



Simudo: a device model for intermediate band materials

Eduard C. Dumitrescu¹ · Matthew M. Wilkins^{1,2} · Jacob J. Krich^{1,2}

Published online: 7 November 2019

© Springer Science+Business Media, LLC, part of Springer Nature 2019

Abstract

We describe Simudo, a free Poisson/drift-diffusion steady state device model for semiconductor and intermediate band materials, including self-consistent optical absorption and generation. Simudo is the first freely available device model that can treat intermediate band materials. Simudo uses the finite element method (FEM) to solve the coupled nonlinear partial differential equations in two dimensions, which is different from the standard choice of the finite volume method in essentially all commercial semiconductor device models. We present the continuous equations that Simudo solves, show the FEM formulations we have developed, and demonstrate how they allow robust convergence with double-precision floating point arithmetic. With a benchmark semiconductor *pn* junction device, we show that Simudo has a higher rate of convergence than Synopsys Sentaurus, converging to high accuracy with a considerably smaller mesh. Simudo includes many semiconductor phenomena and parameters and is designed for extensibility by the user to include many physical processes.

Keywords Device model · Intermediate band materials · Optoelectronics · Finite element method · Photovoltaics

1 Introduction

Device models are essential components of the development of semiconductor devices, from transistors to solar cells to lasers. Standard semiconductor device models, such as Synopsys Sentaurus, treat materials with 0, 1, or 2 bands (i.e., dielectrics, metals, and semiconductors, respectively) along with an electrostatic potential. At a given location in a given material, each band has its own carrier concentration, with particle motion given by diffusion and electric-field-induced drift. Since the electric field itself depends on particle motion, the resulting Poisson/drift-diffusion (PDD) equations are nonlinear and require numerical solution in the general case [1–5].

A new class of materials, called intermediate band (IB) materials, has been developed over the last 20 years with the goal of improving solar cell efficiency and producing effective

infrared photodetectors [6–10]. These IB materials are like semiconductors except they have an extra band of allowed electronic energy levels above the valence band (VB) and below the conduction band (CB), as shown in Fig. 1. Such a band structure permits optical absorption from VB to IB and from IB to CB, which is the key to the increased solar cell efficiency [6]. It is also possible to consider multiple IBs, though such materials have not yet been realized in practice [11].

Where IB devices have been made, they have not generally been highly efficient, which is believed to be largely due to fast nonradiative recombination processes [7–10, 12–15]. It has not been possible, however, to perform standard device modeling to optimize these devices, to determine the ideal layer thicknesses, doping levels, etc., since standard semiconductor device models do not allow the possibility of treating a third band. Therefore, we do not know what efficiencies existing IB materials could permit, if they were optimized. Interpreting experiments on IB materials and designing the best devices require device modeling capabilities.

In order to describe the basic physics of IB devices, one must be able to describe

1. Optical processes between CB, VB, and IB with rates dependent on IB filling fraction f ,
2. Nonradiative processes between CB, VB, and IB with rates dependent on f ,

Electronic supplementary material The online version of this article (<https://doi.org/10.1007/s10825-019-01414-3>) contains supplementary material, which is available to authorized users.

✉ Jacob J. Krich
jkrich@uottawa.ca

¹ Department of Physics, University of Ottawa, Ottawa, ON, Canada

² School of Electrical Engineering and Computer Science, University of Ottawa, Ottawa, ON, Canada

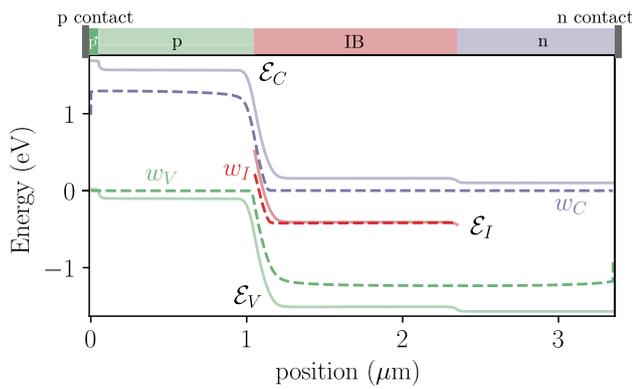


Fig. 1 Band diagram of illuminated 1D intermediate band device at short circuit. Device structure of p -IB- n device shown at top, and the IB material has an extra band contained entirely inside the semiconductor band gap. Solid lines show the band edge energies \mathcal{E}_C , \mathcal{E}_V and the IB energy \mathcal{E}_I . Dashed lines show the quasi-Fermi levels w_C , w_V , and w_I . Parameters as in Sect. 4.3

3. Carrier transport within the IB,
4. Junctions with standard semiconductors.

There is a large array of standard numerical semiconductor device models based on the coupled Poisson and carrier-continuity equations, including general purpose ones, such as Synopsys Sentaurus and Silvaco, as well as more specialized models such as Crosslight, which includes modeling of quantum well physics and a coupled treatment of carrier-density dependent optics for lasers, and TiberCAD [16]. Nextnano++ also includes features specific to quantum structures and can solve an 8-band k . p model self-consistently with the Poisson and drift-diffusion calculations [17]. There are also more focused ones, such as PC1D [18, 19], AFORS-HET [20], SCAPS [21], and Solcore [22], which are 1D models focused on solar cells. Many of these models allow treatment of deep-lying states inside the semiconductor band gap, primarily as Shockley–Read–Hall (SRH) trapping and recombination centers [23]. Sentaurus and PC1D, for example, do not permit optical generation from

the deep-lying states. SCAPS does permit both thermal and optical processes, but does not consider transport of carriers inside the defect band. We do not attempt a full characterization of all the available device models, but Table 1 shows which of these requirements are met by these device models.

There have been a number of device models developed specifically for IB materials, mostly for solar cells, all in steady state. These include traditional [6, 28–30] and Boltzmann approximation [31, 32] detailed balance models, semianalytic models in the drift [33] and diffusion [24, 34, 35] limits, and PDD models [25–27, 36]. The semianalytic models are specific to either the drift or diffusive limits, while the PDD models allow treatment of IB regions that are neither fully depleted nor fully quasi-neutral. A comparison of the features of the PDD models is also included in Table 1. To our knowledge, none has been released as open-source software.

Here, we introduce Simudo, a free and open source steady state PDD solver with self-consistent optics for arbitrary numbers of bands. Simudo uses the finite element method (FEM) to solve the coupled Poisson, drift-diffusion, and Beer-Lambert optical propagation equations self-consistently, when necessary including changing f according to local generation and recombination, with associated changes in the optical absorption coefficient. Simudo has built-in radiative recombination, Shockley–Read trapping, and SRH recombination models in the non-degenerate limit and is straightforward to extend to include other models of generation or recombination. All of the band parameters, from energies to mobilities to cross sections, can vary in space or as functions of other parameters.

Simudo has a number of innovations in its formulation of the problem, described below, and allows high-accuracy simulation of benchmark semiconductor problems while working with 64-bit arithmetic, making it useful both for standard semiconductor simulations as well as for IB devices. It is written in the Python programming language, using the FEniCS platform to solve the FEM problem [37]. It exposes an easy-to-use API for defining problems and extracting results.

Table 1 Comparison of selected device models

	IB optics with photofilling	IB nonradiative processes	IB transport	Junctions	2D
Sentaurus	N	Y	Limited	Y	Y
PC1D [18]	N	Y	N	Y	N
SCAPS [21]	Y	Y	N	Y	N
TiberCAD [16]	N	Y	Y	Y	Y
Martí [24]	N	N	Y	N	N
Strandberg [25]	Y	N	Y	N	N
Tobias [26], Yoshida [27]	Y	N	Y	Y	N
Simudo	Y	Y	Y	Y	Y

It is designed for two-dimensional systems and is available for download at <https://github.com/simudo/simudo>.

Semiconductor device modeling is typically performed using the finite volume method (FVM), which ensures local charge conservation at each cell in the domain [38]. Finite element methods are less common in semiconductor device models, though there are a number of examples [16, 39–41]. FEM is widely used for related advection-diffusion problems in computational fluid dynamics (CFD) studies [42]. Commercial packages for CFD, as for semiconductor device modeling, are generally based on the FVM method. TiberCAD, a commercial device modeling package with many novel features, uses FEM with continuous basis functions [16]. Finite element methods including discontinuous local basis functions, called discontinuous Galerkin (DG) methods, also permit local charge conservation [42], and they have recently begun to be applied to semiconductor device problems [43, 44]. FEM methods simplify consideration of complicated simulation domains and in theory allow higher-order convergence of solutions, but performance of such methods can only be determined with testing. We use such a DG-FEM method here to produce a general purpose steady state PDD solver capable of treating IB systems, and we show that Simudo realizes the higher-order convergence with mesh size, converging much more rapidly than Synopsys Sentaurus as the mesh spacing is reduced. As shown in Sect. 3.5, for a reference pn diode, Simudo demonstrates quartic self-convergence with mesh density, while FVM-based Synopsys Sentaurus demonstrates only quadratic convergence. In the reference problem, Simudo achieves 5–6 digits of convergence with 193 mesh points, while Sentaurus requires more than 3000. Simudo provides both a flexible framework for the study of IB devices and also a freely available example of a DG-FEM semiconductor device model.

In Sect. 2, we define the coupled partial differential equations (PDEs) Simudo solves. Section 3 describes the heart of Simudo, giving in detail the conversion of the equations of Sect. 2 to the weak forms solved using FEM. This section describes the choices for dynamical variables, the weak forms used for FEM, and how these choices enable Simudo to achieve accuracy despite the problems of finite precision arithmetic. This section concludes with a comparison to Synopsys Sentaurus on a benchmark pn diode, showing the high quality of Simudo's results. Section 4.1 gives examples of setting up a simple problem using the API, including examples of its convenient topology definitions, and Sect. 4.2 demonstrates the extensibility of Simudo to include new physical processes (in this case, Auger recombination). Section 4.3 demonstrates the use of Simudo to analyze a system originally studied in Ref. [24], showing that its model works better than had been anticipated in the case with equal subgap optical absorption cross sections, but that unequal subgap absorption cross sections produce more complicated phenomena that require IB transport to describe properly.

2 Statement of problem

In this section, we describe the mathematical model of the steady state PDD and optical problems we use in Simudo. Carriers both drift in response to electric fields and diffuse. Carriers are generated optically and recombine using a variety of models. The local carrier concentration determines both the electric field and the optical absorption coefficients, so the transport, Poisson, and optical propagation equations are all coupled. Symbols used in this manuscript are summarized in Table 2.

2.1 Carrier transport and generation

We consider a CB, a VB, and some number of IBs under the assumption that the carrier population in each band is in local quasi-equilibrium with a temperature T and quasi-Fermi level w_k , where k can be one of $\{C, V, I\}$ for the CB, VB, and IB, respectively. In the case of multiple IBs, k can take values I_1, I_2, \dots , indexing the various IBs, but we simplify the following discussion to consider the case of just one IB, indexed as I .

In the most common approximation of semiconductor device modeling, the carrier dynamics in each band can be described by the drift-diffusion equation and the continuity equation. Letting u_k represent the carrier concentration in band k , u_v and u_c are the hole and electron concentrations, respectively, which we use interchangeably with their standard symbols, p and n . We let $s_k = \pm 1$ give the charge of the carriers in band k , +1 for the VB and -1 for the CB. Then,

$$\mathbf{j}_k = \overbrace{q\mu_k u_k \mathbf{E}}^{\text{drift}} - \overbrace{s_k q \mathcal{D}_k \nabla u_k}^{\text{diffusion}} \quad (1a)$$

$$\frac{\partial u_k}{\partial t} = -s_k \frac{1}{q} \nabla \cdot \mathbf{j}_k + g_k, \quad (1b)$$

where \mathbf{j}_k is the current density of carriers in band k , μ_k is the carrier mobility, \mathcal{D}_k is the carrier diffusion constant, \mathbf{E} is the electric field, q is the elementary charge, and g_k contains all the generation, trapping, and recombination processes (see Sect. 2.1). For non-degenerate bands in which w_k is sufficiently far from the band edge $\mathcal{E}_k - q\phi$, we can write

$$u_k = N_k e^{-s_k(w_k + q\phi - \mathcal{E}_k)/k_B T}, \quad (2)$$

where N_k is the effective density of states of band k , ϕ is the electrostatic potential, and k_B is Boltzmann's constant.

Then, assuming \mathcal{E}_k is spatially constant,

$$\nabla u_k = -s_k N_k e^{-s_k(w_k + q\phi - \mathcal{E}_k)/k_B T} \frac{1}{k_B T} \nabla(w_k + q\phi) \quad (3)$$

$$= -s_k \frac{u_k}{k_B T} \nabla(w_k + q\phi), \quad (4)$$

Table 2 Common symbols used in this manuscript

Symbol	Definition
u_k	Carrier density in band k
\mathcal{E}_k	Band edge energy of band k ; central energy of IB k
s_k	Sign of carriers in band k (+ for VB, – for CB)
w_k	Quasi-Fermi level of carriers in band k
\mathbf{j}_k	Current density in band k
μ_k	Mobility of carriers in band k
\mathcal{D}_k	Diffusion coefficient of carriers in band k
S_k	Surface recombination velocity of carriers in band k
N_k	Effective density of states for nondegenerate band k
	Integrated density of states for intermediate band k
f_k	Filling fraction $f_k = u_k/N_k$ of IB k
$f_{k,\pm}$	$(1 - f_k)$ for positive sign and f_k for negative sign
$f_{k,0}$	Charge neutral filling fraction of IB k
g_k	Net generation in band k due to all generation and recombination processes
T	Temperature
k_B	Boltzmann constant
q	Elementary charge
ϕ, \mathbf{E}, ρ	Electrostatic potential, electric field, and charge density
$\alpha_{fi,\lambda}$	Optical absorption coefficient from band i to f at vacuum wavelength λ
σ_{fi}^{opt}	Optical cross section from band i to f
$\Phi_{\lambda,\hat{s}}$	Photon spectral flux density at wavelength λ in direction \hat{s}
$\Phi_{[\lambda_1,\lambda_2],\hat{s}}$	Photon flux in direction \hat{s} from λ_1 to λ_2 , i.e., $\int_{\lambda_1}^{\lambda_2} \Phi_{\lambda,\hat{s}} d\lambda$
\hat{n}	Surface normal vector
\hat{s}	Direction of light propagation

For such nondegenerate bands, the Einstein relation gives $\mu_k = q\mathcal{D}_k/k_B T$, from which Eq. 1a gives [2]

$$\mathbf{j}_k = \mu_k u_k \nabla w_k, \tag{5}$$

which we use instead of Eq. 1a. Equation 5 also applies to the case of degenerate bands, as shown in [45], even though the Einstein relation requires a modification. Moreover, Eq. 5 applies in the case of spatially varying band structure (e.g., spatially varying N_c, \mathcal{E}_c) [46], so it is considerably more general than this derivation.

Since an intermediate band is often partially filled, we cannot model it using the non-degenerate approximation of Eq. 2. We write $D_I(E)$ for the density of states of the IB, such that $N_I = \int dE D_I(E)$ is the total density of IB states. If the IB has quasi-Fermi level w_I , the electron concentration is

$$u_I = \int dE \frac{D_I(E)}{e^{(E-w_I-q\phi)/k_B T} + 1}. \tag{6}$$

If the bandwidth of the IB is narrow relative to $k_B T$, we can approximate the IB density of states as a Dirac delta $D_I(E) = N_I \delta(E - \mathcal{E}_I)$, and so

$$u_I = N_I \underbrace{\frac{1}{e^{(\mathcal{E}_I-w_I-q\phi)/k_B T} + 1}}_{f_I}, \tag{7}$$

where f_I is the filling fraction of the IB and can be written as $f_I = f(\mathcal{E}_I - w_I - q\phi)$ where $f(E)$ is the Fermi function. We work in this limit for the remainder of this manuscript. Extending beyond this sharp-IB case is not difficult but requires more cumbersome notation.

2.2 Carrier generation and recombination

Each band’s continuity equation (Eq. 1b) has a generation term g_k . This term is the sum of contributions from all generation and recombination processes to the band, which depend on which physical models are included in the simulations. We now specify the details of optical generation g_k^{opt} and a variety of recombination processes r_k , each of which enters either as a negative or positive contribution to g_k , as required for the process.

2.2.1 Optical carrier generation

Modeling optical carrier generation requires modeling the changing light intensity through the device. We use a simple Beer–Lambert model for optical propagation and absorption

$$\nabla \Phi_{\lambda,\hat{s}} \cdot \hat{s} = -\alpha_\lambda \Phi_{\lambda,\hat{s}} \tag{8}$$

where $\Phi_{\lambda,\hat{s}}$ is the photon spectral flux at vacuum wavelength λ and direction of propagation \hat{s} and α_λ is the total absorption coefficient, which can be written as

$$\alpha_\lambda = \sum_{i,f} \alpha_{fi,\lambda},$$

where $\alpha_{fi,\lambda}$ is the absorption coefficient for the optical process at wavelength λ that moves a carrier from band i to band f . In the usual semiconductor case, $\alpha_{VC,\lambda} = 0$ and $\alpha_{CV,\lambda}$ is finite for λ corresponding to energies larger than the band gap. Free carrier absorption is included in $\alpha_{ii,\lambda}$. The carrier generation rate in band k due to optical processes is then

$$g_k^{\text{opt}} = -s_k \int d\lambda \left(\sum_i \alpha_{ki,\lambda} - \sum_f \alpha_{fk,\lambda} \right) \Phi_{\lambda,\hat{s}}. \tag{9}$$

Further details of the optical propagation model are described in Sect. 2.3.

In nondegenerate bands, there are always enough carriers to excite in or out of a band. That is, the valence band always has electrons available, and the conduction band has empty states available to be filled, so the absorption coefficient $\alpha_{CV,\lambda}$ is insensitive to the free carrier density in the bands. In an IB, however, the VB \rightarrow IB process requires empty states in the IB while the IB \rightarrow CB process requires filled states in the IB. To capture this phenomenon, we write

$$\alpha_{CI,\lambda} = \sigma_{CI,\lambda}^{\text{opt}} u_I \tag{10}$$

$$\alpha_{IV,\lambda} = \sigma_{IV,\lambda}^{\text{opt}} (N_I - u_I) \tag{11}$$

where $\sigma_{fi,\lambda}^{\text{opt}}$ is the optical capture cross section from band i to f at wavelength λ of a single intermediate state. We can combine these equations into a single expression,

$$\alpha_{kl,\lambda} = \sigma_{kl,\lambda}^{\text{opt}} u_{I,s_k}, \tag{12}$$

where $u_{I,-}$ is just u_I and $u_{I,+} = N_I - u_I$ is the number of holes in band I , and α_{kl} is understood to be α_{IV} when $k = V$. Since $\alpha_{fi,\lambda}$ depends on the carrier concentrations, and the carrier concentrations depend on $\alpha_{fi,\lambda}$ (through the generation rate g_k^{opt}), the transport and the optical models feed into each other, so they must be solved in a self-consistent manner.

2.2.2 Recombination and trapping

Simudo offers several built-in radiative and nonradiative recombination and trapping mechanisms using the non-degenerate limit for the CB and VB, each including an equivalent thermal generation. An example is the SRH recombination model with a single trap level at energy \mathcal{E}_I [23], in which

two trapping processes (of an electron and a hole) produce a recombination event, with recombination rate

$$r^{\text{SRH}} = \frac{pn - n_i^2}{(p + p_1)\tau_n + (n + n_1)\tau_p}, \tag{13}$$

where τ_p, τ_n are the carrier lifetimes and p_1, n_1 are the carrier concentrations of holes and electrons, respectively, if their quasi-Fermi levels were equal to \mathcal{E}_I . This r^{SRH} appears as a negative contribution to g_k for both CB and VB.

We can model traps as intermediate bands with w_I tracked explicitly, in which case we implement standard Shockley–Read trapping [23],

$$r_{Ik}^{\text{SR}} = [1 - e^{s_k(w_k - w_I)/k_B T}] f_{I,-s_k} u_k / \tau_k, \tag{14}$$

where $f_{I,-s_k}$ is the IB filling fraction of carriers with charge $-s_k$, and τ_k is the Shockley–Read lifetime for band k , as in Eq. 13 [23]. Note that r_{IC}^{SR} makes a negative contribution to g_C and a positive contribution to g_I , while r_{IV}^{SR} makes a negative contribution to both g_V and g_I .

Simudo also implements radiative trapping from band $k = C, V$ to I . When we use Boltzmann statistics rather than Bose statistics for the emitted photons, which is valid when $|w_k - w_I|$ remains at least a few $k_B T$ below $|\mathcal{E}_k - \mathcal{E}_I|$, as in Ref. [25], then the radiative trapping can be written

$$r_{Ik}^{\text{rad}} = [e^{-s_k(w_k - w_I)/k_B T} - 1] u_{I,s_k} \mathcal{I}_{Ik}, \tag{15}$$

where

$$\mathcal{I}_{Ik} = \frac{8\pi n_r^2}{h^3 c^2} \int_0^\infty \sigma_{Ik}^{\text{opt}}(E) E^2 e^{-E/k_B T} dE, \tag{16}$$

where n_r is the index of refraction, and $u_{k,1}$ is either n_1 or p_1 for $k = C, V$, respectively. Note that Ref. [25] includes only the recombination term, and we add the corresponding thermal generation term, which is the -1 in Eq. 15. We can re-express Eq. 15 in a similar form to the nonradiative terms by using the relation

$$u_{I,s_k} = \frac{u_{I,-s_k} u_k}{u_{k,1}} e^{s_k(w_k - w_I)/k_B T},$$

which follows from Eqs. 2, 7. Then,

$$r_{Ik}^{\text{rad}} = [1 - e^{s_k(w_k - w_I)/k_B T}] \frac{u_{I,-s_k} u_k}{u_{k,1}} \mathcal{I}_{Ik}. \tag{17}$$

As with Eq. 2, Eqs. 14–16 are valid in the non-degenerate limit where w_k does not approach \mathcal{E}_k , but full degenerate statistics are used for the IB. Extensions to the degenerate limit can be added, if desired. Simudo also treats standard radiative recombination between conduction and valence bands [47].

We also treat surface recombination at external surfaces Γ of the device, which imposes a boundary condition

$$\mathbf{j}_k \cdot \hat{n}|_{\Gamma} = S_k(u_k - u_{k0})|_{\Gamma}, \quad (18)$$

where S_k is the surface recombination velocity of carriers in band k at boundary Γ , \hat{n} is the normal to Γ , and u_{k0} is the carrier concentration at equilibrium [48]. The current release of Simudo supports only $S_k = 0$ or ∞ , which impose $\mathbf{j}_k \cdot \hat{n}|_{\Gamma} = 0$ or $(u_k - u_{k0})|_{\Gamma} = 0$, respectively.

2.3 Optical equations

For each wavelength, we need to solve the optical propagation according to Eq. 8. For stability of the numerical solution, it is convenient to use a second-order equation so that we can apply boundary conditions on both the inlet and outlet boundaries [49]. We take the derivative of Eq. 8 with respect to the direction of propagation,

$$\hat{s} \cdot \nabla(\hat{s} \cdot \nabla \Phi_{\lambda, \hat{s}}) + \hat{s} \cdot \nabla(\alpha_{\lambda} \Phi_{\lambda, \hat{s}}) = 0, \quad (19)$$

With no reflection from the back, the boundary conditions are then

$$\Phi_{\lambda, \hat{s}} = \Phi_{\lambda, \hat{s}}^0 \quad \mathbf{x} \in \Gamma_i \quad (\text{inlet}) \quad (20a)$$

$$\hat{s} \cdot \nabla \Phi_{\lambda, \hat{s}} + \alpha_{\lambda} \Phi_{\lambda, \hat{s}} = 0 \quad \mathbf{x} \in \Gamma_o \quad (\text{outlet}) \quad (20b)$$

where $\Phi_{\lambda, \hat{s}}^0$ is a spectral photon flux at the inlet boundary.

In the case where α_{λ} is constant for λ in an interval $[\lambda_1, \lambda_2]$, the optical flux at all wavelengths in that range obeys Eq. 19 and can thus be treated together. We can write

$$\Phi_{[\lambda_1, \lambda_2], \hat{s}} = \int_{\lambda_1}^{\lambda_2} \Phi_{\lambda, \hat{s}} d\lambda,$$

where $\Phi_{[\lambda_1, \lambda_2], \hat{s}}$ is a photon flux (where $\Phi_{\lambda, \hat{s}}$ is a spectral photon flux). In this case, we have

$$\hat{s} \cdot \nabla(\hat{s} \cdot \nabla \Phi_{[\lambda_1, \lambda_2], \hat{s}}) + \hat{s} \cdot \nabla(\alpha_{\lambda} \Phi_{[\lambda_1, \lambda_2], \hat{s}}) = 0.$$

Simudo uses this form, which allows simple treatment of piecewise constant absorption coefficients with a small number of optical fields $\Phi_{[\lambda_1, \lambda_2], \hat{s}}$. When optical fields with only one propagation direction \hat{s} are considered, we write the spectral flux density Φ_{λ} and the flux density $\Phi_{[\lambda_1, \lambda_2]}$.

2.4 Poisson's equation

In electrostatics, Poisson's equation relates ϕ , the charge density ρ , and the permittivity ϵ ,

$$\nabla \cdot (\epsilon \nabla \phi) = -\rho. \quad (21)$$

It can also be split into two equations

$$\nabla \phi = -\mathbf{E}, \quad (22a)$$

$$\nabla \cdot (\epsilon \mathbf{E}) = \rho, \quad (22b)$$

where \mathbf{E} is the electric field.

The charge density ρ is the sum of the static charge and the mobile charge in each band. In an IB material,

$$\rho = q[-n + p - N_I(f_I - f_{I,0}) + N_D - N_A], \quad (23)$$

where N_A , N_D are the shallow acceptor and donor doping concentrations, respectively, and the mobile charge in the IB is $qN_I(f_I - f_{I,0})$, with $f_{I,0}$ the IB filling fraction of the bulk IB material at $T = 0$ K. For a donor-type IB, $f_{I,0} = 1$, and for an acceptor-type IB, $f_{I,0} = 0$. Note that in writing the shallow dopant terms N_D and N_A , we are assuming complete ionization of these impurities.

3 Numerical method

Simudo uses the finite element method (FEM) to solve the coupled Poisson/drift-diffusion and optics problems, given by Eqs. 1b, 5, 19, and 22. The FEM method divides the simulation domain Ω into cells, which are generally triangles in 2D and enforces a weak form of the desired PDE's with a set of test functions defined on those cells, with boundary conditions applied on the domain boundary Γ . The method is well-described in many reference texts [50–52]. In this section, we detail the weak forms used for these coupled equations and the solution method for the resulting nonlinear system. We benchmark Simudo against the industry standard Synopsys Sentaurus commercial simulator on a standard semiconductor problem to show the quality of our results.

3.1 Solution method

The PDD problem is a coupled nonlinear system of PDEs, which we solve iteratively using Newton's method as implemented in the FEniCS package. The solution procedure is outlined in Fig. 2. The goal is to find a solution $y = (\phi, \mathbf{E}, w_C, w_I, w_V, \mathbf{j}_C, \mathbf{j}_I, \mathbf{j}_V)$ that satisfies Eqs. 1b, 5, 22 and associated Φ_{λ} that obeys Eq. 19. The optical problem is solved alongside the PDD problem in a self-consistent manner. That is, the PDD subproblem produces the absorption coefficient $\alpha(\lambda)$ (which, for processes involving the IB, depends on the filling fraction). The optical subproblem is then solved using these absorption coefficients, yielding a new photon flux Φ_{λ} , which is fed back into the PDD where it enters in the optical carrier generation process, and the cycle iterates until a self-consistent solution is found.

The convergence of Newton's method depends on the quality of the initial guess. Steps 1 and 2 in Fig. 2 are the pre-solver, which is used once to make the initial guess for

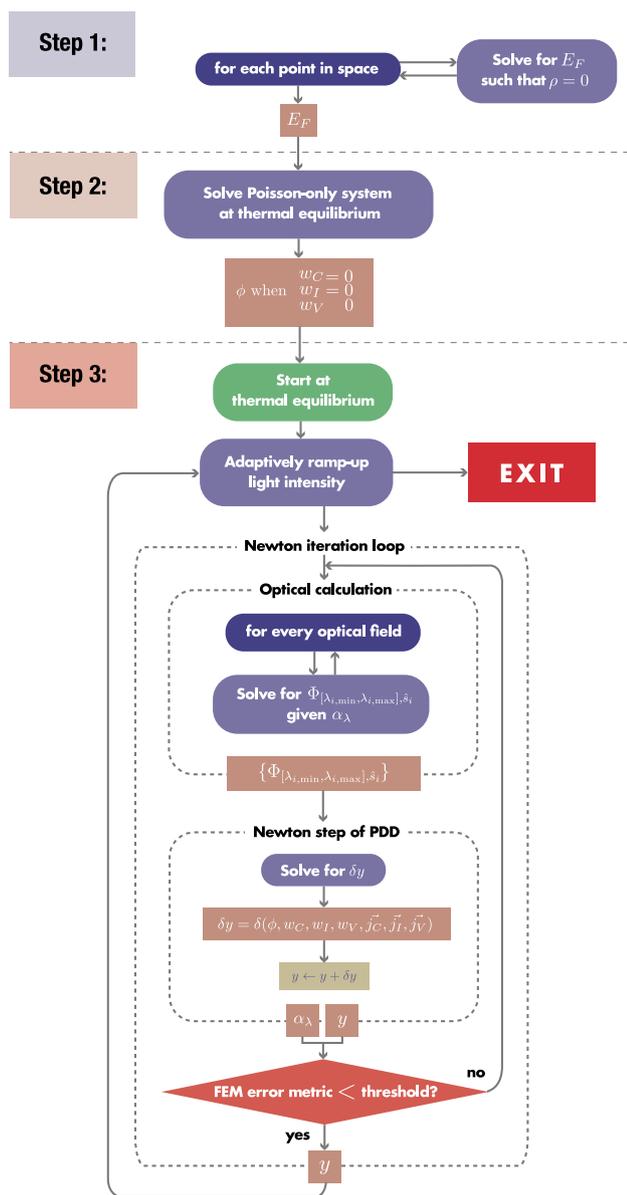


Fig. 2 Workflow of the numerical method used in Simudo. Steps 1 and 2 are pre-solver steps, which construct an initial guess for the main Newton loop (step 3)

the main Newton solver, illustrated in step 3. The full procedure is:

1. For each point in space, calculate the equilibrium Fermi level E_F at that point assuming local charge neutrality and $\phi = 0$. Physically, this step finds the E_F in each location that makes it charge neutral, before any charge is allowed to flow.
2. Determine the built-in potential ϕ_{bi} of the equilibrium system, using $\phi_0 = E_F/q$ as the initial guess. That is, solve only Eq. 22 for ϕ while keeping all $w_k = 0$ and

thus all $\mathbf{j}_k = 0$. Physically, this step allows charge to move, forming depletion regions as the carriers move to achieve a zero-current configuration that satisfies Poisson’s equation. The carrier density inside bulk-like regions of space changes little from the bulk equilibrium value in step 1, making ϕ_0 an excellent guess in large regions of space.

3. *Main solver loop* Adaptively ramp up light intensity and/or bias, starting with thermal equilibrium (dark, no bias). Each solution requires a loop of Newton iterations. Within each Newton iteration, do the following:

- (a) *Optical calculation* For each optical field λ , solve for the photon flux Φ_λ given the latest value of α_λ . Note that Eq. 19 is linear when α_λ is fixed.
- (b) *PDD Newton step* Perform one Newton step of the PDD problem.
 - i. Solve for $\delta y = \delta(\phi, \mathbf{E}, w_C, w_I, w_V, \mathbf{j}_C, \mathbf{j}_I, \mathbf{j}_V)$. Use the value of Φ_λ (and thus optical carrier generation) computed in the previous step.
 - ii. Update $y \leftarrow y + \delta y$. As an option, logarithmic damping can be applied to δy to prevent Newton’s method from diverging, e.g., $y \leftarrow y + \text{LogDamping}(\delta y)$ where $\text{LogDamping}(z) = \text{sgn}(z) \log(1 + c|z|)/c$ for $c = 1.72$ or other user-defined value [53].

We now describe the weak forms that we use for each of Eqs. 1b, 5, 19, and 21. There is much flexibility in the choice of particular weak forms, all of which can be equivalent to the same strong form. In Sect. 3.3.2, we illustrate the use of a partitioned offset representation for w_k , which allows internal currents to be calculated accurately with double-precision arithmetic.

3.2 Poisson equation

Here, we introduce the formulation we use to implement Eq. 22. We use a mixed method to solve for both ϕ and \mathbf{E} explicitly [54, 55]. The potential ϕ is represented as a superposition of discontinuous Galerkin (DG) basis functions of order $d_{\text{poisson}} - 1$ (cell-wise discontinuous polynomials), and \mathbf{E} is represented using Brezzi–Douglas–Marini basis functions of order d_{poisson} (cell-wise discontinuous polynomials with continuous normal component on cell boundaries) [54]. We use the BDM space for all vector quantities, including \mathbf{E} and \mathbf{j}_k . The BDM space is $H(\text{div})$ conforming, meaning

the divergence is accurately calculated and fluxes between cells are preserved, which makes it a natural choice for conserved or almost conserved vector quantities.¹ While \mathbf{j}_k is not a conserved quantity, due to generation and recombination that occur inside of cells, the BDM space ensures that \mathbf{j}_k is accurately preserved when passing between cells. A method using CG or DG functions for \mathbf{j} would be susceptible to numerical errors associated with non-conservation of currents between cells, and we show in Sect. 3.3.2 that Simudo conserves current well in a pn diode. In the results below, $d_{\text{poisson}} = 2$.

We multiply Eq. 22a by test function $\boldsymbol{\psi} \in \text{BDM}(d_{\text{poisson}})$ and Eq. 22b by test function $v \in \text{DG}(d_{\text{poisson}} - 1)$, then integrate each spatially, giving the weak forms

$$\underbrace{\oint_{\Gamma} \boldsymbol{\psi} \cdot \hat{n} \phi_{\text{BC}}}_{\text{natural BC}} - \int_{\Omega} (\nabla \cdot \boldsymbol{\psi}) \phi + \int_{\Omega} \boldsymbol{\psi} \cdot \mathbf{E} = 0 \quad (24)$$

$$\int_{\Omega} v (\nabla \cdot (\epsilon \mathbf{E})) - \int_{\Omega} v \rho = 0, \quad (25)$$

which must hold for every test function $\boldsymbol{\psi}$ and v , where Ω is the full domain and Γ is the boundary. Note that Eq. 24 includes an integration by parts. In this case, the electric field BC is an essential BC, imposed by reducing the set of test functions to those that satisfy the BC, while the potential BC is a natural BC.

3.3 Transport equations

The drift-diffusion equations are often numerically challenging to solve in semiconductors. In carrier density-based formulations, poor resolution of the gradients of u_k makes linear interpolation of current density unstable, which the Scharfetter–Gummel box method corrects for FVM methods [1]. Additionally, catastrophic cancellation can occur in Eq. 1a, e.g., for the majority carrier in a quasi-neutral region of a semiconductor, when the drift and diffusion contributions are nearly equal in magnitude. The current is given by the difference and can be hard to evaluate with finite precision arithmetic. We address these issues by using a quasi-Fermi-level-based representation for carrier density [40, 56]. Calculating ∇w_k in finite precision for Eq. 5 can also be challenging when w_k is very flat, and in Sect. 3.3.2, we introduce a partitioned offset representation for w_k to allow accurate determination of ∇w_k with essentially no extra

computational cost. We use a mixed FEM method that solves explicitly for both w_k and the current density \mathbf{j}_k . As described in Sect. 3.2, the BDM space of basis functions enforces local current conservation in the solutions, which also enables local current densities to be well determined. Without the mixed method, local current conservation is enforced indirectly, and we were not able to obtain well-converged results for local currents.

3.3.1 Quasi-Fermi level formulation

The quasi-Fermi level w_k is represented as a superposition of DG basis functions of order $d_{\text{transport}} - 1$, and the current density \mathbf{j}_k is represented using BDM basis functions of order $d_{\text{transport}}$. Section 3.2 contains a discussion of these functions' properties and of the mixed method. In the results below, $d_{\text{transport}} = 2$.

We derive weak forms of Eqs. 1b and 5, multiplying Eq. 1b by test function $v \in \text{DG}(d_{\text{transport}} - 1)$, taking the dot product of Eq. 5 with the test function $\boldsymbol{\psi} \in \text{BDM}(d_{\text{transport}})$, and integrating each equation spatially, giving

$$0 = \int_{\Omega} v \nabla \cdot \mathbf{j}_k - \int_{\Omega} s_k v q g_k \quad (26a)$$

$$0 = \int_{\Omega} \boldsymbol{\psi} \cdot \mathbf{j}_k / (\mu_k u_k) - \underbrace{\oint_{\Gamma} (\boldsymbol{\psi} \cdot \hat{n}) w_{k,\text{BC}}}_{\text{natural BC}} + \int_{\Omega} (\nabla \cdot \boldsymbol{\psi}) w_k, \quad (26b)$$

where the second equation was obtained by a further integration by parts.

3.3.2 Quasi-Fermi level offset partitioning

As written, Eq. 26b still suffers from a form of catastrophic cancellation in its last term, which corresponds to the gradient term in Eq. 5. Since $\int_{\Omega} \nabla \cdot \boldsymbol{\psi} = 0$ for $\boldsymbol{\psi} \in \text{BDM}(d_{\text{transport}})$, the last term is nonzero only if w_k varies within the domain where $\boldsymbol{\psi}$ is nonzero. w_k can be extremely flat, for example in quasi-neutral regions, which makes this integral hard to calculate with finite arithmetic precision. This difficulty is more apparent in Eq. 5: if $\mathbf{j}_k / \mu_k u_k = \nabla w_k$ is small,² a representation of w_k that stores its value on the nodes of the mesh cannot resolve such small changes in w_k across space.

We circumvent this issue by using an offset representation for w_k . The idea is to give each cell in the domain its own (spatially constant) base quasi-Fermi level w_{k0} relative to which the new dynamical variable δw_k is expressed. That is, $w_k = w_{k0} + \delta w_k$ where δw_k is the quantity we actually solve

¹ In the BDM space, the normal fluxes are shared by adjacent elements. The flux exiting the perimeter of a collection of cells exactly equals the sum of fluxes out of each of the cells, with exact arithmetic.

² relative to $|w_k|/(\text{mesh size})$

for instead of w_k . Before every Newton iteration step, the w_{k0} of each cell is initialized to the cell average of w_k from the previous iteration. This representation allows small spatial changes of δw_k to be accurately represented, enabling accurate determination of the current.

The last remaining question is how to adjoin regions with different base w_{k0} values. We connect them by adding a surface integral jump term to Eq. 26b, resulting in

$$0 = \int_{\Omega} \boldsymbol{\Psi} \cdot \mathbf{j}_k / (\mu_k u_k) - \oint_{\Gamma} (\boldsymbol{\Psi} \cdot \hat{\mathbf{n}}) \delta w_{k,BC} + \int_{\Omega} (\nabla \cdot \boldsymbol{\Psi}) \delta w_k + \underbrace{\sum_{f \in \text{interior facets}} \int_f (\boldsymbol{\Psi} \cdot \hat{\mathbf{n}}) [w_{k0}]}_{\text{region boundary term}} \tag{27}$$

where $[w_{k0}]$ is the jump operator, which takes the difference between the values of a discontinuous expression on either side of a facet. The rest of this section is dedicated to deriving that term and comparing the result to a formulation without the offset representation.

We substitute $w_k = w_{k0} + \delta w_k$ into Eq. 26b, and we obtain

$$0 = \int_{\Omega} \boldsymbol{\Psi} \cdot \mathbf{j}_k / (\mu_k u_k) - \oint_{\Gamma} (\boldsymbol{\Psi} \cdot \hat{\mathbf{n}}) \delta w_{k,BC} + \int_{\Omega} (\nabla \cdot \boldsymbol{\Psi}) \delta w_k - \underbrace{\oint_{\Gamma} (\boldsymbol{\Psi} \cdot \hat{\mathbf{n}}) w_{k0,BC} + \int_{\Omega} (\nabla \cdot \boldsymbol{\Psi}) w_{k0}}_{(\star)} \tag{28}$$

Our goal now is to rewrite the (\star) term. Since w_{k0} is constant on each cell K , $\nabla w_{k0} = \mathbf{0}$ within each cell. We integrate by parts using

$$\int_{\Omega} \boldsymbol{\sigma} \cdot \nabla a + \int_{\Omega} a (\nabla \cdot \boldsymbol{\sigma}) = \oint_{\Gamma} \boldsymbol{\sigma} a \cdot \hat{\mathbf{n}}, \tag{29}$$

yielding

$$\int_K \underbrace{\boldsymbol{\Psi} \cdot \nabla w_{k0}}_0 + \int_K (\nabla \cdot \boldsymbol{\Psi}) w_{k0} = \oint_{\partial K} (\boldsymbol{\Psi} \cdot \hat{\mathbf{n}}) w_{k0} \tag{30}$$

Summing over all cells K ,

$$\underbrace{\sum_K \int_K (\nabla \cdot \boldsymbol{\Psi}) w_{k0}}_{\int_{\Omega}} = \sum_K \oint_{\partial K} (\boldsymbol{\Psi} \cdot \hat{\mathbf{n}}) w_{k0} \\ \int_{\Omega} (\nabla \cdot \boldsymbol{\Psi}) w_{k0} = \oint_{\Gamma} (\boldsymbol{\Psi} \cdot \hat{\mathbf{n}}) w_{k0} + \sum_{f \in \text{interior facets}} \int_f (\boldsymbol{\Psi} \cdot \hat{\mathbf{n}}) [w_{k0}] \\ - \underbrace{\oint_{\Gamma} (\boldsymbol{\Psi} \cdot \hat{\mathbf{n}}) w_{k0} + \int_{\Omega} (\nabla \cdot \boldsymbol{\Psi}) w_{k0}}_{(\ast)} = \sum_{f \in \text{interior facets}} \int_f (\boldsymbol{\Psi} \cdot \hat{\mathbf{n}}) [w_{k0}]$$

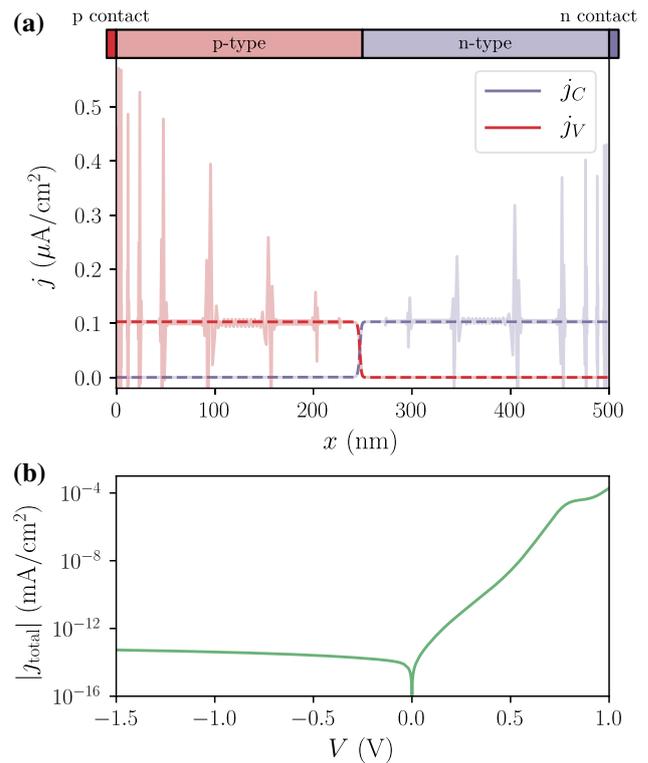


Fig. 3 **a** Current densities j_C and j_V for a silicon pn junction at a bias of 0.16 V. Simudo’s partitioned offset representation (dashed, dark lines) allows robust determination of internal current densities, while calculations without the offset representation (solid, dull lines) cannot determine the majority current densities accurately. **b** Total charge current density at the contact. Results with and without the offset representation agree to 5 digits of precision

which, plugged into Eq. 28, yields Eq. 27.

We perform a test of Simudo, which uses the partitioned offset representation, against the identical model without the offset representation. We consider a standard silicon pn junction diode with symmetric doping of 10^{18} cm^{-3} and SRH lifetimes of 1 ns and 1 μs in the p - and n -type regions, respectively. Each region has a length of 250 nm, for total device length of 500 nm. Although the problem is one-dimensional, we consider a 2D region with a height of 1 μm . At each contact, the majority carrier has an infinite surface recombination velocity, while the minority carrier has zero surface recombination. We use a mesh with 769 points in the x -direction, which is tightest near the contacts and junction and expands out geometrically toward the middle of the quasi-neutral regions. The mesh has 2 points in the y -direction, and further details of the mesh are given in Sect. 3.5.

The offset representation allows internal current densities to be resolved accurately throughout the device. Figure 3a shows the electron and hole currents under 0.16 V bias. Without the offset representation, the majority

currents are poorly resolved, due to the inability to resolve ∇w_k with double-precision arithmetic. The majority currents in the no-offset model become worse as the mesh density increases (not shown), as expected for approximations of ∇w_k . Figure 3b shows the total charge current density at the contact and the results with and without the offset representation agree to 5 digits. We conclude that the offset representation allows robust extraction of internal current densities but does not seem to be important for the overall current density of the test device. The offset representation imposes essentially no extra computational cost on Simudo while enabling robust determination of internal current densities.

3.4 Optics

The optical problem is solved by self-consistently iterating through the optical flux variables $\Phi_{[\lambda_{i,\min}, \lambda_{i,\max}], \hat{s}_i}$ and independently solving Eq. 19 for each one. For convenience, we write $\Phi = \Phi_{[\lambda_{i,\min}, \lambda_{i,\max}], \hat{s}_i}$, $\hat{s} = \hat{s}_i$, and $\alpha = \alpha_{[\lambda_{i,\min}, \lambda_{i,\max}]}$ for the remainder of this section. We represent Φ using CG basis functions of order $d_{\text{optical}} = 2$.

We now derive the weak form used in Simudo to solve each optical propagation problem. We follow closely the derivation in [49] of the modified second-order radiative transfer equation (MSORTE) method, without the scattering matrix. Integrating Eq. 19 with a test function $v \in \text{CG}(d_{\text{optical}})$ gives

$$\int_{\Omega} \mathbf{v} \cdot \nabla \zeta + \int_{\Omega} \mathbf{v} \cdot \nabla (\alpha \Phi) = 0, \tag{31}$$

where $\mathbf{v} = v \hat{s}$ and $\zeta = \hat{s} \cdot \nabla \Phi$. Using Eq. 29, we obtain

$$\oint_{\Gamma} (\mathbf{v} \cdot \hat{n}) \zeta_{\text{BC}} - \int_{\Omega} (\nabla \cdot \mathbf{v}) \zeta + \int_{\Omega} \mathbf{v} \cdot \nabla (\alpha \Phi) = 0. \tag{32}$$

Inserting the outlet boundary condition Eq. 20b into the first term, we obtain the final weak form

$$\oint_{\Gamma} (\mathbf{v} \cdot \hat{n}) (-\alpha \Phi) - \int_{\Omega} (\nabla \cdot \mathbf{v}) \zeta + \int_{\Omega} \mathbf{v} \cdot \nabla (\alpha \Phi) = 0 \tag{33}$$

The inlet boundary condition Eq. 20a is applied directly on Φ as an essential boundary condition.

3.5 Sentaurus benchmark comparison

To validate Simudo, we benchmark it against the industry standard Synopsys Sentaurus device simulator. Since Sentaurus does not support intermediate band materials, the benchmark is limited to standard semiconductors. Our test problem is the same silicon *pn* junction as considered in Sect. 3.3.2, with the overall $J(V)$ as shown in Fig. 3b. See

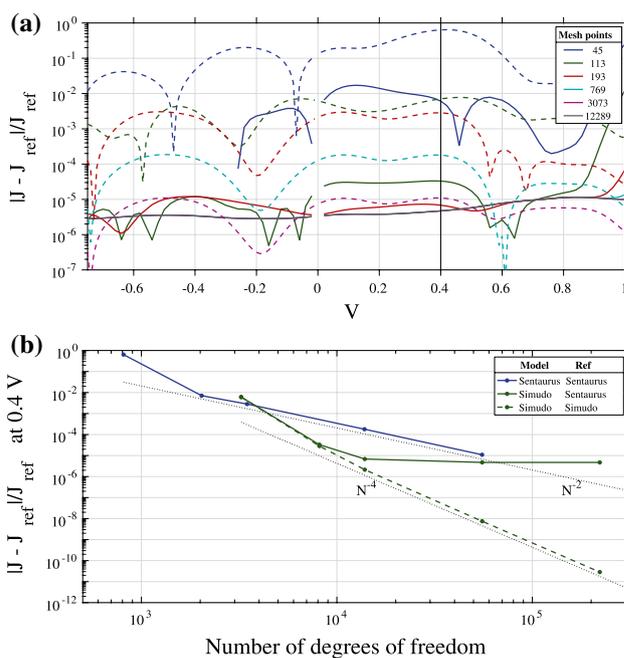


Fig. 4 **a** Relative error in total current J for a test *pn* junction in the dark, at several mesh densities. Dashed lines show results from Sentaurus, and solid lines show results from Simudo, with the same meshes. The Sentaurus result with the finest mesh is taken as J_{ref} for calculating relative error. Points at $V = 0$ are removed to avoid dividing by zero. **b** Relative errors at 0.4 V plotted against the number of degrees of freedom in each simulation. Solid lines use Sentaurus results for J_{ref} , and the dashed line shows Simudo self-convergence. Dotted lines show the scaling trends of the two methods

below for a discussion of the differences in implementation of the Ohmic condition between Simudo and Sentaurus.

We study the convergence of the results as the mesh is refined, using a number of mesh points in the *x*-direction ranging from 45 to 12,289, with the same meshes used for both Sentaurus and Simudo. The mesh spacing is nonuniform in the *x*-direction since the carrier and current densities vary most rapidly near the contacts and the junction. The meshes are generated by splitting the structure into two regions, one extending from the *p*-contact to the junction and one from the *n*-contact to the junction. In each region, a mesh spacing d_0 is applied to the cell adjacent to the contact and the cell adjacent to the junction. The mesh spacing increases geometrically toward the center of each region with a growth factor of 1.2. To generate the finest mesh sizes, these cells are further subdivided into 4, 16 or 64 equal parts. There are 2 points in the *y*-direction. The computational cost generally increases with the number of degrees of freedom rather than with the number of cells in the mesh. Simudo has more degrees of freedom associated with each cell than Sentaurus, due to its higher-order basis functions. For this mesh, Simudo has 36 degrees of freedom per triangle while Sentaurus has 9 per triangle, with 2 triangles

per mesh point. In this mesh, each triangle has at least one edge on the boundary of the device, which increases the number of degrees of freedom per triangle compared to a mesh where most triangles share sides; this effect is similar for both methods.

Figure 4a shows the results of the study, where we plot the relative error in $J(V)$ for each simulation, with the reference current J_{ref} taken from Sentaurus with the densest mesh. With 193 mesh points, Simudo converges approximately as well as Sentaurus with 3073 mesh points. Above 769 mesh points, the Simudo results show no further improvement in error, indicating either that Simudo and Sentaurus converge to results that differ at the 10^{-5} level or that Sentaurus with 12,289 mesh points is only converged to the 10^{-5} level, while Simudo may have converged more precisely. The simulations were performed on different machines, so we do not report timing data.

The figure makes clear that Simudo converges much more rapidly with mesh size than Sentaurus, which demonstrates the higher-order convergence that FEM is supposed to provide over FVM. Figure 4b shows the scaling of the errors with the number of degrees of freedom in each simulation, at 0.4 V. The solid blue line shows that Sentaurus' self-convergence scales like N^{-2} with the number of degrees of freedom. The dashed green line shows Simudo's self-convergence, with J_{ref} taken from the Simudo simulation with 12,289 mesh points. It shows that Simudo's self-convergence scales like N^{-4} with the number of degrees of freedom. For all but the smallest meshes, Simudo's convergence is superior to Sentaurus' at the same number of degrees of freedom. Taken together, these figures show that with 193 mesh points, Simudo's result is as good as Sentaurus' with 64 times as many mesh points, which is equivalent to 16 times as many degrees of freedom.

Note that the boundary conditions at the contacts are not precisely the same for the Sentaurus and Simudo simulations. Both are intended to simulate Ohmic contacts for the majority carrier and surface recombination velocities of 0 for the minority carrier. The Simudo simulations are performed with surface recombination velocity $S = \infty$ and 0 for the majority and minority carriers, respectively, imposing equilibrium carrier concentration at the boundary for majority carriers and setting $\mathbf{j}_k \cdot \hat{n}|_r = 0$ for minority carriers, as described in Sect. 2.2.2. The Sentaurus simulations are performed with $S = 0$ for the minority carriers, in agreement with Simudo, and the default “Ohmic contact” boundary condition for the majority carriers, which imposes charge neutrality and equilibrium carrier concentration at the contact. Under small and reverse bias, these two sets of boundary conditions should be equivalent, but under large forward bias, the default Sentaurus boundary condition is expected to give incorrect results due to its imposition of charge neutrality [57]. Sentaurus provides a “Modified Ohmic” boundary

condition, which should be closer to the Simudo boundary condition, but we were unable to attain convergence using it. As a result, at larger biases the Simudo and Sentaurus results diverge from each other, and we do not include them in Fig. 4. For biases larger than 1 V, the diode is in high injection, and the Boltzmann approximation used in this calculation is not accurate, regardless.

4 Examples and results

In this section, we give examples of using Simudo. Section 4.1 shows how to set up a simple 1-dimensional pn junction device and demonstrates the helpful tools that Simudo provides for defining regions and boundaries. Section 4.2 shows the extensibility of Simudo by illustrating the code required to add a new Auger recombination process. Section 4.3 illustrates the use of Simudo to study a system first considered in Ref. [24].

4.1 PN junction and topology definitions

We include in the supplementary material the code listing `equilibrium.py` describing a simple pn junction device in Simudo. This example constructs the device and implements steps 1–4 of the pre-solver shown in Fig. 2. Here, we discuss some of the pieces of that code and illustrate the useful topology construction operations built in to Simudo.

In the 1-dimensional pn junction example, the object `ls` contains information about the layers, including their sizes, positions, and mesh. The object `pdd` sets up the Poisson/drift-diffusion solver and has information about the bands in each material, including recombination processes and boundary conditions. In this example, there are only two bands (VB, CB); for a problem including an IB, `pdd` would have a third band, too.

Simudo is designed for 2-dimensional simulations, and it has sophisticated tools to define the arrangement of materials, dopings, contacts, meshing regions, or other user-defined spatial properties. In many solvers, interfaces must be tracked manually, including their orientation, to ensure that integrals over those interfaces are added together properly. Simudo introduces a set of topology tools that instead allow users to define the regions and interfaces in which they are interested, and Simudo takes care of all the book-keeping. The user defines regions as desired (e.g., emitter, base, defective-region), which can then be given properties, whether they be doping levels, recombination parameters, or other desired properties. These regions are initially defined abstractly, without having any coordinates in the device, using `CellRegions` and `FacetRegions` and are later connected to geometry and materials by the mesh generator.

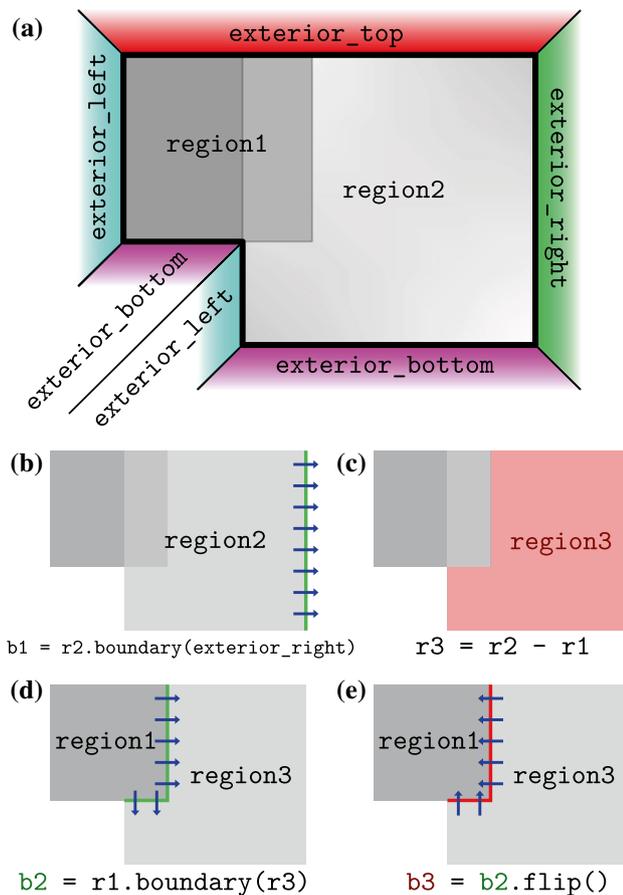


Fig. 5 Simudo allows users to define complicated domains and regions within them, with useful tools to automatically keep track of the required cells and facets in each region. **a** A simulation domain divided into overlapping region1 and region2. Simudo automatically defines the exterior outside the simulation domain and provides the helper regions exterior_left, exterior_right, exterior_top, and exterior_bottom, which may not be helpful, depending on the shape of the domain. Even before the regions are given specific geometries in the simulation domain (e.g., those pictured in **a**), Simudo's topology tools allow construction of further regions and facets. **b–e** Derived regions and facets from the domain shown in **a**, with the Simudo command to produce them shown underneath, where we shorten regionX to rX. Further example code is shown in Fig. 6. **b** The contact FacetRegion b1 is the signed boundary from region2 to exterior_right, with the sign indicated by the arrows. It can be used for determining the current flow out of the device. **c** region3 is region2 with region1 removed. **d** The FacetRegion b2 is the signed boundary from region1 to region3. **e** The flip operation reverses the sign of the boundary

Full details are given in the documentation accompanying Simudo, but we give a further illustration of these methods in Figs. 5 and 6. That example illustrates the creation of arbitrary CellRegion objects, including unions and intersections, and edges that connect them. When R is a CellRegions container, accessing a nonexistent attribute (such as R.domain) causes its creation. The user can define new CellRegion objects by applying Boolean operations

```

from simudo.mesh import CellRegions, FacetRegions
R = CellRegions() ; F = FacetRegions()

# R.region1 is created automatically when
# referenced
print(R)
print(R.region1, type(R.region1))
print(R)
## CellRegions({})
## CellRegionByName('region1') <class 'simudo.mesh.
# topology.CellRegionByName'>
## CellRegions({'region1': CellRegionByName('
# region1')})

# Define new region as set difference of region2
# and region1
R.region3 = R.region2 - R.region1
# The method (R.region1).boundary(R.region2)
# creates the signed boundary from region1 into
# region2, e.g., for calculating flux. R.region2
# is created when referenced.
F.b1 = (R.region2).boundary(R.exterior_right)
F.b2 = (R.region1).boundary(R.region3)
F.b3 = (F.b2).flip()

```

Fig. 6 Example of using CellRegions and FacetRegions to construct regions corresponding to those shown in Fig. 5. The R and F objects are containers for cell and facet regions, respectively. They are both initialized empty, and regions in each one are created when referenced. The boundary method in (R.region1).boundary(R.region3) creates the signed boundary from region1 to region3. The flip() method gives the boundary with the opposite sign. The actual mapping of these regions into the domain occurs in the mesh generation

on previous ones and new FacetRegion objects by using the boundary method. For example, consider the region R.region1. Then, R.region1.boundary(R.region2) creates a signed boundary from region1 to region2, as illustrated in Fig. 5. All of these custom regions are kept as symbolic expressions and evaluated by Simudo only when needed (e.g., when asked to apply a boundary condition or when asked to compute a volume or surface integral). This layer of abstraction allows the user not to worry about the details of mesh markers, entity indices, and facet orientations [58] and is described more fully in the documentation that accompanies Simudo.

The examples in Figs. 5 and 6 illustrate another useful concept. The mesh generation interprets the external region as being outside the simulation domain, allowing convenient definitions for boundary conditions and current flow. The FacetRegions are used in the *pn* junction example shown in the supplementary material to define the boundary conditions, which—in step 2—are conductive at the left and right contacts and nonconductive at the top and bottom surfaces. That example also shows how the mesh can be refined by adding extra mesh points near the contacts.

```

from simudo.physics import (DarkEOPMixer,
                             TwoBandEOPMixer, ElectroOpticalProcess)

class AugerRecombination(DarkEOPMixer,
                         TwoBandEOPMixer, ElectroOpticalProcess):
    """Auger recombination"""
    name = 'Auger'

    def get_Auger_C(self, band):
        return self.pdd.spatial.get('/','.join((
            self.name, band.name, 'C'))))

    def get_generation_user(self, band):
        # In this case, sign is +1 for CB, VB and None
        # for all other bands
        sign = self.get_band_generation_sign(band)
        if sign is None:
            return self._zero_generation

        CB = self.dst_band ; C_n = self.get_Auger_C(CB)
        VB = self.src_band ; C_p = self.get_Auger_C(VB)

        n = CB.u ; n0 = CB.thermal_equilibrium_u
        p = VB.u ; p0 = VB.thermal_equilibrium_u

        r = (C_n*(n**2 * p - n0**2 * p0) +
            C_p*(p**2 * n - p0**2 * n0))

        return (-r) * sign # Contribution to
                           generation rate
...

spatial.add_rule('Auger/CB/C', R.domain,
                U('1.1e-30 cm^6/s'))
spatial.add_rule('Auger/VB/C', R.domain,
                U('0.3e-30 cm^6/s'))

```

Fig. 7 Example Simudo code that implements Auger recombination (Eq. 34), including the AugerRecombination class. The last four lines show the Auger material parameters being set throughout the domain

4.2 Extensibility: adding Auger recombination

The initial release of Simudo contains radiative and Shockley–Read trapping and recombination processes in the non-degenerate limits for VB, CB. The user can easily add modified physics to their problems, which we demonstrate here with an example of adding an Auger recombination process to Simudo, with the form

$$U_A = C_n(n^2p - n_0^2p_0) + C_p(p^2n - p_0^2n_0), \quad (34)$$

where C_n, C_p are the Auger coefficients, and p_0 and n_0 are the hole and electron concentrations at thermal equilibrium, respectively [47]. The code is listed in Fig. 7. The function `get_generation_user(band)` adds a negative local generation rate in the CB and VB and returns 0 for all other bands. This recombination process moves particles between two bands, the `src_band` and the `dst_band`. In this case, where the electrons and holes have opposite charge, the Auger process destroys both particles simultaneously; if both carrier

Table 3 Parameters modeled on the device from [59], for Figs. 8–10

Value	Definition
$\mathcal{E}_C = 1.67 \text{ eV}$	Conduction band edge energy
$\mathcal{E}_I = 1.10 \text{ eV}$	Intermediate band energy
$\mathcal{E}_V = 0 \text{ eV}$	Valence band edge energy
$N_C = N_V = 5 \times 10^{18} \text{ cm}^{-3}$	CB and VB effective density of states
$N_I = 10^{17} \text{ cm}^{-3}$	IB density of states
$\mu_C = \mu_V = 2000 \text{ cm}^2/\text{V/s}$	CB and VB mobility
$\mu_I = 0.001 - 300 \text{ cm}^2/\text{V/s}$	IB mobility
$\alpha_{CV} = 10^4 \text{ cm}^{-1}$	Absorption coefficient for CV process
$\sigma_{CI}^{\text{opt}} = (2 - 10) \times 10^{-13} \text{ cm}^2$	Optical cross section for CI process
$\sigma_{IV}^{\text{opt}} = 2 \times 10^{-13} \text{ cm}^2$	Optical cross section for IV process
$\epsilon = 13\epsilon_0$	Dielectric constant
$T_s = 6000 \text{ K}$	Sun temperature
$T_c = 300 \text{ K}$	Cell temperature
$X = 1000$	Solar concentration factor
$w_{IB} = 1.3 \mu\text{m}$	IB region length
$f_{I,0} = 1/2$	Charge-neutral IB filling fraction

types involved in the process had the same charge (e.g., for a CB-to-IB trapping process), the process would represent a particle-number-conserving transfer (rather than a recombination) from the `src_band` to the `dst_band`, with the appropriate sign for the recombination process determined by the `get_band_generation_sign` method (inherited from `TwoBandEOPMixer`). This method's sign convention is that the `dst_band` always gains carriers through the generation process, while the `src_band` gains or loses as required by conservation of charge.

4.3 P[IB]N junction

In Ref. [59], the authors consider a quantum-dot-based IB solar cell with a p-n-IB-p-n structure. They present a drift-diffusion model for the IB region only, with the carrier density and current density boundary conditions obtained from a depletion approximation and law of the junction. This model assumes that transport is diffusion-dominated in the IB region, and drift can therefore be neglected. This early device model gave important insights into the behavior of IB devices.

In testing the self-consistency of the model, the authors estimate the IB mobility required to remain in the diffusion-dominated regime, finding that an IB mobility greater than $62 \text{ cm}^2/\text{V/s}$ is required to make their model consistent. This claim raises an immediate question: does something interesting happen when the IB mobility goes below that threshold? Since Simudo is a full drift-diffusion device model, we can directly answer that question.

We model a similar device with a simpler p-IB-n structure. This device has the same band and absorption

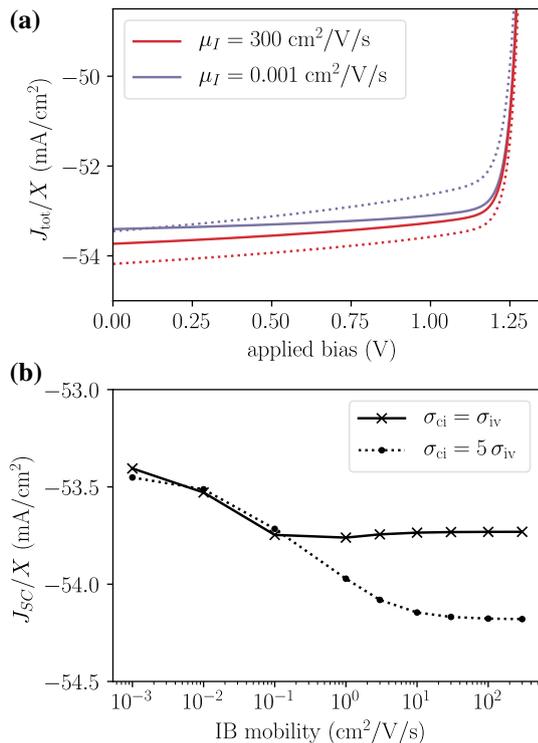


Fig. 8 **a** $J(V)$ curves for devices with parameters of Table 3, under $X = 1000$ suns illumination, modeled on Ref. [24]. Solid lines show $\sigma_{CI}^{opt} = \sigma_{IV}^{opt}$, as in [24], while dotted lines show $\sigma_{CI}^{opt} = 5\sigma_{IV}^{opt}$, which causes the CI absorption to be preferentially at the top of the device. Note the small vertical scale. The mismatched- σ^{opt} case is more strongly influenced by the IB mobility μ_I , but the effects are relatively small throughout. **b** The short-circuit current J_{sc} for devices with varying μ_I shows that the matched- σ^{opt} case is independent of μ_I when μ_I is sufficiently large (≥ 0.1 cm²/V/s), while the mismatched- σ^{opt} case again shows a stronger μ_I -dependence, but note the small vertical scale

parameters as the device described in [59], summarized in Table 3. The incident light is a blackbody spectrum at 6000 K with a solar concentration factor $X = 1000$. The device has equal subgap optical cross sections and nearly current matched (within 10%) incident photon fluxes for the two subgap transitions. In this case, the local IV and CI generation rates are nearly identical throughout the IB device. The code to set up this problem is included in the supplementary material, in `mart_i2002.py`.

We simulate this device with μ_I ranging from 0.001 to 300 cm²/V/s, with resulting J - V curves shown as solid lines in Fig. 8a, which is tightly zoomed and still shows only minor effects of this over-10⁵ change in μ_I . In fact, the IB and drifts currents contribute negligibly to the transport inside the IB region and the current remains diffusion-dominated throughout, as shown in Fig. 9a.

The device behavior is approximately independent of μ_I because the IV and CI generation rates are roughly equal at each point in the device, so IB transport is not

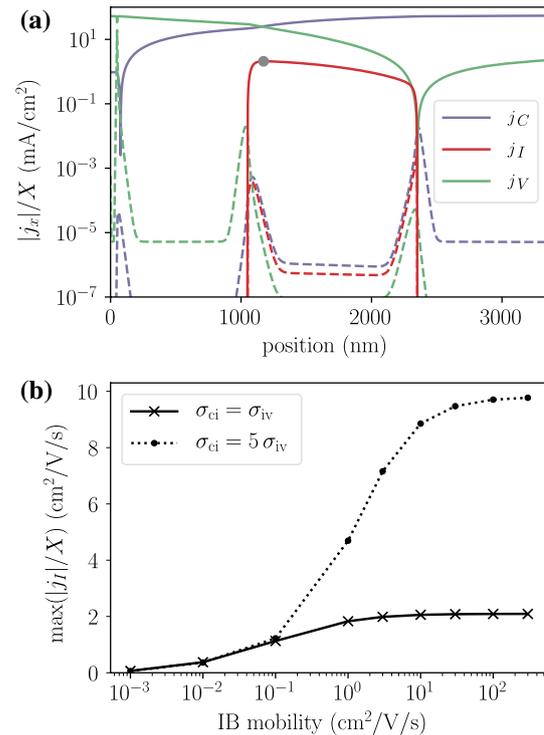


Fig. 9 **a** Drift (dashed) and diffusion (solid) currents for each band, at $\mu_I = 100$ cm²/V/s and $\sigma_{CI}^{opt} = \sigma_{IV}^{opt}$, with other parameters as in Table 3. The IB current density attains its maximum at the marked point. **b** Maximum IB current density as a function of μ_I for the matched and unmatched absorption cross sections, showing the increased importance of j_I and thus μ_I in the case where local absorptions are mismatched

required to enable subgap current matching, but this behavior is not generic for all IB devices. We illustrate this effect by increasing σ_{CI}^{opt} by a factor of five (while keeping σ_{IV}^{opt} unchanged). In this case, the device is still globally approximately current matched, but the CI absorption process occurs preferentially at the top of the device, while the IV generation occurs deeper in the IB region. The device thus relies on IB transport for the CI and IV generation rates to balance over the full device. These effects are shown in the dashed curves of Fig. 8, which show a stronger dependence on μ_I than in the matched case. When IB mobility is low, the excess CI generation in the front of the device instead causes local CI trapping, with equivalent local IV trapping toward the back of the device, reducing overall current. In the high-mobility case, the overall current is slightly larger with the mismatched absorptions, due to the increased optical depth. Figure 8b shows how J_{sc} varies with μ_I in both of these cases, where the greater dependence on μ_I in the mismatched case is apparent. Figure 9b shows that in the matched- σ^{opt} case, j_I is never particularly large, while it grows to be five times

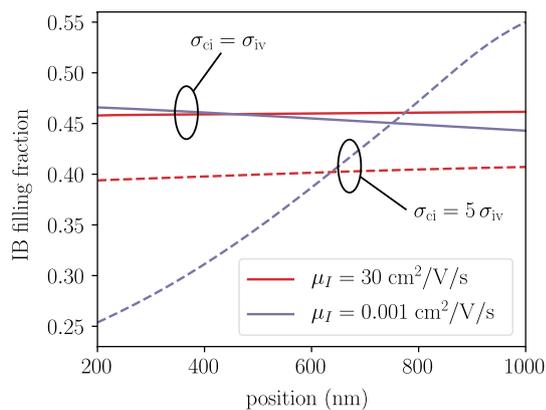


Fig. 10 IB filling fraction $f(x)$ at the maximum power point with the parameters as in Table 3. With mismatched σ^{opt} and low μ_I , the IB filling fraction changes drastically through the depth of the device, as the local generation and recombination rates must come into balance; this balancing increases the local recombination and decreases the total current generation, as seen in Fig. 8b. With matched σ^{opt} , internal IB currents balance the absorptions, and $f(x)$ remains nearly constant

larger in the mismatched case, showing the role of IB currents in internally balancing the optical absorptions.

In the low-mobility limit, where j_i is always small, when local CI and IV current generations are imbalanced, the filling fraction f of the IB must shift to equalize generation and recombination at each point [60]. This effect is visible in Fig. 10, where at low mobility, the mismatched- σ^{opt} case has photodepletion at the front side and photofilling at the back side, consistent with excess CI generation at the front and excess IV generation at the back. Both the matched- σ^{opt} and the high-mobility mismatched- σ^{opt} cases maintain an approximately uniform IB filling fraction.

These examples together show the utility of Simudo to explore the performance of IB devices and resolve an assertion made in earlier device models without the benefit of a coupled PDD/optics solver.

5 Conclusion

The availability of a device model for intermediate band materials should enable both understanding of this new class of materials and optimization of IB devices. Simudo's use of the FEM and its methods for overcoming catastrophic cancellation may also prove useful in standard semiconductor device simulation. Simudo has

been validated against Synopsys Sentaurus for standard semiconductor devices and shown to converge more rapidly with mesh size. This self-consistent solution of the Poisson/drift-diffusion and optical propagation equations provides a platform for studying a wide range of optoelectronic materials and devices, including solar cells and photodetectors, with tools to enable extensibility to arbitrary generation and recombination models, thermal effects, and more. The near-term roadmap for Simudo includes explicit heterojunction support and nonlocal tunneling, which will be available with future releases at github.com/simudo/simudo. We hope that the free and open source nature of this software will enable further development of IB materials and device simulation more broadly.

Acknowledgements We acknowledge funding from US Army Research Laboratory (W911NF-16-2-0167), the Natural Sciences and Engineering Research Council of Canada TOP-SET training program, and computing resources from Compute Canada. We thank Emily Zinnia Zhang for alpha testing Simudo, contributing the first code implementing trapping processes, and valuable conversations.

References

1. Bank, R.E., Rose, D.J., Fichtner, W.: Numerical methods for semiconductor device simulation. *IEEE Trans. Electron Devices* **30**(9), 1031 (1983)
2. Fichtner, W., Rose, D., Bank, R.: Semiconductor device simulation. *IEEE Trans. Electron Devices* **30**(9), 1018 (1983)
3. Markowich, P.A.: *The Stationary Semiconductor Device Equations*. Computational Microelectronics. Springer, Vienna (1986)
4. Piprek, J. (ed.): *Handbook of Optoelectronic Device Modeling & Simulation*. CRC Press, Boca Raton (2018)
5. Schenk, A.: *Advanced Physical Models for Silicon Device Simulation*. Springer, Wien (1998)
6. Luque, A., Martí, A.: Increasing the efficiency of ideal solar cells by photon induced transitions at intermediate levels. *Phys. Rev. Lett.* **78**(26), 5014 (1997)
7. Okada, Y., Ekins-Daukes, N.J., Kita, T., Tamaki, R., Yoshida, M., Pusch, A., Hess, O., Phillips, C.C., Farrell, D.J., Yoshida, K., Ahsan, N., Shoji, Y., Sogabe, T., Guillemoles, J.F.: Intermediate band solar cells: recent progress and future directions. *Appl. Phys. Rev.* **2**(2), 021302 (2015)
8. Mailoa, J.P., Akey, A.J., Simmons, C.B., Hutchinson, D., Mathews, J., Sullivan, J.T., Recht, D., Winkler, M.T., Williams, J.S., Warrender, J.M., Persans, P.D., Aziz, M.J., Buonassisi, T.: Room-temperature sub-band gap optoelectronic response of hyperdoped silicon. *Nat. Commun.* **5**, 3011 (2014)
9. Berencén, Y., Prucnal, S., Liu, F., Skorupa, I., Hübner, R., Rebohle, L., Zhou, S., Schneider, H., Helm, M., Skorupa, W.: Room-temperature short-wavelength infrared Si photodetector. *Sci. Rep.* **7**, 43688 (2017)
10. Wang, M., Berencén, Y., García-Hemme, E., Prucnal, S., Hübner, R., Yuan, Y., Xu, C., Rebohle, L., Böttger, R., Heller, R., Schneider, H., Skorupa, W., Helm, M., Zhou, S.: Extended infrared photoresponse in Te-hyperdoped Si at room temperature. *Phys. Rev. Appl.* **10**(2), 024054 (2018)
11. Brown, A.S., Green, M.A.: Impurity photovoltaic effect: fundamental energy conversion efficiency limits. *J. Appl. Phys.* **92**(3), 1329 (2002)

12. Martí, A., Antolín, E., Stanley, C.R., Farmer, C.D., López, N., Díaz, P., Cánovas, E., Linares, P.G., Luque, A.: Production of photocurrent due to intermediate-to-conduction-band transitions: a demonstration of a key operating principle of the intermediate-band solar cell. *Phys. Rev. Lett.* **97**(24), 247701 (2006)
13. Wang, W., Lin, A.S., Phillips, J.D.: Intermediate-band photovoltaic solar cell based on ZnTe:O. *Appl. Phys. Lett.* **95**(1), 011103 (2009)
14. López, N., Reichertz, L.A., Yu, K.M., Campman, K., Walukiewicz, W.: Engineering the electronic band structure for multi-band solar cells. *Phys. Rev. Lett.* **106**(2), 028701 (2011)
15. Sullivan, J.T., Simmons, C.B., Buonassisi, T., Krich, J.J.: Targeted search for effective intermediate band solar cell materials. *IEEE J. Photovolt.* **5**(1), 212 (2015)
16. der Maur, M.A.: A multiscale simulation environment for electronic and optoelectronic devices. Ph.D. thesis, Università degli Studi di Roma Tor Vergata (2008)
17. Birner, S., Zibold, T., Andlauer, T., Kubis, T., Sabathil, M., Trellakis, A., Vogl, P.: Nextnano: General purpose 3-D simulations. *IEEE Trans. Electron Devices* **54**(9), 2137 (2007)
18. Clugston, D.A., Basore, P.A.: PC1D version 5: 32-bit solar cell modeling on personal computers. In: *Twenty Sixth IEEE Photovoltaic Specialists Conference*, pp. 207–210 (1997)
19. Haug, H., Greulich, J.: PC1Dmod 6.2-improved simulation of c-Si devices with updates on device physics and user interface. *Energy Procedia* **92**(1876), 60 (2016)
20. Varache, R., Leendertz, C., Gueunier-Farret, M., Haschke, J., Muñoz, D., Korte, L.: Investigation of selective junctions using a newly developed tunnel current model for solar cell applications. *Sol. Energy Mater. Sol. Cells* **141**, 14 (2015)
21. Burgelman, M., Nollet, P., Degraeve, S.: Modelling polycrystalline semiconductor solar cells. *Thin Solid Films* **361–362**, 527 (2000)
22. Alonso-Álvarez, D., Wilson, T., Pearce, P., Führer, M., Farrell, D., Ekins-Daukes, N.: Solcore: a multi-scale, Python-based library for modelling solar cells and semiconductor materials. *J. Comput. Electron.* **17**(3), 1099 (2018)
23. Shockley, W., Read, W.T.: Statistics of the recombinations of holes and electrons. *Phys. Rev.* **87**(5), 835 (1952)
24. Martí, A., Cuadra, L., Luque, A.: Quasi-drift diffusion model for the quantum dot intermediate band solar cell. *IEEE Trans. Electron Devices* **49**(9), 1632 (2002)
25. Strandberg, R., Reenaas, T.W.: Drift-diffusion model for intermediate band solar cells including photofilling effects. *Prog. Photovolt.: Res. Appl.* **19**(1), 21 (2011)
26. Tobías, I., Luque, A., Martí, A.: Numerical modeling of intermediate band solar cells. *Semicond. Sci. Technol.* **26**(1), 014031 (2011)
27. Yoshida, K., Okada, Y., Sano, N.: Device simulation of intermediate band solar cells: effects of doping and concentration. *J. Appl. Phys.* **112**(8), 084510 (2012)
28. Cuadra, L., Martí, A., Luque, A.: Influence of the overlap between the absorption coefficients on the efficiency of the intermediate band solar cell. *IEEE Trans. Electron Devices* **51**(6), 1002 (2004)
29. Levy, M.Y., Honsberg, C.: Intraband absorption in solar cells with an intermediate band. *J. Appl. Phys.* **104**(11), 113103 (2008)
30. Hu, W.G., Inoue, T., Kojima, O., Kita, T.: Effects of absorption coefficients and intermediate-band filling in InAs/GaAs quantum dot solar cells. *Appl. Phys. Lett.* **97**(19), 193106 (2010)
31. Strandberg, R.: Analytic *JV*-characteristics of ideal intermediate band solar cells and solar cells with up and downconverters. *IEEE Trans. Electron Devices* **64**(5), 2275 (2017)
32. Strandberg, R.: The *JV*-characteristic of intermediate band solar cells with overlapping absorption coefficients. *IEEE Trans. Electron Devices* **64**(12), 5027 (2017)
33. Lin, A.S., Wang, W., Phillips, J.D.: Model for intermediate band solar cells incorporating carrier transport and recombination. *J. Appl. Phys.* **105**(6), 064512 (2009)
34. Navruz, T., Saritas, M.: Determination of the optimum material parameters for intermediate band solar cells using diffusion model. *Prog. Photovolt.: Res. Appl.* **22**(5), 593 (2014)
35. Krich, J.J., Trojnar, A.H., Feng, L., Hinzer, K., Walker, A.W.: Modeling intermediate band solar cells: a roadmap to high efficiency. In: *Proceedings SPIE 8981, Physics, Simulation, and Photonic Engineering of Photovoltaic Devices III*, p. 898100 (2014)
36. Yoshida, K., Okada, Y., Sano, N.: Self-consistent simulation of intermediate band solar cells: effect of occupation rates on device characteristics. *Appl. Phys. Lett.* **97**(13), 133503 (2010)
37. Alnæs, M.S., Blechta, J., Hake, J., Johansson, A., Kehlet, B., Logg, A., Richardson, C., Ring, J., Rognes, M.E., Wells, G.N.: The FEniCS project version 1.5. *Arch. Numer. Softw.* **3**(100), 9–23 (2015)
38. Eymard, R., Gallouët, T., Herbin, R.: *Handbook of Numerical Analysis*, vol. 7, pp. 713–1018. Elsevier, Amsterdam (2000)
39. He, Y., Cao, G.: A generalized Scharfetter–Gummel method to eliminate crosswind effects (semiconduction device modeling). *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **10**(12), 1579 (1991)
40. Nachaoui, A.: Iterative solution of the drift-diffusion equations. *Numer. Algorithms* **21**, 323 (1999)
41. Bochev, P., Peterson, K., Perego, M.: A multiscale control volume finite element method for advection-diffusion equations. *Int. J. Numer. Methods Fluids* **77**(11), 641 (2015)
42. Cockburn, B., Karniadakis, G.E., Shu, C.W. (eds.): *Discontinuous Galerkin Methods: Theory, Computation, and Applications*. Lecture Notes in Computational Science and Engineering, vol. 11. Springer, Berlin (2000)
43. Kumar, G., Singh, M., Bulusu, A., Trivedi, G.: A framework to simulate semiconductor devices using parallel computer architecture. In: *Journal of Physics: Conference Series*, vol. 759, p. 012098. IOP Publishing, Bristol (2016)
44. Kumar, G., Singh, M., Ray, A., Trivedi, G.: An FEM based framework to simulate semiconductor devices using streamline upwind Petrov–Galerkin stabilization technique. In: *2017 27th International Conference Radioelektronika*, pp. 1–5 (2017)
45. Poupaud, F., Schmeiser, C.: Charge transport in semiconductors with degeneracy effects. *Math. Methods Appl. Sci.* **14**(5), 301 (1991)
46. Marshak, A.H., Van Vliet, C.: Electrical current and carrier density in degenerate materials with nonuniform band structure. *Proc. IEEE* **72**(2), 148 (1984)
47. Nelson, J.: *The Physics of Solar Cells*. Imperial College Press, London (2003)
48. McIntosh, K.R., Black, L.E.: On effective surface recombination parameters. *J. Appl. Phys.* **116**(1), (2014)
49. Zhao, J., Tan, J., Liu, L.: A second order radiative transfer equation and its solution by meshless method with application to strongly inhomogeneous media. *J. Comput. Phys.* **232**(1), 431 (2013)
50. Johnson, C.: *Numerical Solution of Partial Differential Equations by the Finite Element Method*. Cambridge University Press, Cambridge (1987)
51. Gockenbach, M.S.: *Understanding and Implementing the Finite Element Method*. SIAM, New Delhi (2006)
52. Jean Donea, A.H.: *Finite Element Methods for Flow Problems*. Wiley, Hoboken (2003)
53. Gaury, B., Sun, Y., Bermel, P., Haney, P.M.: Sesame: a 2-dimensional solar cell modeling tool. *Sol. Energy Mater. Sol. Cells* **198**, 53 (2019)

54. Brezzi, F., Douglas, J., Marini, L.D.: Two families of mixed finite elements for second order elliptic problems. *Numer. Math.* **47**(2), 217 (1985)
55. Roberts, J., Thomas, J.M.: In *Finite Element Methods (Part 1)*, Handbook of Numerical Analysis, vol. 2, pp. 523–639. Elsevier, Hoboken (1991)
56. Cummings, D.J., Law, M.E., Cea, S., Linton, T.: Comparison of discretization methods for device simulation. *Int. Conf. Simul. Semicond. Process. Devices* **2009**, 1–4 (2009)
57. Synopsys Inc.: *Sentaurus Device User Guide, vK-2015*. Synopsys Inc., Mountain View (2015)
58. Logg, A., Mardal, K.A., Wells, G.N., et al.: *Automated Solution of Differential Equations by the Finite Element Method*. Springer, Berlin (2012)
59. Marti, A., Cuadra, L., Luque, A.: Quasi-drift diffusion model for the quantum dot intermediate band solar cell. *IEEE Trans. Electron Devices* **49**, 1632 (2002)
60. Strandberg, R., Reenaas, T.W.: Photofilling of intermediate bands. *J. Appl. Phys.* **105**(12), 124512 (2009)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.