DISCRETIZATION METHODS FOR EXTREMELY ANISOTROPIC DIFFUSION

Bram van Es[a,b,1], **Barry Koren**[b,c] **and Hugo de Blank**[a]

[a]FOM Institute DIFFER - Dutch Institute for Fundamental Energy Research, Edisonbaan 14, Nieuwegein, The Netherlands
[b] Centrum Wiskunde & Informatica, Science Park 123, Amsterdam, The Netherlands
[c] Eindhoven University of Technology, Den Dolech 2, Eindhoven, The Netherlands

Abstract: Anisotropic diffusion is a common physical phenomenon and describes processes where the diffusion of some scalar quantity is directionally dependent. Anisotropic diffusive processes are for instance Darcy's flow for porous media, large scale turbulence where turbulence scales are anisotropic in size, and heat conduction and momentum dissipation in fusion plasmas. In fusion plasmas there is extreme anisotropy due to the high temperature and large magnetic field strength. This causes diffusive processes, heat diffusion and energy/momentum loss due to viscous friction, to effectively be aligned with the magnetic field lines. This alignment leads to different values for the respective diffusive coefficients in the magnetic field direction and in the perpendicular direction, to the extent that heat diffusion coefficients can be up to 10^{12} times larger in the parallel direction than in the perpendicular direction. This anisotropy puts stringent requirements on the numerical methods used to approximate the MHD-equations since any misalignment of the grid may cause the perpendicular diffusion to be polluted by the numerical error in approximating the parallel diffusion. Currently the common approach is to apply magnetic field-aligned coordinates, an approach that automatically takes care of the directionality of the diffusive coefficients. This approach runs into problems in the case of crossing field lines, e.g., x-points and points where there is magnetic reconnection. It is therefore useful to consider numerical schemes that are more tolerant to the misalignment of the grid with the magnetic field lines, both to improve existing methods and to help open the possibility of applying regular non-aligned grids. To investigate this, several discretization schemes are applied to the unsteady anisotropic heat diffusion equation on a cartesian grid. All methods presented are generic and carry over to any other anisotropic diffusion problem.

[1]Correspondence author; Bram van Es, es@cwi.nl

INTRODUCTION

Numerically, high anisotropy in diffusion may lead to the situation where errors in the direction in which the diffusion coefficient is largest, may significantly influence the diffusion in the perpendicular direction. This may necessitate either a high-order approximation in the direction of the largest coefficient value and/or a limitation on the degree of anisotropy (see e.g. Sovinec et al [1], Meier et al [2]).
Given the high level of anisotropy in tokamak plasmas, a numerical approximation may introduce large perpendicular errors if the magnetic field direction is strongly misaligned with the grid. Here, misaligned means that the directions of diffusion are not aligned with the grid lines. Difficulties that may arise with highly anisotropic diffusion problems on non-aligned meshes are:

- significant numerical diffusion perpendicular to the magnetic field lines due to grid misalignment (see e.g. Umansky et al [3]),
- non-positivity near high gradients (see e.g. Sharma et al [4]),
- mesh locking, stagnation of convergence dependent on anisotropy (see e.g. Babuška and Suri [5]) and
- convergence loss in case of variable diffusion tensor (see e.g. Günter et al [6]).

It is possible to use a field-aligned coordinate system. However, this cannot be maintained throughout the plasma; problems arise at x-points and in regions of highly fluctuating magnetic field directions (for instance in case of edge turbulence). To confidently perform simulations of phenomena which rely heavily on the resolution of the perpendicular temperature gradient we must apply a scheme that is robust in terms of accuracy in case of varying anisotropy and misalignment.
Although anisotropic diffusion is a well studied problem, the numerical methods to approximate it are not fully equipped to handle extremely anisotropic diffusion problems in case of a (sharply) varying diffusion tensor and/or high gradients of the diffusion quantity.
The focus of this paper is on the order of convergence and the perpendicular numerical diffusion for problems with extremely high levels of anisotropy. We present novel interpolation-based schemes and compare these with existing schemes.

PROBLEM DESCRIPTION

Anisotropic thermal diffusion is described by the following model

$$\mathbf{q} = -\mathbf{D} \cdot \nabla T, \quad \frac{\partial T}{\partial t} = -\nabla \cdot \mathbf{q} + f, \tag{1}$$

where T represents the temperature, $\mathbf{b}$ the unit direction vector of the field line, f some source term and $\mathbf{D}$ the diffusion tensor. For a two-dimensional problem the diffusion tensor is given by

$$\begin{aligned} \text{unit direction vector:} \quad & \mathbf{b} = [\cos\alpha,\, \sin\alpha]^T, \\ & \mathbf{D} = D_{\parallel}\mathbf{b}\mathbf{b} + D_{\perp}(\mathcal{I} - \mathbf{b}\mathbf{b}), \\ & \mathbf{D} = \begin{pmatrix} D_{\parallel}b_1^2 + D_{\perp}b_2^2 & (D_{\parallel} - D_{\perp})b_1 b_2 \\ (D_{\parallel} - D_{\perp})b_1 b_2 & D_{\perp}b_1^2 + D_{\parallel}b_2^2 \end{pmatrix}, \end{aligned}$$

where $D_{\parallel}$ and $D_{\perp}$ represent the parallel and the perpendicular diffusion coefficient respectively. We define x, y as the non-aligned coordinate system and s, n as the aligned coordinate system, see figure 1. The boundary conditions are discussed per test case. The diffusion equation is approximated on a uniform cartesian grid, with $\Delta x = \Delta y = h$.
In tokamak fusion plasma simulations the diffusion coefficients are often taken as temperature-dependent. In general the parallel and perpendicular diffusion coefficients are assumed to be proportional to $T^{5/2}$ and $T^{-1/2}$ respectively, i.e., the anisotropy varies strongly with temperature.

FINITE DIFFERENCE SCHEMES

We limit the discussion to finite difference schemes. Given a uniform grid this can be directly translated to a finite volume approach. We consider several second-order accurate finite difference schemes for the approximation of model equation (1). The first two schemes are described in Günter et al [6]. The difference between these

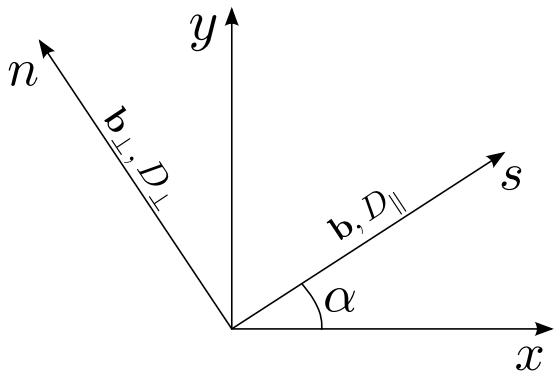


Figure 1: Explanation of symbols

schemes lies in the treatment of the flux, particularly the location of the flux. The new schemes, to be presented here, aim to improve the accuracy of co-located schemes by applying a stencil that lies on an approximation of the field line. We use sub-indices x, y, s, n to denote the respective derivatives.

Asymmetric Finite Differences

The first finite difference scheme for heat diffusion we discuss is depicted in figure 2. For a spatially constant diffusion tensor this scheme reduces to the standard second-order scheme for diffusion. The label asymmetry is

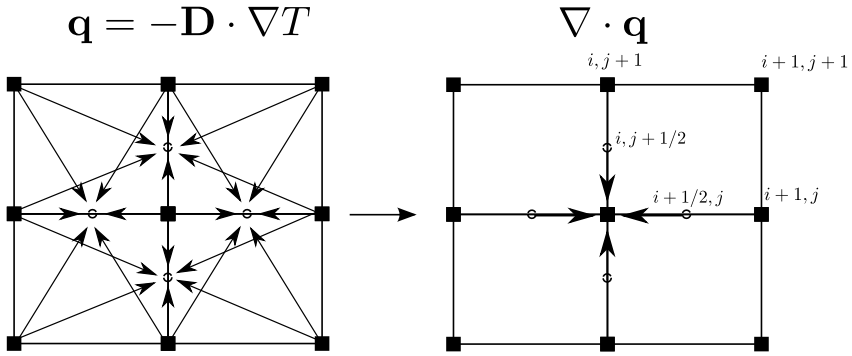


Figure 2: Semi-staggered grid, asymmetric scheme, temperature T is defined on the full indices and the diffusion tensor $\mathbf{D}$ on the half-indices

coined because of the different treatment of the x- versus y-differential in each point. The different treatment is a direct result of taking the flux values in $i \pm \frac{1}{2}, j$ and $i, j \pm \frac{1}{2}$,

$$\left.\frac{\partial T}{\partial x}\right|_{i+\frac{1}{2},j} = \frac{T_{i+1,j} - T_{i,j}}{\Delta x},$$
$$\left.\frac{\partial T}{\partial y}\right|_{i+\frac{1}{2},j} = \frac{T_{i+1,j+1} + T_{i,j+1} - T_{i,j-1} - T_{i+1,j-1}}{4\Delta y},$$
$$\left.\frac{\partial T}{\partial x}\right|_{i,j+\frac{1}{2}} = \frac{T_{i+1,j+1} + T_{i+1,j} - T_{i-1,j+1} - T_{i-1,j}}{4\Delta x},$$
$$\left.\frac{\partial T}{\partial y}\right|_{i,j+\frac{1}{2}} = \frac{T_{i,j+1} - T_{i,j}}{\Delta y},$$

and similar formulas for $\frac{\partial T}{\partial x}\big|_{i-\frac{1}{2},j}$, $\frac{\partial T}{\partial y}\Big|_{i-\frac{1}{2},j}$, $\frac{\partial T}{\partial x}\big|_{i,j-\frac{1}{2}}$, $\frac{\partial T}{\partial y}\Big|_{i,j-\frac{1}{2}}$. For the heat conduction term we have

$$\mathbf{q}_{i+\frac{1}{2},j} = -\mathbf{D}_{i+\frac{1}{2},j} \cdot \left(\frac{\partial T}{\partial x}\Big|_{i+\frac{1}{2},j}, \frac{\partial T}{\partial y}\Big|_{i+\frac{1}{2},j} \right)^T.$$

In case of a co-located grid we use arithmetic averaging for the diffusion tensor, so:

$$\mathbf{D}_{i+\frac{1}{2},j} = \frac{\mathbf{D}_{i+1,j} + \mathbf{D}_{i,j}}{2}.$$

Finally, the diffusion follows from

$$\nabla \cdot \mathbf{q} = \frac{(q_1)_{i+\frac{1}{2},j} - (q_1)_{i-\frac{1}{2},j}}{\Delta x} + \frac{(q_2)_{i,j+\frac{1}{2}} - (q_2)_{i,j-\frac{1}{2}}}{\Delta y}.$$

Besides this semi-staggered grid approach where $\mathbf{q}$ and $\mathbf{D}$ are defined on the half-indices $i \pm \frac{1}{2}, j$ and $i, j \pm \frac{1}{2}$, we also implement the scheme on a co-located grid where $\mathbf{D}$ is defined at the same points as the temperature.

Symmetric Finite Differences

Another approach is taken by Günter et al [6], they use a symmetric scheme (with a symmetric linear operator) that is mimetic by maintaining the self-adjointness of the differential operator. By maintaining the self-adjointness numerically the following integral identity still holds at the discrete level:

$$\int_V \phi \nabla \cdot \mathbf{q} dV + \int_V \mathbf{q} \cdot \nabla \phi dV = \oint_{\partial S} \phi (\mathbf{q} \cdot \mathbf{n}) dS,$$

where ϕ is an arbitrary real-valued function in x, y. The total energy of a system described by the diffusion equation is given by $E = \int_V T dV$. In absence of any surface and source terms this should be constant. This means that $\frac{\partial E}{\partial t} = 0$ or $\int_V \nabla \cdot (\mathbf{D} \cdot \nabla T) dV = 0$. If we take a constant value for ϕ we find that

$$\phi \int_V \nabla \cdot \mathbf{q} dV = \frac{\partial E}{\partial t} = 0,$$

and so energy is preserved exactly.
The approach goes as follows. First, the divergence terms are determined at the center points (see figure 3):

$$\frac{\partial T}{\partial x}\Big|_{i+\frac{1}{2},j+\frac{1}{2}} = \frac{T_{i+1,j+1} + T_{i+1,j} - T_{i,j+1} - T_{i,j}}{2\Delta x},$$
$$\frac{\partial T}{\partial y}\Big|_{i+\frac{1}{2},j+\frac{1}{2}} = \frac{T_{i,j+1} + T_{i+1,j+1} - T_{i+1,j} - T_{i,j}}{2\Delta y}.$$

Next, the diffusion tensor is applied to obtain the heat flux

$$\mathbf{q} = -\mathbf{D} \cdot \nabla T, \quad \mathbf{q}_{i+\frac{1}{2},j+\frac{1}{2}} = -\mathbf{D}_{i+\frac{1}{2},j+\frac{1}{2}} \cdot \left(\frac{\partial T}{\partial x}\Big|_{i+\frac{1}{2},j+\frac{1}{2}}, \frac{\partial T}{\partial y}\Big|_{i+\frac{1}{2},j+\frac{1}{2}} \right)^T,$$

where the diffusion tensor is taken as the arithmetic mean of the four surrounding points, so

$$\mathbf{D}_{i+\frac{1}{2},j+\frac{1}{2}} = \frac{\mathbf{D}_{i+1,j+1} + \mathbf{D}_{i+1,j} + \mathbf{D}_{i,j+1} + \mathbf{D}_{i,j}}{4}.$$

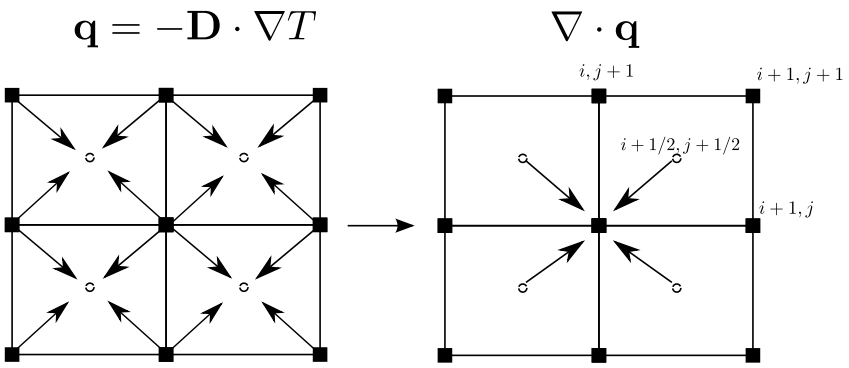


Figure 3: Staggered grid, symmetric scheme, temperature T is defined on the full indices and the diffusion tensor $\mathbf{D}$ on the half-indices

Finally, the divergence is taken over the heat flux

$$\begin{aligned}\nabla \cdot \mathbf{q} = &\frac{(q_1)_{i+\frac{1}{2},j+\frac{1}{2}} + (q_1)_{i+\frac{1}{2},j-\frac{1}{2}} - (q_1)_{i-\frac{1}{2},j+\frac{1}{2}} - (q_1)_{i-\frac{1}{2},j-\frac{1}{2}}}{2\Delta x} \\ &+ \frac{(q_2)_{i+\frac{1}{2},j+\frac{1}{2}} + (q_2)_{i-\frac{1}{2},j+\frac{1}{2}} - (q_2)_{i-\frac{1}{2},j-\frac{1}{2}} - (q_2)_{i+\frac{1}{2},j-\frac{1}{2}}}{2\Delta y}.\end{aligned}$$

Two cases are considered, a fully staggered grid where $\mathbf{q}$ and $\mathbf{D}$ are defined on the half-indices $i \pm \frac{1}{2}, j \pm \frac{1}{2}$ and a co-located grid where $\mathbf{D}$ is defined at the same points as the temperature.

Aligned Finite Differences

The idea is that differencing along the field line yields an approximation less prone to large false perpendicular diffusion. To do this we have to use interpolation to find the values of T and $\mathbf{D}$ on the field line. The field line itself is approximated, by tracing. In the current implementation, the interpolation of $T, \mathbf{b}$ and $\mathbf{D}$ is done on a co-located grid. In the following section we will consider x, y as local coordinates where the origin is located in the stencil point i, j. By applying the product rule and some vector identities we can write the diffusion equation in parts:

$$\nabla \cdot (\mathbf{D} \cdot \nabla T) = \mathcal{A}_1 + \mathcal{A}_2 + \mathcal{A}_3 + \mathcal{A}_4, \tag{2}$$

where the parts are given by

$$\begin{aligned}\text{field line curvature:} \quad & \mathcal{A}_1 = -\left(D_\parallel - D_\perp\right) \nabla \cdot \mathbf{b}_\perp \left(\mathbf{b}_\perp \cdot \nabla T\right), \\ \text{field strength variation:} \quad & \mathcal{A}_2 = \left(D_\parallel - D_\perp\right) \nabla \cdot \mathbf{b} \left(\mathbf{b} \cdot \nabla T\right), \\ \text{temperature diffusion:} \quad & \mathcal{A}_3 = D_\parallel \mathbf{b}\mathbf{b} : \nabla\nabla T + D_\perp \mathbf{b}_\perp \mathbf{b}_\perp : \nabla\nabla T, \\ \text{diffusion variation:} \quad & \mathcal{A}_4 = (\mathbf{b} \cdot \nabla T)(\mathbf{b} \cdot \nabla D_\parallel) + (\mathbf{b}_\perp \cdot \nabla T)(\mathbf{b}_\perp \cdot \nabla D_\perp).\end{aligned}$$

Rewriting this in s, n-coordinates yields

$$\begin{aligned}\mathcal{A}_1 =& -\left(D_\parallel - D_\perp\right) N T_n, \\ \mathcal{A}_2 =& \left(D_\parallel - D_\perp\right) S T_s, \\ \mathcal{A}_3 =& D_\parallel T_{ss} + D_\perp T_{nn}, \\ \mathcal{A}_4 =& D_{\parallel_s} T_s + D_{\perp_n} T_n,\end{aligned} \tag{3}$$

where

$$S = -b_2 b_{1_n} + b_1 b_{2_n}, \quad N = -b_1 b_{2_s} + b_2 b_{1_s}.$$

So we can write

$$\begin{aligned}\nabla \cdot (\mathbf{D} \cdot \nabla T) &= \nabla \cdot \left(D_\parallel (\mathbf{b} \cdot \nabla T)\mathbf{b}\right) + \nabla \cdot \left(D_\perp (\mathbf{b}_\perp \cdot \nabla T)\mathbf{b}_\perp\right), \\ \nabla \cdot \left(D_\parallel (\mathbf{b} \cdot \nabla T)\mathbf{b}\right) &= D_\parallel \left(-N T_n + S T_s + T_{ss}\right) + D_{\parallel_s} T_s, \\ \nabla \cdot \left(D_\perp (\mathbf{b}_\perp \cdot \nabla T)\mathbf{b}_\perp\right) &= D_\perp \left(N T_n - S T_s + T_{nn}\right) + D_{\perp_n} T_n.\end{aligned}$$

Note that $S = \alpha_n$ and $N = -\alpha_s$.
When applying the equations of magnetohydrodynamics to nuclear fusion plasmas, an assumption often made is that the temperature is diffused instantaneously along the field line. This means that the variation of the temperature in the direction of the field line is zero, i.e., $\mathbf{b} \cdot \nabla T = 0$, $T_s = 0$. So in that case our set of equations can be reduced to

$$\begin{aligned}\mathcal{A}_1 =& D_\perp N T_n,\\ \mathcal{A}_2 =& 0,\\ \mathcal{A}_3 =& D_\perp T_{nn},\\ \mathcal{A}_4 =& D_{\perp_n} T_n.\end{aligned}$$

Here, we stick to the more general form with the parts given by (3). We continue by applying an aligned stencil to approximate equation (2) in s, n-coordinates. The stencil points r, l, u, d, c are given in figure 4.

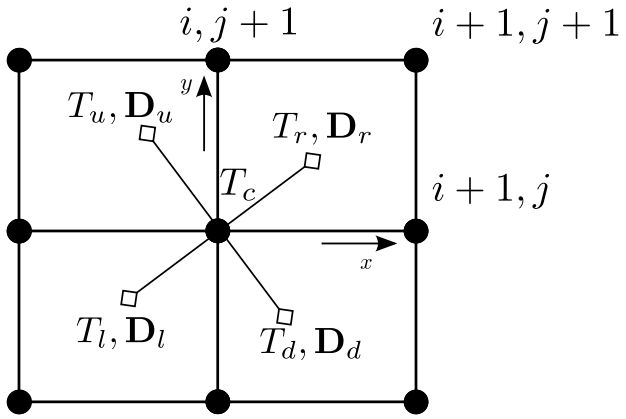


Figure 4: Locally transformed grid, 5-point stencil

The values at the locations r, l, u, d are determined by bi-quadratic interpolation:

$$v(x,y) = c_1 x^2 y^2 + c_2 x^2 y + c_3 y^2 x + c_4 x^2 + c_5 y^2 + c_6 xy + c_7 x + c_8 y + c_9, \quad x, y \in [-h, h], \tag{4}$$

where v can represent $T, b_1, b_2, D_\parallel$ or $D_\perp$. For convenience we assume that we have a uniform Cartesian grid with $\Delta x = \Delta y = h$. Then, for T, the coefficients $c_1, \ldots c_9$ follow from[2]

$$\begin{pmatrix} c_1\\ c_2\\ c_3\\ c_4\\ c_5\\ c_6\\ c_7\\ c_8\\ c_9 \end{pmatrix} = \mathbf{V}^{-1}\mathbf{T}, \quad \mathbf{V} = \begin{pmatrix} h^4 & h^3 & -h^3 & h^2 & h^2 & -h^2 & -h & h & 1\\ h^4 & h^3 & h^3 & h^2 & h^2 & h^2 & h & h & 1\\ h^4 & -h^3 & -h^3 & h^2 & h^2 & h^2 & -h & -h & 1\\ h^4 & -h^3 & h^3 & h^2 & h^2 & -h^2 & h & -h & 1\\ 0 & 0 & 0 & h^2 & 0 & 0 & -h & 0 & 1\\ 0 & 0 & 0 & h^2 & 0 & 0 & h & 0 & 1\\ 0 & 0 & 0 & 0 & h^2 & 0 & 0 & h & 1\\ 0 & 0 & 0 & 0 & h^2 & 0 & 0 & -h & 1\\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \mathbf{T} = \begin{pmatrix} T_{i-1,j+1}\\ T_{i+1,j+1}\\ T_{i-1,j-1}\\ T_{i+1,j-1}\\ T_{i-1,j}\\ T_{i+1,j}\\ T_{i,j+1}\\ T_{i,j-1}\\ T_{i,j} \end{pmatrix},$$

The matrix $\mathbf{V}$ contains the polynomial terms for each node, see figure 4. The coefficients $c_1, \cdots, c_9$ are now given

[2] similarly for $b_1, b_2, D_\parallel, D_\perp$

by

$$
\begin{aligned}
c_1^V &= \frac{1}{h^4}\left(T_{i,j} - \frac{T_{i,j-1}}{2} - \frac{T_{i-1,j}}{2} - \frac{T_{i+1,j}}{2} - \frac{T_{i,j+1}}{2} + \frac{T_{i-1,j-1}}{4} + \frac{T_{i+1,j-1}}{4} + \frac{T_{i+1,j+1}}{4} + \frac{T_{i-1,j+1}}{4}\right),\\
c_2^V &= \frac{1}{4h^3}\left(2T_{i,j-1} - 2T_{i,j+1} + T_{i-1,j+1} + T_{i+1,j+1} - T_{i-1,j-1} - T_{i+1,j-1}\right),\\
c_3^V &= \frac{1}{4h^3}\left(2T_{i-1,j} - 2T_{i+1,j} + T_{i+1,j-1} + T_{i+1,j+1} - T_{i-1,j-1} - T_{i-1,j+1}\right),\\
c_4^V &= \frac{1}{2h^2}\left(T_{i-1,j} - 2T_{i,j} + T_{i+1,j}\right), \quad c_5^V = \frac{1}{2h^2}\left(T_{i,j-1} - 2T_{i,j} + T_{i,j+1}\right),\\
c_6^V &= \frac{1}{4h^2}\left(T_{i-1,j-1} + T_{i+1,j+1} - T_{i+1,j-1} - T_{i-1,j+1}\right),\\
c_7^V &= \frac{T_{i+1,j} - T_{i-1,j}}{2h}, \quad c_8^V = \frac{T_{i,j+1} - T_{i,j-1}}{2h},\\
c_9^V &= T_{i,j},
\end{aligned}
$$

where the superscript V denotes $Vandermonde$. Note that the coefficients $c_1, \ldots c_8$ are all approximations of differential terms in point i, j,

$$
\begin{array}{lll}
c_1 = \frac{1}{4}T_{xxyy} + \mathcal{O}(h^2), & c_2 = \frac{1}{2}T_{xxy} + \mathcal{O}(h^2), & c_3 = \frac{1}{2}T_{yyx} + \mathcal{O}(h^2),\\
c_4 = \frac{1}{2}T_{xx} + \mathcal{O}(h^2), & c_5 = \frac{1}{2}T_{yy} + \mathcal{O}(h^2), & c_6 = T_{xy} + \mathcal{O}(h^2),\\
c_7 = T_x + \mathcal{O}(h^2), & c_8 = T_y + \mathcal{O}(h^2). &
\end{array}
$$

For comparison purposes we change the coefficients that represent T_x, T_y, T_{xx} and T_{yy} to involve more nodes to approximate the respective differentials,

$$
\begin{aligned}
c_4^S &= \frac{1}{8h^2}\left(T_{i-1,j+1} + T_{i-1,j-1} - 2T_{i,j-1} + 2T_{i-1,j} - 4T_{i,j} + 2T_{i+1,j} - 2T_{i,j+1} + T_{i+1,j+1} + T_{i+1,j-1}\right),\\
c_5^S &= \frac{1}{8h^2}\left(T_{i-1,j+1} + T_{i-1,j-1} - 2T_{i-1,j} + 2T_{i,j-1} - 4T_{i,j} + 2T_{i,j+1} - 2T_{i+1,j} + T_{i+1,j+1} + T_{i+1,j-1}\right),\\
c_7^S &= \frac{1}{8h}\left(2T_{i+1,j} + T_{i+1,j+1} + T_{i+1,j-1} - 2T_{i-1,j} - T_{i-1,j+1} - T_{i-1,j-1}\right),\\
c_8^S &= \frac{1}{8h}\left(2T_{i,j+1} + T_{i-1,j+1} + T_{i+1,j+1} - 2T_{i,j-1} - T_{i-1,i-1} - T_{i+1,j-1}\right).
\end{aligned}
$$

This is equivalent to

$$
c_4^S = c_4^V + c_1^V\frac{1}{2}h^2, \quad c_5^S = c_5^V + c_1^V\frac{1}{2}h^2, \quad c_7^S = c_7^V + c_3^V\frac{1}{2}h^2, \quad c_8^S = c_8^V + c_2^V\frac{1}{2}h^2,
$$

where the superscript S denotes symmetric. The reasoning is that the Vandermonde coefficients represent the asymmetric scheme for spatially constant diffusion tensor and likewise the symmetric coefficients represent the mimetic (or symmetric) scheme for a spatially constant diffusion tensor. These are consistent approximations of the differential terms. However, when using these coefficients in the bi-quadratic interpolation they do not exactly yield all nodal values for the given locations.
The locations of r, l, u, d are based on the field line, a first estimate is to apply a single step in the direction of the field line. With s the coordinate in field line direction, n the coordinate normal to it and with Δs and Δn the steps in both directions, the locations then become

$$
\begin{aligned}
(x_r, y_r) = (b_1, b_2)\Delta s, \quad (x_l, y_l) = (-b_1, -b_2)\Delta s,\\
(x_u, y_u) = (-b_2, b_1)\Delta n, \; (x_d, y_d) = (b_2, -b_1)\Delta n.
\end{aligned}
\tag{5}
$$

Now we apply these coordinates (5) to construct discrete schemes in s, n-coordinates for the individual parts $\mathcal{A}_1$,$\mathcal{A}_2$,$\mathcal{A}_3$ and $\mathcal{A}_4$.

Consistency Analysis

The following analysis holds for both the symmetric and the Vandermonde coefficients, the superscripts of the coefficients will denote the variable to which they apply. We remark that although the accuracy requirement holds

for the sum $\mathcal{A}_1 + \mathcal{A}_2 + \mathcal{A}_3 + \mathcal{A}_4$, we choose to impose it on $\mathcal{A}_1$,$\mathcal{A}_2$,$\mathcal{A}_3$ and $\mathcal{A}_4$ individually.
For the approximation of $\mathcal{A}_4$ we have the following expression:

$$\mathcal{A}_4 = \frac{D_{\parallel_r} - D_{\parallel_l}}{2\Delta s}\frac{T_r - T_l}{2\Delta s} + \frac{D_{\perp_u} - D_{\perp_d}}{2\Delta n}\frac{T_u - T_d}{2\Delta n}. \tag{6}$$

To verify that this scheme approximates part $\mathcal{A}_4$ second-order accurately we substitute the interpolation functions in equation (6) and we collect the coefficients:

$$\begin{aligned}
0^{th}\text{-order:}&\frac{1}{4\Delta s^2}\left(c_7^{D_\parallel}(x_r - x_l) + c_8^{D_\parallel}(y_r - y_l)\right)\left(c_7^T(x_r - x_l) + c_8^T(y_r - y_l)\right),\\
&\frac{1}{4\Delta n^2}\left(c_7^{D_\perp}(x_u - x_d) + c_8^{D_\perp}(y_u - y_d)\right)\left(c_7^T(x_u - x_d) + c_8^T(y_u - y_d)\right),\\
1^{st}\text{-order:}&\frac{1}{4\Delta s^2}\left(c_7^{D_\parallel}(x_r - x_l) + c_8^{D_\parallel}(y_r - y_l)\right)\left(c_4^T\left(x_r^2 - x_l^2\right) + c_5^T\left(y_r^2 - y_l^2\right) + c_6^T\left(x_r y_r - x_l y_l\right)\right),\\
&\frac{1}{4\Delta n^2}\left(c_7^{D_\perp}(x_u - x_d) + c_8^{D_\perp}(y_u - y_d)\right)\left(c_4^T\left(x_u^2 - x_d^2\right) + c_5^T\left(y_u^2 - y_d^2\right) + c_6^T\left(x_u y_d - x_u y_d\right)\right),
\end{aligned}$$

where the superscripts of the interpolation coefficients represent the variable to which the interpolation applies. Now the 0^{th}-order expression must be equal to $\mathcal{A}_4$ and the 1^{st}-order expression must be zero. The requirements that can be distilled from this are

$$\begin{array}{lll}
(x_r - x_l)^2 = 4b_1^2\Delta s^2, & (y_r - y_l)^2 = 4b_2^2\Delta s^2, & (x_r - x_l)(y_r - y_l) = 4b_1 b_2 \Delta s^2,\\
(x_u - x_d)^2 = 4b_2^2\Delta n^2, & (y_u - y_d)^2 = 4b_1^2\Delta n^2, & (x_u - x_d)(y_u - y_d) = -4b_1 b_2 \Delta n^2,\\
x_{r,u}^2 - x_{l,d}^2 = 0, & y_{r,u}^2 - y_{l,d}^2 = 0, & x_{r,u}y_{r.u} - x_{l,d}y_{l,d} = 0.
\end{array}$$

This holds for the locations given by equation (5). It appears that the first-order term $\mathcal{A}_4$ can be approximated with second-order accuracy.
For the second-order terms in $\mathcal{A}_3$ we apply the following finite difference formula

$$\mathcal{A}_3 = D_\parallel\frac{T_r - 2T_c + T_l}{\Delta s^2} + D_\perp\frac{T_u - 2T_c + T_d}{\Delta n^2}. \tag{7}$$

Substituting the interpolation values in equation (7) and collecting terms by order in h gives

$$\begin{aligned}
-1^{st}\text{-order:}&\ \frac{D_\parallel}{\Delta s^2}\left(c_7^T(x_r + x_l) + c_8^T(y_r + y_l)\right) + \frac{D_\perp}{\Delta n^2}\left(c_7^T(x_u + x_d) + c_8^T(y_u + y_d)\right),\\
0^{th}\text{-order:}&\ \frac{D_\parallel}{\Delta s^2}\left(c_4^T(x_r^2 + y_l^2) + c_5^T(y_r^2 + y_l^2) + c_6^T(x_r y_r + x_l y_l)\right)\\
&+ \frac{D_\perp}{\Delta n^2}\left(c_4^T(x_u^2 + y_d^2) + c_5^T(y_u^2 + y_d^2) + c_6^T(x_u y_u + x_d y_d)\right),\\
1^{st}\text{-order:}&\ \frac{D_\parallel}{\Delta s^2}\left(c_2^T(x_r^2 y_r + x_l^2 y_l) + c_3^T(y_r^2 x_r + y_l^2 x_l)\right)\\
&+ \frac{D_\perp}{\Delta n^2}\left(c_2^T(x_u^2 y_u + x_d^2 y_d) + c_3^T(y_u^2 x_u + y_d^2 x_d)\right),
\end{aligned}$$

where the -1^{st}- and 1^{st}-order term should be zero, and the 0^{th}-order terms should be equal to $\mathcal{A}_3$. This gives the following requirements

$$\begin{array}{llll}
x_{r,u} + x_{l,d} = 0, & y_{r,u} + y_{l,d} = 0, & x_{r,u}^2 y_{r,u} + x_{l,d}^2 y_{l,d} = 0, & x_{r,u}y_{r,u}^2 + x_{l,d}y_{l,d}^2 = 0,\\
x_r y_r + x_l y_l = 2b_1 b_2 \Delta s^2, & y_r^2 + y_l^2 = 2b_2^2\Delta s^2, & x_r^2 + x_l^2 = 2b_1^2\Delta s^2, &\\
x_u y_u + x_d y_d = -2b_1 b_2 \Delta n^2, & y_u^2 + y_d^2 = 2b_1^2\Delta n^2, & x_u^2 + x_d^2 = 2b_2^2\Delta n^2. &
\end{array}$$

These requirements are fulfilled by the location set described by (5).
We also apply centered differencing for the first-order terms in $\mathcal{A}_2$:

$$\mathcal{A}_2 = \left(D_\parallel - D_\perp\right)\left(-b_2\frac{b_{1_u} - b_{1_d}}{2\Delta n} + b_1\frac{b_{2_u} - b_{2_d}}{2\Delta n}\right)\frac{T_r - T_l}{2\Delta s}. \tag{8}$$

Substituting the interpolation values in equation (8) and collecting terms by order in h gives

$$
\begin{aligned}
0^{th}\text{-order: } & \frac{D_{\parallel}-D_{\perp}}{4\Delta s\Delta n}\left[(-b_2c_7^{b_1}+b_1c_7^{b_2})(x_u-x_d)+(-b_2c_8^{b_1}+b_1c_8^{b_2})(y_u-y_d)\right]\left[c_7^T(x_r-x_l)+c_8^T(y_r-y_l)\right],\\
1^{st}\text{-order: } & -\frac{D_{\parallel}-D_{\perp}}{4\Delta s\Delta n}b_2\left[c_4^{b_1}(x_u^2-x_d^2)+c_5^{b_1}(y_u^2-y_d^2)+c_6^{b_1}(x_uy_u-x_dy_d)\right]\left[c_7^T(x_r-x_l)+c_8^T(y_r-y_l)\right]+\\
& \frac{D_{\parallel}-D_{\perp}}{4\Delta s\Delta n}b_1\left[c_4^{b_2}(x_u^2-x_d^2)+c_5^{b_2}(y_u^2-y_d^2)+c_6^{b_2}(x_uy_u-x_dy_d)\right]\left[c_7^T(x_r-x_l)+c_8^T(y_r-y_l)\right]-\\
& \frac{D_{\parallel}-D_{\perp}}{4\Delta s\Delta n}b_2\left[c_4^T(x_r^2-x_l^2)+c_5^T(y_r^2-y_l^2)+c_6^T(x_ry_r-x_ly_l)\right]\left[c_7^{b_1}(x_u-x_d)+c_8^{b_1}(y_u-y_d)\right]+\\
& \frac{D_{\parallel}-D_{\perp}}{4\Delta s\Delta n}b_1\left[c_4^T(x_r^2-x_l^2)+c_5^T(y_r^2-y_l^2)+c_6^T(x_ry_r-x_ly_l)\right]\left[c_7^{b_2}(x_u-x_d)+c_8^{b_2}(y_u-y_d)\right].
\end{aligned}
$$

After substitution of the location set we have that the 0^{th}-order terms are equal to $\mathcal{A}_2$ and the 1^{st}-order terms are zero.
Finally we apply centered differencing for the first-order terms in $\mathcal{A}_1$ to obtain the approximation

$$
\mathcal{A}_1 = -\left(D_{\parallel}-D_{\perp}\right)\left(-b_1\frac{b_{2_r}-b_{2_l}}{2\Delta s}+b_2\frac{b_{1_r}-b_{1_l}}{2\Delta s}\right)\frac{T_u-T_d}{2\Delta n}. \tag{9}
$$

Substituting the interpolation values in equation (9) and collecting terms by order in h gives

$$
\begin{aligned}
0^{th}\text{-order: } & -\frac{D_{\parallel}-D_{\perp}}{4\Delta s\Delta n}\left[(b_2c_7^{b_1}-b_1c_7^{b_2})(x_r-x_l)+(b_2c_8^{b_1}-b_1c_8^{b_2})(y_r-y_l)\right]\left[c_7^T(x_u-x_d)+c_8^T(y_u-y_d)\right],\\
1^{st}\text{-order: } & -\frac{D_{\parallel}-D_{\perp}}{4\Delta s\Delta n}b_1\left[c_4^{b_2}(x_r^2-x_l^2)+c_5^{b_2}(y_r^2-y_l^2)+c_6^{b_2}(x_ry_r-x_ly_l)\right]\left[c_7^T(x_u-x_d)+c_8^T(y_u-y_d)\right]+\\
& \frac{D_{\parallel}-D_{\perp}}{4\Delta s\Delta n}b_2\left[c_4^{b_1}(x_r^2-x_l^2)+c_5^{b_1}(y_r^2-y_l^2)+c_6^{b_1}(x_ry_r-x_ly_l)\right]\left[c_7^T(x_u-x_d)+c_8^T(y_u-y_d)\right]-\\
& \frac{D_{\parallel}-D_{\perp}}{4\Delta s\Delta n}b_1\left[c_4^T(x_u^2-x_d^2)+c_5^T(y_u^2-y_d^2)+c_6^T(x_uy_u-x_dy_d)\right]\left[c_7^{b_2}(x_r-x_l)+c_8^{b_2}(y_r-y_l)\right]+\\
& \frac{D_{\parallel}-D_{\perp}}{4\Delta s\Delta n}b_2\left[c_4^T(x_u^2-x_d^2)+c_5^T(y_u^2-y_d^2)+c_6^T(x_uy_u-x_dy_d)\right]\left[c_7^{b_1}(x_r-x_l)+c_8^{b_1}(y_r-y_l)\right].
\end{aligned}
$$

After substitution of the location set (5) we have that the 0^{th}-order terms are equal to $\mathcal{A}_1$ and the 1^{st}-order terms are zero.
We call this method *aligned Vandermonde* or *aligned symmetric* depending on the coefficients. In practice we decrease Δs and Δn with increasing anisotropy, and we may simply and safely take $\Delta s = \Delta n$.

Curvature Terms

The aligned schemes presented before assume that the direction does not change up to the interpolation points r, l, u, d. Now we consider a numerical treatment of the terms $b_{1_s}, b_{1_n}, b_{2_s}, b_{2_n}$ based on field line curvature. First we write the terms as

$$
b_{1_s} = x_{ss}, \quad b_{1_n} = y_{nn}, \quad b_{2_s} = y_{ss}, \quad b_{2_n} = -x_{nn}.
$$

This leads to the following equations for S, N:

$$
S = -b_2y_{nn} - b_1x_{nn},, \quad N = -b_1y_{ss} + b_2x_{ss}.
$$

The curvature terms can be approximated by

$$
x_{ss} = \frac{x_r+x_l}{\Delta s^2}, \quad y_{ss} = \frac{y_r+y_l}{\Delta s^2}, \quad x_{nn} = \frac{x_u+x_d}{\Delta n^2}, \quad y_{nn} = \frac{y_u+y_d}{\Delta n^2}, \tag{10}
$$

where the positions r, l, u, d are not to be confused with the positions we used for the aligned stencil depicted in figure 4. We are now explicitly looking for curvature. Given an interpolation function for b_1 and b_2 within the stencil area we can apply tracing to find subsequent points. We go from the center point to the interpolation points

r, l, u, d by applying the (second-order accurate) modified Euler scheme (Heun):

tangential direction:

$$\mathbf{x}_k^* = \mathbf{x}_{k-1} \pm \Delta s^* \mathbf{b}(x_{k-1}, y_{k-1})$$
$$\mathbf{x}_k = \mathbf{x}_{k-1} \pm \frac{1}{2}\Delta s^* \left(\mathbf{b}(x_{k-1}, y_{k-1}) + \mathbf{b}(x_k^*, y_k^*)\right), \quad k = 1, \cdots K,$$

normal direction:

$$\mathbf{x}_k^* = \mathbf{x}_{k-1} \pm \Delta n^* \mathbf{b}_\perp(x_{k-1}, y_{k-1})$$
$$\mathbf{x}_k = \mathbf{x}_{k-1} \pm \frac{1}{2}\Delta n^* \left(\mathbf{b}_\perp(x_{k-1}, y_{k-1}) + \mathbf{b}_\perp(x_k, y_k)\right), \quad k = 1, \cdots K,$$

where K is the number of substeps, and where $x_0 = y_0 = 0$ (see figure 5). The values $\Delta s = K\,\Delta s^*$ and $\Delta n = K\,\Delta n^*$ are used in equation (10).

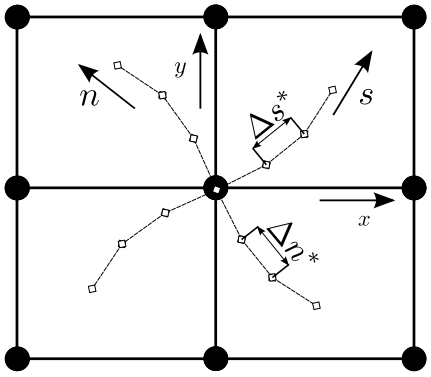


Figure 5: Approximate track of field line

Repeatedly stepping in s, n-direction and applying the interpolation of $\mathbf{b}$ increases the computational cost. The benefit though is that we can easily control the accuracy with which we follow the field line, simply by changing the number of tracing steps.
Still note that the approach to more accurately determine S, N can only improve the accuracy of $\mathcal{A}_1$ and $\mathcal{A}_2$.

Exact Differentiation after Interpolation

We can also find a direct approximation of the various spatial derivatives involved in the anisotropic diffusion operator, by writing the interpolation function (4) in terms of s, n and by taking the appropriate derivatives of this rewritten function. Then, the interpolation functions for b_1 and b_2 need to be applied to find the final form of the approximation. We use the non-conservative form

$$T_t = D_\parallel v_{ss}^T + D_\perp v_{nn}^T + v_s^{D_\parallel} v_s^T + v_n^{D_\perp} v_n^T + \left(D_\parallel - D_\perp\right)\left(S v_s^T - N v_n^T\right),$$

where the terms with v represent the derivatives of the bi-quadratic interpolation functions for the quantities denoted by the superscript, i.e., v^T is the interpolation function for the temperature. The first-order differentials are written as

$$v_s^{D_\parallel} v_s^T + v_n^{D_\perp} v_n^T = (c_7^T b_1 + c_8^T b_2)(c_7^{D_\parallel} b_1 + c_8^{D_\parallel} b_2) + (-c_7^T b_2 + c_8^T b_1)(-c_7^{D_\perp} b_2 + c_8^{D_\perp} b_1).$$

The diffusive terms are given by

$$D_\parallel v_{ss}^T + D_\perp v_{nn}^T = 2D_\parallel \left(c_4 b_1^2 + c_5 b_2^2 + c_6 b_1 b_2\right) + 2D_\perp \left(c_4 b_2^2 + c_5 b_1^2 - c_6 b_1 b_2\right),$$

and the curvature-dependent terms by

$$\left(D_\parallel - D_\perp\right)\left(S v_s^T - N v_n^T\right) = 2D_\parallel \left[c_7\left(b_1 c_7^{b_1} + \frac{1}{2}b_1 c_8^{b_2} + \frac{1}{2}b_2 c_8^{b_1}\right) + c_8\left(b_2 c_8^{b_2} + \frac{1}{2}b_2 c_7^{b_1} + \frac{1}{2}b_1 c_7^{b_2}\right)\right] +$$
$$2D_\perp \left[c_7\left(b_2 c_7^{b_2} - \frac{1}{2}b_1 c_8^{b_2} - \frac{1}{2}b_2 c_8^{b_1}\right) + c_8\left(b_1 c_8^{b_1} - \frac{1}{2}b_2 c_7^{b_1} - \frac{1}{2}b_1 c_7^{b_2}\right)\right].$$

The geometric term is recursive since b_1, b_2 depend on x, y whereas the latter depend on b_1, b_2. We call these methods *interp. Vandermonde* or *interp. symmetric*, depending on the coefficients that are used.

NUMERICAL RESULTS

In this section we show numerical results for four test cases. In all test cases $\mathbf{b}\cdot\nabla T$ is zero. This foreknowledge is not used though; the general expressions $\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3$ and $\mathcal{A}_4$ according to (3) are used. We define the anisotropy as

$$\varsigma = \frac{D_\parallel}{D_\perp},$$

where $D_\perp$ is one by default.

Constant Angle of Misalignment

As an initial test we consider a simple steady diffusion problem. The imposed exact solution reads:

$$T(x,y) = xy\left[\sin(\pi x)\sin(\pi y)\right]^s, \quad x, y \in [0,1],$$

where s is large and the angle of misalignment α is set to a constant value. The solution simulates a temperature peak. Computational results for this test case are given in figure 6.

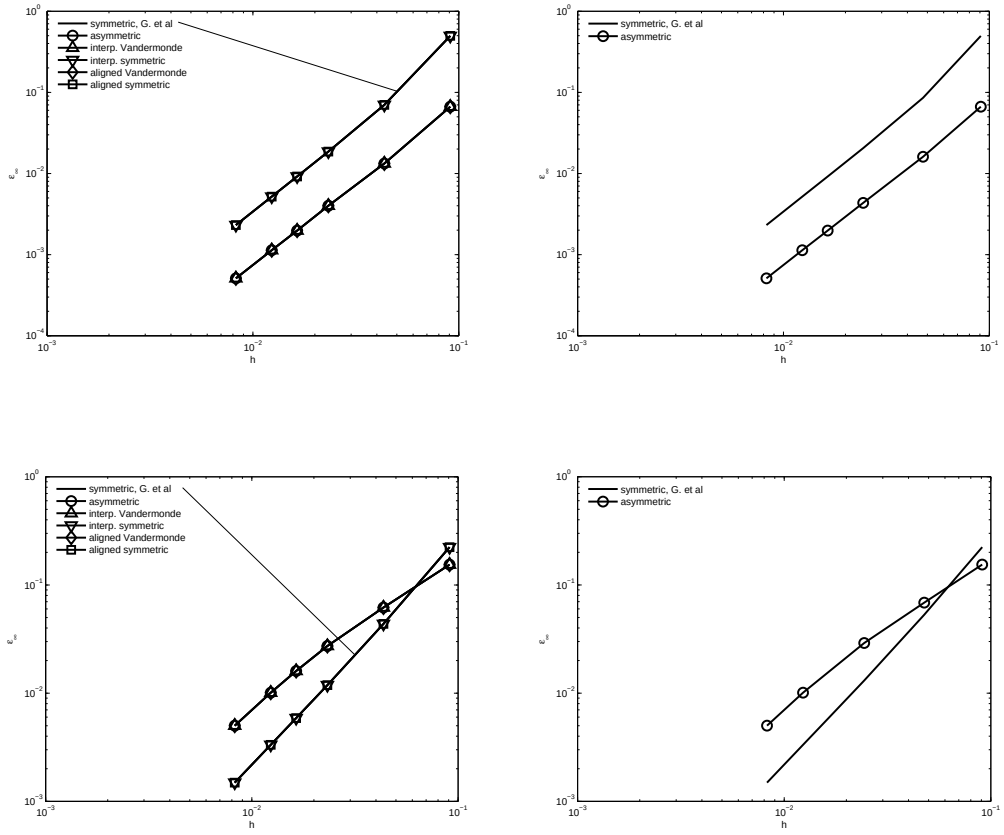


Figure 6: Error ϵ_∞ for test cases with constant angles of misalignment, $s = 10$, $\varsigma = 10^9$, at varying mesh width, top: $\alpha = 5°$, bottom: $\alpha = 30°$, left: co-located, right: staggered. In the plots for the co-located schemes all symmetric schemes overlap and likewise do all asymmetric schemes.

The error norm is defined by

$$\epsilon_\infty = \frac{|\tilde{T} - T|_{max}}{|T|_{max}},$$

where $\tilde{T}$ is the approximate temperature. It is clear from the figure that the symmetric schemes conserve the order of accuracy independent of the anisotropy and angle of misalignment. The co-located schemes are only slightly less accurate than the staggered. For larger values, the asymmetric schemes are less than second-order convergent on coarse grids, but they regain second-order convergence on finer grids.

Varying Angle of Misalignment

Again the problem is considered on a square domain, this time described by $-0.5 \leq x, y \leq 0.5$. The following steady-state solution is assumed on the domain

$$T(x, y) = 1 - (x^2 + y^2)^{3/2}.$$

The direction in which the parallel diffusion acts is given by

$$\mathbf{b} = \frac{1}{\sqrt{x^2 + y^2}} \begin{pmatrix} -y \\ x \end{pmatrix}. \tag{11}$$

Note that both $\nabla \cdot \mathbf{b}$ and $\mathbf{b} \cdot \nabla T$ are zero. This implies that the term $\mathcal{A}_2$ comes into play only due to numerical errors. Term $\mathcal{A}_4$ is exactly zero since $\nabla D_{\parallel}$, $\nabla D_{\perp}$ are zero. Test case 2 stresses terms $\mathcal{A}_1$ and $\mathcal{A}_3$, with added contribution due to numerical errors in term $\mathcal{A}_2$.

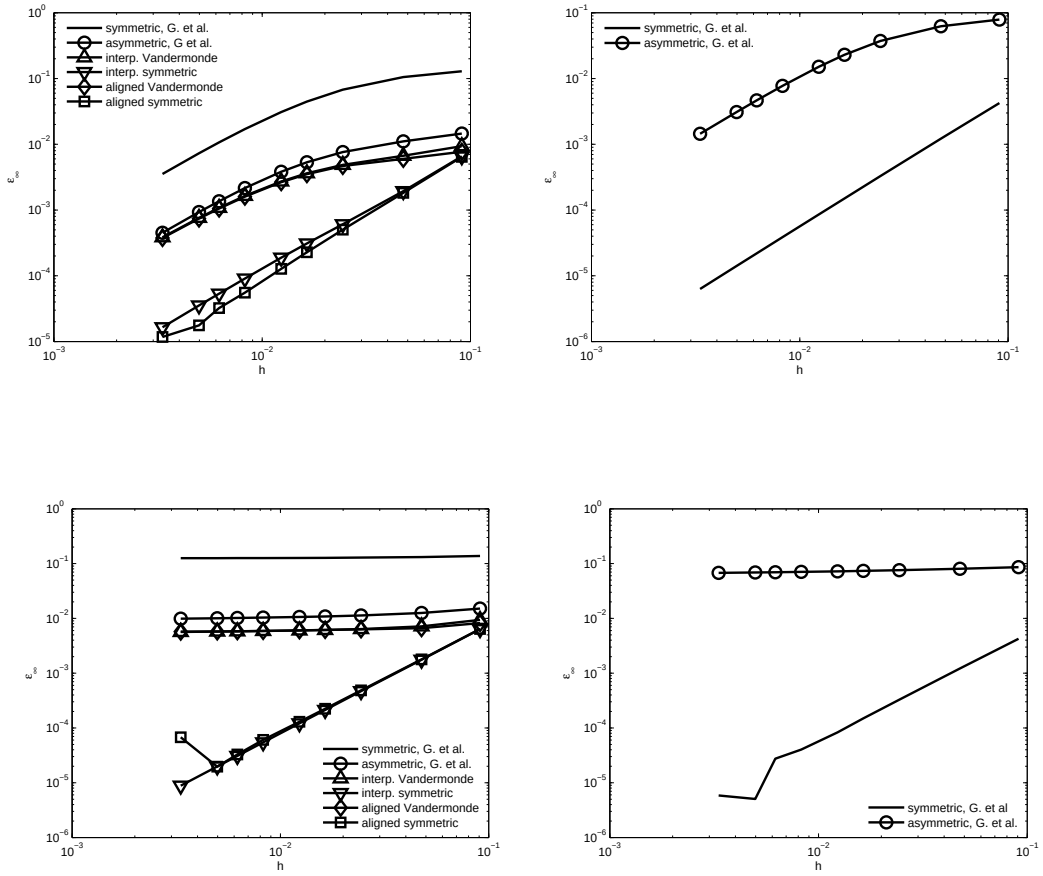


Figure 7: Error ϵ_∞ for test cases with varying misalignment, left: co-located, right: staggered, top: $\varsigma = 10^3$, bottom: $\varsigma = 10^9$

In figure 7, we study the accuracy of the various schemes for two anisotropic cases, one being extremely anisotropic, $\varsigma = 10^9$. The main observation to be made from figure 7 is that for the extremely anisotropic $\varsigma = 10^9$ case only the aligned symmetric scheme and the interpolated symmetric scheme preserve their second-order of accuracy. All other schemes fail completely; they are all inconsistent for the $\varsigma = 10^9$ test case.

A detail to be observed from figure 7 is that for extremely high levels of anisotropy the staggered, symmetric scheme of Günter at al shows a wiggle in the error convergence. This is caused by the fact that this scheme becomes less well-conditioned with increasing resolution. Günter et al [7] had problems with number representation for a fourth-order mimetic finite difference scheme. They resolved this by increasing the number representation accuracy. Further, it can be shown that the analytical problem becomes ill-posed for $\varsigma \to \infty$ (see Degond et al [8]).

Finally, in figure 8 we make a more extensive study of the behavior of the different schemes at varying anisotropy. Here, it appears again the better performance of our interpolated symmetric scheme and aligned symmetric scheme; their errors do not increase at increasing anisotropy. For the following two test cases we will only proceed with the schemes that appear to be consistent in Figure 7.

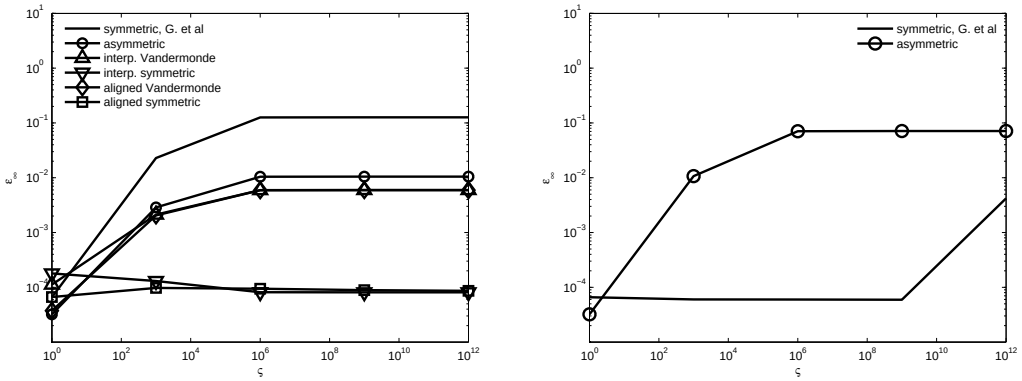


Figure 8: ϵ_∞-error norm versus the anisotropy ς for $h = 0.01$

Perpendicular Numerical Diffusion

Sovinec et al [1] devised a test to directly compare the numerically computed perpendicular diffusion to the exact perpendicular diffusion. This test case is also considered by Günter et al [6] and Sharma et al [4]. The exact solution and the forcing function are given by

$$T = \frac{1}{D_\perp}\psi, \quad f = 2\pi^2\psi, \quad \psi = \cos(\pi x)\cos(\pi y), \quad x, y \in [-0.5, 0.5].$$

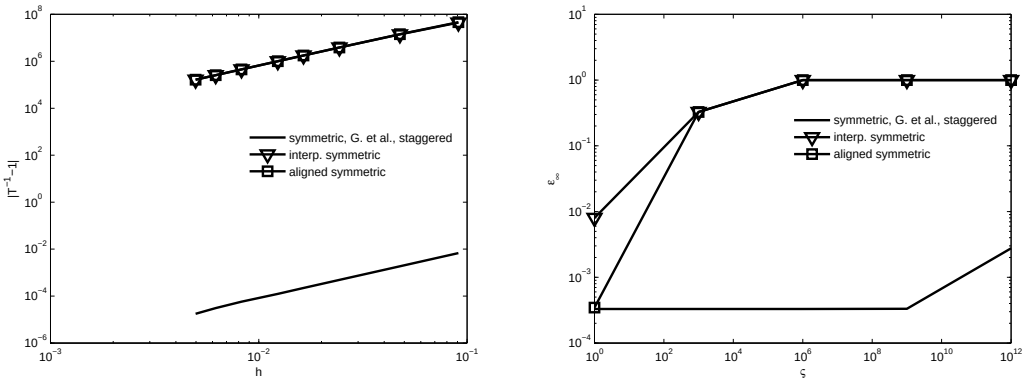
The error in the perpendicular diffusion is given by

$$|T(0,0)^{-1} - D_\perp|.$$

We use homogeneous Dirichlet boundary conditions. The field lines are tangential to the contours of constant temperature, i.e.

$$\mathbf{b} = \frac{1}{\sqrt{\psi_x^2 + \psi_y^2}} \begin{pmatrix} -\psi_y \\ \psi_x \end{pmatrix}.$$

Numerical results, given in Figure 9a, show second-order accuracy for all three schemes. However for this test case we also see some anisotropy dependence of the accuracy, see figure 9b.



(a) varying h, $\varsigma = 10^9$ (b) varying ς, $h = 0.01$

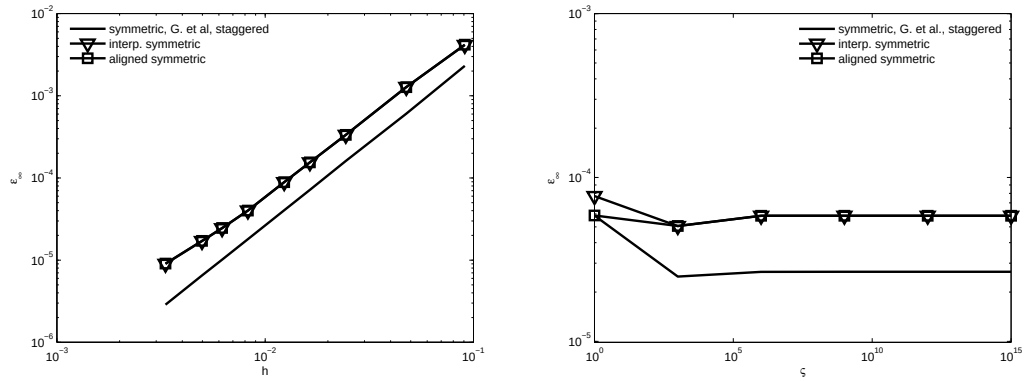
Figure 9: Error in perpendicular diffusion, $|T^{-1} - 1|$, $\varsigma = 10^9$

Tilted Elliptic Temperature Distributions

So far, we have used forcing functions which are spatially symmetric. Now we will apply a forcing function that gives the solution for a tilted elliptic temperature distribution. This distribution has no symmetry axes aligned with the coordinate axes. The tilted elliptic distribution has no rotating field lines, basically the field lines go in the same general direction. The exact solution is given by

$$T(x,y) = 1 + (ax + by)(x^2 + y^2)^{3/2}, \quad \mathbf{b} = \frac{1}{\sqrt{T_x^2 + T_y^2}} \begin{pmatrix} -T_y \\ T_x \end{pmatrix}, \quad x, y \in [-0.5, 0.5].$$

From the numerical results given in figure 10 we see that all three schemes considered have good accuracy behavior.



(a) varying h, $\varsigma = 10^9$ (b) varying ς, $h = 0.01$

Figure 10: ϵ_∞ behavior, $a = 25, b = -75$

CONCLUSION

We have presented a new finite difference approach for problems with strong anisotropic diffusion. The approach uses the concept of following the field line within the stencil area, to obtain the differencing points that are finally used in the discretization. For the test cases considered, the approach works well in maintaining the order of convergence independent of the level of anisotropy.

REFERENCES

[1] Sovinec CR, Glasser AH, Gianakon TA, Barnes DC, Nebel RA, Kruger SE, Schnack DD, Plimpton SJ, Tarditi A, Chu MS, the Nimrod team. Nonlinear magnetohydrodynamics simulation using high-order finite elements. Journal of Computational Physics, 2004; 195: 355-386.

[2] Meier ET, Lukin VS, Shumlak U. Spectral element spatial discretization error in solving highly anisotropic heat conduction equation. Computer Physics Communications, 2010; 181: 837-841.

[3] Umansky MV, Day MS, Rognlien TD. On numerical solution of strongly anisotropic diffusion equation on misaligned grids. Numerical Heat Transfer, Part B, 2007; 47: 533-554.

[4] Sharma P, Hammett GW. Preserving monotonicity in anisotropic diffusion. Journal of Computational Physics, 2007; 227: 123-142.

[5] Babuška I, Suri M. On locking and robustness in the finite element method. SIAM Journal on Numerical Analysis, 1992; 220: 751-771.

[6] Günter S, Yu Q, Krüger J, Lackner K. Modelling of heat transport in magnetised plasmas using non-aligned coordinates. Journal of Computational Physics, 2005; 209: 354-370.

[7] Günter S, Lackner K, Tichmann C. Finite element and higher order difference formulations for modelling heat transport in magnetised plasmas. Journal of Computational Physics, 2007; 226: 2306-2316.

[8] Degond P, Deluzet F, Lozinski A, Narski J, Negulescu C. Duality-based Asymptotic-Preserving method for highly anisotropic diffusion equations. ArXiv e-prints, August 2010.