0))$$

In the rest of this section, we'll be interested in the dark equilibrium situation: there's no carrier generation due to an external energy source, the potential difference corresponds to the internal potential and there's no current generated. So in this case:

$$\phi(x) = \frac{x}{L}V_{\text{int}}$$

Let's further assume that the Einstein relation between mobilities and diffusion coefficients holds:

$$D = \frac{\mu k_B T}{e}$$

The equations for the currents then become:

$$\begin{aligned} J_n &= -\mu_n \frac{k_B T}{e} \frac{dn}{dx} + \mu_n n \frac{d\phi}{dx} \\ &= -\mu_n \frac{k_B T}{e} \frac{dn}{dx} + \mu_n n \frac{V_{\text{int}}}{L} \end{aligned}$$

and

$$\begin{aligned} J_p &= -\mu_p \frac{k_B T}{e} \frac{dp}{dx} - \mu_p p \frac{d\phi}{dx} \\ &= -\mu_p \frac{k_B T}{e} \frac{dp}{dx} - \mu_p p \frac{V_{\text{int}}}{L} \end{aligned}$$

In the dark equilibrium situation, no currents should flow through the device,

$$J_n = J_p = 0,$$

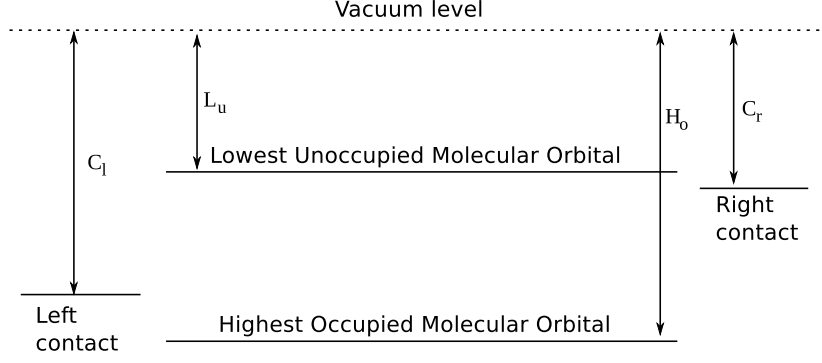


Figure A.1: Energy levels of the device before the different parts come into contact.

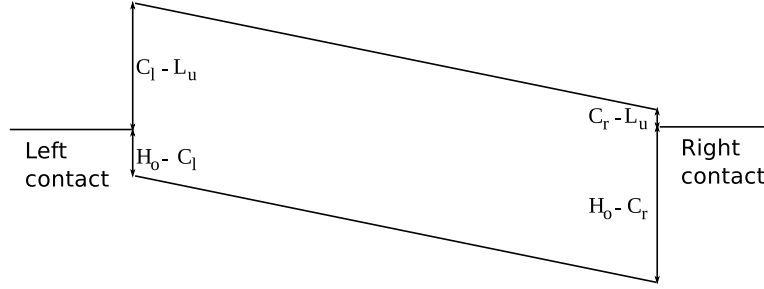


Figure A.2: When the different parts come into contact, the energy levels of the contacts align, causing specific injection barriers.

which then immediately yields equations for the charge carrier densities:

$$\frac{dn}{dx} = \frac{eV_{\text{int}}}{Lk_B T} n \Leftrightarrow n(x) = N_0 \exp\left(\frac{eV_{\text{int}}}{Lk_B T} x\right),$$

$$\frac{dp}{dx} = -\frac{eV_{\text{int}}}{Lk_B T} p \Leftrightarrow p(x) = P_0 \exp\left(-\frac{eV_{\text{int}}}{Lk_B T} x\right),$$

where N_0 and P_0 are constants. From these expressions it is clear that

$$n(0) = N_0 \quad p(0) = P_0.$$

These values, N_0 and P_0 , depend on the boundary conditions, which in turn depend on the Metal-Insulator-Metal model we're using. Suppose that when the metal contacts and the device itself are not connected, the energy bands are those depicted in Fig. A.1. When the different components are brought together, the Fermi levels of the contacts align, creating a diagram such as the one shown in Fig. A.2.

Assuming that the carrier densities at the contacts are described by the injection barriers shown in the figure, the left side boundary conditions become:

$$n(0) = N_{cv} \exp\left(-\frac{C_l - L_u}{k_B T}\right) \equiv N_0$$

$$p(0) = N_{cv} \exp\left(-\frac{H_o - C_l}{k_B T}\right) \equiv P_0$$

From the equations for $n(x)$ and $p(x)$, one finds for the other contact that

$$n(L) = N_0 \exp\left(\frac{eV_{\text{int}}}{Lk_B T} L\right) = N_{cv} \exp\left(-\frac{C_l - L_u}{k_B T}\right) \exp\left(\frac{eV_{\text{int}}}{Lk_B T} L\right),$$

$$p(L) = P_0 \exp\left(-\frac{eV_{\text{int}}}{Lk_B T}L\right) = N_{cv} \exp\left(-\frac{H_o - C_l}{k_B T}\right) \exp\left(-\frac{eV_{\text{int}}}{Lk_B T}L\right).$$

On the other hand, we want the same type of boundary conditions on the right side, i.e. based on the injection barriers:

$$n(L) = N_{cv} \exp\left(-\frac{C_r - L_u}{k_B T}\right),$$

$$p(L) = N_{cv} \exp\left(-\frac{H_o - C_r}{k_B T}\right).$$

Comparing these expressions for $n(L)$ gives us the value of V_{int} :

$$C_l - L_u - eV_{\text{int}} = C_r - L_u \Leftrightarrow V_{\text{int}} = \frac{C_l - C_r}{e}$$

And comparing the expressions for $p(L)$ confirms this:

$$H_o - C_l + eV_{\text{int}} = H_o - C_r \Leftrightarrow V_{\text{int}} = \frac{C_l - C_r}{e}$$

Let's also consider a simple bimolecular recombination model of the type $R = r(np - n_i^2)$. Note that the n_i part actually corresponds to some kind of internal generation $G_{\text{int}} = r n_i^2$ and represents carrier generation due to thermal equilibrium, present even when no generation due to some external energy source is present.

In equilibrium, the carrier densities no longer change:

$$\frac{\partial n}{\partial t} = 0$$

$$\frac{\partial p}{\partial t} = 0$$

Combining this with the fact that G is zero as well as J_n and J_p , then yields an expression for n_i :

$$\begin{aligned} r(np - n_i^2) &= 0 \\ \Rightarrow n_i^2 &= N_0 P_0 \end{aligned}$$

From the results derived earlier, we can fill in the expressions for N_0 and P_0 :

$$N_0 P_0 = N_{cv}^2 \exp\left(-\frac{C_l - L_u + H_o - C_l}{k_B T}\right) = N_{cv}^2 \exp\left(-\frac{H_o - L_u}{k_B T}\right)$$

and since $n_i^2 = N_0 P_0$:

$$\Rightarrow n_i = N_{cv} \exp\left(-\frac{H_o - L_u}{2k_B T}\right).$$

A.2 A bi-layer device

Let's generalize these results for a 1D bi-layer like model, with energy bands as shown in Fig. A.3. For reasons that will become clear later on, suppose we introduce two electrostatic potentials $V_n(x)$ and $V_p(x)$ which only affect electrons and holes respectively, but which are not influenced by the presence of charge carriers (as opposed to ϕ which depends on their

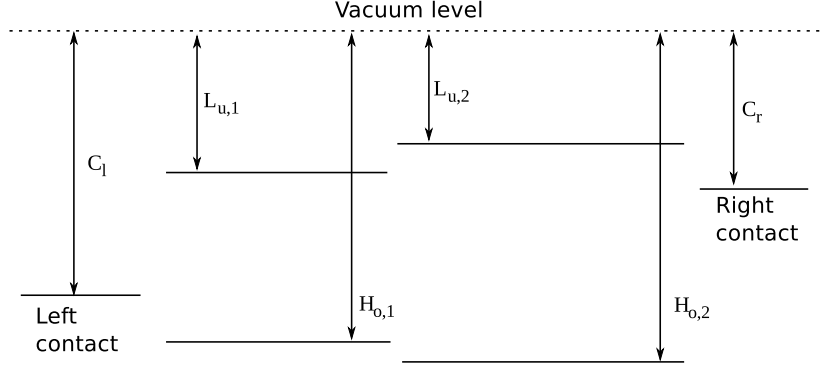


Figure A.3: Energy levels for a bi-layer device.

concentrations through the Poisson equation). We'll also assume again that the charge carrier concentrations are low, meaning that we can forget the Poisson equation and write

$$\phi(x) = \frac{x}{L} V_{\text{int}}$$

The 1D current equations

$$\begin{aligned} J_n &= -\mu_n \frac{k_B T}{e} \frac{dn}{dx} + \mu_n n \left(\frac{d\phi}{dx} + \frac{dV_n}{dx} \right) \\ J_p &= -\mu_p \frac{k_B T}{e} \frac{dp}{dx} - \mu_p p \left(\frac{d\phi}{dx} + \frac{dV_p}{dx} \right) \end{aligned}$$

then become

$$\begin{aligned} J_n &= -\mu_n \frac{k_B T}{e} \frac{dn}{dx} + \mu_n n \left(\frac{V_{\text{int}}}{L} + \frac{dV_n}{dx} \right) \\ J_p &= -\mu_p \frac{k_B T}{e} \frac{dp}{dx} - \mu_p p \left(\frac{V_{\text{int}}}{L} + \frac{dV_p}{dx} \right) \end{aligned}$$

Now, suppose that V_n is zero in the left part of the device, a constant value V_n^0 in the right part and varies linearly in a transition region:

$$V_n(x) = \begin{cases} 0 & \text{in } [0, d_1] \\ V_n^0 \frac{x-d_1}{d_2-d_1} & \text{in } [d_1, d_2] \\ V_n^0 & \text{in } [d_2, L] \end{cases}$$

For each of these regions separately, one can easily find the following solutions for the left part (n_l), center part (n_c) and right part (n_r):

$$\begin{aligned} n_l(x) &= A_l \exp\left(\frac{eV_{\text{int}}}{Lk_B T} x\right) && \text{in } [0, d_1] \\ n_c(x) &= A_c \exp\left(\frac{e}{k_B T} \left(\frac{V_{\text{int}}}{L} + \frac{V_n^0}{d_2-d_1}\right) x\right) && \text{in } [d_1, d_2] \\ n_r(x) &= A_r \exp\left(\frac{eV_{\text{int}}}{Lk_B T} x\right) && \text{in } [d_2, L] \end{aligned}$$

Making sure that these different solutions match at d_1 and d_2

$$n_l(d_1) = n_c(d_1) \quad n_c(d_2) = n_r(d_2)$$

one then finds:

$$\begin{aligned} n_l(x) &= N_0 \exp\left(\frac{eV_{\text{int}} x}{k_B T L}\right) \\ n_r(x) &= N_0 \exp\left(\frac{eV_n^0}{k_B T}\right) \exp\left(\frac{eV_{\text{int}} x}{k_B T L}\right) \end{aligned} \quad (\text{A.1})$$

As we're not really interested in the transition region, the solution for n_c is not shown (although it is equally straightforward). Defining $V_p(x)$ for the holes in a similar way, using the same boundaries d_1 and d_2 , one finds:

$$\begin{aligned} p_l(x) &= P_0 \exp\left(-\frac{eV_{\text{int}} x}{k_B T L}\right) \\ p_r(x) &= P_0 \exp\left(-\frac{eV_p^0}{k_B T}\right) \exp\left(-\frac{eV_{\text{int}} x}{k_B T L}\right) \end{aligned} \quad (\text{A.2})$$

These equations for electrons and holes illustrate the use of such potentials V_n^0 and V_p^0 . When the energy levels for a charge carrier are different on left and right side, one would expect a corresponding concentration difference. By comparing n_l and n_r , or similarly p_l and p_r , this is precisely what V_n^0 and V_p^0 do: they introduce concentration differences between the two sides of the bi-layer model.

At the left side boundary we again find, based on the equations, that

$$n(0) = N_0 \quad p(0) = P_0.$$

while based on the injection barrier boundary conditions:

$$\begin{aligned} n(0) &= N_{cv} \exp\left(-\frac{C_l - L_{u,1}}{k_B T}\right) \equiv N_0 \\ p(0) &= N_{cv} \exp\left(-\frac{H_{o,1} - C_l}{k_B T}\right) \equiv P_0 \end{aligned}$$

For the right side, we find

$$\begin{aligned} n_r(L) &= N_0 \exp\left(\frac{eV_n^0}{k_B T}\right) \exp\left(\frac{eV_{\text{int}}}{k_B T}\right) \\ p_r(L) &= P_0 \exp\left(-\frac{eV_p^0}{k_B T}\right) \exp\left(-\frac{eV_{\text{int}}}{k_B T}\right) \end{aligned}$$

based on the equations, and

$$\begin{aligned} n(L) &= N_{cv} \exp\left(-\frac{C_r - L_{u,2}}{k_B T}\right), \\ p(L) &= N_{cv} \exp\left(-\frac{H_{o,2} - C_r}{k_B T}\right) \end{aligned}$$

based on the boundary conditons.

Comparing these expressions while substituting the values of N_0 and P_0 based on the left boundary conditions, we find:

$$N_{cv} \exp\left(-\frac{C_l - L_{u,1}}{k_B T}\right) \exp\left(\frac{eV_n^0}{k_B T}\right) \exp\left(\frac{eV_{\text{int}}}{k_B T}\right) = N_{cv} \exp\left(-\frac{C_r - L_{u,2}}{k_B T}\right)$$

as well as

$$N_{cv} \exp\left(-\frac{H_{o,1} - C_l}{k_B T}\right) \exp\left(-\frac{eV_p^0}{k_B T}\right) \exp\left(-\frac{eV_{\text{int}}}{k_B T}\right) = N_{cv} \exp\left(-\frac{H_{o,2} - C_r}{k_B T}\right).$$

These equations simplify to

$$\begin{aligned} -C_l + L_{u,1} + eV_n^0 + eV_{\text{int}} &= -C_r + L_{u,2} \\ -H_{o,1} + C_l - eV_p^0 - eV_{\text{int}} &= -H_{o,2} + C_r \end{aligned}$$

which can be suggestively rewritten as

$$\begin{aligned} eV_{\text{int}} + eV_n^0 &= (C_l - C_r) + (L_{u,2} - L_{u,1}) \\ eV_{\text{int}} + eV_p^0 &= (C_l - C_r) + (H_{o,2} - H_{o,1}). \end{aligned}$$

Is is obvious that a solution to these equations can be obtained by setting

$$\begin{aligned} eV_{\text{int}} &= C_l - C_r \\ eV_n^0 &= L_{u,2} - L_{u,1} \\ eV_p^0 &= H_{o,2} - H_{o,1} \end{aligned}$$

Not only does this solution simplify to the earlier results in case there's only a single LUMO and HOMO level, this solution also produces the desired changes in electron and hole concentration between regions of the device (see (A.1) and (A.2)).

Now let us explore what the value of n_i should be on each side. Starting from the equation we encountered earlier

$$r(np - n_i^2) = 0$$

one finds on the left side

$$n_{i,l}^2 = n_l(0)p_l(0)$$

On the other hand, at the left boundary one finds

$$n_l(0)p_l(0) = N_{cv}^2 \exp\left(-\frac{H_{o,1} - L_{u,1}}{k_B T}\right)$$

so that we can write:

$$\begin{aligned} n_{i,l}^2 &= N_{cv}^2 \exp\left(-\frac{H_{o,1} - L_{u,1}}{k_B T}\right) \\ \Rightarrow n_{i,l} &= N_{cv} \exp\left(-\frac{H_{o,1} - L_{u,1}}{2k_B T}\right) \end{aligned}$$

Evaluating the equation using the expressions for the right side, one finds:

$$n_{i,r}^2 = n_r(L)p_r(L)$$

From the boundary conditions on the other hand, we would like to find

$$n_r(L)p_r(L) = N_{cv}^2 \exp\left(-\frac{H_{o,2} - L_{u,2}}{k_B T}\right)$$

immediately yielding the expression for n_i on the right side:

$$\Rightarrow n_{i,r} = N_{cv} \exp\left(-\frac{H_{o,2} - L_{u,2}}{2k_B T}\right)$$

A.3 General results

For a consistent description, where the absence of illumination leads to a solution which does not produce an external current, several conditions must be met when using injection barrier type boundary conditions, e.g. $n_l(0) = N_{cv} \exp\left(-\frac{C_l - L_{u,1}}{k_B T}\right)$

- Use additional potentials V_n and V_p , determined by the energy level differences to account for carrier concentration differences.
- Set the internal voltage of the device to the difference in energy levels of the contacts: $V_{\text{int}} = \frac{C_l - C_r}{e}$.
- For bimolecular recombination, set n_i in each region according to the difference between HOMO and LUMO energy level in that region, e.g. $n_{i,l} = N_{cv} \exp\left(-\frac{H_{o,1} - L_{u,1}}{2k_B T}\right)$.

The results above were derived using a straightforward 1D model with low carrier concentrations. We thoroughly verified these results, not only in a 1D setting with low carrier concentrations, but also using concentrations that cause a significant deviation of the linear potential as well as in a 2D setting using various energy levels and various morphologies. The conditions above always appear to create a consistent device description.