



HAL
open science

Structure-preserving schemes for drift-diffusion systems on general meshes: DDFV vs HFV

Stella Krell, Julien Moatti

► **To cite this version:**

Stella Krell, Julien Moatti. Structure-preserving schemes for drift-diffusion systems on general meshes: DDFV vs HFV. 2023. hal-04037264v1

HAL Id: hal-04037264

<https://hal.science/hal-04037264v1>

Preprint submitted on 20 Mar 2023 (v1), last revised 20 Jul 2023 (v2)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Structure-preserving schemes for drift-diffusion systems on general meshes: DDFV vs HFV

Stella Krell¹ and Julien Moatti²

¹ Université Côte d'Azur, CNRS, Inria, LJAD, France,
stella.krell@univ-cotedazur.fr

² Inria, Univ. Lille, CNRS, UMR 8524 - Laboratoire Paul Painlevé, F-59000 Lille, France
julien.moatti@inria.fr

Abstract. We made a comparison between a Discrete Duality Finite Volume (DDFV) scheme and a Hybrid Finite Volume (HFV) scheme for a drift-diffusion model with mixed boundary conditions on general meshes. Both schemes are based on a nonlinear discretisation of the convection-diffusion fluxes, which ensures the positivity of the discrete densities. We investigate the behaviours of the schemes on various numerical test cases.

Keywords: Discrete Duality Finite Volume, Hybrid Finite Volume, positivity preserving methods, discrete entropy/dissipation relation, long-time behaviour.

1 Motivation

We are interested in the numerical discretization of drift-diffusion model. Let Ω be a polygonal connected open bounded subset of \mathbb{R}^2 , whose boundary $\Gamma = \partial\Omega$ is divided into two parts $\Gamma = \Gamma^D \cup \Gamma^N$ with $m(\Gamma^D) > 0$. The problem writes:

$$\left\{ \begin{array}{ll} \partial_t N - \operatorname{div}(\nabla N - N\nabla\phi) = 0 & \text{in } \mathbb{R}_+ \times \Omega, \\ \partial_t P - \operatorname{div}(\nabla P + P\nabla\phi) = 0 & \text{in } \mathbb{R}_+ \times \Omega, \\ -\lambda^2 \operatorname{div}(\nabla\phi) = C + P - N & \text{in } \mathbb{R}_+ \times \Omega, \\ N = N^D, P = P^D \text{ and } \phi = \phi^D & \text{on } \mathbb{R}_+ \times \Gamma^D, \\ (\nabla N - N\nabla\phi) \cdot n = (\nabla P + P\nabla\phi) \cdot n = \nabla\phi \cdot n = 0 & \text{on } \mathbb{R}_+ \times \Gamma^N, \\ N(0, \cdot) = N^{in} \text{ and } P(0, \cdot) = P^{in} & \text{in } \Omega, \end{array} \right. \quad (1)$$

where n denotes the unit normal vector to $\partial\Omega$ pointing outward Ω . Regarding the data, (i) the parameter $\lambda > 0$ is the rescaled Debye length of the system, which accounts for the nondimensionalisation (relevant values of this parameter can be very small, inducing some stiff behaviours), (ii) the initial conditions N^{in} and P^{in} belong to $L^\infty(\Omega)$ and are positive, (iii) the doping profile C is in $L^\infty(\Omega)$, and characterises the semiconductor device used. In the following, we also assume that the boundary conditions are the trace of some H^1 function on Ω , such that the following relation holds:

$$\log(N^D) - \phi^D = \alpha_N \text{ and } \log(P^D) + \phi^D = \alpha_P \text{ on } \Gamma^D, \quad (2)$$

where α_N and α_P are two real constants. It follows that N^D and P^D are positive.

The solution to (1) enjoys some natural physical properties: the densities N and P are positive for all time, and the solution converges exponentially fast towards some thermal equilibrium (N^e, P^e, ϕ^e) -which is a stationary solution to (1)- where $N^e = e^{\alpha_N + \phi^e}$, $P^e = e^{\alpha_P - \phi^e}$ and ϕ^e is the solution to the Poisson-Boltzmann equation

$$\begin{cases} -\lambda^2 \operatorname{div}(\nabla \phi^e) = C + \exp(\alpha_P - \phi^e) - \exp(\alpha_N + \phi^e) & \text{in } \Omega, \\ \phi^e = \phi^D \text{ on } \Gamma^D \quad \text{and} \quad \nabla \phi^e \cdot n = 0 & \text{on } \Gamma^N. \end{cases} \quad (3)$$

Relation (2) is a compatibility condition in order to ensure the existence of the thermal equilibrium (3). When designing numerical schemes for (1), it is crucial to ensure that the scheme preserves these properties at the discrete level. This structure preserving feature is ensured by classical TPFA schemes on admissible orthogonal meshes (see [1]). Unfortunately, these schemes cannot be used on general meshes. Following the ideas introduced in [3], a nonlinear positivity preserving DDFV scheme for Fokker-Planck equations has been introduced in [2]. In the spirit of these works, a nonlinear structure preserving HFV scheme was introduced and partially analysed in [5]. The aim of this paper is to introduce a nonlinear structure preserving DDFV scheme for (1) based on the scheme of [2] and to compare it numerically with the HFV scheme of [5].

2 Descriptions of the schemes

The schemes used here are based on the same nonlinear strategy, introduced in [3], consisting in the reformulation of the convection-diffusion fluxes:

$$\nabla N - N \nabla \phi = N \nabla (\log(N) - \phi) \quad \text{and} \quad \nabla P + P \nabla \phi = P \nabla (\log(P) + \phi).$$

At the discrete level, both schemes rely on discrete gradients operators to approximate the continuous gradients. The major issue lies in the discretisation of the prefactors P and N , which will be handled by local reconstruction operators. Both schemes are based on a backward Euler discretisation in time. To fix ideas, we will use a constant time step $\Delta t > 0$. For more precise descriptions and statements about the schemes and the meshes, we refer to [2] (DDFV) and [5] (HFV).

Remark 1 (Generalisation to anisotropic models). In this paper, we consider isotropic convection-diffusion equations for the charges carriers for the sake of brevity. One could add anisotropic diffusion tensors and consider the framework described in [5].

Both schemes rely on a spatial discretisation (or mesh) of the domain Ω . The (primal interior) mesh \mathfrak{M} is a partition of Ω in polygonal control volumes (or cells). We let $\partial \mathfrak{M}$ be the set of boundary edges, seen either as degenerate control volumes (DDFV framework) or as edges (HFV framework). The primal mesh $\overline{\mathfrak{M}}$ is defined as the union of \mathfrak{M} and $\partial \mathfrak{M}$. Given a cell $K \in \overline{\mathfrak{M}}$, we fix a point $x_K \in K$, called the center of K . For all neighboring primal cells K and L , we assume that $\partial K \cap \partial L$ is a segment, corresponding to an internal edge of the mesh \mathfrak{M} , denoted by $\sigma = K|L$ and we let \mathcal{E}_{int} be the set of such edges. We denote by $\mathcal{E} = \mathcal{E}_{int} \cup \partial \mathfrak{M}$ the set of all (internal and exterior) edges of the mesh, and define \mathcal{E}_K the set of edges of the cell $K \in \mathfrak{M}$. For any $K \in \mathfrak{M}$ and $\sigma \in \mathcal{E}_K$, we define $\mathbf{n}_{\sigma K}$ as the unit normal to σ outward K .

Given any measurable $X \subset \mathbb{R}^2$, we denote by m_X the measure of the object X .

2.1 The DDFV scheme

In order to define the DDFV scheme, we need to introduce two other meshes: the dual mesh denoted $\overline{\mathcal{M}}^*$ and the diamond mesh denoted \mathcal{D} (see [2] for more details). The dual mesh $\overline{\mathcal{M}}^*$ is also composed of interior dual mesh \mathcal{M}^* (corresponding of cells around vertex in Ω) and of boundary dual mesh $\partial\mathcal{M}^*$ (corresponding of cells around vertex on $\partial\Omega$). For any vertex x_{K^*} of the primal mesh satisfying $x_{K^*} \in \Omega$, we define a polygonal control volume K^* by connecting all the centers of the primal cells sharing x_{K^*} as vertex. For any vertex $x_{K^*} \in \partial\Omega$, we define a polygonal control volume K^* by connecting the centers x_K of the interior primal cells and the midpoints of the boundary edges sharing x_{K^*} as vertex and x_{K^*} . We define the set \mathcal{E}_{int}^* of internal edges of the dual mesh similarly as \mathcal{E}_{int} . We denote by $\mathbf{n}_{\sigma^*K^*}$ the unit normal to σ^* outward K^* . For each couple $(\sigma, \sigma^*) \in \mathcal{E} \times \mathcal{E}_{int}^*$ such that $\sigma = [x_{K^*}, x_{L^*}]$ and $\sigma^* = K^*|L^*$, we define the quadrilateral diamond $\mathcal{D}_{\sigma, \sigma^*}$ whose diagonals are σ and σ^* (if $\sigma \subset \partial\Omega$, it degenerates into a triangle). The set of the diamonds defines the diamond mesh \mathcal{D} , which is a partition of Ω .

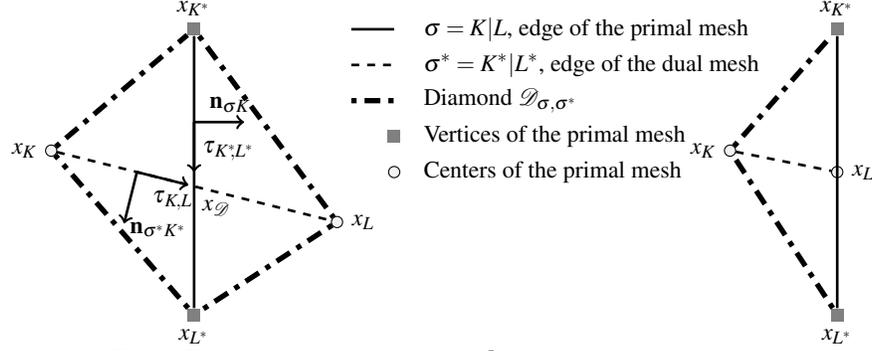


Fig. 1: Definition of the diamonds $\mathcal{D}_{\sigma, \sigma^*}$ and related notations.

Finally, the DDFV mesh is made of $\mathcal{T} = (\overline{\mathcal{M}}, \overline{\mathcal{M}}^*)$ and \mathcal{D} .

We now introduce the space of scalar fields which are associated to each primal and dual cell $\mathbb{R}^{\mathcal{T}}$, and space of vector fields constant on the diamonds $(\mathbb{R}^2)^{\mathcal{D}}$:

$$u_{\mathcal{T}} \in \mathbb{R}^{\mathcal{T}} \iff u_{\mathcal{T}} = ((u_K)_{K \in \overline{\mathcal{M}}}, (u_{K^*})_{K^* \in \overline{\mathcal{M}}^*}) \text{ and } \xi_{\mathcal{D}} \in (\mathbb{R}^2)^{\mathcal{D}} \iff \xi_{\mathcal{D}} = (\xi_{\mathcal{D}})_{\mathcal{D} \in \mathcal{D}}.$$

To enforce Dirichlet boundary conditions, we introduce the set of Dirichlet boundary primal and dual cells: $\partial\mathcal{M}_D = \{K \in \partial\mathcal{M} : K \subset \Gamma_D\}$ and $\partial\mathcal{M}_D^* = \{K^* \in \partial\mathcal{M}^* : x_{K^*} \in \overline{\Gamma}_D\}$, and, for a given $v \in C(\Gamma^D)$, we define

$$E_v^{\Gamma_D} = \{u_{\mathcal{T}} \in \mathbb{R}^{\mathcal{T}} \mid \forall K \in \partial\mathcal{M}_D, u_K = v(x_K) \text{ and } \forall K^* \in \partial\mathcal{M}_D^*, u_{K^*} = v(x_{K^*})\}.$$

We also define discrete bilinear forms on $\mathbb{R}^{\mathcal{T}}$ and $(\mathbb{R}^2)^{\mathcal{D}}$ by

$$\begin{aligned} \llbracket v_{\mathcal{T}}, u_{\mathcal{T}} \rrbracket_{\mathcal{T}} &= \frac{1}{2} \left(\sum_{K \in \overline{\mathcal{M}}} m_K u_K v_K + \sum_{K^* \in \overline{\mathcal{M}}^*} m_{K^*} u_{K^*} v_{K^*} \right), \quad \forall (u_{\mathcal{T}}, v_{\mathcal{T}}) \in (\mathbb{R}^{\mathcal{T}})^2, \\ (\xi_{\mathcal{D}}, \varphi_{\mathcal{D}})_{\mathcal{D}} &= \sum_{\mathcal{D} \in \mathcal{D}} m_{\mathcal{D}} \xi_{\mathcal{D}} \cdot \varphi_{\mathcal{D}}, \quad \forall (\xi_{\mathcal{D}}, \varphi_{\mathcal{D}}) \in ((\mathbb{R}^2)^{\mathcal{D}})^2. \end{aligned}$$

The DDFV method is based on the definition of a discrete gradient operator $\nabla^{\mathcal{D}} : \mathbb{R}^{\mathcal{T}} \rightarrow (\mathbb{R}^2)^{\mathcal{D}}$, defined by $\nabla^{\mathcal{D}} u_{\mathcal{T}} = \left(\nabla^{\mathcal{D}} u_{\mathcal{T}} \right)_{\mathcal{D} \in \mathcal{D}}$, where

$$\nabla^{\mathcal{D}} u_{\mathcal{T}} = \frac{1}{2m_{\mathcal{D}}} (m_{\sigma}(u_L - u_K) \mathbf{n}_{\sigma K} + m_{\sigma^*}(u_{L^*} - u_{K^*}) \mathbf{n}_{\sigma^* K^*}) \quad \forall \mathcal{D} \in \mathcal{D}. \quad (4)$$

Finally, we introduce a reconstruction operator on diamonds $r^{\mathcal{D}}$. It is a mapping from $\mathbb{R}^{\mathcal{T}}$ to $\mathbb{R}^{\mathcal{D}}$ defined for all $u_{\mathcal{T}} \in \mathbb{R}^{\mathcal{T}}$ by $r^{\mathcal{D}} u_{\mathcal{T}} = \left(r^{\mathcal{D}} u_{\mathcal{T}} \right)_{\mathcal{D} \in \mathcal{D}}$, where for $\mathcal{D} \in \mathcal{D}$, whose vertices are $x_K, x_L, x_{K^*}, x_{L^*}$, $r^{\mathcal{D}} u_{\mathcal{T}} = \frac{1}{4}(u_K + u_L + u_{K^*} + u_{L^*})$. One can now introduce a DDFV discretisation of $(u, w, v) \mapsto \int_{\Omega} u \nabla w \cdot \nabla v$, defined by

$$T_{\mathcal{D}} : (u_{\mathcal{T}}, w_{\mathcal{T}}, v_{\mathcal{T}}) \mapsto \sum_{\mathcal{D} \in \mathcal{D}} m_{\mathcal{D}} r^{\mathcal{D}} u_{\mathcal{T}} \nabla^{\mathcal{D}} w_{\mathcal{T}} \cdot \nabla^{\mathcal{D}} v_{\mathcal{T}}.$$

Now, we first discretise the data by taking the mean values of N^{in} , P^{in} and C on the primal and dual cells, which define $N_{\mathcal{T}}^0$, $P_{\mathcal{T}}^0$ and $C_{\mathcal{T}}$. Then, for all $n \geq 0$, we look for $(N_{\mathcal{T}}^{n+1}, P_{\mathcal{T}}^{n+1}, \phi_{\mathcal{T}}^{n+1}) \in \mathbf{E}_{ND}^{\Gamma_D} \times \mathbf{E}_{pD}^{\Gamma_D} \times \mathbf{E}_{\phi D}^{\Gamma_D}$ solution to:

$$\left[\left[\frac{N_{\mathcal{T}}^{n+1} - N_{\mathcal{T}}^n}{\Delta t}, v_{\mathcal{T}} \right]_{\mathcal{T}} + T_{\mathcal{D}}(N_{\mathcal{T}}^{n+1}, \log(N_{\mathcal{T}}^{n+1}) - \phi_{\mathcal{T}}^{n+1}, v_{\mathcal{T}}) = 0 \quad \forall v_{\mathcal{T}} \in \mathbf{E}_0^{\Gamma_D}, \quad (5a)$$

$$\left[\left[\frac{P_{\mathcal{T}}^{n+1} - P_{\mathcal{T}}^n}{\Delta t}, v_{\mathcal{T}} \right]_{\mathcal{T}} + T_{\mathcal{D}}(P_{\mathcal{T}}^{n+1}, \log(P_{\mathcal{T}}^{n+1}) + \phi_{\mathcal{T}}^{n+1}, v_{\mathcal{T}}) = 0 \quad \forall v_{\mathcal{T}} \in \mathbf{E}_0^{\Gamma_D}, \quad (5b)$$

$$\lambda^2 \left(\nabla^{\mathcal{D}} \phi_{\mathcal{T}}^{n+1}, \nabla^{\mathcal{D}} v_{\mathcal{T}} \right)_{\mathcal{D}} = \left[\left[C_{\mathcal{T}} + P_{\mathcal{T}}^{n+1} - N_{\mathcal{T}}^{n+1}, v_{\mathcal{T}} \right]_{\mathcal{T}} \quad \forall v_{\mathcal{T}} \in \mathbf{E}_0^{\Gamma_D}. \quad (5c)$$

In (5a) and (5b), we use the notation $\log(u_{\mathcal{T}}) = ((\log(u_K))_{K \in \mathfrak{M}}, (\log(u_{K^*}))_{K^* \in \overline{\mathfrak{M}}})$.

2.2 The HFV scheme

In order to define the HFV scheme, we need to introduce a pyramidal submesh. To do so, one has to assume that each cell $K \in \mathfrak{M}$ is star-shaped with respect to its center x_K (we recall that x_K is not necessarily the barycentre of K). We then define $P_{K,\sigma}$ as the pyramid (triangle) of base σ and apex x_K . Given any $\sigma \in \mathcal{E}$, we denote by \bar{x}_{σ} the barycentre of σ , and by $d_{K,\sigma}$ the euclidean distance between σ and x_K . Finally, we define the hybrid discretisation (or mesh) as $\mathcal{D} = (\mathfrak{M}, \mathcal{E})$.

We now introduce the space of discrete (scalar) hybrid unknowns $\underline{V}_{\mathcal{D}}$:

$$\underline{u}_{\mathcal{D}} \in \underline{V}_{\mathcal{D}} \iff \underline{u}_{\mathcal{D}} = ((u_K)_{K \in \mathfrak{M}}, (u_{\sigma})_{\sigma \in \mathcal{E}}),$$

where the $u_K \in \mathbb{R}$ are the cell unknowns and the $u_{\sigma} \in \mathbb{R}$ are the edges unknowns (approximation of the trace of the solutions on the edges). To enforce Dirichlet boundary conditions, for a given $v \in C(\Gamma^D)$, we define

$$\underline{V}_{\mathcal{D},v}^{\Gamma_D} = \{ \underline{u}_{\mathcal{D}} \in \underline{V}_{\mathcal{D}} \mid \forall \sigma \in \partial \mathfrak{M}_D, u_{\sigma} = v(\bar{x}_{\sigma}) \}.$$

As for the DDFV framework, we define a bilinear form on $\underline{V}_{\mathcal{D}}$, discrete counterpart of the inner product on $L^2(\Omega)$ as

$$\llbracket \underline{u}_{\mathcal{D}}, \underline{v}_{\mathcal{D}} \rrbracket_{\mathfrak{M}} = \sum_{K \in \mathfrak{M}} m_K u_K v_K, \quad \forall (\underline{u}_{\mathcal{D}}, \underline{v}_{\mathcal{D}}) \in \underline{V}_{\mathcal{D}}^2.$$

The HFV method is based on the definition of a discrete gradient operator $\nabla_{\mathcal{D}} : \underline{V}_{\mathcal{D}} \rightarrow (\mathbb{R}^2)^{\Omega}$ which maps discrete hybrid unknowns onto piecewise constant functions on the pyramidal submesh. More precisely, given $\underline{v}_{\mathcal{D}} \in \underline{V}_{\mathcal{D}}$, $K \in \mathfrak{M}$ and $\sigma \in \mathcal{E}_K$,

$$\nabla_{\mathcal{D}} \underline{v}_{\mathcal{D}}|_{P_{K,\sigma}} = G_K \underline{v}_{\mathcal{D}} + S_{K,\sigma} \underline{v}_{\mathcal{D}},$$

where, for some $\eta > 0$, the consistent and stabilisation parts of the gradient are given by

$$G_K \underline{v}_{\mathcal{D}} = \frac{1}{m_K} \sum_{\sigma' \in \mathcal{E}_K} m_{\sigma'} \nu_{\sigma'} n_{K,\sigma'} \text{ and } S_{K,\sigma} \underline{v}_{\mathcal{D}} = \frac{\eta}{d_{K,\sigma}} (\nu_{\sigma} - \nu_K - G_K \underline{v}_{\mathcal{D}} \cdot (\bar{x}_{\sigma} - x_K)) n_{K,\sigma}.$$

One can now define the discrete counterpart of $(u, v) \mapsto \int_{\Omega} \nabla u \cdot \nabla v$ as

$$a_{\mathcal{D}} : (\underline{u}_{\mathcal{D}}, \underline{v}_{\mathcal{D}}) \mapsto \int_{\Omega} \nabla_{\mathcal{D}} \underline{u}_{\mathcal{D}} \cdot \nabla_{\mathcal{D}} \underline{v}_{\mathcal{D}}.$$

Finally, we introduce as previously local reconstruction operators on cells $r^K : \underline{V}_{\mathcal{D}} \rightarrow \mathbb{R}$, such that for any $\underline{u}_{\mathcal{D}} \in \underline{V}_{\mathcal{D}}$, $r^K(\underline{u}_{\mathcal{D}}) = \frac{1}{|\mathcal{E}_K|} \sum_{\sigma \in \mathcal{E}_K} \frac{u_K + u_{\sigma}}{2}$, where $|\mathcal{E}_K|$ is the cardinal of the finite set \mathcal{E}_K . One can now introduce a HFV discretisation of $(u, w, v) \mapsto \int_{\Omega} u \nabla w \cdot \nabla v$, defined by

$$T_{\mathcal{D}} : (\underline{u}_{\mathcal{D}}, \underline{w}_{\mathcal{D}}, \underline{v}_{\mathcal{D}}) \mapsto \sum_{K \in \mathfrak{M}} r^K(\underline{u}_{\mathcal{D}}) \int_K \nabla_{\mathcal{D}} \underline{w}_{\mathcal{D}} \cdot \nabla_{\mathcal{D}} \underline{v}_{\mathcal{D}}.$$

We now discretise the data by taking the mean values of N^{in} , P^{in} and C on the cells and edges, which define $\underline{P}_{\mathcal{D}}^0$, $\underline{N}_{\mathcal{D}}^0$ and $\underline{C}_{\mathcal{D}}$. Then, for all $n \geq 0$, we look for $(\underline{N}_{\mathcal{D}}^{n+1}, \underline{P}_{\mathcal{D}}^{n+1}, \underline{\phi}_{\mathcal{D}}^{n+1}) \in \underline{V}_{\mathcal{D},ND}^{I_D} \times \underline{V}_{\mathcal{D},PD}^{I_D} \times \underline{V}_{\mathcal{D},\phi^D}^{I_D}$ solution to:

$$\left[\left[\frac{\underline{N}_{\mathcal{D}}^{n+1} - \underline{N}_{\mathcal{D}}^n}{\Delta t}, \underline{v}_{\mathcal{D}} \right]_{\mathfrak{M}} + T_{\mathcal{D}}(\underline{N}_{\mathcal{D}}^{n+1}, \log(\underline{N}_{\mathcal{D}}^{n+1}) - \underline{\phi}_{\mathcal{D}}^{n+1}, \underline{v}_{\mathcal{D}}) = 0 \quad \forall \underline{v}_{\mathcal{D}} \in \underline{V}_{\mathcal{D},0}^{I_D}, \quad (6a)$$

$$\left[\left[\frac{\underline{P}_{\mathcal{D}}^{n+1} - \underline{P}_{\mathcal{D}}^n}{\Delta t}, \underline{v}_{\mathcal{D}} \right]_{\mathfrak{M}} + T_{\mathcal{D}}(\underline{P}_{\mathcal{D}}^{n+1}, \log(\underline{P}_{\mathcal{D}}^{n+1}) + \underline{\phi}_{\mathcal{D}}^{n+1}, \underline{v}_{\mathcal{D}}) = 0 \quad \forall \underline{v}_{\mathcal{D}} \in \underline{V}_{\mathcal{D},0}^{I_D}, \quad (6b)$$

$$\lambda^2 a_{\mathcal{D}}(\underline{\phi}_{\mathcal{D}}^{n+1}, \underline{v}_{\mathcal{D}}) = \left[\left[\underline{C}_{\mathcal{D}} + \underline{P}_{\mathcal{D}}^{n+1} - \underline{N}_{\mathcal{D}}^{n+1}, \underline{v}_{\mathcal{D}} \right]_{\mathfrak{M}} \quad \forall \underline{v}_{\mathcal{D}} \in \underline{V}_{\mathcal{D},0}^{I_D}. \quad (6c)$$

As previously, we use the notation $\log(\underline{u}_{\mathcal{D}}) = ((\log(u_K))_{K \in \mathfrak{M}}, (\log(u_{\sigma}))_{\sigma \in \mathcal{E}})$.

2.3 Some structural differences between schemes

As highlighted by the unified presentation above, both schemes are very similar and rely on the same features. Note that both local reconstruction operators $r^{\mathcal{D}}$ and r^K take into account all the local unknowns of the geometric entity considered (diamond or cells), this property is the key point of the analysis of this kind of schemes, see [2,5].

However, the schemes exhibit differences, some of which are listed below:

- the discrete HFV gradient $\nabla_{\mathcal{D}}$ includes a stabilisation term for the sake of coercivity and the stabilisation parameter η has to be chosen a priori, whereas the DDFV one is simpler and do not need any choice of parameter;

- the DDFV unknowns are all "volumic", in the sense that there are associated to geometric entities with non-zero two-dimensional measures, whereas the faces unknowns of the HFV method have no mass and have no influence on the discrete time derivative terms $\llbracket \underline{N}_{\mathcal{D}}^{n+1} - \underline{N}_{\mathcal{D}}^n, \underline{\nu}_{\mathcal{D}} \rrbracket_{\mathfrak{M}}$ and $\llbracket \underline{P}_{\mathcal{D}}^{n+1} - \underline{P}_{\mathcal{D}}^n, \underline{\nu}_{\mathcal{D}} \rrbracket_{\mathfrak{M}}$;
- the cells unknowns of the HFV scheme can be eliminated before solving linear systems, using a static condensation procedure (see [5, Section 5.1.2.]), whereas one has to solve a system including all primal and dual unknowns for DDFV;
- the HFV scheme can be used in 3D without any modification (the edges become faces), whereas using a DDFV method in 3D requires more sophisticated changes (see [4]).

3 Numerical experiments

The two numerical schemes described here are nonlinear, hence their algebraic realisations boil down to the resolution of nonlinear systems of equations. To do so, we use Newton method, with an adaptative time stepping strategies: if the Newton method does not converge, we try to compute the solution for a smaller time step $0.5 \times \Delta t$. If the method converges, we use a bigger time step $1.4 \times \Delta t$. The initial time step is the maximal time step allowed. For the HFV scheme, at each system resolution, a static condensation is used to eliminate the cell unknowns (see [5, Section 5.1.2.]), and we use $\eta = 1.5$.

The test case used below follow the framework used in [5] to describe a 2D PN-junction, whose geometry is described in Figure 2.

The domain Ω is the unit square $]0, 1[{}^2$. For the boundary conditions, we split $\Gamma^D = \Gamma_0^D \cup \Gamma_1^D$ with $\Gamma_0^D = [0, 1] \times \{0\}$ and $\Gamma_1^D = [0, 0.25] \times \{1\}$. For $i \in \{0, 1\}$, we let $N^D = N_i^D$, $P^D = P_i^D$ and $\phi^D = \frac{\log(N_i^D) - \log(P_i^D)}{2}$ on Γ_i^D . To be consistent with the compatibility condition (2) we assume that there exists a constant α_0 such that $\log(N^D \times P^D) = \alpha_0$. Therefore for given N^D and α_0 we set $P^D = \frac{e^{\alpha_0}}{N^D}$ on Γ^D . Thus, one has $\alpha_N = \alpha_P = \frac{\alpha_0}{2}$. The doping profile C is piecewise constant, equal to -1 in the P-region and 1 in the N-region (see Figure 2). Last, we use the following smooth initial conditions: $N_0(x, y) = N_1^D + (N_0^D - N_1^D)(1 - \sqrt{y})$ and $P_0(x, y) = P_1^D + (P_0^D - P_1^D)(1 - \sqrt{y})$.

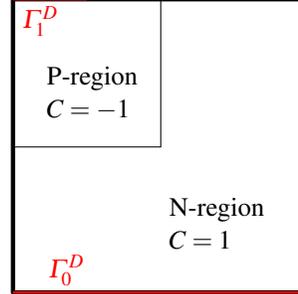


Fig. 2: PN diode geometry.

3.1 Positivity

In this section, we want to assert and compare the positivity preserving features of the schemes. The test case used here corresponds to the following values: $\lambda = 0.05$, $N_0^D = 0.1$, $N_1^D = 1$ and $\alpha_0 = -7.5$. To give quantitative informations, we show in Figure 3 the evolution of the minimal values of P and N , along with the time step and the number of Newton's iterations needed to compute the solutions at a given time for each time step.

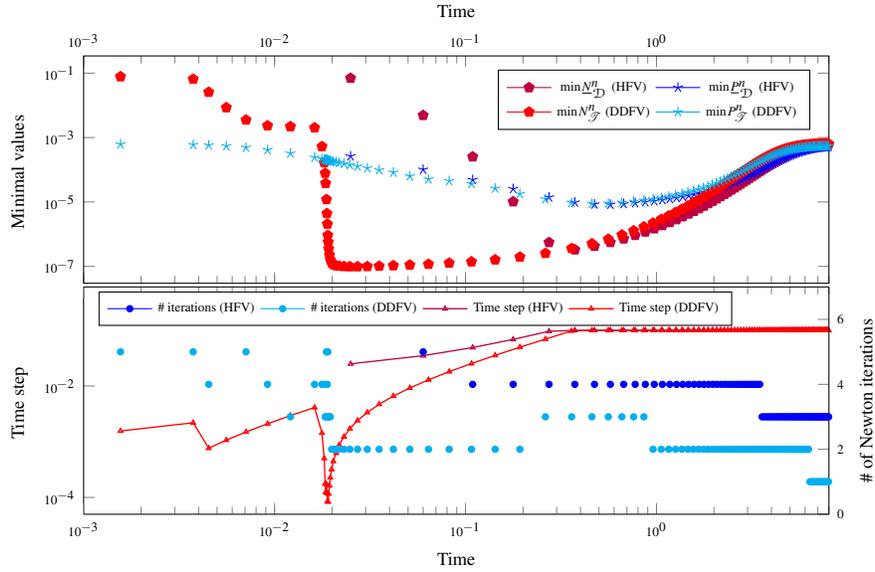


Fig. 3: **Test-case.** Evolution of the discrete minimal values, time step and cost

The minimal values are taken on every unknowns (primal and dual cells for the DDFV scheme, cells and faces for the HFV one).

For this simulation, both schemes need to perform time step reductions in order to compute solutions. One can see that the DDFV scheme needs a smaller initial time step than the HFV one to compute the first iteration in time. Moreover, the values of the minimal density N are different for small times. This difference between the two schemes vanishes for $t > 3 \cdot 10^{-1}$. However, if one looks at the minimal values for P , both schemes give similar results. In fact, the difference for small time is not so significant. Indeed the profiles of the solutions -not shown here- are very similar for both schemes, and it appears that the smallest values are reached only on few unknowns. We suspect that this difference is due to the different spatial positions of some unknowns for the two schemes, and that the DDFV “sees” some piece of the solution with very small values. One can also notice that the number of Newton iterations needed to compute one time step stays reasonable (5 iteration at most). Moreover, this number decreases along time since the solution converges exponentially fast towards the equilibrium: once the equilibrium is (almost) reached, a time step corresponds to a very small evolution of the solution. Overall, these results show that the schemes are rather robust, since they are able to compute solutions whose densities reach 10^{-7} .

3.2 Long-time behaviour

Here, we investigate the long-time behaviour of the schemes. At the continuous level, one usually quantify the distance between the solution (N, P, ϕ) and the equilibrium (N^e, P^e, ϕ^e) by looking at the relative entropy, defined as

$$\mathbb{E}(t) = \int_{\Omega} N^e H\left(\frac{N}{N^e}\right) + \int_{\Omega} P^e H\left(\frac{P}{P^e}\right) + \frac{\lambda^2}{2} \|\nabla(\phi - \phi^e)\|_{L^2(\Omega)}^2,$$

with $H : s \mapsto s \log(s) - s + 1$. One can check that (N, P, ϕ) coincides with the equilibrium if and only if the relative entropy cancels. In the following, we are interested in the evolution of the discrete counterparts of this quantities.

We consider a test case with physical data $N_0^D = e$, $N_1^D = 1$ and $\alpha_0 = 0$. We also use two different values of the Debye length λ , respectively 1 and 0.01. We perform simulations on a triangular mesh, with a initial time step $\Delta t = 0.1$. On Figure 4, we show

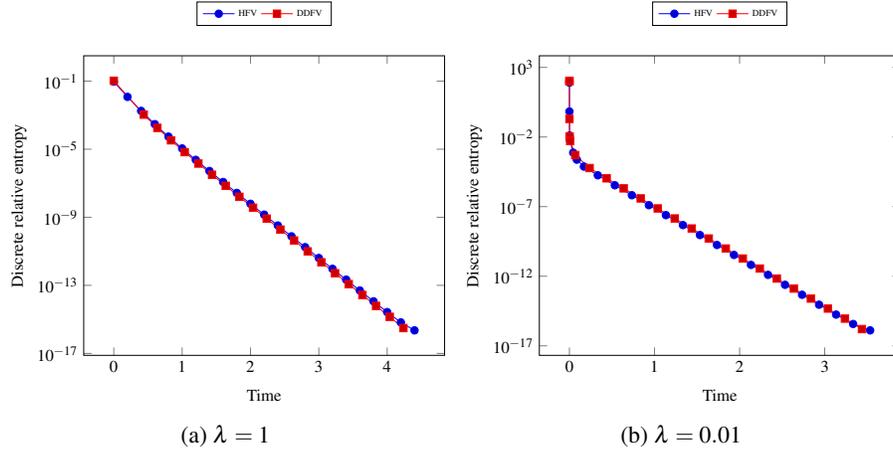


Fig. 4: **Long-time behaviour.** Evolution of the discrete relative entropies.

the evolutions of the discrete relative entropies along time, for the two values of λ and both schemes. As expected, the convergence towards the equilibrium is exponentially fast, as in the continuous framework. Moreover, it is remarkable to notice that the decay rates are almost the same for both schemes. Moreover, with the small Debye length (Figure 4b), both schemes are able to capture the behaviour with a very fast evolution far from the equilibrium, then slower once close to it.

References

1. Bessemoulin-Chatard M., Chainais-Hillairet C.: Exponential decay of a finite volume scheme to the thermal equilibrium for drift-diffusion systems. *J. Numer. Math.* 25, 3, 147–168 (2017).
2. Cancès, C., Chainais-Hillairet, C., Krell, S.: Numerical analysis of a nonlinear free-energy diminishing discrete duality finite volume scheme for convection diffusion equations. *Comput. Methods Appl. Math.*, **18**, 407–432 (2018).
3. Cancès C., Guichard C.: Numerical analysis of a robust free energy diminishing finite volume scheme for parabolic equations with gradient structure. *Found. Comput. Math.* 17, 6, 1525–1584 (2017).
4. Coudière Y., Hubert F.: A 3D Discrete Duality Finite Volume method for non-linear elliptic equations. *SIAM Journal on Scientific Computing* 33, no. 4, 1739–1764 (2011).
5. Moatti, J.: A structure preserving hybrid finite volume scheme for semiconductor models with magnetic field on general meshes. (2022). <https://hal.science/hal-03715313v1>