Freie Universität Berlin

Iterative Solvers Respecting the Discrete Maximum Principle for Steady-State Convection-Diffusion-Reaction Equations

Master's Thesis submitted for the degree of Master of Science (M.Sc.) at the Department of Mathematics and Computer Science of Freie Universität Berlin

Student: Shi Xu

Supervisor: Univ.-Prof. Dr. Volker John

Second Reviewer: Dr. Alfonso Caiazzo

Berlin, June 26, 2026

Declaration of Authorship

I hereby declare that I have written this thesis independently and that I have used no sources or aids other than those cited. All passages that have been taken from other works, either directly or indirectly, have been clearly marked and properly referenced.

I further declare that this thesis has not been submitted, either in identical or similar form, to any other university or examination authority, and that it has not been published elsewhere.

During the preparation of this thesis, I used AI-assisted tools, such as ChatGPT, solely to improve the readability, language quality, and formatting of the text. After using these tools, I carefully reviewed and edited the content. I take full responsibility for all mathematical content, analysis, arguments, and conclusions presented in this thesis.

Berlin, 26.06.2026

Date

Shi Xu

Signature

Abstract

This thesis investigates iterative solution strategies for finite element discretizations of steady-state convection-diffusion-reaction equations, with particular emphasis on discrete maximum principles and the preservation of physically relevant solution bounds. In convection-dominated regimes, standard Galerkin finite element approximations may exhibit non-physical undershoots and overshoots. Monotone low-order discretizations and nonlinear algebraic stabilization methods provide alternatives, but the resulting nonlinear systems require iterative solution techniques.

The thesis studies a fixed-point right-hand-side iteration for the monotone upwind-type algebraically stabilized method and considers four modifications: right-hand-side clipping, positivity-budget-based limiter damping, projection onto prescribed bounds after every iteration, and projection of the final iterate. Particular attention is paid to intermediate iterates and to solutions obtained by terminating the nonlinear iteration early. A main contribution is the construction and numerical assessment of the positivity-budget-based limiter-damping strategy.

The methods are compared on a smooth problem, a layer problem, and the Hemker problem. The numerical results show that the projection-based methods preserve prescribed bounds while generally remaining close to the standard approach. Limiter damping can substantially reduce the number of nonlinear iterations and preserve the expected bounds, but it may produce considerably more diffusive solutions in the presence of sharp layers. Thus, nonlinear convergence, bound preservation, and spatial accuracy must be assessed separately.

Keywords: steady-state convection-diffusion-reaction equation; finite element method; discrete maximum principle; M-matrix; algebraic flux correction; monotone upwind-type algebraically stabilized method; fixed-point iteration; bound preservation; positivity preservation.

Acknowledgments

I would like to express my deepest gratitude to my supervisor, Prof. Dr. Volker John. He is one of the very few people who have sincerely encouraged me in my study of mathematics. He was my first research mentor and introduced me to the field of numerical analysis. Becoming his student has been one of the most fortunate experiences of my more than four years of studying mathematics.

My gratitude to him extends beyond his continuous guidance, valuable suggestions, and careful feedback during the preparation of this thesis. He has also provided me with invaluable advice and support concerning my future academic development. I have learned a great deal from his lecture notes, books, and research articles. In all of these works, I have been deeply impressed by his mathematical rigor, his strong sense of responsibility, and his persistent pursuit of high-quality research. His attitude toward mathematics has had a lasting influence on me. I sincerely hope that one day I may become a mathematician with the same rigor, sense of responsibility, and passion.

I am also grateful to Dr. Alfonso Caiazzo for serving as the second reviewer of this thesis.

Finally, I would like to express my heartfelt gratitude to my parents and my older sister. They supported my decision to come to Germany and pursue my studies in mathematics. My family is not wealthy, and my parents did not have the opportunity to receive a formal education. Although the mathematics I study is entirely unfamiliar to them, they have always supported me selflessly, both emotionally and financially, simply because they wanted me to pursue something that genuinely interests me. Without their sacrifices, understanding, and unconditional support, this journey would not have been possible.

I am grateful to everyone who has helped and encouraged me along the way. I hope that, with their support and inspiration, I will be able to continue moving forward on the path of mathematics.

Contents

List of Notations

Notation	Description	Section
A		
$a(\cdot,\cdot)$	Bilinear form of the weak formulation	2.3
a_{ij}	Entries of the global system matrix A	3.2
A	Global system matrix	3.2
A^i	Active stiffness matrix associated with active nodes	3.2
A^b	Coupling matrix with Dirichlet boundary nodes	3.2
$\mathbb{A}_c$	Convective matrix	3.2.3
$\mathbb{A}_d$	Standard Galerkin diffusion matrix	3.2.3
$\bar{A}$	Stabilized system matrix $A + D$	4.1
$\bar{a}_{ij}$	Entries of the low-order matrix $\bar{A} = A + D$	4.1
$\bar{A}^i$	Active block of the low-order stabilized matrix $\bar{A} = A + D$	4.1
$\bar{A}^b$	Coupling block of the low-order stabilized matrix associated with Dirichlet nodes	4.1
B		
$\mathbf{b}$	Convection field (convective transport)	2.1
$B(\underline{u})$	Solution-dependent nonlinear stabilization matrix	4.1
$B^i(\underline{u})$	Active block of the nonlinear stabilization matrix $B(\underline{u})$	4.1
$b_{ij}(\underline{u})$	Entries of the nonlinear stabilization matrix $B(\underline{u})$	4.1
$\mathcal{B}(\underline{u}^{(\nu)})$	Nonsingular iteration matrix in the general damped nonlinear iteration	5.1
C		
c	Reaction coefficient	2.1
C_P	Poincaré-Friedrichs constant	2.3
c_{ij}	Entries of the convective matrix $\mathbb{A}_c$	3.3.2
D		
d	Dimension of the domain ($d \in \{2, 3\}$)	2.1
D	Symmetric artificial diffusion matrix	4.1
d_{ij}	Entries of the artificial diffusion matrix	4.1
D_i	Barycentric dual cell associated with node $\boldsymbol{x}_i$	3.3.1

Continued on next page

Continued from previous page

Notation	Description	Section
$\lvert D_i \rvert$	Measure (area or volume) of the dual cell D_i	3.3.1
E		
$\underline{e}_i$	i-th canonical unit vector	3.2
E	An edge (for $d = 2$) or a face-edge (for $d = 3$) of a simplex	3.2.3
F		
f	Source term (right-hand side function)	2.1
f_{ij}	Anti-diffusive fluxes in AFC scheme	4.1
f_{ij}^+	Positive part of the anti-diffusive flux f_{ij}	4.2
f_{ij}^-	Negative part of the anti-diffusive flux f_{ij}	4.2
$F(\underline{u})$	Nonlinear residual of the AFC or MUAS algebraic system	5.1
$F_i(\underline{u})$	i-th component of the nonlinear residual	5.1
$f_{ij}^{(\nu)}$	Anti-diffusive flux evaluated at the nonlinear iterate $\underline{u}^{(\nu)}$	5.1
G		
g_N	Prescribed Neumann boundary data	2.1
H		
h	Global mesh parameter (maximum diameter)	3.1.1
h_K	Diameter of a simplex element K	3.1.1
$h_i^\perp$	Height of simplex K corresponding to vertex x_i	3.2.3
K		
$\mathcal{K}_E$	Set of all simplices $K \in \mathcal{T}_h$ sharing the edge E	3.2.3
κ_E^K	The $(d-2)$-dimensional sub-simplex in K opposite to edge E	3.2.3
$\lvert \kappa_E^K \rvert$	Measure (length or area) of the sub-simplex κ_E^K	3.2.3
K_i^{up}	Direction-based upwind simplex for node $\boldsymbol{x}_i$	3.3.2
$K_{i,k}^{\mathrm{up}}$	Component-wise upwind simplex for direction k	3.3.2
L		
$\mathcal{L}$	Second-order linear differential operator	2.2
$l(\cdot)$	Linear functional of the weak formulation	2.3
l_i	i-th component of the load vector, $l_i = l(\phi_i)$	3.2
$\mathcal{L}_h$	Mass lumping operator	3.3.1
$\underline{l}$	Load vector restricted to the active equations	5.2
$\underline{l}^{NL}(\underline{v})$	Nonlinear flux-correction contribution evaluated at the current iterate	5.2

Continued on next page

Continued from previous page

Notation	Description	Section
$l_i^{NL}(\underline{v})$	i-th nonlinear right-hand side contribution generated by flux correction	5.2
M		
m	Number of active degrees of freedom, i.e. non-Dirichlet nodes	3.1.1
$\mathbb{M}$	Consistent mass matrix	3.2.3
$\tilde{m}_{ij}$	Entries of the lumped mass matrix M_L	3.3.1
m_{ij}	Entries of the consistent mass matrix $\mathbb{M}$	3.3.1
M_L	Strictly diagonal lumped mass matrix	3.3.1
$\mathbb{M}_r$	Strictly diagonal lumped reaction matrix	3.3.2
N		
$\mathbf{n}$	Outward pointing unit normal vector	2.1
$\mathcal{N}_h$	Set of vertices of the triangulation, $\mathcal{N}_h = \{\boldsymbol{x}_i\}_{i=1}^n$	3.3.1
$n_i(\underline{v})$	Negative budget of the i-th active equation	5.2.2
$N_{\max}$	Prescribed maximum number of nonlinear iterations	6.1
P		
Pe_K	Local Mesh Péclet Number	3.2.3
P_i^+, P_i^-	Positive and negative nodal flux sums used in the limiter construction	4.2
p_i	Positivity budget of the i-th active equation	5.2.2
Q_i^+, Q_i^-	Admissible positive and negative correction amounts in the Kuzmin limiter	4.2
R		
r_{ij}	Entries of the lumped reaction matrix $\mathbb{M}_r$	3.3.2
R_i^+, R_i^-	Positive and negative nodal limiting factors	4.2
$\underline{r}(\underline{v})$	Assembled right-hand side before clipping	5.2.1
$r_i(\underline{v})$	i-th component of the assembled right-hand side before clipping	5.2.1
$\widetilde{\underline{r}}(\underline{v})$	Clipped assembled right-hand side	5.2.1
$\widetilde{r}_i(\underline{v})$	i-th component of the clipped assembled right-hand side	5.2.1
S		
S_i	Set of neighbor nodes for node i (stencil)	3.1.1
$S_K^{(i)}$	Polyhedral subset of simplex K assigned to node $\boldsymbol{x}_i$	3.3.1
s_{ij}	MUAS weight used in the definition of $Q_i^{\pm}$	4.3
T		

Continued on next page

Continued from previous page

Notation	Description	Section
$\mathcal{T}_h$	Triangulation of the domain	3.1.1
U		
u	Solution of the continuous convection-diffusion-reaction problem	2.1
u_D	Prescribed Dirichlet boundary data	2.1
u^+, u^-	Positive and negative parts of a function	2.4
u_h	Discrete finite element solution	3.1.1
$\underline{u}^b$	Vector of Dirichlet boundary nodal values	3.2
u_i^b	Prescribed Dirichlet value at the i-th Dirichlet node	3.2
$\underline{u}^i$	Vector of active nodal values	3.2
$U_{\text{up,Tab}}$	Tabata upwind operator	3.3.2
$\underline{u}^{(\nu)}$	Nonlinear iterate at iteration step ν	5.1
$\underline{u}^{(0)}$	Initial nonlinear iterate	5.1
$\underline{u}_{\text{Tab}}$	Tabata upwind solution used as initial iterate	5.1
$u_{\min}$	Prescribed lower bound used in the projection step	5.3.1
$u_{\max}$	Prescribed upper bound used in the projection step	5.3.1
$\widehat{\underline{u}}$	Corrected final solution after the final projection step	5.3.2
u_{ref}	Reference cross-section profile used in the layer solution example	6.3
V		
V	Trial space incorporating Dirichlet data u_D	2.3
V_0	Test space with homogeneous boundary conditions	2.3
V_h	Discrete test space associated with active nodes	3.1.1
$\underline{v}$	Current iterate used to evaluate nonlinear coefficients and flux corrections	5.2
W		
W_h	Continuous, piecewise linear FE space	3.1.1
$w_h(4)$	Computed width of the upper interior layer at $x = 4$	6.4
$w_{\text{ref}}(4)$	Reference width of the upper interior layer at $x = 4$	6.4
Y		
$\boldsymbol{y}_i$	Auxiliary point on the upstream half-line within K_i^{up}	3.3.2
$\boldsymbol{\alpha}$		
α_{ij}	Solution-dependent limiters for AFC schemes	4.1
$\tilde{\alpha}_{ij}$	Preliminary edge limiter before symmetrization in the Kuzmin limiter	4.2

Continued on next page

Continued from previous page

Notation	Description	Section
$\alpha_{ij}^{(\nu)}$	Limiter coefficient evaluated at the nonlinear iterate $\underline{u}^{(\nu)}$	5.1
$\widetilde{\alpha}_{ij}$	Damped limiter coefficient used in Algorithm 2	5.2.2
β		
$\beta_{ij}(\underline{u})$	MUAS limiter coefficient	4.3
δ		
δ_{ij}	Kronecker delta function	3.1.1
σ		
σ	Shape-regularity constant of the triangulation family	3.1.1
ε		
ε	Diffusion coefficient (positive constant)	2.1
θ		
θ_E^K	Dihedral angle (or interior angle) of simplex K at edge E	3.2.3
θ_{ij}^K	Dihedral angle opposite to edge connecting i, j	3.2.3
$\theta_i(\underline{v})$	Local limiter damping factor for the i-th degree of freedom	5.2.2
θ_{ij}	Pairwise damping factor applied to a limiter contribution	5.2.2
Π		
Π_h	Standard Lagrange nodal interpolation operator	3.3.2
Γ		
Γ	Boundary of the domain $\partial\Omega$	2.1
Γ_D	Dirichlet boundary	2.1
Γ_N	Neumann boundary	2.1
Γ_{cs}	Cross-section used for evaluating the reference-profile errors	6.3
Ψ		
Ψ_h	Dual space spanned by piecewise constant functions ψ_i	3.3.1
ψ_i	Characteristic function of the dual cell D_i	3.3.1
ρ		
ρ_K	Diameter of the largest inscribed ball in K	3.1.1
ϕ		
ϕ_i	Linear nodal basis function corresponding to $\mathbf{x}_i$	3.1.1
Ω		
Ω	Bounded domain in $\mathbb{R}^d$	2.1

Continued on next page

Continued from previous page

Notation	Description	Section
ω_E	Geometric weight factor associated with edge E	3.2.3
ω_i	Macro-element (patch) sharing the vertex $\boldsymbol{x}_i$	3.3.1
$\omega^{(\nu)}$	Damping parameter in the general nonlinear iteration	5.1
Operators		
∇	Gradient operator	2.1
Δ	Laplace operator	2.1
$\nabla\cdot$	Divergence operator	2.1
$\langle\cdot,\cdot\rangle$	Euclidean inner product	2.3
$(\cdot,\cdot)_h$	The lumped L^2 inner product	3.3
$(\cdot,\cdot)_K$	L^2 inner product in K	3.3
$(\cdot,\cdot)_{L^2}$	L^2 inner product	2.3
$\langle\cdot,\cdot\rangle_{X',X}$	Duality pairing between X' and X	2.3
Norms		
$\|\cdot\|_{L^2(\Omega)}$	Standard L^2-norm	2.3
$\|\cdot\|_{L^\infty(\Omega)}$	Essential supremum norm (L^∞-norm)	2.3
$\|\cdot\|_{H^1(\Omega)}$	Standard Sobolev H^1-norm	2.3
$\|\cdot\|_{L^q(\Omega)}$	L^q-norm for $q \in [1,\infty)$	2.4
$\|\cdot\|_2$	Euclidean norm in $\mathbb{R}^d$	2.2
$\|\cdot\|_{\infty,h}$	Discrete maximum norm	3.2
$\lvert\cdot\rvert_{H^1(\Omega)}$	H^1 seminorm	2.3
Measures		
$\lvert\cdot\rvert$	Lebesgue measure of a set (length, area, or volume)	2.1
$\mathrm{meas}_d(\cdot)$	d-dimensional measure of a set	2.1
$\mathrm{diam}(K)$	Diameter of a simplex element K	3.1.1

List of Figures

List of Tables

1. Introduction

Convection-diffusion-reaction equations arise in the mathematical modelling of transport processes in which diffusion, directed convection, and reaction act simultaneously. In this thesis, we consider the steady-state boundary value problem

$$\begin{aligned} -\varepsilon\Delta u + \mathbf{b}\cdot\nabla u + cu &= f && \text{in } \Omega, \\ u &= u_D && \text{on } \Gamma_D, \\ \varepsilon\nabla u\cdot\mathbf{n} &= g_N && \text{on } \Gamma_N, \end{aligned}$$

where $\Omega \subset \mathbb{R}^d$, $d \in \{2,3\}$, is a bounded Lipschitz domain, $\varepsilon > 0$ is the diffusion coefficient, $\mathbf{b}$ is the convection field, c is the reaction coefficient, and f is the source term. A detailed mathematical treatment of this class of problems is given in [BJK25, Chapter 2].

Besides well-posedness, maximum principles are fundamental qualitative properties of the continuous problem. They provide bounds for the solution in terms of the data and, under appropriate assumptions, imply positivity preservation. Classical maximum principles for second-order elliptic equations are presented in [Eva10, Section 6.4]. Corresponding maximum principles for weak solutions of convection-diffusion-reaction problems are discussed in [BJK25, Section 2.4]. These results motivate the requirement that numerical approximations should preserve analogous bounds.

After having performed a discretization, for example by a finite element method, this requirement leads to the concept of a discrete maximum principle. For linear discrete problems, DMPs are closely related to the algebraic structure of the system matrix. Matrices of non-negative type, monotone matrices, inverse-positive matrices, and M-matrices provide important tools for this analysis; see [BJK25, Chapter 3]. The connection between discrete maximum principles and monotone matrices was studied in [Cia70], while classical results on M-matrices can be found in [Ost37].

The standard Galerkin finite element method does not generally possess the sign structure needed for a discrete maximum principle. In convection-dominated regimes, positive off-diagonal entries may lead to non-physical undershoots and overshoots. Stabilized finite element methods are therefore required. Classical approaches include the streamline-upwind Petrov–Galerkin method [BH82] and SOLD methods. Although these methods are designed to improve stability and reduce oscillations near layers, SUPG solutions may still exhibit localized oscillations, and most classical SOLD methods do not satisfy a discrete maximum principle; see [BJK25, pp. 231–232 and 326–327].

A different approach is to construct monotone low-order discretizations. In 1977, Tabata introduced an upwind finite element method for convection-diffusion problems [Tab77]. Under suitable assumptions, such upwind discretizations can yield a monotone algebraic structure and may therefore satisfy local and global discrete maximum principles. However, the artificial diffusion required by a linear monotone scheme may smear steep boundary and interior layers. In the presence of such layers, the solution exhibits a multiscale structure that cannot generally be resolved uniformly on a given mesh. Since the location and strength of the layers depend on the solution itself, an effective stabilization should adapt to the local solution behavior. This motivates nonlinear stabilization techniques that combine the robustness of a monotone low-order method with a less diffusive correction.

Algebraic flux correction builds on flux-corrected transport methods [Zal79] and was developed for finite element discretizations, among others, by Kuzmin [Kuz07]. AFC methods start from a monotone low-order operator and restore part of the removed high-order contribution through limited

anti-diffusive fluxes. Analytical results for AFC discretizations are given in [BJK15, BJK16, BJK17], while a systematic finite element treatment can be found in [BJK25, Chapter 10].

The monotone upwind-type algebraically stabilized method considered in this thesis belongs to the class of nonlinear algebraic stabilization methods. It defines the nonlinear stabilization directly on the algebraic level and is designed to satisfy discrete maximum principles without the additional mesh-dependent sign condition required by the classical Kuzmin limiter. The MUAS method and its mathematical properties are developed in [Kno21, JK22]; see also [BJK25, Section 10.5].

AFC and MUAS discretizations lead to nonlinear algebraic systems because their limiter coefficients or stabilization matrices depend on the unknown discrete solution. Moreover, the limiting procedures contain non-smooth operations such as minima, maxima, and positive or negative parts. Fixed-point methods are therefore natural solvers for these systems. Basic nonlinear iteration schemes and their numerical behavior have been investigated in [JJ20, JJ19].

The numerical computations in this thesis use the fixed-point right-hand-side iteration introduced in Section 5.1. The low-order matrix is kept fixed, while the nonlinear flux-correction contribution is evaluated at the current iterate and moved to the right-hand side. Thus, only the right-hand side changes during the nonlinear iteration. This fixed-point strategy is studied in [JJ20]. The Tabata upwind solution is used as the initial iterate because it provides a robust and accurate approximation with favorable discrete maximum-principle properties; see the numerical comparisons in [BJK25, Section 8.7].

An important distinction has to be made between the qualitative properties of the converged nonlinear solution and those of the intermediate iterates. Even if the nonlinear discretization satisfies a DMP at convergence, an intermediate iterate or an approximation obtained by early termination may violate positivity or prescribed solution bounds. This observation motivates the iteration modifications studied in this thesis.

Four modifications of the fixed-point iteration are considered. Algorithm 1 clips those components of the assembled right-hand side for which the nonnegativity condition is violated. Algorithm 2 damps limiter contributions before the nonlinear right-hand side is assembled. Algorithms 3 and 4 use a component-wise projection onto prescribed lower and upper bounds, either after every nonlinear iteration or only after termination. The mathematical construction of these algorithms is presented in Chapter 5.

A particular contribution of this thesis is the construction of the positivity-budget-based limiter-damping strategy in Algorithm 2 and its comparison with right-hand-side clipping and projection-based corrections. The numerical investigation systematically distinguishes nonlinear convergence, preservation of prescribed bounds, and spatial accuracy, with special attention to solutions obtained by early termination of the nonlinear iteration.

The numerical study compares the four modifications with the unmodified fixed-point iteration, which is referred to as the standard approach. The spatial discretization is kept fixed so that the observed differences can be attributed to the nonlinear iteration strategy. The main questions concern early stopping, nonlinear convergence, preservation of positivity and prescribed bounds, and the influence of the corrections on spatial accuracy and layer resolution.

Three numerical experiments are considered. The first one is the smooth prescribed-solution problem from [BJK25, Examples 8.63 and 10.108]. The second one is the layer problem from [BJK25, Examples 2.25 and 10.109], where the computed profiles are compared with the reference cross-sections. The final experiment is the Hemker problem introduced in [Hem96], using the bounded obstacle configuration from [JKP23, Example 3.2]. It is a convection-dominated benchmark with a circular obstacle and thin layers downstream of the obstacle.

The thesis is organized as follows. Chapter 2 introduces the steady-state convection-diffusion-reaction problem, its weak formulation, and the continuous maximum principles. Chapter 3 discusses the conforming finite element discretization, algebraic conditions for local and global DMPs, and the

Tabata upwind method. Chapter 4 presents the AFC framework, the Kuzmin limiter, and the MUAS method. Chapter 5 introduces the fixed-point iteration and the four modifications. The numerical experiments are presented in Chapter 6. The main conclusions and directions for future research are summarized in the final chapter 7, while Appendix A contains the detailed numerical data.

2. Steady-State Convection-Diffusion-Reaction Equations and Maximum Principles

This chapter is devoted to the mathematical formulation of the model problem considered in this thesis – the scalar steady-state convection-diffusion-reaction boundary value problem.

In Section 2.1, we first introduce the classical strong formulation of the boundary value problem, along with the necessary assumptions on the problem data to ensure well-posedness. Since the physical quantities modelled by this equation (e.g., concentration or temperature) are typically non-negative, it is crucial that the mathematical model preserves this property. Therefore, in Section 2.2, we review the classical maximum principles for the strong solution, which provide the theoretical foundation for positivity preservation.

Subsequently, in Section 2.3, we derive the variational (weak) formulation of the problem, which serves as the starting point for the finite element discretization. We also discuss the existence and uniqueness of the weak solution under standard coercivity assumptions. Finally, Section 2.4 extends the concept of the maximum principles to the weak formulation, stating the conditions under which the weak solution satisfies the maximum principle.

2.1. The Steady-State Model Problem

Let $\Omega \subset \mathbb{R}^d$ ($d \in \{2, 3\}$) be a bounded domain with a Lipschitz continuous boundary $\Gamma = \partial\Omega$. We denote by $\mathbf{n}$ the outward pointing unit normal vector on Γ. We consider the scalar steady-state convection-diffusion-reaction equation: Find $u : \overline{\Omega} \to \mathbb{R}$ such that

$$\begin{aligned} -\varepsilon\Delta u + \mathbf{b}\cdot\nabla u + cu &= f && \text{in } \Omega, \\ u &= u_D && \text{on } \Gamma_D, \\ \varepsilon\nabla u\cdot\mathbf{n} &= g_N && \text{on } \Gamma_N. \end{aligned} \tag{2.1}$$

The boundary Γ is decomposed into two disjoint measurable subsets: the Dirichlet boundary Γ_D and the Neumann boundary Γ_N, such that $\Gamma = \overline{\Gamma}_D \cup \overline{\Gamma}_N$ and $\Gamma_D \cap \Gamma_N = \emptyset$. We assume that the $(d-1)$-dimensional measure of Γ_D is positive, i.e., $\text{meas}_{d-1}(\Gamma_D) > 0$, to ensure the uniqueness of the solution.

The problem data satisfy the following assumptions:

- The diffusion coefficient is a positive constant: $\varepsilon > 0$.
- The convection field $\mathbf{b} \in (W^{1,\infty}(\Omega))^d$ and the reaction coefficient $c \in L^\infty(\Omega)$.
- Coercivity Condition: To ensure the coercivity of the associated bilinear form, we assume that the reaction coefficient and the divergence of the convection field satisfy the following condition (Friedrichs condition):
$$c(\mathbf{x}) - \frac{1}{2}\nabla\cdot\mathbf{b}(\mathbf{x}) \geq 0 \quad \text{a.e. in } \Omega. \tag{2.2}$$
- The boundary data are sufficiently smooth: $u_D \in H^{1/2}(\Gamma_D)$ and $g_N \in H^{-1/2}(\Gamma_N)$.

2.2. Classical Maximum Principles for the Continuous Problem

The solution of the continuous problem is required to be physically consistent, which implies that non-negative input data should yield a non-negative exact solution. This requirement is mathematically governed by the maximum principles. In this section, we introduce the differential operator and establish the necessary continuous maximum principles.

Definition 2.1 (Differential Operator and Ellipticity)**.** *Let Ω be defined as above. We introduce the second-order linear differential operator $\mathcal{L}$, acting on a function $u \in C^2(\Omega)$, by*

$$\mathcal{L}u := -\varepsilon\Delta u + \mathbf{b} \cdot \nabla u + cu. \tag{2.3}$$

The principal part of $\mathcal{L}$ is given by $-\varepsilon\Delta$. Hence, the associated diffusion matrix is $A = \varepsilon I$. Since $\varepsilon > 0$, the operator $\mathcal{L}$ is uniformly elliptic. Indeed, for all $\xi \in \mathbb{R}^d$,

$$\xi^T A\xi = \sum_{i,j=1}^{d} \varepsilon\delta_{ij}\xi_i\xi_j = \varepsilon\|\xi\|_2^2 \geq \theta\|\xi\|_2^2, \tag{2.4}$$

with ellipticity constant $\theta = \varepsilon > 0$, where $\|\cdot\|_2$ denotes the Euclidean norm.

The weak and strong maximum principles recalled below are classical. Their proofs follow the standard perturbation arguments in [Eva10, Section 6.4.1–Section 6.4.2], adapted to the convection-diffusion-reaction operator used in this thesis.

Theorem 2.2 (Weak Maximum Principle)**.** *Let $\Omega \subset \mathbb{R}^d$ be a bounded domain. Assume that $u \in C^2(\Omega) \cap C(\bar{\Omega})$ satisfies the differential inequality:*

$$\mathcal{L}u \leq 0 \quad \text{in } \Omega. \tag{2.5}$$

Under the assumptions of $\varepsilon > 0$ and $c(\mathbf{x}) \geq 0$, the maximum of u satisfies:

$$\max_{\bar{\Omega}} u \leq \max\left\{0, \max_{\partial\Omega} u\right\}. \tag{2.6}$$

Proof. The proof is divided into two steps: first for the strict inequality by contradiction, and second for the general case.

First step, assume first that $\mathcal{L}u < 0$ in Ω. Suppose that u attains a positive maximum at an interior point $\mathbf{x}_0 \in \Omega$:

$$u(\mathbf{x}_0) = \max_{\bar{\Omega}} u > \max\left\{0, \max_{\partial\Omega} u\right\} \geq 0. \tag{2.7}$$

At the interior maximum $\mathbf{x}_0$, the necessary conditions imply:

$$\nabla u(\mathbf{x}_0) = \mathbf{0} \quad \text{and} \quad \Delta u(\mathbf{x}_0) \leq 0. \tag{2.8}$$

Evaluating the operator $\mathcal{L}$ at $\mathbf{x}_0$ yields:

$$\begin{aligned} \mathcal{L}u(\mathbf{x}_0) &= -\varepsilon\Delta u(\mathbf{x}_0) + \mathbf{b}(\mathbf{x}_0) \cdot \nabla u(\mathbf{x}_0) + c(\mathbf{x}_0)u(\mathbf{x}_0) \\ &= \underbrace{-\varepsilon}_{<0}\underbrace{\Delta u(\mathbf{x}_0)}_{\leq 0} + \mathbf{b} \cdot \mathbf{0} + \underbrace{c(\mathbf{x}_0)}_{\geq 0}\underbrace{u(\mathbf{x}_0)}_{>0} \\ &\geq 0. \end{aligned} \tag{2.9}$$

This contradicts the assumption $\mathcal{L}u(\mathbf{x}_0) < 0$. Thus, u cannot have a positive interior maximum.

Second step, assume now $\mathcal{L}u \leq 0$. Let $\mathbf{x} = (x_1, \ldots, x_d)^T \in \Omega$ be the coordinate vector. Define the auxiliary function v_{n_1} for $n_1 > 0$ as:

$$v_{n_1}(\mathbf{x}) := u(\mathbf{x}) + n_1 e^{n_2 x_1}, \tag{2.10}$$

where x_1 is the first component of $\mathbf{x}$ and $n_2 > 0$ is a constant. Applying $\mathcal{L}$ to the perturbation term:

$$\mathcal{L}(e^{n_2 x_1}) = e^{n_2 x_1}(-\varepsilon n_2^2 + b_1 n_2 + c). \tag{2.11}$$

where b_1 is the first component of $\mathbf{b}$. Since $b_1, c \in L^\infty(\Omega)$ and $\varepsilon > 0$, one can choose n_2 sufficiently large such that $-\varepsilon n_2^2$ dominates, ensuring $(-\varepsilon n_2^2 + b_1 n_2 + c) < 0$. Then we get $n_1 \mathcal{L}(e^{n_2 x_1}) < 0$. Consequently:

$$\mathcal{L}v_{n_1} = \mathcal{L}u + n_1 \mathcal{L}(e^{n_2 x_1}) < 0. \tag{2.12}$$

According to the conclusion of the first step($\mathcal{L}(u) < 0$), v_{n_1} satisfies the maximum principle:

$$\max_{\bar{\Omega}} u \leq \max_{\bar{\Omega}} v_{n_1} \leq \max\left\{0, \max_{\partial\Omega} v_{n_1}\right\}. \tag{2.13}$$

Taking the limit $n_1 \to 0$ completes the proof. □

While the Weak Maximum Principle provides global bounds, the Strong Maximum Principle characterizes the behavior of the solution in the interior of the domain. This is intimately related to the behavior of the normal derivative at the boundary, governed by Hopf's Lemma.

Theorem 2.3 (Strong Maximum Principle)**.** *Let $\Omega \subset \mathbb{R}^d$ be a bounded, open, and connected domain. Assume $u \in C^2(\Omega) \cap C(\bar{\Omega})$ satisfies the differential inequality:*

$$\mathcal{L}u := -\varepsilon \Delta u + \mathbf{b} \cdot \nabla u + cu \leq 0 \quad \text{in } \Omega, \tag{2.14}$$

with coefficients $\varepsilon > 0$ and $c(\mathbf{x}) \geq 0$. If u attains its maximum at an interior point $\mathbf{x}_0 \in \Omega$, that is,

$$u(\mathbf{x}_0) = \max_{\bar{\Omega}} u \geq 0, \tag{2.15}$$

then u must be constant throughout $\bar{\Omega}$.

Proof. The proof proceeds by contradiction: assuming u is not constant, one constructs an interior ball within the domain strictly touching the set of maxima to invoke Lemma 2.4, which yields a strictly positive normal derivative that directly contradicts the necessary condition $\nabla u = 0$ at an interior maximum. For the complete mathematical details, we refer to [Eva10, Theorem 3, Section 6.4.3]. □

The Strong Maximum Principle implies that non-constant solutions must attain their positive maxima strictly on the boundary. The behavior at such boundary points is described by the following lemma, which provides the necessary tool to handle Neumann boundary conditions.

Lemma 2.4 (Hopf's Lemma)**.** *Let the assumptions of Theorem 2.3 hold, and additionally assume that $u \in C^1(\bar{\Omega})$. Suppose there exists a boundary point $\mathbf{x}_0 \in \partial\Omega$ such that $u(\mathbf{x}_0) \geq 0$ and $u(\mathbf{x}) < u(\mathbf{x}_0)$ for all $\mathbf{x} \in \Omega$. If Ω satisfies the interior ball condition at $\mathbf{x}_0$[1], then the outward normal derivative at $\mathbf{x}_0$ is strictly positive:*

$$\frac{\partial u}{\partial \mathbf{n}}(\mathbf{x}_0) > 0. \tag{2.16}$$

Proof. The detailed proof based on the construction of an auxiliary barrier function on the interior ball is classical. We refer to [Eva10, Hopf's Lemma, Section 6.4.2]. □

[1] This means there exists an open ball $B \subset \Omega$ with $\mathbf{x}_0 \in \partial B$.

The following result is a direct consequence of the above maximum principles.

Corollary 2.5 (Positivity Preservation)**.** *Let the assumptions of Theorem 2.3 and Lemma 2.4 hold. If the source term is non-negative ($f \geq 0$) and the boundary data satisfy $u_D \geq 0$ on Γ_D and $g_N \geq 0$ on Γ_N, then the solution u is non-negative everywhere in the domain:*

$$u(\mathbf{x}) \geq 0 \quad \forall \mathbf{x} \in \bar{\Omega}. \tag{2.17}$$

Proof. We proceed by contradiction. Suppose that u attains a strictly negative minimum in $\bar{\Omega}$. Let $v = -u$, then v satisfies $\mathcal{L}v = -f \leq 0$ in Ω, and v attains a strictly positive maximum $M > 0$ in $\bar{\Omega}$.

According to Theorem 2.3, if v attains its maximum in the interior, it must be a constant function that then $v(\mathbf{x}) = M$ for all $\mathbf{x} \in \bar{\Omega}$. If v is a constant function, then for any $\mathbf{x} \in \Gamma_D$, $v(\mathbf{x}) = -u_D(\mathbf{x}) \leq 0$ (since $u_D \geq 0$), which contradicts $M > 0$. Thus, the maximum must be attained on the boundary $\partial\Omega = \Gamma_D \cup \Gamma_N$.

If the maximum is attained at $\mathbf{x}_0 \in \Gamma_D$, then $v(\mathbf{x}_0) = -u_D(\mathbf{x}_0) \leq 0$, which contradicts $v(\mathbf{x}_0) = M > 0$. Therefore, the maximum must be attained at some $\mathbf{x}_0 \in \Gamma_N$. According to Lemma 2.4, the outward normal derivative at $\mathbf{x}_0$ satisfies:

$$\frac{\partial v}{\partial \mathbf{n}}(\mathbf{x}_0) > 0. \tag{2.18}$$

On the other hand, substituting $v = -u$ into the Neumann boundary condition $\varepsilon \nabla u \cdot \mathbf{n} = g_N$ yields:

$$\frac{\partial v}{\partial \mathbf{n}}(\mathbf{x}_0) = -\frac{\partial u}{\partial \mathbf{n}}(\mathbf{x}_0) = -\frac{g_N(\mathbf{x}_0)}{\varepsilon}. \tag{2.19}$$

Since $\varepsilon > 0$ and $g_N \geq 0$ on Γ_N, we have $\frac{\partial v}{\partial \mathbf{n}}(\mathbf{x}_0) \leq 0$. This contradicts the strictly positive result from Lemma 2.4. This completes the proof that u cannot have a negative minimum, hence $u(\mathbf{x}) \geq 0$ for all $\mathbf{x} \in \bar{\Omega}$. □

Remark 2.6 (Minimum Principles and Positivity)**.** *Analogous to the maximum principles established in Theorem 2.2 and Theorem 2.3, the corresponding weak and strong minimum principles are obtained by applying the maximum principles to $-u$.*

In the context of the physical problem, these principles imply the following qualitative properties:

1. ***Positivity preservation:*** *If the source term f and the boundary data are non-negative in the sense stated in Corollary 2.5, then the solution is non-negative in $\bar{\Omega}$, namely*

$$u(\mathbf{x}) \geq 0 \quad \forall \mathbf{x} \in \bar{\Omega}. \tag{2.20}$$

2. ***Strict positivity:*** *If the data are non-trivial and the solution is not identically zero, then the strong minimum principle implies that the solution cannot attain the value 0 at an interior minimum. Hence,*

$$u(\mathbf{x}) > 0 \quad \forall \mathbf{x} \in \Omega. \tag{2.21}$$

2.3. Derivation of the Weak Formulation

To derive the finite element formulation, we first define the appropriate function spaces. Let $H^1(\Omega)$ denote the standard Sobolev space equipped with the norm $\|v\|_{H^1} = (\|v\|_{L^2}^2 + \|\nabla v\|_{L^2}^2)^{1/2}$. We introduce the trial space V (incorporating the Dirichlet data) and the test space V_0 (with homogeneous boundary conditions) as:

$$V := \{v \in H^1(\Omega) : v|_{\Gamma_D} = u_D\}, \tag{2.22}$$

$$V_0 := \{v \in H^1(\Omega) : v|_{\Gamma_D} = 0\}. \tag{2.23}$$

We multiply the strong form equation (2.1) by a test function $v \in V_0$ and integrate over the domain Ω. The transformation to the weak form proceeds as follows:

$$\begin{aligned}
\int_\Omega f v \, dx &= \int_\Omega (-\varepsilon \Delta u + \mathbf{b} \cdot \nabla u + cu) v \, dx \\
&= \int_\Omega \varepsilon \nabla u \cdot \nabla v \, dx - \int_{\partial\Omega} \varepsilon (\nabla u \cdot \mathbf{n}) v \, ds + \int_\Omega (\mathbf{b} \cdot \nabla u + cu) v \, dx \quad \text{(First Green's formula)} \\
&= \int_\Omega (\varepsilon \nabla u \cdot \nabla v + \mathbf{b} \cdot \nabla u v + cuv) \, dx - \int_{\Gamma_N} \varepsilon (\nabla u \cdot \mathbf{n}) v \, ds \qquad \text{(Since } v|_{\Gamma_D} = 0\text{)} \\
&= \int_\Omega (\varepsilon \nabla u \cdot \nabla v + \mathbf{b} \cdot \nabla u v + cuv) \, dx - \int_{\Gamma_N} g_N v \, ds. \qquad \text{(Neumann condition)}
\end{aligned}$$

Rearranging the terms to separate the bilinear form (terms involving u) and the linear functional (data terms), we arrive at the formal definition.

Definition 2.7 (Weak Formulation)**.** *Find a solution $u \in V$ such that*

$$a(u, v) = l(v) \qquad \forall v \in V_0, \tag{2.24}$$

where the bilinear form $a(\cdot, \cdot)$ and the linear functional $l(\cdot)$ are defined by

$$a(u, v) := (\varepsilon \nabla u, \nabla v)_{L^2(\Omega)} + (\mathbf{b} \cdot \nabla u, v)_{L^2(\Omega)} + (cu, v)_{L^2(\Omega)}, \tag{2.25}$$

$$l(v) := \langle f, v \rangle_{V_0', V_0} + \langle g_N, v \rangle_{H^{-1/2}(\Gamma_N), H^{1/2}(\Gamma_N)}. \tag{2.26}$$

Here, $(\cdot, \cdot)_{L^2(\Omega)}$ denotes the standard $L^2(\Omega)$ inner product, and $\langle \cdot, \cdot \rangle_{X', X}$ denotes the duality pairing between a Banach space X and its dual space X'. In particular:

- *$\langle f, v \rangle_{V_0', V_0}$ represents the action of the source term $f \in V_0'$ on the test function $v \in V_0$;*
- *$\langle g_N, v \rangle_{H^{-1/2}(\Gamma_N), H^{1/2}(\Gamma_N)}$ represents the action of the Neumann datum $g_N \in H^{-1/2}(\Gamma_N)$ on the trace of v on Γ_N.*

Theorem 2.8 (Existence and Uniqueness under Generalized Stability Conditions)**.** *Let Ω be a bounded domain with Lipschitz boundary $\partial\Omega = \Gamma_D \cup \Gamma_N$. Assume that the Dirichlet boundary has a strictly positive measure, $|\Gamma_D| > 0$. Let the data satisfy $f \in V_0'$, $\mathbf{b} \in (W^{1,\infty}(\Omega))^d$, and $c \in L^\infty(\Omega)$. Assume that the problem data satisfy the following two stability conditions:*

1. ***Friedrichs Condition*** *in the domain:*

$$c(\mathbf{x}) - \frac{1}{2} \nabla \cdot \mathbf{b}(\mathbf{x}) \geq 0 \quad \text{a.e. in } \Omega. \tag{2.27}$$

2. ***Non-inflow Condition*** *on the Neumann boundary:*

$$\mathbf{b}(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) \geq 0 \quad \text{a.e. on } \Gamma_N. \tag{2.28}$$

Then the bilinear form $a(\cdot, \cdot)$ is continuous and coercive on V_0. Consequently, the weak formulation in Definition 2.7 admits a unique solution $u \in V$.

Proof. The proof relies on the Lax–Milgram theorem [Eva10, Theorem 1, Section 6.2.1]. Since the trial space V is affine, we first introduce a fixed extension $u_{D,\text{ext}} \in H^1(\Omega)$ of the Dirichlet datum satisfying

$$u_{D,\text{ext}}|_{\Gamma_D} = u_D. \tag{2.29}$$

For $u \in V$, define

$$w := u - u_{D,\text{ext}}. \tag{2.30}$$

Then $w \in V_0$. Hence, the weak formulation is equivalent to the problem: find $w \in V_0$ such that

$$a(w, v) = l(v) - a(u_{D,\text{ext}}, v) \qquad \forall v \in V_0. \tag{2.31}$$

The right-hand side defines a bounded linear functional on V_0. It therefore remains to verify the continuity and coercivity of the bilinear form $a(\cdot, \cdot)$ on V_0.

Continuity: For any $w, v \in V_0$, we estimate the absolute value of the bilinear form. Using standard integral inequalities, we obtain the following chain of estimates:

$$\begin{aligned}
|a(w, v)| &= |(\varepsilon \nabla w, \nabla v)_{L^2} + (\mathbf{b} \cdot \nabla w, v)_{L^2} + (cw, v)_{L^2}| \\
&\leq \varepsilon |(\nabla w, \nabla v)_{L^2}| + |(\mathbf{b} \cdot \nabla w, v)_{L^2}| + |(cw, v)_{L^2}| && \text{(Triangle inequality)} \\
&\leq \varepsilon \|\nabla w\|_{L^2} \|\nabla v\|_{L^2} + \|\mathbf{b}\|_{L^\infty} \|\nabla w\|_{L^2} \|v\|_{L^2} + \|c\|_{L^\infty} \|w\|_{L^2} \|v\|_{L^2} && \text{(Hölder and Cauchy–Schwarz)} \\
&\leq (\varepsilon + \|\mathbf{b}\|_{L^\infty} + \|c\|_{L^\infty}) \, \|w\|_{H^1} \|v\|_{H^1}. && \text{(Since } \|\cdot\|_{L^2} \leq \|\cdot\|_{H^1}\text{)}
\end{aligned}$$

Thus, continuity holds with $C_1 := \varepsilon + \|\mathbf{b}\|_{L^\infty} + \|c\|_{L^\infty}$.

Coercivity: For any $v \in V_0$, we analyze the convection term using integration by parts and the identity $v \nabla v = \frac{1}{2} \nabla (v^2)$:

$$\begin{aligned}
(\mathbf{b} \cdot \nabla v, v)_{L^2} &= \int_\Omega \mathbf{b} \cdot \nabla \left(\frac{1}{2} v^2 \right) d\mathbf{x} \\
&= \frac{1}{2} \int_{\partial\Omega} v^2 (\mathbf{b} \cdot \mathbf{n}) \, ds - \frac{1}{2} \int_\Omega v^2 (\nabla \cdot \mathbf{b}) \, d\mathbf{x}.
\end{aligned} \tag{2.32}$$

Since $v|_{\Gamma_D} = 0$, the boundary integral is restricted to Γ_N. Substituting this back into the bilinear form definition yields:

$$\begin{aligned}
a(v, v) &= \varepsilon \|\nabla v\|_{L^2}^2 + (\mathbf{b} \cdot \nabla v, v)_{L^2} + (cv, v)_{L^2} \\
&= \varepsilon \|\nabla v\|_{L^2}^2 + \frac{1}{2} \int_{\Gamma_N} v^2 (\mathbf{b} \cdot \mathbf{n}) \, ds - \frac{1}{2} \int_\Omega v^2 (\nabla \cdot \mathbf{b}) \, d\mathbf{x} + \int_\Omega c v^2 \, d\mathbf{x} \\
&= \varepsilon \|\nabla v\|_{L^2}^2 + \underbrace{\frac{1}{2} \int_{\Gamma_N} v^2 (\mathbf{b} \cdot \mathbf{n}) \, ds}_{\geq 0 \text{ (by Assumption 2)}} + \int_\Omega \underbrace{\left(c - \frac{1}{2} \nabla \cdot \mathbf{b} \right)}_{\geq 0 \text{ (by Assumption 1)}} v^2 \, d\mathbf{x}.
\end{aligned} \tag{2.33}$$

Since both the boundary term and the generalized reaction term are non-negative, we can drop them to obtain the lower bound:

$$a(v, v) \geq \varepsilon \|\nabla v\|_{L^2}^2. \tag{2.34}$$

Because $|\Gamma_D| > 0$, we can apply the Poincaré–Friedrichs inequality ($\|v\|_{L^2} \leq C_P \|\nabla v\|_{L^2}$ for $v \in V_0$) to establish coercivity in the full H^1-norm:

$$a(v, v) \geq \alpha \|v\|_{H^1}^2, \qquad \alpha = \frac{\varepsilon}{1 + C_P^2}. \tag{2.35}$$

With continuity and coercivity established, the Lax–Milgram theorem yields a unique solution $w \in V_0$. Defining

$$u := w + u_{D,\text{ext}}, \tag{2.36}$$

we obtain a unique function $u \in V$ satisfying

$$a(u, v) = l(v) \qquad \forall v \in V_0. \tag{2.37}$$

This completes the proof of existence and uniqueness of the weak solution. □

Remark 2.9. *It is important to note that in the widely used case of an incompressible flow field ($\nabla \cdot \mathbf{b} = 0$), the Friedrichs condition (2.27) in Theorem 2.8 simplifies directly to the standard non-negativity constraint on the reaction coefficient:*

$$c(\mathbf{x}) \geq 0 \quad \text{a.e. in } \Omega. \tag{2.38}$$

2.4. Maximum Principles for the Weak Solution

Having established the existence and uniqueness of the weak solution, we now turn to its qualitative properties. In this section, we extend the classical maximum principles discussed in Section 2.2 to the variational setting. We show that the weak solution $u \in V$ satisfies the same global bounds as the classical solution, ensuring physical consistency such as positivity preservation. These results are derived by testing the weak formulation in Definition 2.7 with appropriate test functions (e.g., truncated functions $v = u^-$ or $v = (u - k)^+$) in the Sobolev space $H^1(\Omega)$.

Now, we introduce the concept of the *essential supremum* and the properties of truncated functions.

Definition 2.10 (Essential Supremum and Infimum)**.** *Let u be a measurable function on Ω. The essential supremum and infimum are defined as:*

$$\operatorname*{ess\,sup}_{\Omega} u := \inf\{C \in \mathbb{R} : u(x) \leq C \text{ a.e. in } \Omega\}, \tag{2.39}$$

$$\operatorname*{ess\,inf}_{\Omega} u := \sup\{C \in \mathbb{R} : u(x) \geq C \text{ a.e. in } \Omega\}. \tag{2.40}$$

To construct appropriate test functions for the maximum principle, we use the positive and negative part functions:

$$u^+(\mathbf{x}) := \max(u(\mathbf{x}), 0), \quad u^-(\mathbf{x}) := \min(u(\mathbf{x}), 0). \tag{2.41}$$

We now state and prove the maximum principle for weak solutions in the test space V_0.

Theorem 2.11 (Maximum Principle for Weak Solutions)**.** *[BJK25, Theorem 2.21, Section 2.4] Let the reaction coefficient satisfy $c(\mathbf{x}) \geq 0$ almost everywhere in Ω. Then, for any weak solution $u \in H^1(\Omega)$, the following implications hold:*

$$a(u, v) \leq 0 \quad \forall v \in V_0, v \geq 0 \quad \Longrightarrow \quad \operatorname*{ess\,sup}_{\Omega} u \leq \operatorname*{ess\,sup}_{\Gamma_D} u^+, \tag{2.42}$$

$$a(u, v) \geq 0 \quad \forall v \in V_0, v \geq 0 \quad \Longrightarrow \quad \operatorname*{ess\,inf}_{\Omega} u \geq \operatorname*{ess\,inf}_{\Gamma_D} u^-. \tag{2.43}$$

In particular, for the homogeneous equation ($a(u, v) = 0$), we have:

$$\operatorname*{ess\,sup}_{\Omega} |u| = \operatorname*{ess\,sup}_{\Gamma_D} |u|. \tag{2.44}$$

Moreover, if $c = 0$ a.e. in Ω, the bounds simplify to:

$$a(u, v) \leq 0 \quad \forall v \in V_0, v \geq 0 \quad \Longrightarrow \quad \operatorname*{ess\,sup}_{\Omega} u = \operatorname*{ess\,sup}_{\Gamma_D} u, \tag{2.45}$$

$$a(u, v) \geq 0 \quad \forall v \in V_0, v \geq 0 \quad \Longrightarrow \quad \operatorname*{ess\,inf}_{\Omega} u = \operatorname*{ess\,inf}_{\Gamma_D} u. \tag{2.46}$$

Proof. The proof follows the argument in [BJK25, Theorem 2.21, Section 2.4]. We give the proof for the upper bound (2.42). The proof relies on a contradiction argument involving the measure of the set where the solution exceeds the boundary bound.

Consider $u \in H^1(\Omega)$. We define the boundary bound s_D:

$$s_D = \begin{cases} \operatorname{ess\,sup}_{\Gamma_D} u & \text{if } c = 0 \text{ a.e. in } \Omega, \\ \operatorname{ess\,sup}_{\Gamma_D} u^+ & \text{otherwise.} \end{cases} \tag{2.47}$$

Let $s_\Omega := \operatorname{ess\,sup}_\Omega u$. The proof aims to show $s_\Omega \leq s_D$. We proceed by contradiction: assume that $s_\Omega > s_D$.

Define the difference δ and a sequence of levels k_n:

$$\delta = \frac{s_\Omega - s_D}{2}, \quad k_n = s_\Omega - \frac{\delta}{n} \quad \text{for } n \in \mathbb{N}. \tag{2.48}$$

Then we have the strict ordering $s_D < k_n < s_\Omega$ for any $n \in \mathbb{N}$, and $k_n \to s_\Omega$ as $n \to \infty$.

We define the test function sequence:

$$v_n = (u - k_n)^+. \tag{2.49}$$

Since $u \in H^1(\Omega)$, by Stampacchia's Lemma [KS80], $v_n \in H^1(\Omega)$. Crucially, on the boundary Γ_D, $u \leq s_D < k_n$ a.e., which implies $u - k_n < 0$, so $v_n = 0$ on Γ_D. Thus, $v_n \in V_0$. Also, $v_n \geq 0$.

The gradient of the test function is:

$$\nabla v_n = \begin{cases} \nabla u & \text{if } u > k_n, \\ \mathbf{0} & \text{if } u \leq k_n. \end{cases} \tag{2.50}$$

Furthermore, $\|\nabla v_n\|_{L^2(\Omega)} \neq 0$, otherwise v_n would be constant (zero, due to boundary conditions), implying $u \leq k_n$ a.e., which contradicts $s_\Omega > k_n$.

Using the hypothesis $a(u, v_n) \leq 0$ and the definition $a(u, v) = (\varepsilon \nabla u, \nabla v) + (\mathbf{b} \cdot \nabla u + cu, v)$, we have:

$$(\varepsilon \nabla u, \nabla v_n) \leq -(\mathbf{b} \cdot \nabla u, v_n) - (cu, v_n). \tag{2.51}$$

Noting that $\nabla v_n = \nabla u$ on the support of v_n, we also have $cuv_n \geq 0$. Indeed, this is immediate if $c = 0$ almost everywhere in Ω. Otherwise, $s_D = \operatorname{ess\,sup}_{\Gamma_D} u^+ \geq 0$, and $v_n > 0$ implies $u > k_n > s_D \geq 0$. Hence, in both cases, $(cu, v_n) \geq 0$, and the non-negative reaction term may be omitted from the estimate. Thus,

$$(\varepsilon \nabla v_n, \nabla v_n) \leq -(\mathbf{b} \cdot \nabla v_n, v_n) - (cu, v_n) \leq -(\mathbf{b} \cdot \nabla v_n, v_n). \tag{2.52}$$

Let us introduce the sets:

$$Q_n = \{\mathbf{x} \in \Omega : v_n(\mathbf{x}) \neq 0\} = \{\mathbf{x} \in \Omega : u(\mathbf{x}) > k_n\}, \tag{2.53}$$

$$G_n = \{\mathbf{x} \in Q_n : \nabla v_n(\mathbf{x}) \neq \mathbf{0}\} = \{\mathbf{x} \in Q_n : \nabla u(\mathbf{x}) \neq \mathbf{0}\}. \tag{2.54}$$

Observe that for $n > m$, $k_n > k_m$, so $Q_n \subset Q_m$ and consequently $G_n \subset G_m$.

The convection term can be estimated on the set G_n:

$$|(\mathbf{b} \cdot \nabla v_n, v_n)| = |(\mathbf{b} \cdot \nabla v_n, v_n)_{G_n}| \leq \|\mathbf{b}\|_{L^\infty(\Omega)} \|\nabla v_n\|_{L^2(\Omega)} \|v_n\|_{L^2(G_n)}. \tag{2.55}$$

Using the identity $\varepsilon \|\nabla v_n\|^2_{L^2(\Omega)} = (\varepsilon \nabla v_n, \nabla v_n)$ and combining with (2.52), we obtain:

$$\varepsilon \|\nabla v_n\|^2_{L^2(\Omega)} \leq \|\mathbf{b}\|_{L^\infty(\Omega)} \|\nabla v_n\|_{L^2(\Omega)} \|v_n\|_{L^2(G_n)}. \tag{2.56}$$

Dividing by $\|\nabla v_n\|_{L^2(\Omega)}$ (which is non-zero) yields:

$$\|\nabla v_n\|_{L^2(\Omega)} \leq \varepsilon^{-1}\|\mathbf{b}\|_{L^\infty(\Omega)}\|v_n\|_{L^2(G_n)}. \tag{2.57}$$

If $\|\mathbf{b}\|_{L^\infty} = 0$, we have an immediate contradiction $\nabla v_n = 0$. So assume $\|\mathbf{b}\|_{L^\infty} \neq 0$. We use the Hölder inequality, Sobolev inequality (for dimension $d \geq 2$, with Sobolev conjugate $q > 2$) and the Friedrichs inequality to estimate $\|v_n\|_{L^2(G_n)}$:

$$\|v_n\|_{L^2(G_n)} \leq |G_n|^{\frac{1}{2}-\frac{1}{q}}\|v_n\|_{L^q(\Omega)} \leq C_{\text{imb}}C_{\text{Fr}}|G_n|^{\frac{1}{2}-\frac{1}{q}}\|\nabla v_n\|_{L^2(\Omega)}, \tag{2.58}$$

where $|G_n|$ denotes the measure of G_n. Substituting this back into (2.57) and canceling $\|\nabla v_n\|_{L^2(\Omega)}$:

$$1 \leq \varepsilon^{-1}\|\mathbf{b}\|_{L^\infty(\Omega)}C_{\text{imb}}C_{\text{Fr}}|G_n|^{\frac{1}{2}-\frac{1}{q}}. \tag{2.59}$$

Rearranging for $|G_n|$, we find a strictly positive lower bound C_0:

$$|G_n| \geq C_0 := \left(\varepsilon^{-1}\|\mathbf{b}\|_{L^\infty(\Omega)}C_{\text{imb}}C_{\text{Fr}}\right)^{-\frac{2q}{q-2}} > 0, \quad \forall n \in \mathbb{N}. \tag{2.60}$$

Let $G = \bigcap_{n=1}^{\infty} G_n$. Since G_n is a decreasing sequence of sets with measure bounded from below by C_0, the limit set has measure $|G| \geq C_0 > 0$.

For any point $\mathbf{x} \in G$, we have $\mathbf{x} \in Q_n$ for all n, meaning:

$$u(\mathbf{x}) > k_n \quad \forall n \in \mathbb{N}. \tag{2.61}$$

Taking the limit as $n \to \infty$ (where $k_n \to s_\Omega$), we get $u(\mathbf{x}) \geq s_\Omega$. Since s_Ω is the essential supremum, this implies $u = s_\Omega$ almost everywhere in G.

If u is constant (s_Ω) in G, then its gradient must be zero:

$$\nabla u = \mathbf{0} \quad \text{a.e. in } G. \tag{2.62}$$

However, by definition, $G \subset G_1 = \{\mathbf{x} : \nabla u(\mathbf{x}) \neq \mathbf{0}\}$. This implies $\nabla u \neq \mathbf{0}$ in G.

This is a contradiction. Therefore, the assumption $s_\Omega > s_D$ is false, and the maximum principle holds. □

3. Linear Finite Element Methods and Discrete Maximum Principles

The transition from a continuous mathematical model to a numerical simulation requires a discretization process that is not only accurate in terms of error norms but also physically reliable. Building upon the continuous maximum principles established in Chapter 2, this chapter focuses on the finite element discretization of the steady-state convection-diffusion-reaction equation.

This chapter is structured to systematically investigate the interplay between finite element discretization and algebraic stability constraints. We begin in Section 3.1 by establishing the numerical framework, detailing the admissible triangulations and finite element spaces, and subsequently deriving the linear algebraic system from the standard Galerkin variational formulation.

Building on this foundation, Section 3.2 provides a rigorous analysis of linear discrete problems. We establish the algebraic conditions necessary for the discrete solution to preserve the qualitative properties of the continuous problem. Specifically, we examine the properties of matrices of nonnegative type required for the satisfaction of local DMPs, as well as the theory of inverse-positive matrices (including M-matrices), which governs global DMPs.

However, the standard linear approach encounters significant limitations in specific regimes. Section 3.2.3 demonstrates the deficiency of the standard Galerkin method when applied to convection-dominated problems, analyzing why the resulting stiffness matrix generally fails to maintain the required M-matrix property. To address this within a linear framework, Section 3.3 introduces the upwind finite element method of Tabata [Tab77], which restores the non-positive off-diagonal sign pattern through geometric upwinding and mass lumping.

3.1. Galerkin Discretization

In this section, we formulate the discrete counterpart to the continuous variational problem derived in Chapter 2. The essence of the Standard Galerkin Finite Element Method is the approximation of the infinite-dimensional space V by a finite-dimensional subspace $V_h \subset V$. We seek a discrete solution $u_h \in V_h$ that satisfies the weak formulation for all test functions within the same subspace. This approach effectively projects the continuous problem onto V_h, transforming the partial differential equation into a system of linear algebraic equations.

The construction of this discrete system proceeds in two steps. First, in Section 3.1.1, we define the triangulation of the domain and the corresponding finite element basis functions. Subsequently, in Section 3.1.2, we derive the explicit algebraic coefficients of the stiffness matrix and load vector, establishing the linear system $A\underline{u} = \underline{l}$ that serves as the baseline for our analysis.

3.1.1. Triangulations and Finite Element Spaces

Let $\Omega \subset \mathbb{R}^d$ $(d = 2, 3)$ be a bounded domain with Lipschitz boundary $\partial\Omega$. We introduce an admissible triangulation $\mathcal{T}_h$ of Ω, consisting of a finite collection of closed simplices with mutually disjoint interiors (triangles in two dimensions and tetrahedra in three dimensions), denoted by K. The triangulation is assumed to satisfy the following geometric conformity conditions:

1. **Partition of the Domain:** The closure of the domain is the union of the closures of all elements in $\mathcal{T}_h$:
$$\bar{\Omega} = \bigcup_{K \in \mathcal{T}_h} \bar{K}. \tag{3.1}$$
2. **Admissibility (No Hanging Nodes):** For any two distinct elements $K_1, K_2 \in \mathcal{T}_h$, the intersection $\bar{K}_1 \cap \bar{K}_2$ is either:
   - the empty set $\emptyset$,
   - a single common vertex,
   - a complete common edge (if $d \geq 2$), or
   - a complete common face (if $d = 3$).

For each element $K \in \mathcal{T}_h$, we denote its diameter by $h_K = \operatorname{diam}(K)$ and the diameter of its largest inscribed ball by ρ_K. The global mesh parameter h is defined as the maximum element diameter:

$$h = \max_{K \in \mathcal{T}_h} h_K. \tag{3.2}$$

Furthermore, we assume that the family of triangulations $\{\mathcal{T}_h\}_{h>0}$ is *shape-regular*. That is, there exists a constant $\sigma > 0$, independent of h and K, such that the aspect ratio condition holds:

$$\frac{h_K}{\rho_K} \leq \sigma \quad \forall K \in \mathcal{T}_h. \tag{3.3}$$

This regularity condition ensures that the elements do not degenerate as the mesh is refined ($h \to 0$), which is a necessary condition for the optimal convergence rates of the finite element approximation.

Following the standard Galerkin discretization approach, we replace the infinite-dimensional spaces $H^1(\Omega)$ and V with finite-dimensional subspaces. Based on the triangulation $\mathcal{T}_h$, we first construct the space W_h of continuous, piecewise linear functions:

Definition 3.1 (Continuous Piecewise Linear Finite Element Space)**.** *Based on the triangulation $\mathcal{T}_h$, we construct the space W_h of continuous, piecewise linear functions as:*

$$W_h := \{v \in C^0(\overline{\Omega}) : v|_K \in \mathbb{P}_1(K), \quad \forall K \in \mathcal{T}_h\}, \tag{3.4}$$

where $\mathbb{P}_1(K)$ denotes the space of polynomials of degree at most 1 defined on the simplex K.

Let n denote the total number of vertices in the mesh, and let $\{\phi_1, \ldots, \phi_n\}$ be the standard nodal basis of W_h, defined by the Lagrange property:

$$\phi_i(\mathbf{x}_j) = \delta_{ij}, \quad \forall i, j = 1, \ldots, n. \tag{3.5}$$

Thus, we can explicitly express the space as the span of these basis functions:

$$W_h = \operatorname{span}\{\phi_1, \ldots, \phi_n\}. \tag{3.6}$$

To incorporate the Dirichlet boundary conditions, we use the full finite element space W_h together with the homogeneous subspace V_h for testing. We assume the nodes are ordered such that indices $1, \ldots, m$ correspond to the active degrees of freedom, and indices $m+1, \ldots, n$ correspond to the Dirichlet boundary nodes.

The test space $V_h \subset W_h$, which consists of functions vanishing on the Dirichlet boundary Γ_D, is defined as:

$$V_h = \operatorname{span}\{\phi_1, \ldots, \phi_m\}. \tag{3.7}$$

Accordingly, the approximate solution $u_h \in W_h$ is represented as:

$$u_h(\mathbf{x}) = \underbrace{\sum_{i=1}^{m} u_i \phi_i(\mathbf{x})}_{\text{Unknowns (active)}} + \underbrace{\sum_{i=m+1}^{n} u_i^b \phi_i(\mathbf{x})}_{\text{Prescribed (Dirichlet boundary)}}, \tag{3.8}$$

where u_i are the unknown coefficients to be determined, and u_i^b are fixed values given by the Dirichlet boundary data u_D. We define the set of neighbor nodes (stencil) for a node x_i as:

$$S_i = \{j \in \{1, \dots, n\} \setminus \{i\} : \text{supp}(\phi_i) \cap \text{supp}(\phi_j) \neq \emptyset\}. \tag{3.9}$$

Geometrically, $j \in S_i$ implies that the vertices $\mathbf{x}_i$ and $\mathbf{x}_j$ are connected by an edge in the triangulation.

3.1.2. Derivation of the Linear Discrete System

The Galerkin finite element discretization is based on the weak formulation derived in Definition 2.7. We seek a discrete solution $u_h \in W_h$ such that $u_h - u_{D,h} \in V_h$ and

$$a(u_h, v_h) = l(v_h) \qquad \forall v_h \in V_h. \tag{3.10}$$

Here, $u_{D,h} \in W_h$ denotes a discrete lifting of the Dirichlet boundary data. The test space V_h consists of functions that vanish on the Dirichlet boundary Γ_D and are unconstrained on the Neumann boundary Γ_N.

We express the discrete solution u_h using the global nodal basis $\{\phi_1, \dots, \phi_n\}$ of W_h:

$$u_h(\mathbf{x}) = \sum_{j=1}^{n} u_j \phi_j(\mathbf{x}). \tag{3.11}$$

The coefficient vector $\underline{u} = (u_1, \dots, u_n)^T$ contains the values for degrees of freedom located in the interior and on the Neumann boundary, representing the unknowns, as well as those on the Dirichlet boundary, representing the prescribed values.

Substituting this expansion into the variational equation and testing with the basis functions ϕ_i associated with the active nodes ($i = 1, \dots, m$), we obtain the algebraic constraints:

$$\sum_{j=1}^{n} a(\phi_j, \phi_i) u_j = l(\phi_i), \quad i = 1, \dots, m. \tag{3.12}$$

For the Dirichlet nodes ($i = m+1, \dots, n$), the values are explicitly determined by the boundary data u_D. Following the standard Lagrange interpolation, we set:

$$u_i = u_i^b = u_D(\mathbf{x}_i), \quad i = m+1, \dots, n. \tag{3.13}$$

The coefficients of the linear system are determined by the integral forms defined in Definition 2.7. Specifically, the stiffness matrix entries a_{ij} are given by:

$$a_{ij} = a(\phi_j, \phi_i) = \int_\Omega \left(\varepsilon \nabla \phi_j \cdot \nabla \phi_i + (\mathbf{b} \cdot \nabla \phi_j) \phi_i + c \phi_j \phi_i \right) d\mathbf{x}. \tag{3.14}$$

Consistent with Definition 2.7, we define

$$l_i := l(\phi_i) = \int_\Omega f \phi_i \, d\mathbf{x} + \int_{\Gamma_N} g_N \phi_i \, ds, \qquad i = 1, \dots, m, \tag{3.15}$$

where the boundary integral over Γ_N vanishes if the trace of ϕ_i has no support on Γ_N.

Remark 3.2 (Partition of Unity)**.** *A fundamental algebraic characteristic of the global nodal basis is the partition of unity:*

$$\sum_{j=1}^{n} \phi_j(\mathbf{x}) = 1, \quad \forall \mathbf{x} \in \bar{\Omega}. \tag{3.16}$$

Since W_h inherently contains the globally constant function $v(\mathbf{x}) \equiv 1$, this function admits a unique representation through its finite element nodal expansion:

$$v(\mathbf{x}) = \sum_{j=1}^{n} v(\mathbf{x}_j)\phi_j(\mathbf{x}). \tag{3.17}$$

Given $v(\mathbf{x}) \equiv 1$, evaluating this function at any node yields $v(\mathbf{x}_j) = 1$ for all $j = 1, \ldots, n$. Substituting these constant nodal values directly into the expansion gives:

$$1 = \sum_{j=1}^{n} 1 \cdot \phi_j(\mathbf{x}) = \sum_{j=1}^{n} \phi_j(\mathbf{x}). \tag{3.18}$$

This proves the partition of unity property.

Equations (3.12)–(3.15) fully define the linear discrete system. The matrix-vector representation of these equations, including the specific partitioning of active and Dirichlet degrees of freedom, is detailed in Section 3.2.

3.2. Linear Discrete Problems

The finite element discretization derived in Section 3.1.2 leads to a linear algebraic system. The algebraic notation and the discrete maximum principle framework used in this section follow the presentation in [BJK25, Section 3.1]. Let n be the total number of degrees of freedom, ordered such that the indices $1, \ldots, m$ correspond to the active degrees of freedom and the indices $m + 1, \ldots, n$ correspond to the Dirichlet boundary nodes. Then the discrete problem reads

$$\sum_{j=1}^{n} a_{ij}u_j = l_i, \qquad i = 1, \ldots, m, \tag{3.19}$$

$$u_i = u_i^b, \qquad i = m + 1, \ldots, n. \tag{3.20}$$

Introducing the vectors

$$\underline{u}^i := (u_1, \ldots, u_m)^T \in \mathbb{R}^m, \qquad \underline{u}^b := (u_{m+1}^b, \ldots, u_n^b)^T \in \mathbb{R}^{n-m}, \tag{3.21}$$

and

$$\underline{l} := (l_1, \ldots, l_m)^T \in \mathbb{R}^m, \tag{3.22}$$

the system can be written in block form as

$$A\underline{u} = \begin{pmatrix} A^i & A^b \\ 0 & I \end{pmatrix} \begin{pmatrix} \underline{u}^i \\ \underline{u}^b \end{pmatrix} = \begin{pmatrix} \underline{l} \\ \underline{u}^b \end{pmatrix}, \tag{3.23}$$

where $A \in \mathbb{R}^{n\times n}$ is the global system matrix, $A^i \in \mathbb{R}^{m\times m}$ is the submatrix associated with the active degrees of freedom, $A^b \in \mathbb{R}^{m\times(n-m)}$ describes the coupling with the Dirichlet boundary nodes, and $I \in \mathbb{R}^{(n-m)\times(n-m)}$ is the identity matrix.

Lemma 3.3 (Nonsingularity of A and A^i)**.** *The matrix A is nonsingular if and only if the matrix A^i is nonsingular. If A is nonsingular, its inverse is given by*

$$A^{-1} = \begin{pmatrix} (A^i)^{-1} & -(A^i)^{-1}A^b \\ 0 & I \end{pmatrix}. \tag{3.24}$$

Proof. The first statement follows immediately from the determinant property of block triangular matrices. Indeed, the matrix

$$A = \begin{pmatrix} A^i & A^b \\ 0 & I \end{pmatrix} \tag{3.25}$$

is block upper triangular. Therefore,

$$\det(A) = \det(A^i)\det(I) = \det(A^i). \tag{3.26}$$

Thus, $\det(A) \neq 0$ if and only if $\det(A^i) \neq 0$.

The formula for the inverse can be verified by direct matrix multiplication:

$$\begin{pmatrix} A^i & A^b \\ 0 & I \end{pmatrix} \begin{pmatrix} (A^i)^{-1} & -(A^i)^{-1}A^b \\ 0 & I \end{pmatrix} = \begin{pmatrix} A^i(A^i)^{-1} & -A^i(A^i)^{-1}A^b + A^bI \\ 0 & I \end{pmatrix} = \begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix}. \tag{3.27}$$

This completes the proof. □

A critical property for the well-posedness of the discrete problem is the positive definiteness of the inner matrix A^i. We assume that the active matrix $A^i = (a_{kl})_{k,l=1}^m \in \mathbb{R}^{m\times m}$ satisfies the condition:

$$\sum_{k=1}^{m}\sum_{l=1}^{m} v_k a_{kl} v_l > 0, \quad \forall \underline{v} \in \mathbb{R}^m \setminus \{\underline{0}\}. \tag{3.28}$$

Lemma 3.4 (Positive Definiteness of A^i)**.** *Let the bilinear form $a(\cdot,\cdot)$ be coercive on V_0 with constant $\alpha > 0$. Then the inner stiffness matrix A^i is positive definite.*

Proof. Let $\underline{v} \in \mathbb{R}^m \setminus \{\underline{0}\}$ be an arbitrary non-zero vector. Let $v_h \in V_h$ be the corresponding finite element function defined by the basis expansion $v_h(x) = \sum_{k=1}^m v_k\phi_k(x)$. Since the basis functions are linearly independent and $\underline{v} \neq \underline{0}$, we have $v_h \not\equiv 0$. By the definition of the stiffness matrix entries, the quadratic form satisfies:

$$\langle \underline{v}, A^i\underline{v}\rangle = \sum_{k=1}^{m}\sum_{l=1}^{m} v_k a(\phi_l,\phi_k)v_l = a\left(\sum_{l=1}^{m} v_l\phi_l, \sum_{k=1}^{m} v_k\phi_k\right) = a(v_h, v_h). \tag{3.29}$$

Using the coercivity of the bilinear form (2.34), we obtain:

$$\langle \underline{v}, A^i\underline{v}\rangle = a(v_h, v_h) \geq \alpha\|v_h\|^2_{H^1(\Omega)} > 0. \tag{3.30}$$

Thus, A^i is positive definite. □

We introduce the following notation for component-wise inequalities and special vectors. For any two vectors $\underline{v}, \underline{w} \in \mathbb{R}^n$, the notation $\underline{v} \leq \underline{w}$ signifies that $v_i \leq w_i$ holds for all indices $i = 1,\ldots,n$. Similarly, for a matrix $A = (a_{ij}) \in \mathbb{R}^{n\times n}$, the inequality $A \geq 0$ indicates that all entries satisfy $a_{ij} \geq 0$ (respectively, $a_{ij} > 0$).

We denote the vector of all zeros by $\underline{0}$ and the vector of all ones by $\underline{1}$. Where necessary to avoid ambiguity, the dimension will be indicated by a subscript (e.g., $\underline{0}_n$, $\underline{1}_n$). The i-th canonical unit vector is denoted by $\underline{e}_i$. Finally, to simplify notation, the expression all $v_i \geq 0$ is used as a shorthand for $\underline{v} \geq \underline{0}$.

Having established the algebraic structure and solvability conditions, now we proceed to define the discrete maximum principles.

3.2.1. Local and Global DMPs

This section establishes general algebraic sufficient conditions for the validity of local and global discrete maximum principles. The theoretical framework is built upon the concept of *matrices of non-negative type*.

Definition 3.5 (Matrix of Non-negative Type)**.** *A matrix $A = (a_{ij}) \in \mathbb{R}^{m\times n}$ is called a matrix of non-negative type (classically known as a Minkowski matrix) if it satisfies the following two conditions:*

$$a_{ij} \leq 0, \quad \forall j \neq i, \quad (1 \leq i \leq m,\ 1 \leq j \leq n), \tag{3.31}$$

$$\sum_{j=1}^{n} a_{ij} \geq 0, \quad \forall i = 1, \ldots, m. \tag{3.32}$$

The first condition (3.31) requires non-positive off-diagonal entries, while the second condition (3.32) requires non-negative row sums. Note that the term "matrix of non-negative type" is distinct from a "non-negative matrix" (where all entries are non-negative).

Theorem 3.6 (Sufficient and Necessary Conditions for the Local DMP)**.** *[BJK25, Theorem 3.3, Section 3.1.1] Let $A \in \mathbb{R}^{m\times n}$ with $0 < m < n$. The local DMPs are satisfied if and only if A is a matrix of non-negative type (satisfying conditions (3.31) and (3.32)) and possesses strictly positive diagonal entries:*

$$a_{ii} > 0, \quad \forall i = 1, \ldots, m. \tag{3.33}$$

Specifically, the implication

$$\sum_{j=1}^{n} a_{ij}u_j \leq 0 \quad \Longrightarrow \quad u_i \leq \max_{j\neq i, a_{ij}\neq 0} u_j^+, \tag{3.34}$$

$$\sum_{j=1}^{n} a_{ij}u_j \geq 0 \quad \Longrightarrow \quad u_i \geq \min_{j\neq i, a_{ij}\neq 0} u_j^- \tag{3.35}$$

hold for any arbitrary vector $\underline{u} \in \mathbb{R}^n$ if and only if the matrix properties hold.

Proof. Since the minimum principle follows from the maximum principle by considering $-\underline{u}$, we restrict the proof to the implication (3.34).
Necessity ($\Rightarrow$): We assume that one of the matrix conditions is violated and construct a vector $\underline{u}$ for which (3.34) fails.

1. *Violation of positive diagonal:* Suppose $a_{ii} \leq 0$ for some index i. Choose $\underline{u}$ such that $u_i = 1$ and $u_j = 0$ for all $j \neq i$. Then $\sum_{j=1}^{n} a_{ij}u_j = a_{ii} \leq 0$, but $u_i = 1 > 0 = \max_{j\neq i} u_j^+$. The DMP is violated.

2. *Violation of sign condition:* Suppose $a_{ik} > 0$ for some $k \neq i$. We assume $a_{ii} > 0$ (otherwise case 1 applies). Define $\underline{u}$ by:

$$u_i = 1, \quad u_k = -\frac{a_{ii}}{a_{ik}}, \quad u_j = 0 \quad \forall j \in \{1, \ldots, n\} \setminus \{i, k\}. \tag{3.36}$$

   Here, $u_k < 0$, so $\max_{j\neq i, a_{ij}\neq 0} u_j^+ = 0$. However, the row sum yields $\sum_{j=1}^{n} a_{ij}u_j = a_{ii}(1) + a_{ik}(-a_{ii}/a_{ik}) = 0$. From (3.34), since $u_i = 1 > 0$, the DMP is violated.

3. *Violation of row sum:* Suppose $\sum_{j=1}^{n} a_{ij} < 0$ for some i, while the other conditions hold. We define $\underline{u}$ by:

$$u_i = 1 - \frac{1}{a_{ii}} \sum_{j=1}^{n} a_{ij} > 1, \quad u_j = 1 \quad \forall j \neq i. \tag{3.37}$$

Computing the matrix-vector product yields:

$$\sum_{j=1}^{n} a_{ij} u_j = a_{ii} u_i + \sum_{j \neq i} a_{ij} = a_{ii} - \sum_{k=1}^{n} a_{ik} + \sum_{j \neq i} a_{ij} = 0. \tag{3.38}$$

The left-hand side of (3.34) equals zero, yet $u_i > 1$. According to (3.34), the DMP is violated.

Sufficiency ($\Leftarrow$): Assume A is a matrix of non-negative type with $a_{ii} > 0$. Let $\underline{u}$ be a vector satisfying $\sum_{j=1}^{n} a_{ij} u_j \leq 0$. We define the bound $r := \max_{j \neq i, a_{ij} \neq 0} u_j^+$. Since $r \geq 0$, and using the non-positive off-diagonal entries ($a_{ij} \leq 0$ for $j \neq i$), we have:

$$a_{ii} u_i \leq \sum_{j \neq i} (-a_{ij}) u_j = \sum_{j \neq i} (-a_{ij})(u_j - r) + \sum_{j \neq i} (-a_{ij}) r. \tag{3.39}$$

Observing that $u_j \leq u_j^+ \leq r$ implies $(u_j - r) \leq 0$, and since $-a_{ij} \geq 0$, the first summation is non-positive. We can bound the expression by:

$$a_{ii} u_i \leq r \sum_{j \neq i} (-a_{ij}). \tag{3.40}$$

Using the row sum condition $\sum_{j=1}^{n} a_{ij} \geq 0$, we have $a_{ii} \geq \sum_{j \neq i} (-a_{ij})$. Substituting this into the inequality:

$$a_{ii} u_i \leq r a_{ii}. \tag{3.41}$$

Since $a_{ii} > 0$, dividing by the diagonal entry yields $u_i \leq r$, which completes the proof. □

Remark 3.7 (Standard Local DMPs vs. Stronger Local DMPs)**.** *Theorem 3.6 characterizes the Standard Local DMPs, where the solution bounds depend on the zero level. However, Stronger Local DMPs [BJK25, Theorem 3.4, Section 3.1.1] can be established if the numerical scheme yields a stiffness matrix with zero row sums:*

$$\sum_{j=1}^{n} a_{ij} = 0. \tag{3.42}$$

Under condition (3.42), one has $a_{ii} = -\sum_{j \neq i} a_{ij}$. *Hence, if the homogeneous discrete equation* $\sum_{j=1}^{n} a_{ij} u_j = 0$ *is satisfied at node* x_i, *then*

$$a_{ii} u_i = -\sum_{j \neq i} a_{ij} u_j. \tag{3.43}$$

Therefore,

$$u_i = \sum_{j \neq i} \omega_{ij} u_j, \qquad \omega_{ij} = \frac{-a_{ij}}{a_{ii}}. \tag{3.44}$$

Since the matrix A is of non-negative type, the weights satisfy $\omega_{ij} \geq 0$ and $\sum_{j\neq i} \omega_{ij} = 1$. This implies that u_i is a convex combination of its neighboring values, directly leading to the stronger bounds:

$$\min_{j\neq i, a_{ij}\neq 0} u_j \leq u_i \leq \max_{j\neq i, a_{ij}\neq 0} u_j. \tag{3.45}$$

It is important to emphasize that the vanishing of row sums is a matrix property dependent on the discretization; for instance, the EAFE scheme [XZ99] may not satisfy (3.42) even for purely convective-diffusive problems. For general convection-diffusion-reaction problems, the reaction term typically results in $\sum_{j=1}^{n} a_{ij} > 0$, making the standard local DMPs the relevant stability criterion.

Theorem 3.8 (Sufficient Conditions for the Global DMP)**.** *[BJK25, Theorem 3.5, Section 3.1.1]*

Consider a matrix $A = (a_{ij}) \in \mathbb{R}^{m\times n}$ with $0 < m < n$ satisfying the non-negative type conditions (3.31) and (3.32), and let the active matrix $A^i = (a_{ij})_{i,j=1}^{m}$ be nonsingular. Then, for any vector $\underline{u} \in \mathbb{R}^n$, the following global DMPs hold:

$$\sum_{j=1}^{n} a_{ij}u_j \leq 0, \quad \forall i = 1,\dots,m \quad \Longrightarrow \quad \max_{i=1,\dots,n} u_i \leq \max_{k=m+1,\dots,n} u_k^+, \tag{3.46}$$

$$\sum_{j=1}^{n} a_{ij}u_j \geq 0, \quad \forall i = 1,\dots,m \quad \Longrightarrow \quad \min_{i=1,\dots,n} u_i \geq \min_{k=m+1,\dots,n} u_k^-. \tag{3.47}$$

If condition (3.42) is satisfied, then

$$\sum_{j=1}^{n} a_{ij}u_j \leq 0, \quad \forall i = 1,\dots,m \quad \Longrightarrow \quad \max_{i=1,\dots,n} u_i = \max_{k=m+1,\dots,n} u_k, \tag{3.48}$$

$$\sum_{j=1}^{n} a_{ij}u_j \geq 0, \quad \forall i = 1,\dots,m \quad \Longrightarrow \quad \min_{i=1,\dots,n} u_i = \min_{k=m+1,\dots,n} u_k. \tag{3.49}$$

Proof. The proof follows the argument in [BJK25, Theorem 3.5, Section 3.1.1]. We proceed by contradiction to prove (3.46). Assume the implication is false. This implies that there exists a vector $\underline{u}$ such that

$$\sum_{j=1}^{n} a_{ij}u_j \leq 0, \qquad i = 1,\dots,m, \tag{3.50}$$

while

$$\max_{i=1,\dots,n} u_i > \max_{k=m+1,\dots,n} u_k^+. \tag{3.51}$$

Let

$$S := \max_{i=1,\dots,n} u_i. \tag{3.52}$$

By our assumption, $S > 0$, and the maximum cannot be attained at a Dirichlet boundary node. We define the set of active indices where this maximum is attained as

$$\mathcal{J} := \{i \in \{1,\dots,m\} : u_i = S\}. \tag{3.53}$$

Then $\mathcal{J}$ is non-empty.

We first show that there exists an index $k \in \mathcal{J}$ such that

$$\mu_k := \sum_{j\in\mathcal{J}} a_{kj} > 0. \tag{3.54}$$

Suppose, to the contrary, that

$$\sum_{j\in\mathcal{J}} a_{ij} \le 0 \qquad \forall i \in \mathcal{J}. \tag{3.55}$$

Since A is of non-negative type, its row sums are non-negative and its off-diagonal entries are non-positive. Therefore, for each $i \in \mathcal{J}$,

$$0 \le \sum_{j=1}^{n} a_{ij} = \sum_{j\in\mathcal{J}} a_{ij} + \sum_{j\notin\mathcal{J}} a_{ij}. \tag{3.56}$$

The first term on the right-hand side is non-positive by assumption. The second term is also non-positive, because $a_{ij} \le 0$ for $j \ne i$ and $j \notin \mathcal{J}$. Hence both terms must vanish:

$$\sum_{j\in\mathcal{J}} a_{ij} = 0 \qquad \forall i \in \mathcal{J}, \tag{3.57}$$

and

$$a_{ij} = 0 \qquad \forall i \in \mathcal{J}, \quad j \notin \mathcal{J}. \tag{3.58}$$

In particular, the submatrix $(a_{ij})_{i,j\in\mathcal{J}}$ is singular, since the sum of its rows is zero. Consequently, its transpose is also singular. Hence, there exists a non-zero vector $(v_i)_{i\in\mathcal{J}}$ such that

$$\sum_{i\in\mathcal{J}} a_{ij} v_i = 0 \qquad \forall j \in \mathcal{J}. \tag{3.59}$$

We extend this vector to a vector $\underline{\tilde{v}} \in \mathbb{R}^m$ by setting

$$\tilde{v}_i = \begin{cases} v_i, & i \in \mathcal{J}, \\ 0, & i \notin \mathcal{J}. \end{cases} \tag{3.60}$$

Using (3.58) and (3.59), we obtain, for all $j = 1, \dots, m$,

$$\sum_{i=1}^{m} a_{ij}\tilde{v}_i = \sum_{i\in\mathcal{J}} a_{ij} v_i = 0. \tag{3.61}$$

Thus $(A^i)^T \underline{\tilde{v}} = \underline{0}$. Since $\underline{\tilde{v}} \ne \underline{0}$, the matrix $(A^i)^T$ is singular, and therefore A^i is singular. This contradicts the assumption that A^i is nonsingular. Hence there must exist $k \in \mathcal{J}$ satisfying (3.54).

Now choose such an index $k \in \mathcal{J}$ and define

$$r := \max_{j\notin\mathcal{J}} u_j. \tag{3.62}$$

Since $u_j = S$ for $j \in \mathcal{J}$, the discrete inequality at the index k gives

$$0 \ge \sum_{j=1}^{n} a_{kj} u_j = S \sum_{j\in\mathcal{J}} a_{kj} + \sum_{j\notin\mathcal{J}} a_{kj} u_j. \tag{3.63}$$

Hence

$$S\mu_k \le -\sum_{j\notin\mathcal{J}} a_{kj} u_j. \tag{3.64}$$

Since $a_{kj} \le 0$ for $j \notin \mathcal{J}$ and $u_j \le r$, we have

$$-\sum_{j\notin\mathcal{J}} a_{kj} u_j \le r \sum_{j\notin\mathcal{J}} (-a_{kj}). \tag{3.65}$$

Furthermore, the row sum condition implies

$$0 \leq \sum_{j=1}^{n} a_{kj} = \sum_{j \in \mathcal{J}} a_{kj} + \sum_{j \notin \mathcal{J}} a_{kj} = \mu_k + \sum_{j \notin \mathcal{J}} a_{kj}, \tag{3.66}$$

and therefore

$$\sum_{j \notin \mathcal{J}} (-a_{kj}) \leq \mu_k. \tag{3.67}$$

Combining the preceding inequalities yields

$$S\mu_k \leq r\mu_k. \tag{3.68}$$

Since $\mu_k > 0$, it follows that $S \leq r$. This contradicts the definition of $\mathcal{J}$ as the set of active indices attaining the global maximum, unless the maximum is also attained at a Dirichlet boundary node. Hence

$$\max_{i=1,\dots,n} u_i \leq \max_{k=m+1,\dots,n} u_k^+. \tag{3.69}$$

This proves (3.46).

Finally, if the zero row sum condition (3.42) is satisfied, we define a shifted vector $\underline{\tilde{u}}$, where $\tilde{u}_i = u_i + q$ and $q > 0$ is chosen sufficiently large such that $\tilde{u}_i > 0$ for all $i = 1, \dots, n$. Since $\sum_{j=1}^{n} a_{ij} = 0$, the shifted vector satisfies

$$\sum_{j=1}^{n} a_{ij}\tilde{u}_j = \sum_{j=1}^{n} a_{ij}u_j + q\sum_{j=1}^{n} a_{ij} = \sum_{j=1}^{n} a_{ij}u_j \leq 0. \tag{3.70}$$

Applying (3.46) to $\underline{\tilde{u}}$ yields

$$\max_{i=1,\dots,n} \tilde{u}_i \leq \max_{k=m+1,\dots,n} \tilde{u}_k^+ = \max_{k=m+1,\dots,n} \tilde{u}_k, \tag{3.71}$$

where the equality holds because $\tilde{u}_i > 0$ for all i. Subtracting the constant q from both sides, we obtain

$$\max_{i=1,\dots,n} u_i \leq \max_{k=m+1,\dots,n} u_k. \tag{3.72}$$

Since the set of Dirichlet nodes $\{m+1, \dots, n\}$ is a subset of all nodes $\{1, \dots, n\}$, the reverse inequality holds by definition. Thus, the equality is established. The proof for the minimum principle follows by an analogous argument applied to $-\underline{u}$. □

Remark 3.9 (Weak and Strong Global DMPs)**.** *The distinction between weak and strong global DMPs follows the terminology in [BJK25, Remark 3.7, Section 3.1.1]. The global DMPs in Theorem 3.8 control the global maximum and minimum by the prescribed boundary values, but they still allow the extremal value to be attained at an active degree of freedom. In this sense, they are weak global DMPs.*

A strong global DMP excludes this situation. If the row sums vanish, $\sum_{j=1}^{n} a_{ij} = 0$ for $i = 1, \dots, m$, and if the corresponding discrete residual inequalities are strict, then

$$\sum_{j=1}^{n} a_{ij}u_j < 0, \quad i = 1, \dots, m \quad \Longrightarrow \quad \max_{i=1,\dots,m} u_i < \max_{k=m+1,\dots,n} u_k, \tag{3.73}$$

$$\sum_{j=1}^{n} a_{ij}u_j > 0, \quad i = 1, \dots, m \quad \Longrightarrow \quad \min_{i=1,\dots,m} u_i > \min_{k=m+1,\dots,n} u_k. \tag{3.74}$$

The argument is short. If an active node k attained the global maximum, then $u_j - u_k \leq 0$ for all j. Using the zero row sum condition gives

$$\sum_{j=1}^{n} a_{kj} u_j = \sum_{j=1}^{n} a_{kj}(u_j - u_k). \tag{3.75}$$

Since $a_{kj} \leq 0$ for $j \neq k$, the right-hand side is non-negative, contradicting the strict assumption $\sum_{j=1}^{n} a_{kj} u_j < 0$. The minimum case is analogous.

Remark 3.10 (Independence of Local and Global DMPs)**.** *It is important to note that local and global DMPs are distinct mathematical properties. According to Theorem 3.6, the satisfaction of local DMPs is characterized by the matrix being of non-negative type together with positive diagonal entries. However, this does not logically imply the validity of global DMPs, which, as shown in Theorem 3.8 and Theorem 3.13, additionally relies on global algebraic properties such as nonsingularity and inverse-positivity. Conversely, a discrete system may satisfy global DMPs even if the local DMP conditions are violated. This is because the assumption that A is of non-negative type is a sufficient, but not necessary, condition for global stability of the system.*

3.2.2. Global DMPs via Inverse Positive Matrices

The theory of monotone matrices (also referred to as inverse-positive matrices) serves as the fundamental algebraic framework for analyzing global DMPs. This subsection introduces this class of operators and establishes the necessary and sufficient criteria for the satisfaction of the global maximum principles. Special emphasis is placed on *M-matrices*, a vital subset of monotone matrices that facilitates the practical verification of these stability properties in finite element schemes. The presentation in this subsection follows the theoretical framework of [BJK25, Section 3.1.2].

Definition 3.11 (Monotone Matrix)**.** *A matrix $A = (a_{ij})_{i,j=1}^{n}$ is called monotone if:*

(i) A is nonsingular;

(ii) $A^{-1} \geq 0$.

Lemma 3.12 (Equivalent Characterization of Monotone Matrices)**.** *A matrix $A \in \mathbb{R}^{n\times n}$ is a monotone matrix if and only if the following implication holds for all vectors $\underline{v} \in \mathbb{R}^n$:*

$$A\underline{v} \geq 0 \quad \Longrightarrow \quad \underline{v} \geq 0. \tag{3.76}$$

Proof. **Necessity ($\Rightarrow$):** Let A be monotone. Then A^{-1} exists and $A^{-1} \geq 0$. For any vector $\underline{v}$ satisfying $A\underline{v} \geq 0$, we have

$$\underline{v} = A^{-1}(A\underline{v}) \geq A^{-1}\underline{0} = \underline{0}. \tag{3.77}$$

Sufficiency ($\Leftarrow$): Assume that implication (3.76) holds. First, we verify nonsingularity. Let $\underline{v} \in \ker(A)$, meaning $A\underline{v} = \underline{0}$. This implies both $A\underline{v} \geq 0$ and $A(-\underline{v}) \geq 0$. By the hypothesis, we must have $\underline{v} \geq 0$ and $-\underline{v} \geq 0$, which necessitates $\underline{v} = \underline{0}$. Thus, $\ker(A)$ is trivial and A is invertible.

Next, to show $A^{-1} \geq 0$, consider the j-th canonical unit vector $\underline{e}_j \geq 0$. Let $\underline{x}_j = A^{-1}\underline{e}_j$ denote the j-th column of the inverse. Since $A\underline{x}_j = \underline{e}_j \geq 0$, the hypothesis implies $\underline{x}_j \geq 0$. Since this holds for all columns $j = 1, \ldots, n$, it follows that $A^{-1} \geq 0$. □

This result serves as a fundamental algebraic tool in the analysis of DMPs. In particular, the implication (3.76) provides an equivalent characterization of monotone matrices, which will be used in Theorem 3.13 to derive the necessary and sufficient conditions for the validity of the global DMPs. The following characterization is due to [Cia70] and we use the formulation presented in [BJK25, Theorem 3.16, Section 3.1.2].

Theorem 3.13 (Sufficient and Necessary Conditions for Global DMPs)**.** *[Cia70] Let the system matrix A be partitioned into internal and boundary blocks as defined in (3.23), i.e., $A^i \in \mathbb{R}^{m\times m}$ and $A^b \in \mathbb{R}^{m\times(n-m)}$. Let the vector $\underline{u}$ be partitioned as $\underline{u} = (\underline{u}^i, \underline{u}^b)^T$. The global DMP (in matrix-vector form):*

$$A^i\underline{u}^i + A^b\underline{u}^b \leq 0 \quad \Longrightarrow \quad \max_{j=1,\dots,n} u_j \leq \max_{k=m+1,\dots,n} u_k^+, \tag{3.78}$$

$$A^i\underline{u}^i + A^b\underline{u}^b \geq 0 \quad \Longrightarrow \quad \min_{j=1,\dots,n} u_j \geq \min_{k=m+1,\dots,n} u_k^-, \tag{3.79}$$

*hold **if and only if** the following two conditions are satisfied:*

1. *A is monotone,*
2. *The row sums of the matrix $-(A^i)^{-1}A^b$ satisfy:*

$$-(A^i)^{-1}A^b\underline{1}_{n-m} \leq \underline{1}_m, \tag{3.80}$$

where $\underline{1}_k$ denotes a vector of ones of size k.

Proof. The detailed proof can be found in the classical work of [Cia70, Theorem 1]. □

The conditions in Theorem 3.13 can be significantly simplified if the matrix satisfies the standard row sum condition (consistent with the conservation property).

Corollary 3.14 (Simplified Criteria under Row Sum Condition)**.** *Assume that the matrix A satisfies the non-negative row sum condition (3.32). Then the global DMPs (3.78) and (3.79) hold **if and only if** A is monotone.*

Proof. **Necessity ($\Rightarrow$):** If the global DMPs are satisfied, Condition 1 of Theorem 3.13 must hold. Thus, A must be monotone.

Sufficiency ($\Leftarrow$): Assume A is monotone, then A^i is monotone (i.e., $(A^i)^{-1} \geq 0$). We only need to verify that Condition 2 of Theorem 3.13 is automatically satisfied under the row sum assumption. The row sum condition $\sum_{j=1}^{n} a_{ij} \geq 0$ can be written in block-vector form as:

$$A^i\underline{1}_m + A^b\underline{1}_{n-m} \geq \underline{0}_m. \tag{3.81}$$

Rearranging the terms yields $A^i\underline{1}_m \geq -A^b\underline{1}_{n-m}$. Since $(A^i)^{-1} \geq 0$, we can pre-multiply by the inverse without changing the inequality direction:

$$\underline{1}_m \geq -(A^i)^{-1}A^b\underline{1}_{n-m}. \tag{3.82}$$

This is precisely Condition 2 of Theorem 3.13. Thus, monotonicity combined with non-negative row sums is sufficient for the validity of the global DMPs. □

Directly verifying the monotonicity condition in Theorem 3.13 is computationally prohibitive for large systems, as it requires the inverse matrix A^{-1}. To avoid this, we focus on the class of M-matrices. Since every M-matrix is monotone by definition, this property provides a practical sufficient condition for the global DMPs.

Definition 3.15 (M-Matrix [Ost37])**.** *A matrix $A = (a_{ij})_{i,j=1}^N$ is an M-matrix if:*

1. *$a_{ii} \geq 0$ for all $i = 1, \dots, N$;*
2. *$a_{ij} \leq 0$ for all $i \neq j$;*

3. *All principal minors of A are non-negative;*

4. $\det(A) > 0$.

Although this algebraic characterization is mathematically equivalent to Definition 3.16, the inverse-positivity condition ($A^{-1} \geq 0$) provides a more direct link to the Discrete Maximum Principle.

Definition 3.16 (M-matrix)**.** *A matrix $A = (a_{ij})_{i,j=1}^{n}$ is an M-matrix if*

(i) the off-diagonal entries are nonpositive, i.e., $a_{ij} \leq 0$, $i, j = 1, \ldots, n$, $i \neq j$;

(ii) A is nonsingular; and

(iii) $A^{-1} \geq 0$.

In Definition 3.11, a matrix that satisfies conditions (ii) and (iii) is called a monotone matrix.

Since verifying monotonicity directly is difficult, we now show that the M-matrix property provides a sufficient algebraic condition. First, we establish that for the block structures considered here, the M-matrix property of the global matrix A is determined solely by its inner part A^i.

Lemma 3.17 (Restriction to Inner Nodes)**.** *Let A be a nonsingular matrix of the form (3.23) with the condition (i) of Definition 3.16. Then the full matrix A is an M-matrix if and only if the active matrix A^i is an M-matrix.*

Proof. By the determinant property of block triangular matrices, A is nonsingular if and only if A^i is nonsingular. If A^i is nonsingular, then

$$A^{-1} = \begin{pmatrix} (A^i)^{-1} & -(A^i)^{-1}A^b \\ 0 & I \end{pmatrix}. \tag{3.83}$$

If A^i is an M-matrix, then $(A^i)^{-1} \geq 0$. Since condition (i) of Definition 3.16 implies $A^b \leq 0$, it follows that $-(A^i)^{-1}A^b \geq 0$. Hence $A^{-1} \geq 0$, and therefore A is an M-matrix.

Conversely, if A is an M-matrix, then $A^{-1} \geq 0$. Since the upper-left block of A^{-1} is $(A^i)^{-1}$, we obtain $(A^i)^{-1} \geq 0$. Together with the nonsingularity of A^i and the nonpositive off-diagonal entries inherited from A, this shows that A^i is an M-matrix. □

Corollary 3.18 (M-Matrices and Global DMPs)**.** *Let the discretization matrix A satisfy Condition (3.32). If A is an M-matrix, then the discrete solution satisfies the Global DMPs (3.78) and (3.79).*

Proof. Since A is an M-matrix, it is by definition a monotone matrix (i.e., $A^{-1} \geq 0$). The statement then follows directly from Corollary 3.14, which states that for matrices with non-negative row sums, monotonicity of A is sufficient to guarantee the validity of the global DMPs. □

3.2.3. Violation of the DMP in the Galerkin Method

The satisfaction of a Discrete Maximum Principle requires the global stiffness matrix A to follow a specific sign pattern. According to Theorem 3.6, a necessary condition for the discrete solution to respect the bounds prescribed by the continuous Maximum Principle is that the off-diagonal entries of A are non-positive:

$$a_{ij} \leq 0, \quad \forall i, j = 1, \ldots, n, \quad i \neq j. \tag{3.84}$$

It should be noted that the sign condition (3.84) is an essential algebraic requirement for the DMP. However, even if the solution remains within the prescribed physical bounds, it may still exhibit oscillations. To analyze this behavior, we first define the consistent finite element matrices.

Definition 3.19. *The diffusive, convective, and reactive operators in the standard Galerkin framework are represented by the matrices* $\mathbb{A}_d, \mathbb{A}_c, \mathbb{M} \in \mathbb{R}^{n\times n}$*, whose entries for* $i, j = 1, \dots, n$ *are defined as:*

- ***Diffusion Matrix:*** $\mathbb{A}_d = (a_{ij}^d)_{i,j=1}^n$*, where* $a_{ij}^d = \varepsilon\,(\nabla\phi_j, \nabla\phi_i)_{L^2}$.
- ***Convection Matrix:*** $\mathbb{A}_c = (c_{ij})_{i,j=1}^n$*, where* $c_{ij} = (\boldsymbol{b}\cdot\nabla\phi_j, \phi_i)_{L^2}$.
- ***Reaction Matrix or the consistent mass matrix:*** $\mathbb{M} = (m_{ij})_{i,j=1}^n$*, where* $m_{ij} = (\phi_j, \phi_i)_{L^2}$.

In the standard Galerkin method, the global entry a_{ij} is obtained by assembling the local contributions from all elements K in the support of the basis functions ϕ_i and ϕ_j. To clearly identify the source of instability, we focus on the interplay between diffusion and convection. Following standard analyses, we consider the convection-dominated regime where the reaction term is vanishing ($c = 0$). Under this assumption, the local stiffness entry a_{ij}^K on an element K is given by:

$$a_{ij}^K = \underbrace{\int_K \varepsilon\nabla\phi_j\cdot\nabla\phi_i\,d\mathbf{x}}_{a_{ij}^{\mathrm{diff},K}} + \underbrace{\int_K (\mathbf{b}\cdot\nabla\phi_j)\phi_i\,d\mathbf{x}}_{a_{ij}^{\mathrm{conv},K}}. \tag{3.85}$$

The diffusive part $a_{ij}^{\mathrm{diff},K}$ depends solely on the mesh geometry. Its sign property can be derived explicitly from the properties of simplicial elements. Let F_i^K denote the face of the simplex K opposite to vertex x_i, and let $\mathbf{n}_i^K$ be the corresponding outward unit normal vector. Since the linear basis function ϕ_i vanishes on F_i^K and attains the value 1 at vertex x_i, its gradient is constant and parallel to the normal vector:

$$\nabla\phi_i|_K = -\frac{|F_i^K|}{d|K|}\mathbf{n}_i^K, \tag{3.86}$$

where $|F_i^K|$ is the measure (area/volume) of the face, $|K|$ is the measure of the element, and d is the dimension.

Substituting this into the diffusion integral, the scalar product of gradients becomes related to the angle between the face normals:

$$\begin{aligned}\int_K \nabla\phi_j\cdot\nabla\phi_i\,d\mathbf{x} &= |K|\left(-\frac{|F_j^K|}{d|K|}\mathbf{n}_j^K\right)\cdot\left(-\frac{|F_i^K|}{d|K|}\mathbf{n}_i^K\right)\\ &= \frac{|F_i^K||F_j^K|}{d^2|K|}(\mathbf{n}_i^K\cdot\mathbf{n}_j^K) = -\frac{|F_i^K||F_j^K|}{d^2|K|}\cos(\theta_{ij}^K).\end{aligned} \tag{3.87}$$

Geometrically, the dot product of the outward normals corresponds to the negative cosine of the dihedral angle θ_{ij}^K between the faces F_i^K and F_j^K, which means $\mathbf{n}_i^K\cdot\mathbf{n}_j^K = -\cos(\theta_{ij}^K)$. This yields the explicit geometric representation:

$$a_{ij}^{\mathrm{diff},K} = -\varepsilon\underbrace{\frac{|F_i^K||F_j^K|}{d^2|K|}}_{>0}\cos(\theta_{ij}^K). \tag{3.88}$$

Consequently, a sufficient condition for the diffusive contribution to be non-positive ($a_{ij}^{\mathrm{diff},K} \le 0$) is that the mesh satisfies the *acuteness condition* ($\theta_{ij}^K \le \pi/2$), which ensures $\cos(\theta_{ij}^K) \ge 0$.

However, the condition for the global stiffness matrix to be an M-matrix is less restrictive than requiring every element to be acute. It relies on the assembled entries across the edge shared by neighboring elements. This necessary and sufficient condition was rigorously established by Xu and Zikatanov in [XZ99]:

Lemma 3.20 (Geometric Condition for Poisson Equation)**.** *[XZ99, Lemma 2.1] The stiffness matrix for the Poisson equation (pure diffusion) discretized by linear finite elements is an M-matrix if and only if for any fixed edge E, the following inequality holds:*

$$\omega_E := \frac{1}{d(d-1)} \sum_{K \in \mathcal{K}_E} |\kappa_E^K| \cot(\theta_E^K) \geq 0, \tag{3.89}$$

where $\mathcal{K}_E$ is the set of simplices sharing edge E, $|\kappa_E^K|$ denotes the measure of the $(d-2)$-dimensional sub-simplex opposite to edge E, and θ_E^K is the dihedral angle of K at the edge E.

Definition 3.21 (XZ-type Triangulation)**.** *A simplicial triangulation $\mathcal{T}_h$ of the domain Ω is defined to be of **XZ-type** if the geometric weight condition (3.89) is satisfied for every interior edge E of the triangulation.*

Theorem 3.22. *[BJK25, Theorem 6.8]. The system matrix $\mathbb{A}_d = (a_{ij}^d)$ of the diffusion term discretized by conforming P_1 finite elements is of non-negative type if and only if the triangulation $\mathcal{T}_h$ is of XZ-type.*

Proof. First, we establish that the row sum condition for a non-negative type matrix is unconditionally satisfied. Since $\sum_{j=1}^{n} \phi_j \equiv 1$, it follows that $\nabla \sum_{j=1}^{n} \phi_j = \mathbf{0}$. Consequently, the row sums of the global diffusion matrix are identically zero:

$$\sum_{j=1}^{n} a_{ij}^d = \int_\Omega \varepsilon \nabla \left(\sum_{j=1}^{n} \phi_j \right) \cdot \nabla \phi_i \, \mathrm{d}\boldsymbol{x} = 0 \geq 0, \quad \forall i. \tag{3.90}$$

Therefore, the classification of $\mathbb{A}_d$ as a non-negative type matrix depends entirely on whether its off-diagonal entries are non-positive.

Let E be an interior edge connecting distinct adjacent nodes x_i and x_j. The global off-diagonal entry a_{ij}^d is the standard finite element assembly of the local diffusive contributions from the set of simplices $\mathcal{K}_E$ sharing the edge E:

$$a_{ij}^d = \sum_{K \in \mathcal{K}_E} a_{ij}^{\mathrm{diff},K} = -\varepsilon \sum_{K \in \mathcal{K}_E} \frac{|F_i^K||F_j^K|}{d^2|K|} \cos(\theta_{ij}^K). \tag{3.91}$$

According to the geometric identity established in Lemma 3.20 [XZ99], we have:

$$a_{ij}^d = -\varepsilon \omega_E = -\varepsilon \left(\frac{1}{d(d-1)} \sum_{K \in \mathcal{K}_E} |\kappa_E^K| \cot(\theta_E^K) \right). \tag{3.92}$$

($\Longleftarrow$ Sufficiency): Assume $\mathcal{T}_h$ is of XZ-type. By Definition 3.21, $\omega_E \geq 0$ for all interior edges E. Since $\varepsilon > 0$, substituting this into (3.92) yields $a_{ij}^d = -\varepsilon \omega_E \leq 0$ for all $i \neq j$. For any non-adjacent nodes, $a_{ij}^d = 0 \leq 0$ holds trivially. Combined with the zero row sums, $\mathbb{A}_d$ is of non-negative type.

($\Longrightarrow$ Necessity): Assume $\mathbb{A}_d$ is of non-negative type, implying $a_{ij}^d \leq 0$ for all $i \neq j$. For any adjacent nodes sharing an edge E, (3.92) dictates $-\varepsilon \omega_E \leq 0$. Given $\varepsilon > 0$, it follows that $\omega_E \geq 0$. The triangulation $\mathcal{T}_h$ strictly satisfies Definition 3.21 and is thus of XZ-type. □

Remark 3.23. *In two dimensions, condition (3.89) simplifies to $\theta_E^{K_1} + \theta_E^{K_2} \leq \pi$. While a Delaunay triangulation is defined by the empty circumcircle property, this angle sum is an equivalent property in two dimensions. The XZ-condition is more general because it is less restrictive for cells at the domain boundary, thereby providing greater flexibility in ensuring the M-matrix property.*

Thus, the violation of the M-matrix property in the standard Galerkin method is not inherent to the diffusion discretization provided the mesh is XZ-type, but is driven by the convective term, as analyzed next.

In contrast to the symmetric and stabilizing diffusion term, the globally assembled convective contribution is skew-symmetric for internal basis functions under the assumption $\nabla \cdot \mathbf{b} = 0$. To see this, let us first consider the globally assembled convective matrix entry a_{ij}^{conv} for any two active nodes x_i and x_j. Integration by parts over the entire domain Ω yields:

$$\begin{aligned} a_{ij}^{\text{conv}} &= \int_\Omega (\mathbf{b} \cdot \nabla \phi_j)\phi_i \, d\mathbf{x} \\ &= \int_{\partial\Omega} (\mathbf{b} \cdot \mathbf{n})\phi_i\phi_j \, ds - \int_\Omega (\mathbf{b} \cdot \nabla \phi_i)\phi_j \, d\mathbf{x} - \int_\Omega (\nabla \cdot \mathbf{b})\phi_i\phi_j \, d\mathbf{x}. \end{aligned} \tag{3.93}$$

Since ϕ_i and ϕ_j are basis functions associated with internal nodes, they vanish on the global boundary $\partial\Omega$, eliminating the first term. The solenoidal assumption eliminates the third term. Thus, the global convective matrix exhibits exact skew-symmetry:

$$a_{ij}^{\text{conv}} = -\int_\Omega (\mathbf{b} \cdot \nabla \phi_i)\phi_j \, d\mathbf{x} = -a_{ji}^{\text{conv}}. \tag{3.94}$$

The globally assembled convective contribution is sign-indefinite and does not provide any stabilizing effect. Therefore, the non-positivity of the off-diagonal entries cannot be expected from the convective term itself, but must result from a sufficiently strong diffusive contribution.

We now derive a sufficient criterion ensuring that the diffusive contribution is strong enough for the contribution of each element to the off-diagonal entries to be non-positive. For a single element K, the convective contribution is given by

$$a_{ij}^{\text{conv},K} = \int_K (\mathbf{b} \cdot \nabla \phi_j)\phi_i \, d\mathbf{x}. \tag{3.95}$$

To satisfy the condition $a_{ij} \leq 0$ for $i \neq j$, it is sufficient to require that the contribution of each element K to the corresponding off-diagonal entry be non-positive, that is,

$$a_{ij}^{\text{diff},K} + a_{ij}^{\text{conv},K} \leq 0, \qquad i \neq j. \tag{3.96}$$

The diffusive contribution must therefore be sufficiently strong to compensate for a possible positive convective contribution. A convenient sufficient condition is

$$-a_{ij}^{\text{diff},K} \geq \left| a_{ij}^{\text{conv},K} \right|. \tag{3.97}$$

To this end, we employ the following upper bound:

$$|a_{ij}^{\text{conv},K}| \leq \int_K |\mathbf{b} \cdot \nabla \phi_j||\phi_i| \, d\mathbf{x} \tag{3.98}$$

$$\leq \|\mathbf{b}\|_{L^\infty(K)} |\nabla \phi_j| \int_K \phi_i \, d\mathbf{x} \qquad (\phi_i \geq 0 \text{ on } K) \tag{3.99}$$

$$= \|\mathbf{b}\|_{L^\infty(K)} \left(\frac{|F_j^K|}{d|K|} \right) \left(\frac{|K|}{d+1} \right) \qquad (\text{using (3.86) and } \int_K \phi_i = \frac{|K|}{d+1}) \tag{3.100}$$

$$= \frac{\|\mathbf{b}\|_{L^\infty(K)} |F_j^K|}{d(d+1)}. \tag{3.101}$$

Substituting the estimate and using (3.88) into the stability condition (3.97) leads to:

$$\varepsilon \frac{|F_i^K|}{d|K|} \cos(\theta_{ij}^K) \geq \frac{\|\mathbf{b}\|_{L^\infty(K)}}{d+1}. \tag{3.102}$$

Recalling the geometric relationship for a simplex, the height corresponding to vertex x_i is given by $h_i^\perp = \frac{d|K|}{|F_i^K|}$. For a shape-regular family of meshes, $h_i^\perp$ and h_K are of the same order. Let us approximate the local mesh width by this characteristic height, i.e., $h_K \approx h_i^\perp$. Substituting $\frac{|F_i^K|}{d|K|} = \frac{1}{h_K}$ into the inequality and rearranging the terms to isolate the ratio of physical coefficients on the left-hand side, we obtain:

$$\frac{\|\mathbf{b}\|_{L^\infty(K)} h_K}{\varepsilon} \leq (d+1)\cos(\theta_{ij}^K). \tag{3.103}$$

The fraction on the left-hand side represents the ratio of convective transport to diffusive transport at the element level. This dimensionless quantity is of fundamental importance in the analysis of convection-diffusion problems and is formally defined as follows:

Definition 3.24 (Local Mesh Péclet Number)**.** *The local mesh Péclet number, denoted by Pe_K, is defined for each element $K \in \mathcal{T}_h$ as:*

$$Pe_K := \frac{\|\mathbf{b}\|_{L^\infty(K)} h_K}{\varepsilon}. \tag{3.104}$$

The problem is said to be in the convection-dominated regime locally if $Pe_K \gg 1$. If $Pe_K \leq 1$, then one says that locally diffusion dominates convection.

While the general local mesh Péclet number is often defined using the element diameter h_K, the rigorous stability analysis in Theorem 3.25 requires the specific geometric height $h_i^\perp$. For a shape-regular family of meshes, these quantities are comparable ($h_K \sim h_i^\perp$). The heuristic derivation above suggests that the stability of the numerical scheme depends critically on the local mesh Péclet number and the mesh geometry. This dependence is rigorously formalized by the following sufficient condition for matrices of non-negative type:

Theorem 3.25 (Sufficient Condition for Matrices of Non-negative Type)**.** *[BJK25, Theorem 8.5, Section 8.1] Let W_h be the Lagrange P_1 finite element space on a simplicial mesh $\mathcal{T}_h$ as defined in Definition 3.1. The global stiffness matrix A is of non-negative type, if for all elements $K \in \mathcal{T}_h$, the following local geometric condition holds:*

$$\frac{\|\mathbf{b}\|_{L^\infty(K)} h_i^\perp}{\varepsilon} \leq (d+1)\cos(\theta_{ij}^K), \tag{3.105}$$

where $h_i^\perp$ denotes the height of the simplex K corresponding to vertex x_i (i.e., the distance from vertex x_i to the opposite facet), and θ_{ij}^K is the dihedral angle opposite to the edge connecting vertices x_i and x_j.

Proof. The proof follows the same scaling arguments as presented in the derivation above. The detailed formal verification can be found in [BJK25, Theorem 8.5, Section 8.1]. □

Remark 3.26 (Fundamental Deficiency of the Standard Method)**.** *The rigorous stability condition (3.105) derived in Theorem 3.25 exposes two critical failure modes of the standard Galerkin method. Let us denote the term on the left-hand side as the local geometric Péclet number $Pe_{K,i} := \frac{\|\mathbf{b}\|_{L^\infty(K)} h_i^\perp}{\varepsilon}$.*

- ***Convection-Dominated Failure:*** *In the regime where convection dominates diffusion ($\varepsilon \ll \|\mathbf{b}\|_{L^\infty(K)}$) and the mesh is comparatively coarse, the local mesh Péclet number becomes excessively large ($Pe_{K,i} \gg 1$). Under these conditions, the standard Galerkin method fails to resolve the sharp layers, leading to non-physical oscillations. While stability could be theoretically maintained if one could afford a prohibitively fine mesh ($h \sim \varepsilon$) to resolve such layers, this resolution is generally computationally intractable in most practical applications.*

- ***Geometric Failure:*** *Even for moderate local mesh Péclet numbers, the stability condition becomes impossible to satisfy if the mesh angle θ_{ij}^K approaches $\pi/2$. This failure originates from the diffusion term, as the vanishing of the geometric bound $(d+1)\cos(\theta_{ij}^K)$ prevents the diffusive contribution from counteracting the convective part, regardless of the mesh size.*

Example 3.27. *Consider the 1D convection-diffusion problem:*

$$-\varepsilon u'' + u' = 1 \quad \text{in } \Omega = (0,1), \quad u(0) = u(1) = 0. \tag{3.106}$$

We use a uniform mesh with $N = 10$ ($h = 0.1$) and set $\varepsilon = 10^{-2}$. The resulting local mesh Péclet number is:

$$Pe_K = \frac{\|\mathbf{b}\|_{L^\infty(K)} h}{\varepsilon} = \frac{1 \cdot 0.1}{0.01} = 10.0. \tag{3.107}$$

As shown in Figure 3.1, this condition ($Pe_K = 10.0 \gg 1$) leads to significant spurious oscillations.

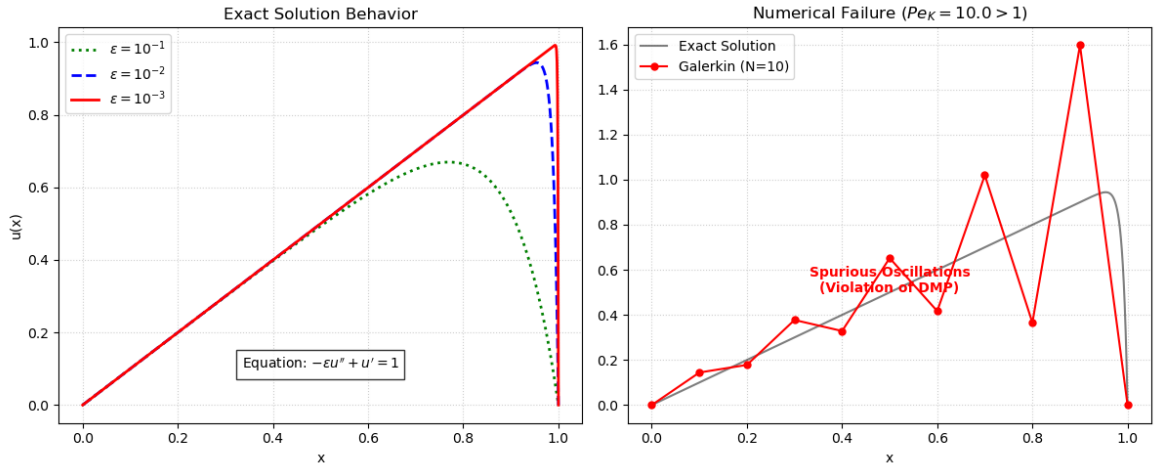


Figure 3.1.: Analysis of the convection-diffusion model problem $-\varepsilon u'' + u' = 1$. **(Left)** Behavior of the exact solution for varying $\varepsilon \in \{10^{-1}, 10^{-2}, 10^{-3}\}$, showing the formation of a sharp boundary layer. **(Right)** Numerical failure of the standard Galerkin method ($N = 10, \varepsilon = 10^{-2}$). The condition $Pe_K = 10.0 \gg 1$ leads to significant spurious oscillations, violating the Discrete Maximum Principle.

Consequently, for practical convection-dominated problems on meshes that do not satisfy the restrictive stability condition above, the stiffness matrix may acquire positive off-diagonal entries ($a_{ij} > 0$). This algebraic deficiency explains why the standard Galerkin method may fail to satisfy the Discrete Maximum Principle and motivates the development of stabilization techniques.

3.3. Tabata's Upwind Finite Element Method

To overcome the algebraic deficiencies of the standard Galerkin method discussed in Section 3.2.3, additional stabilization is required in order to recover the sign structure needed for a discrete maximum

principle. A classical linear approach is the upwind finite element method proposed by Tabata [Tab77], which was designed for unstructured simplicial meshes. The method modifies the discretization of the convective term by using an upwind simplex associated with each node. In addition, the reactive term is treated by mass lumping, so that no positive off-diagonal entries are introduced by the reaction contribution. In this sense, the construction combines finite element ideas with finite difference and finite volume type stabilization mechanisms.

The literature usually distinguishes between two early classes of steady-state upwind finite element schemes: methods based on upwind simplices [Tab77] and methods based on upwind fluxes on dual grids [BT81]. Both approaches can be designed to satisfy DMPs and typically exhibit first-order behavior in the $L^\infty(\Omega)$ norm; see [BJK25, Section 8.4]. In the present work, we focus on Tabata's upwind simplex method. This choice is motivated by the numerical comparison in [BJK25, Section 8.7], which indicates that, among the linear discretizations considered there, Tabata's method performs particularly well on Delaunay grids, especially with respect to convergence in the $H^1(\Omega)$ seminorm and accuracy in the $L^\infty(\Omega)$ norm. Moreover, the method fits naturally into the variational framework used throughout this thesis.

3.3.1. Dual Mesh and Mass Lumping

To stabilize the zero-th order reactive term and eliminate the generation of positive off-diagonal entries, the mass lumping technique is employed. This method approximates standard finite element integrals by evaluating the integrands exclusively at the mesh vertices, thereby producing a strictly diagonal mass matrix. Following the framework discussed in [Tab77], the geometric foundation for this numerical approximation involves the construction of a barycentric dual mesh, which partitions the domain into non-overlapping control volumes assigned to each node.

Let $\mathcal{N}_h = \{\boldsymbol{x}_i\}_{i=1}^n$ denote the set of vertices in the primary simplicial mesh $\mathcal{T}_h$. For each node $\boldsymbol{x}_i$, we define a macro-element ω_i as the union of all simplices sharing this vertex:

$$\omega_i = \bigcup \{K \in \mathcal{T}_h : \boldsymbol{x}_i \in K\}. \tag{3.108}$$

Within each simplex $K \subset \omega_i$, a specific polyhedral subset $S_K^{(i)}$ is uniquely determined and assigned to the node $\boldsymbol{x}_i$. To ensure a consistent topological closure, the vertices constituting this subset are defined as follows:

- the target vertex node $\boldsymbol{x}_i$ itself;
- the geometric barycenter of the simplex K;
- the midpoints of all edges of K incident to $\boldsymbol{x}_i$;
- the barycenters of all two-dimensional faces of K sharing the vertex $\boldsymbol{x}_i$, which is strictly required for three-dimensional domains where $d = 3$.

By this construction, the volume of the resulting subset satisfies $|S_K^{(i)}| = |K|/(d+1)$. The complete dual cell D_i is subsequently defined as the union of these subsets over the macro-element:

$$D_i = \bigcup_{K \subset \omega_i} S_K^{(i)}. \tag{3.109}$$

The measure of the global dual cell is then given by

$$|D_i| = \sum_{K \subset \omega_i} \frac{|K|}{d+1} = |\omega_i|/(d+1). \tag{3.110}$$

Associated with this dual partition, we introduce a space $\Psi_h = \operatorname{span}\{\psi_i\}_{i=1}^n$ consisting of piecewise constant functions $\psi_i(\boldsymbol{x})$ defined as:

$$\psi_i(\boldsymbol{x}) = \begin{cases} 1 & \text{if } \boldsymbol{x} \in D_i, \\ 0 & \text{otherwise,} \end{cases} \quad i = 1, \dots, n. \tag{3.111}$$

Utilizing these functions, the mass lumping operator $\mathcal{L}_h : C(\bar{\Omega}) \to \Psi_h$ is formally defined to map a given function with well-defined nodal values to its piecewise constant approximation over the dual mesh:

$$\mathcal{L}_h v = \sum_{i=1}^n v(\boldsymbol{x}_i)\psi_i. \tag{3.112}$$

Then

$$\mathcal{L}_h \phi_i = \psi_i. \tag{3.113}$$

This operator also induces a lumped inner product $(\cdot,\cdot)_h$. Since the dual cells are disjoint up to sets of measure zero, we have $(\psi_i, \psi_j)_{L^2} = |D_i|\delta_{ij}$. Hence, the lumped inner product simplifies to:

$$(f, g)_h = (\mathcal{L}_h f, \mathcal{L}_h g)_{L^2} = \sum_{i,j=1}^n f(\boldsymbol{x}_i)g(\boldsymbol{x}_j)(\psi_i, \psi_j)_{L^2} = \sum_{i=1}^n |D_i| f(\boldsymbol{x}_i)g(\boldsymbol{x}_i). \tag{3.114}$$

Applying this inner product to the standard finite element basis functions $\phi_i \in V_h$, the entries of the lumped mass matrix $M_L = (\tilde{m}_{ij})$ become:

$$\tilde{m}_{ij} = (\phi_j, \phi_i)_h = |D_i|\delta_{ij}. \tag{3.115}$$

This explicitly demonstrates that the lumped mass matrix is strictly diagonal, effectively eliminating any positive off-diagonal entries. The diagonal entries $\tilde{m}_{ii}$ can be rigorously derived through the following sequence of equalities:

$$\begin{aligned}
\tilde{m}_{ii} = |D_i| &= \sum_{K \subset \omega_i} \frac{|K|}{d+1} = \sum_{K \subset \omega_i} \int_K \phi_i(\mathbf{x})\, d\mathbf{x} && \left(\textstyle\int_K \phi_i = \frac{|K|}{d+1}\right) \\
&= \sum_{K \subset \omega_i} \int_K \left(\sum_{j=1}^n \phi_j(\mathbf{x})\right) \phi_i(\mathbf{x})\, d\mathbf{x} && \left(\text{using } \textstyle\sum_{j=1}^n \phi_j(\mathbf{x}) = 1 \text{ in Remark 3.2}\right) \\
&= \sum_{j=1}^n \sum_{K \subset \omega_i} \int_K \phi_j(\mathbf{x})\phi_i(\mathbf{x})\, d\mathbf{x} = \sum_{j=1}^n \int_{\omega_i} \phi_j(\mathbf{x})\phi_i(\mathbf{x})\, d\mathbf{x} \\
&= \sum_{j=1}^n \int_\Omega \phi_j(\mathbf{x})\phi_i(\mathbf{x})\, d\mathbf{x} && (\text{since } \operatorname{supp}\phi_i \subset \omega_i) \\
&= \sum_{j=1}^n (\phi_j, \phi_i)_{L^2} = \sum_{j=1}^n m_{ij},
\end{aligned}$$

where m_{ij} is the (i, j)-th entry of the consistent mass matrix $\mathbb{M}$ defined in Definition 3.19.

Remark 3.28. *For piecewise linear basis functions on a d-simplex K, the explicit values of the local entries m_{ij}^K are given by [VS18]:*

$$m_{ij}^K = (\phi_j, \phi_i)_K = \frac{|K|}{(d+1)(d+2)}(1 + \delta_{ij}), \tag{3.116}$$

where $(\cdot,\cdot)_K$ denotes the L^2-inner product restricted to the simplex K. This explicit formula confirms that $m_{ij}^K > 0$ for all $i, j = 1, \ldots, d+1$. Such positive off-diagonal entries directly compromise the M-matrix property of the global system, thereby necessitating the application of the mass lumping technique to ensure numerical stability and satisfy the discrete maximum principle. Consequently, from

$$\tilde{m}_{ii} = \sum_{j=1}^{n} m_{ij}, \tag{3.117}$$

the lumped mass matrix M_L can be efficiently assembled by directly summing the rows of the consistent mass matrix M, i.e., $\tilde{m}_{ii} = \sum_{j=1}^{n} m_{ij}$. This effectively bypasses the need for the explicit geometric construction of the dual mesh in practical numerical implementations.

Remark 3.29. *With the barycentric dual construction described above, the integration of the standard P_1 basis function ϕ_i and its lumped counterpart $\psi_i = \mathcal{L}_h\phi_i$ satisfies the following geometric and algebraic equivalence:*

$$\int_\Omega \phi_i(\mathbf{x})\,d\mathbf{x} = \sum_{K\subset\omega_i} \frac{|K|}{d+1} = |D_i| = \int_\Omega \psi_i(\mathbf{x})\,d\mathbf{x} = \int_\Omega \mathcal{L}_h\phi_i(\mathbf{x})\,d\mathbf{x}. \tag{3.118}$$

This fundamental identity seamlessly bridges the continuous Galerkin formulation and the lumped discrete formulation, serving as a crucial tool for the subsequent derivations of the stabilized algebraic system.

3.3.2. Tabata's Upwind Scheme and Discrete Maximum Principles

To achieve a monotone discretization of the convection term, we follow Tabata's approach [Tab77], identifying an upwind simplex for each node $\boldsymbol{x}_i$. Based on the treatment of the convection field $\boldsymbol{b}(\boldsymbol{x}_i) = (b_1(\boldsymbol{x}_i), \ldots, b_d(\boldsymbol{x}_i))^T$, this selection is performed either component-wise or along the full velocity vector.

In the component-wise approach, for each coordinate direction $k = 1, \ldots, d$, a simplex $K_{i,k}^{\mathrm{up}} \subset \omega_i$ containing $\boldsymbol{x}_i$ is chosen such that its interior intersects the upstream half-line along $-b_k(\boldsymbol{x}_i)\boldsymbol{e}_k$:

$$\{\boldsymbol{x} \in \Omega : \boldsymbol{x} = \boldsymbol{x}_i - t b_k(\boldsymbol{x}_i)\boldsymbol{e}_k,\ t > 0\} \cap \mathrm{int}(K_{i,k}^{\mathrm{up}}) \neq \emptyset, \tag{3.119}$$

where $\mathrm{int}(\cdot)$ denotes the simplex interior. Consequently, as illustrated in Figure 3.2, a single node $\boldsymbol{x}_i$ may require up to d distinct upwind simplices depending on grid alignment.

To avoid this grid-dependency, our study adopts a coordinate-independent formulation. A simplex $K_i^{\mathrm{up}} \subset \omega_i$ containing $\boldsymbol{x}_i$ is selected to intersect the upstream half-line of the full velocity vector:

$$\{\boldsymbol{x} \in \Omega : \boldsymbol{x} = \boldsymbol{x}_i - t\boldsymbol{b}(\boldsymbol{x}_i),\ t > 0\} \cap \mathrm{int}(K_i^{\mathrm{up}}) \neq \emptyset. \tag{3.120}$$

As illustrated in Figure 3.2, this criterion localizes the convection contribution to the upstream flow region.

If $-\boldsymbol{b}(\boldsymbol{x}_i)$ aligns with a shared element face, K_i^{up} is not unique; any simplex in ω_i satisfying (3.120) is admissible and preserves stability. If $\boldsymbol{b}(\boldsymbol{x}_i) = \boldsymbol{0}$, any simplex $K_i^{\mathrm{up}} \ni \boldsymbol{x}_i$ in ω_i can be used. These conventions apply analogously to the component-wise formulation.

The direction-based approach (3.120) is advantageous because it yields a coordinate-independent discretization. Furthermore, the component-wise criterion (3.119) exhibits a critical limitation at Neumann boundaries. Even if the vector $\boldsymbol{b}(\boldsymbol{x}_i)$ points outside Ω at a vertex $\boldsymbol{x}_i \in \Gamma_N$, the direction-based upwind simplex K_i^{up} remains well-defined. This is not guaranteed for the component-wise variant: a

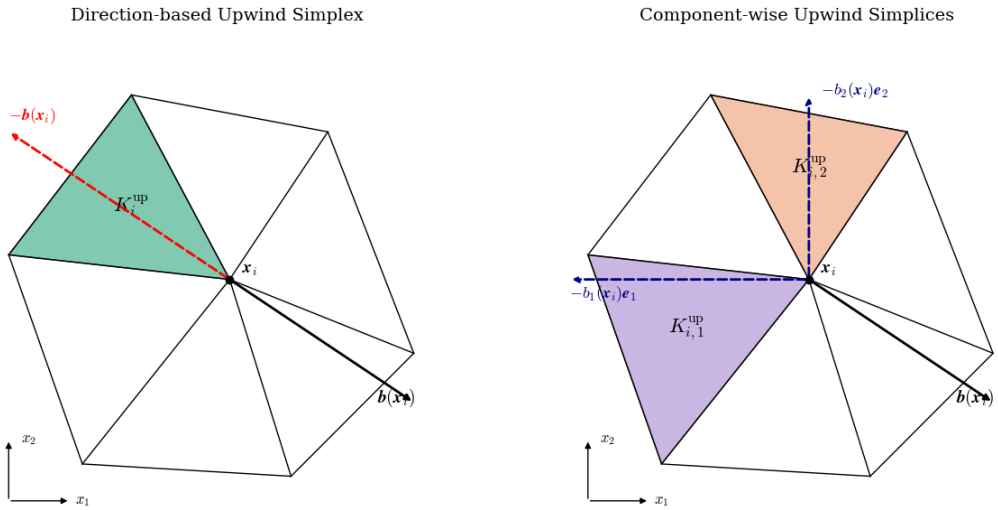


Figure 3.2.: Upwind triangles for methods from Tabata [Tab77], component-wise approach (right) and coordinate-independent approach (left)

line parallel to a coordinate axis passing through $\boldsymbol{x}_i \in \Gamma_N$ might not intersect Ω. Consequently, the corresponding component-wise upwind simplex $K_{i,k}^{\mathrm{up}}$ is undefined, even though $\boldsymbol{b}(\boldsymbol{x}_i)$ points outside Ω.

The construction of the upwind simplex K_i^{up} introduces artificial diffusion to ensure numerical stability. Furthermore, we explicitly assume that the reaction coefficient satisfies $c(\mathbf{x}) \geq 0$ almost everywhere in Ω. This non-negativity constraint is an algebraic necessity for the discrete problem to fulfill the maximum principles. Building upon the dual mesh D_i and the mass lumping operator $\mathcal{L}_h$ established in Section 3.3.1, we now derive the stabilized representations of the convective and reactive matrices defined in Definition 3.19.

To this end, we formulate the stabilized upwind finite element scheme as follows: Find $u_h \in W_h$ such that $u_h - u_{D,h} \in V_h$, and for all test functions $v_h \in V_h$,

$$\varepsilon(\nabla u_h, \nabla v_h)_{L^2} + (U_{\mathrm{up,Tab}}(u_h), \mathcal{L}_h v_h)_{L^2} + (cu_h, v_h)_{L^2} = (f, v_h)_{L^2}. \tag{3.121}$$

To assemble the corresponding algebraic linear system, we evaluate this variational problem by setting the test function $v_h = \phi_i$ for $i = 1, \ldots, m$.

For the convection term, $U_{\mathrm{up,Tab}}(u_h)$ is defined as a piecewise constant function on the dual mesh using the dual basis functions $\psi_i \in \Psi_h$:

$$U_{\mathrm{up,Tab}}(u_h)(\mathbf{x}) := \sum_{k=1}^{n} U_{\mathrm{up,Tab}}(u_h)(\boldsymbol{x}_k)\psi_k(\mathbf{x}), \tag{3.122}$$

where the nodal value is given by

$$U_{\mathrm{up,Tab}}(u_h)(\boldsymbol{x}_i) := \begin{cases} \boldsymbol{b}(\boldsymbol{x}_i) \cdot \nabla u_h\big|_{K_i^{\mathrm{up}}}, & \text{if } i = 1, \ldots, m, \\ 0, & \text{if } i = m+1, \ldots, n. \end{cases} \tag{3.123}$$

Since u_h is a P_1 finite element function, ∇u_h is constant on each element. Hence, the nodal value $U_{\mathrm{up,Tab}}(u_h)(\boldsymbol{x}_i)$ is well-defined, and the extension via $\{\psi_i\}$ yields a piecewise constant function on the dual cells D_i.

By utilizing this global piecewise constant definition, the stabilized convective inner product for the active test basis function ϕ_i $(i = 1, \ldots, m)$ evaluates explicitly to:

$$(U_{\text{up,Tab}}(u_h), \mathcal{L}_h\phi_i)_{L^2} = \int_\Omega \left(\sum_{k=1}^{n} U_{\text{up,Tab}}(u_h)(\boldsymbol{x}_k)\psi_k(\mathbf{x}) \right) \psi_i(\mathbf{x})\, d\mathbf{x} = |D_i| U_{\text{up,Tab}}(u_h)(\boldsymbol{x}_i), \tag{3.124}$$

where we use the property $\mathcal{L}_h\phi_i = \psi_i$ and the fact that the disjoint supports imply $\int_\Omega \psi_k\psi_i\, d\mathbf{x} = \delta_{ki}|D_i|$. Substituting the trial basis expansion $u_h = \sum_{j=1}^{n} u_j\phi_j$ into the evaluation (3.124), we obtain:

$$(U_{\text{up,Tab}}(u_h), \mathcal{L}_h\phi_i)_{L^2} = \sum_{j=1}^{n} u_j \left(|D_i| \boldsymbol{b}(\boldsymbol{x}_i) \cdot \nabla\phi_j\big|_{K_i^{\text{up}}} \right). \tag{3.125}$$

Consequently, the convection matrix $\mathbb{A}_c = (c_{ij}) \in \mathbb{R}^{m\times n}$ is given by

$$c_{ij} = \left(U_{\text{up,Tab}}(\phi_j),\ \mathcal{L}_h\phi_i\right)_{L^2} = \left(U_{\text{up,Tab}}(\phi_j),\ \psi_i\right)_{L^2} = |D_i| \left(\boldsymbol{b}(\boldsymbol{x}_i) \cdot \nabla\phi_j\big|_{K_i^{\text{up}}} \right), \tag{3.126}$$

Here, $i = 1, \ldots, m$ and $j = 1, \ldots, n$. By utilizing the geometric identity $|D_i| = \int_\Omega \phi_i\, d\mathbf{x}$ and Remark 3.29, any scalar constant evaluated at the node $\boldsymbol{x}_i$, such as the upwind derivative $U_{\text{up,Tab}}(\phi_j)(\boldsymbol{x}_i) \in \mathbb{R}$, satisfies:

$$c_{ij} = (U_{\text{up,Tab}}(\phi_j), \mathcal{L}_h\phi_i)_{L^2} = (U_{\text{up,Tab}}(\phi_j)(\boldsymbol{x}_i), \phi_i)_{L^2} \quad \text{for } i = 1, \ldots, m,\ j = 1, \ldots, n. \tag{3.127}$$

Similarly, the reaction term in (3.121) is approximated. To obtain a diagonal lumped reaction matrix, we employ the standard Lagrange nodal interpolation operator Π_h. For any continuous function $w \in C^0(\bar{\Omega})$, this operator projects onto the finite element space W_h and is defined as:

$$\Pi_h w(\mathbf{x}) := \sum_{k=1}^{n} w(\boldsymbol{x}_k)\phi_k(\mathbf{x}). \tag{3.128}$$

Applying Π_h exclusively to the product of the trial and test functions, we approximate the exact reaction inner product as:

$$\begin{aligned} (cu_h, \phi_i)_{L^2} &\approx \int_\Omega c(\mathbf{x})\Pi_h(u_h\phi_i)\, d\mathbf{x} \\ &= \int_\Omega c(\mathbf{x}) \left(\sum_{k=1}^{n} u_h(\boldsymbol{x}_k)\phi_i(\boldsymbol{x}_k)\phi_k(\mathbf{x}) \right) d\mathbf{x}. \end{aligned} \tag{3.129}$$

Due to the nodal property $\phi_i(\boldsymbol{x}_k) = \delta_{ik}$, only the term corresponding to $k = i$ remains non-zero in the summation. The approximated integral then simplifies directly to:

$$\int_\Omega c(\mathbf{x})\Pi_h(u_h\phi_i)\, d\mathbf{x} = u_h(\boldsymbol{x}_i) \int_\Omega c(\mathbf{x})\phi_i(\mathbf{x})\, d\mathbf{x} = u_h(\boldsymbol{x}_i)(c, \phi_i)_{L^2}, \quad \text{for } i = 1, \ldots, m. \tag{3.130}$$

Consequently, the modified reaction matrix $\mathbb{M}_r = (r_{ij}) \in \mathbb{R}^{m\times n}$ is diagonal in the sense that

$$(c\phi_j, \phi_i)_{L^2} \approx \phi_j(\boldsymbol{x}_i)(c, \phi_i)_{L^2} = \delta_{ij}(c, \phi_i)_{L^2} = r_{ij}, \qquad i = 1, \ldots, m,\ j = 1, \ldots, n. \tag{3.131}$$

For the source term f on the right-hand side, the entries of $\mathbf{f} = (f_i)$ are evaluated using the standard exact L^2 inner product:

$$f_i = (f, \phi_i)_{L^2} = \int_\Omega f(\mathbf{x})\phi_i(\mathbf{x})\, d\mathbf{x}, \quad \text{for } i = 1, \ldots, m. \tag{3.132}$$

Then the global variational problem translates into the following algebraic linear system:

$$(\mathbb{A}_d + \mathbb{A}_c + \mathbb{M}_r)\mathbf{u} = \mathbf{f}, \tag{3.133}$$

where to distinguish it from the standard Galerkin matrix A, we denote the matrix obtained from the Tabata upwind discretization by $\mathbb{A}$. Component-wise, the i-th equation of this stabilized global system, corresponding to the active node $\boldsymbol{x}_i$, can be explicitly expanded as:

$$\sum_{j=1}^{n} a_{ij}^d u_j + \sum_{j=1}^{n} \left(|D_i| \boldsymbol{b}(\boldsymbol{x}_i) \cdot \nabla \phi_j \big|_{K_i^{\mathrm{up}}} \right) u_j + (c, \phi_i)_{L^2} u_i = (f, \phi_i)_{L^2}, \quad \text{for } i = 1, \ldots, m, \tag{3.134}$$

where $a_{ij}^d := \varepsilon(\nabla\phi_j, \nabla\phi_i)_{L^2}$ as defined in Definition 3.19. Let $\mathbb{A} = \mathbb{A}_d + \mathbb{A}_c + \mathbb{M}_r$ denote the matrix of active equations. The individual entries a_{ij} of the matrix $\mathbb{A}$ are explicitly given by:

$$a_{ij} = \varepsilon(\nabla\phi_j, \nabla\phi_i)_{L^2} + (U_{\mathrm{up,Tab}}(\phi_j)(\boldsymbol{x}_i), \phi_i)_{L^2} + (c, \phi_i)_{L^2}\delta_{ij}, \quad \text{for } i = 1, \ldots, m,\ j = 1, \ldots, n. \tag{3.135}$$

To establish the M-matrix properties of the global system matrix A, which are essential for satisfying the DMP, we first analyze the convective matrix $\mathbb{A}_c$, whose entries are defined by:

$$c_{ij} = (U_{\mathrm{up,Tab}}(\phi_j)(\boldsymbol{x}_i), \phi_i)_{L^2}. \tag{3.136}$$

The following lemma demonstrates that Tabata's upwind discretization inherently guarantees the crucial sign and row sum conditions for this convective term.

Lemma 3.30. *[BJK25, Lemma 8.20, Section 8.4] The matrix $\mathbb{A}_c$ with entries $c_{ij} = (U_{up,Tab}(\phi_j)(\boldsymbol{x}_i), \phi_i)_{L^2}$ corresponding to the upwind discretization (3.135) of the convective term is of non-negative type and possesses zero row sums.*

Proof. The proof follows the argument in [BJK25, Lemma 8.20, Section 8.4], adapted to the notation used in this thesis. First, we demonstrate that the row sums are zero. Using the linearity of the upwind operator and the partition of unity property of the standard finite element basis functions outlined in Remark 3.2, the row sums of the convective matrix evaluate to:

$$\begin{aligned}
\sum_{j=1}^{n} c_{ij} &= \sum_{j=1}^{n} \left(U_{\mathrm{up,Tab}}(\phi_j)(\boldsymbol{x}_i) \int_\Omega \phi_i \, \mathrm{d}\boldsymbol{x} \right) \\
&= U_{\mathrm{up,Tab}}\Big(\sum_{j=1}^{n} \phi_j \Big)(\boldsymbol{x}_i) \int_\Omega \phi_i \, \mathrm{d}\boldsymbol{x} \\
&= U_{\mathrm{up,Tab}}(1)(\boldsymbol{x}_i) \int_\Omega \phi_i \, \mathrm{d}\boldsymbol{x} = 0, \quad \text{for } i = 1, \ldots, m,
\end{aligned}$$

which holds because the upwind derivative operator applied to a globally constant function identically vanishes.

Next, we prove that the off-diagonal entries are non-positive, i.e., $c_{ij} \leq 0$ for $i \neq j$. By definition, we have:

$$c_{ij} = U_{\mathrm{up,Tab}}(\phi_j)(\boldsymbol{x}_i) \int_\Omega \phi_i \, \mathrm{d}\boldsymbol{x}. \tag{3.137}$$

Since the standard P_1 basis function satisfies $\phi_i(\boldsymbol{x}) \geq 0$ for all $\boldsymbol{x} \in \Omega$ and is strictly positive in the interior of its support, we obtain $\int_\Omega \phi_i \, \mathrm{d}\boldsymbol{x} > 0$. Consequently, the sign of c_{ij} is uniquely determined by the scalar convective contribution $U_{\mathrm{up,Tab}}(\phi_j)(\boldsymbol{x}_i) = \boldsymbol{b}(\boldsymbol{x}_i) \cdot \nabla\phi_j\big|_{K_i^{\mathrm{up}}}$.

If $\boldsymbol{b}(\boldsymbol{x}_i) = \mathbf{0}$, then $U_{\mathrm{up,Tab}}(\phi_j)(\boldsymbol{x}_i) = 0 \leq 0$ holds trivially. Assuming $\boldsymbol{b}(\boldsymbol{x}_i) \neq \mathbf{0}$, the geometric definition of the direction-based upwind simplex K_i^{up} guarantees the existence of a distinct point

$\boldsymbol{y}_i \in K_i^{\text{up}}$ located on the upstream half-line. Thus, $\boldsymbol{y}_i = \boldsymbol{x}_i - \alpha_i \boldsymbol{b}(\boldsymbol{x}_i)$ for some scalar $\alpha_i > 0$. Rearranging this geometric relation yields:

$$\boldsymbol{b}(\boldsymbol{x}_i) = \frac{\boldsymbol{x}_i - \boldsymbol{y}_i}{\alpha_i}. \tag{3.138}$$

Considering the restriction of the global basis function ϕ_j to the simplex K_i^{up}, it behaves as a purely affine function, meaning its gradient $\nabla\phi_j|_{K_i^{\text{up}}}$ is a constant vector. By the definition of affine mappings, the directional derivative satisfies $\nabla\phi_j|_{K_i^{\text{up}}} \cdot (\boldsymbol{x}_i - \boldsymbol{y}_i) = \phi_j(\boldsymbol{x}_i) - \phi_j(\boldsymbol{y}_i)$. Substituting the representation of $\boldsymbol{b}(\boldsymbol{x}_i)$ into the convective term yields:

$$\begin{aligned} \boldsymbol{b}(\boldsymbol{x}_i) \cdot \nabla\phi_j\big|_{K_i^{\text{up}}} &= \left(\frac{\boldsymbol{x}_i - \boldsymbol{y}_i}{\alpha_i}\right) \cdot \nabla\phi_j\big|_{K_i^{\text{up}}} \\ &= \frac{1}{\alpha_i}\Big((\boldsymbol{x}_i - \boldsymbol{y}_i) \cdot \nabla\phi_j\big|_{K_i^{\text{up}}}\Big) \\ &= \frac{1}{\alpha_i}\big(\phi_j(\boldsymbol{x}_i) - \phi_j(\boldsymbol{y}_i)\big). \end{aligned} \tag{3.139}$$

For the off-diagonal entries ($j \neq i$), the Kronecker delta property of the nodal basis dictates that $\phi_j(\boldsymbol{x}_i) = 0$. Since $\boldsymbol{y}_i \in K_i^{\text{up}}$ and the P_1 basis functions are globally non-negative ($\phi_j \geq 0$), it immediately follows that:

$$U_{\text{up,Tab}}(\phi_j)(\boldsymbol{x}_i) = \frac{1}{\alpha_i}\big(0 - \phi_j(\boldsymbol{y}_i)\big) = -\frac{1}{\alpha_i}\phi_j(\boldsymbol{y}_i) \leq 0. \tag{3.140}$$

Therefore, we conclude that $c_{ij} \leq 0$ for all $i \neq j$. This completes the proof. □

Corollary 3.31 (Matrix of Non-negative Type)**.** *[BJK25, Corollary 8.21, Section 8.4] Consider the matrix $\mathbb{A} = (a_{ij}) \in \mathbb{R}^{m\times n}$ resulting from the discretization (3.135), incorporating mass lumping for the reactive term. If the simplicial triangulation $\mathcal{T}_h$ is of XZ-type (see Definition 3.21), then $\mathbb{A}$ is of non-negative type.*

Proof. The statement follows directly from the properties of the constituent matrices. According to Theorem 3.22, the standard Galerkin diffusion matrix $\mathbb{A}_d$ yields non-positive off-diagonal entries under the assumption of an XZ-type triangulation. Furthermore, Lemma 3.30 guarantees that the convective matrix $\mathbb{A}_c$ discretized via the Tabata upwind scheme is of non-negative type. Since the lumped mass matrix $\mathbb{M}_r$ for the reaction term is strictly diagonal, which means its off-diagonal entries are identically zero, their combination $\mathbb{A} = \mathbb{A}_d + \mathbb{A}_c + \mathbb{M}_r$ with $\varepsilon > 0$ strictly preserves the non-positivity of the off-diagonal entries. For a detailed rigorous proof, we refer the reader to [BJK25, Corollary 8.21]. □

Theorem 3.32 (Existence and Uniqueness of the Discrete Solution)**.** *[BJK25, Theorem 8.22, Section 8.4] Let the triangulation $\mathcal{T}_h$ be of XZ-type and assume that $c(\mathbf{x}) \geq 0$ in Ω. Then the discrete linear system resulting from the discretization (3.135) admits a unique solution.*

Proof. The existence and uniqueness of the discrete solution require the global system matrix A in (3.23) to be nonsingular. According to Lemma 3.3, A is nonsingular if and only if its principal submatrix $A^i \in \mathbb{R}^{m\times m}$, which is associated solely with the active degrees of freedom, is nonsingular. To prove the nonsingularity of A^i, we decompose it as $A^i = B^i + C^i$, where $B^i = \mathbb{A}_d^i$ represents the diffusion submatrix restricted to the active degrees of freedom, and $C^i = \mathbb{A}_c^i + \mathbb{M}_r^i$ denotes the corresponding combined convective and lumped reactive submatrix.

By Theorem 3.22, under the assumption of an XZ-type triangulation, B^i is a matrix of non-negative type. Furthermore, the coercivity of the diffusion operator over the active degrees of freedom ensures that B^i is strictly positive definite, and consequently nonsingular.

Based on our previous derivations (see Corollary 3.31 and Lemma 3.30), the upwind convective matrix and the lumped reactive matrix yield non-positive off-diagonal entries and non-negative row sums. Consequently, their restriction to the active degrees of freedom guarantees that the remainder matrix C^i is also of non-negative type.

We now invoke [BJK25, Theorem 3.9]: if B^i and C^i are square matrices of non-negative type and B^i is nonsingular, their sum $A^i = B^i + C^i$ is necessarily nonsingular. Since A^i is nonsingular, it directly follows from Lemma 3.3 that the global matrix A is nonsingular. Thus, the linear system $\mathbb{A}\underline{u} = \underline{f}$ is uniquely solvable. □

Remark 3.33. *The non-negativity of the reaction coefficient $c(\mathbf{x}) \geq 0$ is essential for the matrix C^i in the proof of Theorem 3.32 to be of non-negative type. For active nodes, the row sum of the submatrix $C^i = \mathbb{A}_c^i + \mathbb{M}_r^i$ is given by:*

$$\sum_{j=1}^{m}(C^i)_{ij} = \sum_{j=1}^{m}(\mathbb{A}_c^i)_{ij} + (\mathbb{M}_r)_{ii} = -\sum_{j=m+1}^{n}(\mathbb{A}_c)_{ij} + \int_\Omega c(\mathbf{x})\phi_i \,\mathrm{d}\mathbf{x}.$$

Since the upwind convective matrix satisfies $(\mathbb{A}_c)_{ij} \leq 0$ for $i \neq j$, we have the term $-\sum_{j=m+1}^{n}(\mathbb{A}_c)_{ij} \geq 0$. To ensure that the total row sum remains non-negative for all active nodes, particularly those not adjacent to the boundary where the convective contribution may vanish such that $-\sum_{j=m+1}^{n}(\mathbb{A}_c)_{ij} = 0$, the condition $c(\mathbf{x}) \geq 0$ almost everywhere in Ω is required.

Theorem 3.34 (Local and Global DMPs)**.** *[BJK25, Theorem 8.23, Section 8.4] Let the triangulation $\mathcal{T}_h$ be of XZ-type and assume that $c(\mathbf{x}) \geq 0$ in Ω. Then the solutions of the upwind finite element problem satisfy the local and global DMPs.*

Proof. The discrete problem corresponding to the active degrees of freedom can be expressed using the block matrices defined in (3.23) as $A^i\underline{u}^i + A^b\underline{u}^b = \underline{f}^i$. According to Corollary 3.31, the horizontal block (A^i, A^b) is of non-negative type, inherently ensuring that its principal submatrix A^i is also of non-negative type. Furthermore, Theorem 3.32 establishes that A^i is nonsingular. Consequently, following [BJK25, Lemma 3.45], the nonsingular non-negative type matrix A^i is rigorously identified as an M-matrix. Finally, given that the global discrete matrix A satisfies condition (i) of Definition 3.16 and is nonsingular, the combined application of Lemma 3.17 and Theorem 3.32 extends this property, confirming that the full matrix A is also a nonsingular M-matrix, which inherently possesses strictly positive diagonal entries. Therefore, the local DMPs follow from Theorem 3.6, and the global DMPs follow from Theorem 3.8. □

In summary, under the assumptions discussed above, the Tabata upwind finite element method [Tab77] satisfies both the local and the global DMPs and thus provides a stable, bound-preserving linear approximation with strongly reduced spurious oscillations. According to the numerical study in [BJK25, Section 8.7], it is among the most accurate linear discretizations considered there, which makes it a natural choice for initializing the nonlinear iterations in the later numerical experiments. Although the method can resolve sharp layers reasonably well on suitably flow-aligned meshes [BJK25, Section 8.4], its built-in artificial diffusion may still smear steep gradients on more general unaligned meshes. Therefore, in the present work, the Tabata solution is used primarily as a robust DMP-preserving initial iterate for the subsequent nonlinear methods.

4. Nonlinear Finite Element Methods

As shown in Chapter 3, linear upwind discretizations such as Tabata's method [Tab77] can enforce discrete maximum principles by adding suitable artificial diffusion. Although such schemes provide stable and non-oscillatory approximations, the added diffusion may smear sharp layers and reduce accuracy on general meshes. This motivates nonlinear stabilization techniques, whose aim is to combine the robustness of monotone low-order methods with the higher resolution of less diffusive Galerkin-type discretizations.

Classical stabilization methods such as the streamline-upwind Petrov–Galerkin (SUPG) method [BH82] improve the robustness of standard Galerkin finite element approximations, but they do not generally guarantee the DMP. Spurious oscillations at layers diminishing (SOLD) methods were developed to further reduce localized oscillations; however, many classical SOLD approaches still lack a rigorous DMP guarantee [JK07, JK08].

This chapter therefore focuses on algebraic flux correction (AFC) and related nonlinear algebraic stabilization techniques. The AFC methodology, rooted in flux-corrected transport [Zal79] and developed in the finite element context by Kuzmin [Kuz07], starts from a sufficiently diffusive low-order discretization and adds limited anti-diffusive fluxes in a solution-dependent way. We first introduce the abstract AFC framework following the matrix-based formulation in [BJK25, Chapter 10]. We then discuss the classical Kuzmin limiter and finally present the monotone upwind-type algebraically stabilized (MUAS) method, which modifies the stabilization matrix directly and yields a DMP-preserving nonlinear scheme on general admissible meshes [Kno21, JK22].

4.1. Algebraic Flux Correction (AFC) Scheme

Throughout this section, we restrict ourselves to the AFC formulation for conforming P_1 finite elements on simplicial meshes. As shown in Section 3.2, the Galerkin finite element discretization leads to a linear algebraic system of the form

$$\sum_{j=1}^{n} a_{ij}u_j = l_i, \qquad i = 1, \dots, m, \tag{4.1}$$

$$u_i = u_i^b, \qquad i = m+1, \dots, n. \tag{4.2}$$

To construct an algebraically stabilized scheme, it is convenient to work with an $n \times n$ matrix $A = (a_{ij})_{i,j=1}^{n}$. In accordance with the discrete formulation (3.10), we define

$$a_{ij} := a(\phi_j, \phi_i), \qquad i, j = 1, \dots, n. \tag{4.3}$$

Under the assumption $c \geq 0$ and Remark 3.2, the row sums for $i = 1, \dots, n$ satisfy

$$\begin{aligned}\sum_{j=1}^{n} a_{ij} &= \sum_{j=1}^{n} a(\phi_j, \phi_i) = a\left(\sum_{j=1}^{n} \phi_j, \phi_i\right) = a(1, \phi_i) \\ &= \int_\Omega \varepsilon \nabla 1 \cdot \nabla \phi_i \, d\mathbf{x} + \int_\Omega (\mathbf{b} \cdot \nabla 1)\phi_i \, d\mathbf{x} + \int_\Omega c \cdot 1 \cdot \phi_i \, d\mathbf{x} \\ &= \int_\Omega c\phi_i \, d\mathbf{x} \geq 0.\end{aligned}$$

To enforce the required monotonicity on this system, we introduce a symmetric artificial diffusion matrix $D = (d_{ij})_{i,j=1}^{n}$. The entries of D are constructed explicitly from the entries of A defined in (4.3) to eliminate any destabilizing positive off-diagonal contributions:

$$d_{ij} = d_{ji} = -\max\{a_{ij}, 0, a_{ji}\} \quad \forall i \neq j, \tag{4.4}$$

and the diagonal entries are defined to ensure the row sum conservation property:

$$d_{ii} = -\sum_{j \neq i} d_{ij}. \tag{4.5}$$

Based on the construction in (4.4) and (4.5), the positive semidefiniteness of the artificial diffusion matrix D can be explicitly verified. For any arbitrary vector $\underline{v} \in \mathbb{R}^n$, the quadratic form associated with D satisfies:

$$\begin{aligned}
\langle \underline{v}, D\underline{v} \rangle &= \sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij} v_i v_j \\
&= \sum_{i=1}^{n} d_{ii} v_i^2 + \sum_{i=1}^{n} \sum_{j \neq i} d_{ij} v_i v_j \\
&= \sum_{i=1}^{n} \left(-\sum_{j \neq i} d_{ij} \right) v_i^2 + \sum_{i=1}^{n} \sum_{j \neq i} d_{ij} v_i v_j \quad \text{(using (4.5))} \\
&= \sum_{i=1}^{n} \sum_{j \neq i} d_{ij} (v_i v_j - v_i^2) \\
&= \frac{1}{2} \sum_{i=1}^{n} \sum_{j \neq i} d_{ij} (2 v_i v_j - v_i^2 - v_j^2) \quad \text{(by symmetry } d_{ij} = d_{ji}\text{)} \\
&= \frac{1}{2} \sum_{i=1}^{n} \sum_{j \neq i} \underbrace{(-d_{ij})}_{\geq 0} \underbrace{(v_i - v_j)^2}_{\geq 0} \\
&\geq 0.
\end{aligned} \tag{4.6}$$

Therefore, D is symmetric positive semidefinite. Consequently, since the original inner stiffness matrix A^i is positive definite (see (3.28)), the modified active matrix $(A + D)^i$ remains positive definite, ensuring the unique solvability of the stabilized algebraic system:

$$\sum_{j=1}^{n} (a_{ij} + d_{ij}) u_j = l_i, \quad i = 1, \ldots, m, \tag{4.7}$$

$$u_i = u_i^b, \quad i = m+1, \ldots, n. \tag{4.8}$$

Crucially, the stabilized operator $\bar{A} = A + D$ satisfies the conditions of a matrix of non-negative type. First, the off-diagonal entries are non-positive:

$$\bar{a}_{ij} = a_{ij} + d_{ij} = a_{ij} - \max\{a_{ij}, 0, a_{ji}\} \leq 0, \quad \forall i \neq j. \tag{4.9}$$

Second, due to the zero row sum property of D, the row sums of the stabilized matrix inherit the non-negativity of the original system:

$$\sum_{j=1}^{n} \bar{a}_{ij} = \sum_{j=1}^{n} a_{ij} + \sum_{j=1}^{n} d_{ij} = \sum_{j=1}^{n} a_{ij} \geq 0, \quad \forall i = 1, \ldots, m. \tag{4.10}$$

Moreover, since $(A + D)^i$ is positive definite, its diagonal entries corresponding to the active degrees of freedom are strictly positive. Therefore, it follows directly from Theorem 3.6 that the solution of the stabilized system (4.7)–(4.8) satisfies local discrete maximum principles.

Remark 4.1. *The preceding arguments show that the low-order AFC matrix $\bar{A} = A + D$ is of non-negative type and that its active block $\bar{A}^i = (A + D)^i$ is nonsingular. Hence, by the characterization of nonsingular matrices of non-negative type in [BJK25, Lemma 3.45], the matrix $\bar{A}^i$ is an M-matrix. In particular, $(\bar{A}^i)^{-1} \geq 0$.*

Moreover, the corresponding full low-order matrix $\bar{A} = A + D$ has the block form

$$\bar{A} = \begin{pmatrix} \bar{A}^i & \bar{A}^b \\ 0 & I \end{pmatrix}. \tag{4.11}$$

This is the block form considered in Lemma 3.17. Hence, since $\bar{A}^i$ is an M-matrix, the full matrix $\bar{A}$ is an M-matrix as well.

To define the flux correction, the original high-order system $A\underline{u} = \underline{l}$ is reformulated using the stabilized low-order operator $\bar{A} = A + D$ as follows:

$$(\bar{A}\underline{u})_i = l_i + (D\underline{u})_i, \quad i = 1, \ldots, m. \tag{4.12}$$

Using $\sum_{j=1}^{n} d_{ij} = 0$, the term $(D\underline{u})_i$ can be written as a sum of anti-diffusive fluxes:

$$(D\underline{u})_i = \sum_{j \neq i} f_{ij}, \qquad \text{with} \qquad f_{ij} = d_{ij}(u_j - u_i). \tag{4.13}$$

Since D is symmetric, these fluxes are antisymmetric, that is,

$$f_{ij} = -f_{ji}, \qquad \forall i, j. \tag{4.14}$$

To prevent the reappearance of spurious oscillations, the AFC scheme limits these anti-diffusive fluxes by introducing solution-dependent correction factors $\alpha_{ij} \in [0, 1]$. These coefficients are chosen in such a way that the anti-diffusive fluxes are retained as much as possible in smooth regions, while they are reduced near steep gradients or local extrema where oscillations may occur. The limited low-order system then takes the form

$$(\bar{A}\underline{u})_i = l_i + \sum_{j \neq i} \alpha_{ij} f_{ij}, \quad i = 1, \ldots, m. \tag{4.15}$$

For conservation, the correction factors are required to be symmetric:

$$\alpha_{ij} = \alpha_{ji}, \quad \forall i, j. \tag{4.16}$$

Substituting $\bar{A} = A + D$ into (4.15) and using the definition of f_{ij} (4.13) yields the nonlinear algebraic AFC system

$$\sum_{j=1}^{n} a_{ij} u_j + \sum_{j \neq i} (1 - \alpha_{ij}(\underline{u}))\, d_{ij}(u_j - u_i) = l_i, \qquad i = 1, \ldots, m, \tag{4.17}$$

$$u_i = u_i^b, \qquad i = m + 1, \ldots, n. \tag{4.18}$$

Here, the coefficients α_{ij} determine the amount of anti-diffusive correction admitted along each edge of the mesh, and their precise definition depends on the choice of the limiter.

Assumption 4.2 (Continuity assumption for the AFC scheme)**.** *For every $i = 1, \dots, m$ and every $j \in S_i$, the function $\alpha_{ij}(\underline{u})(u_j - u_i)$ is continuous with respect to $\underline{u} = (u_1, \dots, u_n)^T \in \mathbb{R}^n$.*

Under Assumption 4.2, the nonlinear AFC problem (4.17)–(4.18) admits at least one solution; see [BJK25, Theorem 10.1, Section 10.1].

To cast the AFC scheme into the abstract nonlinear stabilization framework, we define the matrix $B(\underline{u}) = (b_{ij}(\underline{u}))_{i,j=1}^n$ by

$$b_{ij}(\underline{u}) = (1 - \alpha_{ij}(\underline{u}))\, d_{ij}, \qquad i \neq j, \tag{4.19}$$

$$b_{ii}(\underline{u}) = -\sum_{j \neq i} b_{ij}(\underline{u}). \tag{4.20}$$

With this definition, the nonlinear stabilization term can be written in matrix form, and the matrix $B(\underline{u})$ satisfies the structural properties

$$b_{ij}(\underline{u}) = b_{ji}(\underline{u}), \qquad i, j = 1, \dots, n, \tag{4.21}$$

$$b_{ij}(\underline{u}) \leq 0, \qquad i \neq j, \tag{4.22}$$

$$\sum_{j=1}^n b_{ij}(\underline{u}) = 0, \qquad i = 1, \dots, n. \tag{4.23}$$

These properties follow directly from the symmetry condition (4.16), the symmetry of the artificial diffusion matrix $D = (d_{ij})$, the bound $0 \leq \alpha_{ij} \leq 1$, and the definition (4.20). Indeed, arguing as in (4.6), one obtains for any $\underline{v} \in \mathbb{R}^n$ that

$$\underline{v}^T B(\underline{u})\underline{v} = \frac{1}{2} \sum_{i=1}^n \sum_{j \neq i} \big(-b_{ij}(\underline{u})\big)(v_i - v_j)^2 \geq 0. \tag{4.24}$$

Hence, the principal submatrix $B^i(\underline{u}) = \big(b_{ij}(\underline{u})\big)_{i,j=1}^m$ is also positive semidefinite. Since the active block $A^i = \big(a_{ij}\big)_{i,j=1}^m$ is positive definite, it follows that the matrix

$$A^i + B^i(\underline{u}) = \big(a_{ij} + b_{ij}(\underline{u})\big)_{i,j=1}^m \tag{4.25}$$

is positive definite for every fixed $\underline{u} \in \mathbb{R}^n$. Consequently, the nonlinear AFC system (4.17)–(4.18) can be rewritten in the abstract form

$$\sum_{j=1}^n \big(a_{ij} + b_{ij}(\underline{u})\big) u_j = l_i, \qquad i = 1, \dots, m, \tag{4.26}$$

$$u_i = u_i^b, \qquad i = m+1, \dots, n. \tag{4.27}$$

Remark 4.3 (Role of symmetry in the nonlinear stabilization)**.** *The symmetry condition (4.16) plays a central role in the classical AFC construction for three reasons.*

First, together with the antisymmetry of the fluxes in (4.14), it guarantees that the correction term in (4.15) is conservative.

Second, owing to (4.16), (4.19), and (4.20), the matrix $B(\underline{u})$ satisfies the structural properties (4.21)–(4.23). More precisely, the essential property is the symmetry of the stabilization matrix, namely $b_{ij}(\underline{u}) = b_{ji}(\underline{u})$. In the AFC construction, this follows from $d_{ij} = d_{ji}$ and $\alpha_{ij} = \alpha_{ji}$. Hence, by (4.24), $B(\underline{u})$ is symmetric positive semidefinite for every fixed $\underline{u} \in \mathbb{R}^n$, so that the nonlinear stabilization contributes in a genuinely stabilizing way.

Finally, the symmetry of the nonlinear stabilization is also important for solvability. As shown in [BJK25, Remark 10.3, Section 10.1], without the appropriate symmetry of $B(\underline{u})$, the nonlinear algebraic problem (4.17)–(4.18) is, in general, not solvable.

Hence, the AFC scheme fits into the abstract framework of nonlinear discrete problems with matrix-based nonlinearity; see [BJK25, Section 3.2]. Under Assumption 4.2, the resulting nonlinear AFC problem admits at least one solution. Moreover, sufficient conditions for the validity of the local and global DMPs are given in [BJK15].

4.2. Design of the Kuzmin Limiter

To complete the AFC construction, it remains to specify the limiter coefficients $\alpha_{ij} \in [0,1]$. In this work, we employ the classical Kuzmin limiter [Kuz07], whose formulation is rooted in the FCT methodology of Zalesak [Zal79]. The limiter is defined entirely at the algebraic level.

For every active node $i = 1, \dots, m$, we first introduce the aggregated incoming positive and negative anti-diffusive fluxes, together with the corresponding admissible bounds:

$$P_i^+ = \sum_{\substack{j \in S_i \\ a_{ji} \le a_{ij}}} f_{ij}^+, \qquad P_i^- = \sum_{\substack{j \in S_i \\ a_{ji} \le a_{ij}}} f_{ij}^-, \tag{4.28}$$

$$Q_i^+ = -\sum_{j \in S_i} f_{ij}^-, \qquad Q_i^- = -\sum_{j \in S_i} f_{ij}^+, \tag{4.29}$$

where S_i is given by (3.9), and

$$f_{ij}^+ := \max\{0, f_{ij}\}, \qquad f_{ij}^- := \min\{0, f_{ij}\}. \tag{4.30}$$

The condition $a_{ji} \le a_{ij}$ selects one orientation on each edge and thereby prevents the same flux from being restricted twice in conflicting ways.

The nodal limiting factors are then defined by comparing the admissible amounts $Q_i^\pm$ with the proposed flux sums $P_i^\pm$. For $i = 1, \dots, m$, we set

$$R_i^+ = \min\left\{1, \frac{Q_i^+}{P_i^+}\right\}, \qquad R_i^- = \min\left\{1, \frac{Q_i^-}{P_i^-}\right\}. \tag{4.31}$$

Whenever $P_i^+ = 0$ or $P_i^- = 0$, the corresponding factor is defined by $R_i^+ = 1$ or $R_i^- = 1$. For Dirichlet nodes, the boundary values are prescribed and therefore no further nodal limitation is required. We therefore choose

$$R_i^+ = 1, \qquad R_i^- = 1, \qquad i = m+1, \dots, n. \tag{4.32}$$

These nodal quantities induce preliminary edge factors $\tilde{\alpha}_{ij}$, defined according to the sign of the flux:

$$\tilde{\alpha}_{ij} = \begin{cases} R_i^+, & \text{if } f_{ij} > 0, \\ 1, & \text{if } f_{ij} = 0, \\ R_i^-, & \text{if } f_{ij} < 0. \end{cases} \tag{4.33}$$

In the final step, the limiter must be symmetrized in order to preserve conservation. Using again the orientation $a_{ji} \le a_{ij}$, we define

$$\alpha_{ij} = \alpha_{ji} = \tilde{\alpha}_{ij} \qquad \text{whenever } a_{ji} \le a_{ij}, \quad i, j = 1, \dots, n. \tag{4.34}$$

By construction, the coefficients satisfy

$$0 \le \alpha_{ij} \le 1, \qquad \alpha_{ij} = \alpha_{ji}, \quad i, j = 1, \dots, n. \tag{4.35}$$

Remark 4.4 (Technical conventions in the Kuzmin limiter)**.** *Several technical ambiguities in the above definition are harmless and do not affect the resulting AFC scheme. First, if $j \notin S_i$, then the corresponding flux satisfies $f_{ij} = 0$. Therefore, enlarging the index set in the sums defining $P_i^{\pm}$ and $Q_i^{\pm}$ does not change their values. Second, if one of the denominators in (4.31) vanishes, the corresponding factor may be prescribed arbitrarily. In this work, we choose*

$$R_i^+ = 1 \qquad \text{if } P_i^+ = 0, \qquad R_i^- = 1 \qquad \text{if } P_i^- = 0. \tag{4.36}$$

This convention does not affect the resulting limiter, since the corresponding factor is not used for any nonzero flux contribution.

Finally, if $a_{ij} = a_{ji}$, the orientation condition in (4.34) is not unique. This ambiguity is irrelevant if $a_{ij} = a_{ji} \leq 0$, because then $d_{ij} = 0$ and the corresponding flux contribution vanishes. If $a_{ij} = a_{ji} > 0$, the sign condition (4.37) fails, and the DMP result in Corollary 4.6 is not applicable.

Theorem 4.5 (Continuity property of the Kuzmin limiter)**.** *The Kuzmin limiter defined by (4.28)–(4.34) satisfies Assumption 4.2.*

Proof. See [BJK25, Theorem 10.26, Section 10.2]. □

Corollary 4.6 (The DMP for the AFC scheme with the Kuzmin limiter)**.** *If*

$$\min\{a_{ij}, a_{ji}\} \leq 0 \qquad \text{for all } i = 1, \ldots, m,\ j = 1, \ldots, n,\ i \neq j, \tag{4.37}$$

then the AFC scheme (4.26)–(4.27), equipped with the Kuzmin limiter, satisfies the local and global DMPs.

Proof. This corollary is based on [BJK25, Theorem 10.22]. The AFC scheme with the Kuzmin limiter is shown to satisfy the matrix-based DMP properties for strict and nonstrict extrema in the sense of [BJK25, Definitions 3.71 and 3.75]. Combined with the general abstract results for nonlinear discrete problems with matrix-based nonlinearity in [BJK25, Section 3.2], this yields the asserted local and global DMPs. □

Remark 4.7 (Mesh-dependent upwind character of the Kuzmin limiter)**.** *The directional choice in the Kuzmin limiter is governed by the inequality $a_{ji} < a_{ij}$. On many regular meshes, this criterion leads to a limiter determined by quantities associated with the upwind vertex of the edge; see [BJK25, Remark 10.21]. However, this upwind character is not guaranteed on distorted meshes. This limitation motivates the modified algebraically stabilized construction introduced in the next section.*

4.3. Monotone Upwind-Type Algebraically Stabilized (MUAS) Method

As discussed in Section 4.2, the classical Kuzmin limiter does not ensure DMPs on arbitrary meshes. The essential difficulty lies in the final symmetrization step (4.34), which may destroy the upwind character of the method; see Remark 4.7. This motivates a modified upwind-type algebraic stabilization [Kno21].

The presentation in this section follows the matrix-based formulation of the MUAS method in [BJK25, Section 10.5], which is based on the construction of John and Knobloch [JK22], adapted here to the notation and structure used in this thesis. The MUAS approach is based on a modified definition of the nonlinear stabilization matrix $B(\underline{u})$ in (4.26)–(4.27). In this way, the required monotonicity properties are enforced directly at the matrix level, leading to a DMP-preserving method on general meshes.

We now introduce the corresponding stabilization matrix and the resulting nonlinear discrete problem. Using (4.19), the definition of the artificial diffusion matrix (4.4), and the symmetry condition (4.16), the off-diagonal entries of the AFC stabilization matrix can be written in the equivalent form

$$b_{ij}(\underline{u}) = -\max\{(1-\alpha_{ij}(\underline{u}))a_{ij},\ 0,\ (1-\alpha_{ji}(\underline{u}))a_{ji}\}, \qquad i \neq j. \tag{4.38}$$

This formulation shows that the relevant object in the AFC analysis is the matrix $B(\underline{u})$. Indeed, the AFC scheme is written in the abstract form (4.26)–(4.27), and the corresponding structural conditions are imposed on $B(\underline{u})$ through (4.21)–(4.23). In the classical AFC construction, these properties are enforced by the symmetry condition (4.16), equivalently by the symmetrization step (4.34). In the MUAS approach, by contrast, the matrix $B(\underline{u})$ is defined directly so that the required structure is imposed at the matrix level. This motivates the introduction of a new family of solution-dependent coefficients

$$\beta_{ij} : \mathbb{R}^n \to [0,1], \qquad i,j = 1,\dots,n, \tag{4.39}$$

and the direct definition of the nonlinear stabilization matrix by

$$b_{ij}(\underline{u}) = -\max\{\beta_{ij}(\underline{u})a_{ij},\ 0,\ \beta_{ji}(\underline{u})a_{ji}\}, \qquad i \neq j, \tag{4.40}$$

$$b_{ii}(\underline{u}) = -\sum_{j\neq i} b_{ij}(\underline{u}), \qquad i = 1,\dots,n. \tag{4.41}$$

For pairs of Dirichlet boundary nodes, we set

$$\beta_{ij}(\underline{u}) = 0, \qquad i,j = m+1,\dots,n. \tag{4.42}$$

Moreover, if the coefficients β_{ij} are chosen symmetrically and satisfy

$$\beta_{ij}(\underline{u}) = 1-\alpha_{ij}(\underline{u}), \tag{4.43}$$

then (4.40) coincides with (4.19). Thus, the standard AFC stabilization is recovered as a special case, and the MUAS construction may be viewed as a matrix-based generalization of the AFC framework.

Accordingly, the MUAS scheme is defined by the nonlinear discrete problem

$$\sum_{j=1}^{n}\big(a_{ij}+b_{ij}(\underline{u})\big)u_j = l_i, \qquad i = 1,\dots,m, \tag{4.44}$$

$$u_i = u_i^b, \qquad i = m+1,\dots,n. \tag{4.45}$$

In particular, (4.44)–(4.45) has the same abstract form as (4.26)–(4.27), but with the nonlinear stabilization matrix defined by (4.40)–(4.41).

Theorem 4.8 (Solvability of the MUAS scheme)**.** *[BJK25, Theorem 10.47] Let the coefficients $\beta_{ij} : \mathbb{R}^n \to [0,1]$ be such that, for every $i,j = 1,\dots,n$ with $a_{ij} > 0$, the function*

$$\beta_{ij}(\underline{u})(u_j - u_i) \tag{4.46}$$

is continuous with respect to $\underline{u} = (u_1,\dots,u_n)^T \in \mathbb{R}^n$. Then the nonlinear discrete problem (4.44)–(4.45), with the matrix $B(\underline{u})$ defined by (4.40)–(4.41), admits at least one solution.

Proof. Using the row sum property (4.41), one obtains

$$\sum_{j=1}^{n} b_{ij}(\underline{u})u_j = \sum_{j=1}^{n} b_{ij}(\underline{u})(u_j - u_i), \qquad i = 1,\dots,m. \tag{4.47}$$

Hence, it suffices to verify the continuity of the mappings

$$\underline{u} \mapsto b_{ij}(\underline{u})(u_j - u_i), \qquad i = 1, \dots, m, \quad j = 1, \dots, n. \tag{4.48}$$

This is proved exactly as in [BJK25, Theorem 10.47], using the stated continuity assumption on

$$\beta_{ij}(\underline{u})(u_j - u_i) \tag{4.49}$$

together with the definition (4.40)–(4.41) of the entries $b_{ij}(\underline{u})$. Consequently, for every $i = 1, \dots, m$, the mapping

$$\underline{u} \mapsto \sum_{j=1}^{n} b_{ij}(\underline{u})u_j \tag{4.50}$$

is continuous. Therefore, the abstract solvability theory for nonlinear discrete problems with matrix-based nonlinearity [BJK25, Section 3.2] applies and yields the existence of at least one solution. □

Remark 4.9 (Relation to AFC under the sign condition)**.** *Several important observations follow from [BJK25, Section 10.5]. If the sign condition* (4.37) *holds, then the MUAS stabilization matrix defined by* (4.40)–(4.41) *can be rewritten in the AFC form* (4.19)–(4.20)*, with the artificial diffusion matrix $D = (d_{ij})$ given by* (4.4)–(4.5)*. Hence, the MUAS scheme* (4.44)–(4.45) *belongs to the same abstract matrix-based framework as the AFC scheme* (4.26)–(4.27)*. This observation clarifies the relation between the MUAS construction and the classical AFC framework. Furthermore, the main advantage of the MUAS construction is that it can be designed to satisfy DMP properties even when the condition* (4.37) *is not fulfilled.*

As discussed above, the definition (4.40)–(4.41) makes it possible to avoid the explicit symmetrization step (4.34) in the classical Kuzmin limiter. Nevertheless, as observed in [JK22, Remark 8], the MUAS coefficients β_{ij} cannot in general be defined directly by the preliminary correction factors $\tilde{\alpha}_{ij}$ from (4.33) if one seeks a well-posed scheme. For this reason, the nodal quantities underlying the Kuzmin limiter have to be reformulated first. The next lemma provides the corresponding identities.

Lemma 4.10 (Reformulation of the nodal quantities in the Kuzmin limiter)**.** *[BJK25, Lemma 10.50] Assume that the sign condition* (4.37) *holds. Then, for every $i = 1, \dots, m$, the quantities $P_i^{\pm}$ and $Q_i^{\pm}$ defined in* (4.28)–(4.29) *satisfy*

$$P_i^{+} = \sum_{\substack{j \in S_i \\ a_{ij} > 0}} a_{ij}(u_i - u_j)^{+}, \qquad P_i^{-} = \sum_{\substack{j \in S_i \\ a_{ij} > 0}} a_{ij}(u_i - u_j)^{-}, \tag{4.51}$$

and

$$Q_i^{+} = \sum_{j \in S_i} |d_{ij}|(u_j - u_i)^{+}, \qquad Q_i^{-} = \sum_{j \in S_i} |d_{ij}|(u_j - u_i)^{-}. \tag{4.52}$$

Moreover, the identities (4.52) *remain valid even if the sign condition* (4.37) *is not satisfied.*

Proof. We first prove (4.52). By (4.4), one has

$$d_{ij} \le 0, \qquad i \ne j, \tag{4.53}$$

and therefore

$$f_{ij} = d_{ij}(u_j - u_i) = |d_{ij}|(u_i - u_j). \tag{4.54}$$

Using the identities

$$-a^{-} = (-a)^{+}, \qquad -a^{+} = (-a)^{-}, \qquad a \in \mathbb{R}, \tag{4.55}$$

it follows from (4.29) that

$$Q_i^+ = -\sum_{j\in S_i} f_{ij}^- = -\sum_{j\in S_i} |d_{ij}|(u_i-u_j)^- = \sum_{j\in S_i} |d_{ij}|(u_j-u_i)^+, \tag{4.56}$$

$$Q_i^- = -\sum_{j\in S_i} f_{ij}^+ = -\sum_{j\in S_i} |d_{ij}|(u_i-u_j)^+ = \sum_{j\in S_i} |d_{ij}|(u_j-u_i)^-. \tag{4.57}$$

Hence (4.52) holds. Note that this argument does not use the sign condition (4.37).

Next, we prove (4.51). Assume that (4.37) holds. For any $i = 1,\dots,m$ and $j \in S_i$, we claim that

$$a_{ji} \le a_{ij} \text{ and } d_{ij} \ne 0 \qquad \Longleftrightarrow \qquad a_{ij} > 0. \tag{4.58}$$

Indeed, if $a_{ij} > 0$, then (4.37) implies $a_{ji} \le 0$, hence $a_{ji} \le a_{ij}$, and by (4.4),

$$d_{ij} = -\max\{a_{ij}, 0, a_{ji}\} = -a_{ij} \ne 0. \tag{4.59}$$

Conversely, if $a_{ji} \le a_{ij}$ and $d_{ij} \ne 0$, then

$$\max\{a_{ij}, 0, a_{ji}\} > 0. \tag{4.60}$$

If $a_{ij} \le 0$, then $a_{ji} \le a_{ij} \le 0$, so the above maximum would be zero, a contradiction. Hence $a_{ij} > 0$, and (4.58) follows.

Therefore, using (4.28), only the indices with $a_{ij} > 0$ contribute, and for such indices one has $d_{ij} = -a_{ij}$. Consequently,

$$f_{ij} = d_{ij}(u_j - u_i) = a_{ij}(u_i - u_j). \tag{4.61}$$

Thus,

$$P_i^+ = \sum_{\substack{j\in S_i \\ a_{ji}\le a_{ij}}} f_{ij}^+ = \sum_{\substack{j\in S_i \\ a_{ij}>0}} a_{ij}(u_i-u_j)^+, \tag{4.62}$$

$$P_i^- = \sum_{\substack{j\in S_i \\ a_{ji}\le a_{ij}}} f_{ij}^- = \sum_{\substack{j\in S_i \\ a_{ij}>0}} a_{ij}(u_i-u_j)^-. \tag{4.63}$$

This proves (4.51). □

Motivated by Lemma 4.10, we now give the concrete definition of the coefficients β_{ij} used in the MUAS method. The construction follows the idea of John and Knobloch [JK22]; see also [BJK25, Section 10.5].

Definition 4.11 (MUAS limiter coefficients)**.** *For every active node $i = 1,\dots,m$, define the quantities*

$$P_i^+ = \sum_{\substack{j\in S_i \\ a_{ij}>0}} a_{ij}(u_i-u_j)^+, \qquad P_i^- = \sum_{\substack{j\in S_i \\ a_{ij}>0}} a_{ij}(u_i-u_j)^-, \tag{4.64}$$

and

$$Q_i^+ = \sum_{j\in S_i} s_{ij}(u_j-u_i)^+, \qquad Q_i^- = \sum_{j\in S_i} s_{ij}(u_j-u_i)^-, \tag{4.65}$$

where

$$s_{ij} := \max\{|a_{ij}|, a_{ji}\}, \qquad j \in S_i. \tag{4.66}$$

The nodal limiting factors are then defined by

$$R_i^+ = \min\left\{1, \frac{Q_i^+}{P_i^+}\right\}, \qquad R_i^- = \min\left\{1, \frac{Q_i^-}{P_i^-}\right\}, \qquad i = 1, \dots, m. \tag{4.67}$$

If $P_i^+ = 0$ or $P_i^- = 0$, the corresponding limiting factor is set to

$$R_i^+ = 1 \qquad or \qquad R_i^- = 1, \tag{4.68}$$

respectively. For Dirichlet nodes, no additional limitation is required, and we set

$$R_i^+ = 1, \qquad R_i^- = 1, \qquad i = m+1, \dots, n. \tag{4.69}$$

The coefficients β_{ij} are finally defined by

$$\beta_{ij}(\underline{u}) = \begin{cases} 1 - R_i^+, & \text{if } u_i > u_j, \\ 0, & \text{if } u_i = u_j, \\ 1 - R_i^-, & \text{if } u_i < u_j, \end{cases} \qquad i, j = 1, \dots, n. \tag{4.70}$$

By construction, the coefficients satisfy

$$0 \le \beta_{ij}(\underline{u}) \le 1, \qquad i, j = 1, \dots, n. \tag{4.71}$$

Moreover, for boundary nodes $i, j = m+1, \dots, n$, the choice (4.69) implies

$$\beta_{ij}(\underline{u}) = 0, \qquad i, j = m+1, \dots, n, \tag{4.72}$$

which is consistent with the boundary condition (4.42). Combining Definition 4.11 with the matrix definition (4.40)–(4.41) yields the monotone upwind-type algebraically stabilized method.

Remark 4.12. *The definition of $Q_i^\pm$ in (4.65) differs from the reformulated Kuzmin quantities in (4.52) by replacing $|d_{ij}|$ with the weights s_{ij} defined in (4.66). Since*

$$|d_{ij}| = \max\{a_{ij}, 0, a_{ji}\}, \qquad s_{ij} = \max\{|a_{ij}|, a_{ji}\}, \tag{4.73}$$

one has $s_{ij} \ge |d_{ij}|$. This modification was observed in [JK22] to be beneficial for the accuracy and convergence behavior in diffusion-dominated problems on non-Delaunay meshes.

The convention (4.68) is analogous to the corresponding convention in the Kuzmin limiter; see Remark 4.4. It is harmless in the present construction as well. Indeed, if $P_i^+ = 0$, then for all $j \in S_i$ with $u_i > u_j$ one necessarily has $a_{ij} \le 0$. Hence, $\beta_{ij}(\underline{u})a_{ij} \le 0$, and therefore this term does not affect the value of the maximum in (4.40). Consequently, the value assigned to R_i^+ is irrelevant for the entries of $B(\underline{u})$. The same argument applies to R_i^- when $P_i^- = 0$. Hence, the convention (4.68) does not affect the entries of the stabilization matrix $B(\underline{u})$.

In the following result, the DMP properties are understood in the sense of the local and global discrete maximum principles for nonlinear matrix-based problems formulated in [BJK25, Definitions 3.71 and 3.75]. The proof relies only on the definition of the MUAS coefficients and does not use the sign condition (4.37).

Theorem 4.13 (DMP properties of the MUAS method)**.** *[BJK25, Theorem 10.53] Let the matrix $B(\underline{u})$ be defined by (4.40)–(4.41), with the coefficients β_{ij} given by Definition 4.11. Then the MUAS scheme (4.44)–(4.45) satisfies these DMP properties associated with the sets S_i in (3.9).*

Proof. We verify the algebraic DMP condition for the nonlinear matrix $A + B(\underline{u})$. Let $i = 1, \dots, m$ and $j \in S_i$. Suppose that u_i is a local extremum of $\underline{u}$ with respect to S_i and that $u_i \neq u_j$. Since

$$b_{ij}(\underline{u}) = -\max\{\beta_{ij}(\underline{u})a_{ij},\, 0,\, \beta_{ji}(\underline{u})a_{ji}\}, \qquad i \neq j, \tag{4.74}$$

it is sufficient to show that

$$a_{ij} - \max\{\beta_{ij}(\underline{u})a_{ij},\, 0,\, \beta_{ji}(\underline{u})a_{ji}\} \leq 0. \tag{4.75}$$

If $a_{ij} \leq 0$, then (4.75) is immediate, since the maximum is non-negative.

It remains to consider the case $a_{ij} > 0$. Since u_i is a local maximum, $u_i \geq u_k$ for all $k \in S_i$. For the fixed index j with $u_i \neq u_j$, this gives $u_i > u_j$. Hence

$$P_i^+ \geq a_{ij}(u_i - u_j)^+ > 0. \tag{4.76}$$

Moreover, for every $k \in S_i$, one has $u_k - u_i \leq 0$, and therefore

$$Q_i^+ = \sum_{k \in S_i} s_{ik}(u_k - u_i)^+ = 0. \tag{4.77}$$

Consequently,

$$R_i^+ = \min\left\{1, \frac{Q_i^+}{P_i^+}\right\} = 0, \tag{4.78}$$

and by (4.70),

$$\beta_{ij}(\underline{u}) = 1 - R_i^+ = 1. \tag{4.79}$$

Thus,

$$\max\{\beta_{ij}(\underline{u})a_{ij},\, 0,\, \beta_{ji}(\underline{u})a_{ji}\} \geq a_{ij}, \tag{4.80}$$

which proves (4.75).

The case where u_i is a local minimum is analogous. Then $u_i < u_j$, and

$$P_i^- \leq a_{ij}(u_i - u_j)^- < 0, \tag{4.81}$$

whereas

$$Q_i^- = \sum_{k \in S_i} s_{ik}(u_k - u_i)^- = 0. \tag{4.82}$$

Hence $R_i^- = 0$, and by (4.70),

$$\beta_{ij}(\underline{u}) = 1 - R_i^- = 1. \tag{4.83}$$

Again, the maximum in (4.75) is at least a_{ij}, and therefore (4.75) follows.

Thus, the required algebraic condition holds for every active node $i = 1, \dots, m$ and every $j \in S_i$. Hence, the MUAS scheme satisfies the DMP properties in the sense of [BJK25, Definitions 3.71 and 3.75]. □

Remark 4.14 (Solvability and DMP property of the MUAS method)**.** *The preceding results summarize the main advantage of the MUAS construction. Under the continuity assumption in Theorem 4.8, the nonlinear MUAS problem (4.44)–(4.45) admits at least one solution. For the concrete coefficients β_{ij} introduced in Definition 4.11, this continuity property is established in [BJK25, Theorem 10.54]. Moreover, Theorem 4.13 shows that the MUAS scheme satisfies the DMP properties in the sense of [BJK25, Definitions 3.71 and 3.75]. In contrast to the AFC scheme with the Kuzmin limiter, this DMP result does not require the sign condition (4.37). Hence, the MUAS method provides a solvable nonlinear algebraically stabilized scheme whose DMP property holds on arbitrary admissible meshes.*

The preceding sections introduced the nonlinear AFC and MUAS schemes considered in this work. Since the resulting algebraic systems are nonlinear, they have to be solved iteratively. The next chapter addresses practical issues arising during the nonlinear iteration process, in particular the preservation of qualitative solution properties at intermediate or final iterates.

5. Iterative Solution of Nonlinear Algebraically Stabilized Systems

The AFC and MUAS schemes introduced in Chapter 4 give rise to nonlinear algebraic systems. In the AFC case, the nonlinearity is caused by the solution-dependent limiter coefficients $\alpha_{ij}(\underline{u})$, while in the MUAS construction it is contained in the matrix-valued stabilization term $B(\underline{u})$. Hence, the discrete problem has to be solved iteratively.

The solution of these nonlinear systems is nontrivial because the limiting procedures involve non-smooth operations such as minima, maxima, and positive or negative parts. In practice, fixed-point iterations are therefore commonly used for nonlinear algebraically stabilized discretizations. The performance of different nonlinear iteration schemes for AFC discretizations was studied in [JJ20, JJ19].

This chapter introduces the residual formulation of the nonlinear systems and focuses on the fixed-point right-hand-side iteration. Particular attention is paid to practical modifications that aim to preserve qualitative properties, such as positivity or prescribed bounds, during the nonlinear iteration process.

5.1. Residual Formulation and Fixed-Point Right-Hand-Side Iteration

We first consider the AFC scheme. Based on the nonlinear AFC system (4.17)–(4.18), we define the nonlinear residual $F : \mathbb{R}^n \to \mathbb{R}^n$ by

$$F_i(\underline{u}) := \sum_{j=1}^{n} a_{ij}u_j + \sum_{j=1}^{n}\big(1 - \alpha_{ij}(\underline{u})\big)d_{ij}(u_j - u_i) - l_i, \qquad i = 1, \dots, m, \tag{5.1}$$

and, for the prescribed Dirichlet degrees of freedom,

$$F_i(\underline{u}) := u_i - u_i^b, \qquad i = m+1, \dots, n. \tag{5.2}$$

Together with the boundary residual equations (5.2), the active residual equations (5.1) define a nonlinear system equivalent to the AFC problem, namely

$$F(\underline{u}) = 0. \tag{5.3}$$

The same idea applies to the MUAS scheme. Recalling (4.44)–(4.45), the residual takes the abstract form

$$F_i(\underline{u}) := \sum_{j=1}^{n}\big(a_{ij} + b_{ij}(\underline{u})\big)u_j - l_i, \qquad i = 1, \dots, m, \tag{5.4}$$

together with the boundary equations (5.2). Hence, both the AFC and MUAS schemes can be treated as nonlinear systems of the form (5.3).

Following the formulation in [JJ19], a general damped nonlinear iteration for solving (5.3) is given by

$$\underline{u}^{(\nu+1)} = \underline{u}^{(\nu)} - \omega^{(\nu)}\mathcal{B}\big(\underline{u}^{(\nu)}\big)^{-1}F\big(\underline{u}^{(\nu)}\big), \qquad \nu = 0, 1, 2, \dots, \tag{5.5}$$

where $\mathcal{B}(\underline{u}^{(\nu)})$ is a nonsingular iteration matrix and $\omega^{(\nu)} \in (0, 1]$ is a damping parameter. Different choices of $\mathcal{B}(\underline{u}^{(\nu)})$ lead to different nonlinear solvers. In the fixed-point approaches considered below, the limiter coefficients or stabilization matrices are first computed from the current iterate and then treated as fixed quantities when solving for the next iterate.

For the AFC scheme, a natural fixed-point iteration is obtained by evaluating the limiter coefficients at the current iterate,

$$\alpha_{ij}^{(\nu)} := \alpha_{ij}\big(\underline{u}^{(\nu)}\big), \tag{5.6}$$

and then solving for the next iterate $\underline{u}^{(\nu+1)}$ from

$$\sum_{j=1}^{n} a_{ij} u_j^{(\nu+1)} + \sum_{j=1}^{n} \big(1 - \alpha_{ij}^{(\nu)}\big) d_{ij} \big(u_j^{(\nu+1)} - u_i^{(\nu+1)}\big) = l_i, \qquad i = 1, \dots, m, \tag{5.7}$$

$$u_i^{(\nu+1)} = u_i^b, \qquad i = m+1, \dots, n. \tag{5.8}$$

This iteration corresponds to assembling a new linear system at every nonlinear step, using the limiter values computed from the previous iterate. In the terminology of [JJ20], this approach is referred to as a fixed-point matrix iteration.

An alternative fixed-point formulation uses the low-order matrix

$$\bar{A} := A + D. \tag{5.9}$$

Following [JJ20], the zero row sums of the artificial diffusion matrix D allow the nonlinear AFC problem to be written in fixed-point right-hand-side form. With

$$f_{ij}^{(\nu)} := d_{ij}\big(u_j^{(\nu)} - u_i^{(\nu)}\big), \tag{5.10}$$

one obtains the fixed-point iteration

$$\sum_{j=1}^{n} \big(a_{ij} + d_{ij}\big) u_j^{(\nu+1)} = l_i + \sum_{j=1}^{n} \alpha_{ij}^{(\nu)} f_{ij}^{(\nu)}, \qquad i = 1, \dots, m, \tag{5.11}$$

$$u_i^{(\nu+1)} = u_i^b, \qquad i = m+1, \dots, n. \tag{5.12}$$

Consequently, only the right-hand side changes during the nonlinear iteration, which can be computationally useful if the same linear solver or preconditioner can be reused. Although this iteration is written in the classical AFC flux notation, the same fixed-point right-hand-side strategy is used for the MUAS scheme, whose nonlinear stabilization matrix is defined in (4.40)–(4.41).

In the numerical experiments of this thesis, the Tabata upwind solution introduced in Section 3.3 will be used as the initial iterate,

$$\underline{u}^{(0)} := \underline{u}_{\mathrm{Tab}}. \tag{5.13}$$

This choice is motivated by the robustness of the Tabata method and by its good accuracy among stabilized linear finite element methods; see, for instance, the numerical comparisons in [BJK25, Section 8.7]. Thus, the Tabata solution provides a stable and accurate starting point for the nonlinear iteration. The fixed-point right-hand-side iteration (5.11)–(5.12) will serve as the basis for the positivity-preserving modifications introduced in the following section.

5.2. Positivity-Preserving Modifications of the Fixed-Point Iteration

The modifications considered in this section are based on the fixed-point right-hand-side iteration (5.11)–(5.12). In this formulation, the low-order matrix $\bar{A} = A + D$ is fixed during the nonlinear

iteration, whereas the nonlinear flux-correction contribution is evaluated at the current iterate and moved to the right-hand side. This structure makes it possible to modify the right-hand side in order to preserve positivity of the iterates.

Let $\underline{v}$ denote the current iterate used to evaluate the nonlinear correction term. At one step of the fixed-point right-hand-side iteration, the linear system can be written in block form as

$$\begin{pmatrix} \bar{A}^i & \bar{A}^b \\ 0 & I \end{pmatrix} \begin{pmatrix} \underline{u}^i \\ \underline{u}^b \end{pmatrix} = \begin{pmatrix} \underline{l} \\ \underline{u}^b \end{pmatrix} + \begin{pmatrix} \underline{l}^{NL}(\underline{v}) \\ 0 \end{pmatrix}, \tag{5.14}$$

where $\underline{u}^i$ denotes the vector of active degrees of freedom and $\underline{u}^b$ denotes the vector of prescribed Dirichlet values. The nonlinear vector $\underline{l}^{NL}(\underline{v})$ contains the flux-correction contribution evaluated at the current iterate. In terms of (5.11), its components are given by

$$l_i^{NL}(\underline{v}) = \sum_{j=1}^{n} \alpha_{ij}(\underline{v}) f_{ij}(\underline{v}), \qquad i = 1, \dots, m. \tag{5.15}$$

The block structure in (5.14) is the same as the low-order AFC block form (4.11). By Remark 4.1, the active block $\bar{A}^i$ is an M-matrix. The active part of (5.14) is given by

$$\bar{A}^i \underline{u}^i = \underline{l} - \bar{A}^b \underline{u}^b + \underline{l}^{NL}(\underline{v}). \tag{5.16}$$

Since $(\bar{A}^i)^{-1} \geq 0$, positivity of the next iterate is guaranteed if the right-hand side in (5.16) is component-wise nonnegative, that is,

$$\underline{l} - \bar{A}^b \underline{u}^b + \underline{l}^{NL}(\underline{v}) \geq 0 \tag{5.17}$$

component-wise.

The two modifications introduced below are designed to enforce or support (5.17). The first one modifies the assembled right-hand side directly, whereas the second one damps those limiter contributions that may produce negative components in the right-hand side.

5.2.1. Algorithm 1: Right-Hand-Side Clipping

The first modification acts directly on the assembled right-hand side of the fixed-point right-hand-side iteration. Recall from (5.16) that positivity of the active part of the next iterate follows if

$$\underline{l} - \bar{A}^b \underline{u}^b + \underline{l}^{NL}(\underline{v}) \geq 0 \tag{5.18}$$

component-wise. By Remark 4.1, $\bar{A}^i$ is an M-matrix, and hence this condition implies nonnegativity of the active part of the next iterate. The idea of right-hand-side clipping is therefore to modify the assembled right-hand side only in those components where this condition is violated.

Let

$$\underline{r}(\underline{v}) := \underline{l} + \underline{l}^{NL}(\underline{v}) \tag{5.19}$$

denote the assembled right-hand side before the Dirichlet contribution is moved to the right-hand side. Then the active system can be written as

$$\bar{A}^i \underline{u}^i = \underline{r}(\underline{v}) - \bar{A}^b \underline{u}^b. \tag{5.20}$$

Thus, for an active degree of freedom $i = 1, \dots, m$, the positivity condition is violated if

$$r_i(\underline{v}) - \left(\bar{A}^b \underline{u}^b\right)_i < 0, \tag{5.21}$$

or equivalently,

$$r_i(\underline{v}) < \left(\bar{A}^b \underline{u}^b\right)_i. \tag{5.22}$$

In this case, the assembled right-hand side is clipped by setting

$$\widetilde{r}_i(\underline{v}) := \left(\bar{A}^b \underline{u}^b\right)_i. \tag{5.23}$$

If condition (5.21) is not satisfied, the component is left unchanged, that is,

$$\widetilde{r}_i(\underline{v}) := r_i(\underline{v}). \tag{5.24}$$

Equivalently, the clipped right-hand side can be written as

$$\widetilde{r}_i(\underline{v}) = \max\left\{r_i(\underline{v}), \left(\bar{A}^b \underline{u}^b\right)_i\right\}, \qquad i = 1, \ldots, m. \tag{5.25}$$

The next iterate is then computed from

$$\bar{A}^i \underline{u}^{i,(\nu+1)} = \widetilde{\underline{r}}\left(\underline{u}^{(\nu)}\right) - \bar{A}^b \underline{u}^b. \tag{5.26}$$

By construction, the right-hand side in (5.26) is component-wise nonnegative. Hence, by Remark 4.1, $\underline{u}^{i,(\nu+1)} \geq 0$.

In the special case of homogeneous Dirichlet boundary data, the boundary contribution vanishes and the modification reduces to replacing negative components of the assembled right-hand side by zero. Thus, the method is simple to implement and preserves the matrix structure of the fixed-point right-hand-side iteration. However, the modification changes the nonlinear right-hand side after assembly and should therefore be regarded as a practical positivity-preserving correction rather than as a modification of the limiter itself.

5.2.2. Algorithm 2: Limiter Damping

The second modification acts on the limiter coefficients before the nonlinear right-hand side is assembled. In contrast to Algorithm 1, the assembled right-hand side is not clipped directly. Instead, the limiter contributions that may lead to negative components of the right-hand side are damped. The aim is again to enforce the component-wise nonnegativity condition (5.17).

Using the notation introduced in (5.15), the active right-hand side in (5.16) can be written component-wise as

$$l_i - \left(\bar{A}^b \underline{u}^b\right)_i + \sum_{j=1}^{n} \alpha_{ij}(\underline{v}) f_{ij}(\underline{v}) \geq 0, \qquad i = 1, \ldots, m. \tag{5.27}$$

For each active degree of freedom, we split the flux-correction contribution into positive and negative parts. Thus, condition (5.27) can be rewritten as

$$l_i - \left(\bar{A}^b \underline{u}^b\right)_i + \sum_{\alpha_{ij}(\underline{v}) f_{ij}(\underline{v}) > 0} \alpha_{ij}(\underline{v}) f_{ij}(\underline{v}) + \sum_{\alpha_{ij}(\underline{v}) f_{ij}(\underline{v}) < 0} \alpha_{ij}(\underline{v}) f_{ij}(\underline{v}) \geq 0. \tag{5.28}$$

A damping strategy based directly on (5.28) would lead to a coupled problem for the damping factors. Indeed, in order to preserve the symmetry of the limiter coefficients, damping the coefficient α_{ij} also implies damping α_{ji}. Hence, a damping chosen for the i-th active equation may also affect the flux-correction contribution in the j-th active equation. Connected active degrees of freedom can therefore not be treated independently if positive flux-correction contributions are used to compensate negative ones.

For this reason, Algorithm 2 uses a stronger sufficient condition. Only flux-correction terms with $\alpha_{ij}(\underline{v})f_{ij}(\underline{v}) < 0$ are restricted, and positive contributions are not used to compensate negative ones. Therefore, for the damped limiter coefficients it is sufficient to require

$$l_i - \left(\bar{A}^b \underline{u}^b\right)_i + \sum_{\substack{j=1 \\ \alpha_{ij}(\underline{v})f_{ij}(\underline{v})<0}}^{n} \theta_{ij}\alpha_{ij}(\underline{v})f_{ij}(\underline{v}) \geq 0, \qquad i = 1, \ldots, m, \tag{5.29}$$

where $0 \leq \theta_{ij} \leq 1$. This condition is stronger than (5.28), since positive flux-correction terms are not used to compensate negative ones.

The quantity

$$p_i := l_i - \left(\bar{A}^b \underline{u}^b\right)_i, \qquad i = 1, \ldots, m, \tag{5.30}$$

is called the positivity budget of the i-th active equation. Under the positivity assumptions on the data and the prescribed Dirichlet values, one has $p_i \geq 0$. Indeed, $l_i \geq 0$, $\underline{u}^b \geq 0$, and, by (4.9), the Dirichlet block satisfies $\bar{A}^b \leq 0$ component-wise. Hence $(\bar{A}^b \underline{u}^b)_i \leq 0$, and therefore $p_i = l_i - (\bar{A}^b \underline{u}^b)_i \geq 0$.

For a given iterate $\underline{v}$, we define the negative budget by

$$n_i(\underline{v}) := \sum_{j \neq i} \max\left\{0, -\alpha_{ij}(\underline{v})f_{ij}(\underline{v})\right\}, \qquad i = 1, \ldots, m. \tag{5.31}$$

$$\theta_i(\underline{v}) := \begin{cases} \min\left\{1, \dfrac{p_i}{n_i(\underline{v})}\right\}, & i = 1, \ldots, m, \quad n_i(\underline{v}) > 0, \\ 1, & i = 1, \ldots, m, \quad n_i(\underline{v}) = 0, \\ 1, & i = m+1, \ldots, n. \end{cases} \tag{5.32}$$

The damped limiter coefficients are then defined by

$$\widetilde{\alpha}_{ij}(\underline{v}) := \min\left\{\theta_i(\underline{v}), \theta_j(\underline{v})\right\} \alpha_{ij}(\underline{v}), \qquad i, j = 1, \ldots, n. \tag{5.33}$$

This choice preserves the symmetry of the limiter coefficients. Indeed, if $\alpha_{ij} = \alpha_{ji}$ as in (4.16), then $\widetilde{\alpha}_{ij} = \widetilde{\alpha}_{ji}$.

The next iterate is computed by replacing the original limiter coefficients in the fixed-point right-hand-side iteration (5.11) by the damped coefficients:

$$\sum_{j=1}^{n} \left(a_{ij} + d_{ij}\right) u_j^{(\nu+1)} = l_i + \sum_{j=1}^{n} \widetilde{\alpha}_{ij}\left(\underline{u}^{(\nu)}\right) f_{ij}^{(\nu)}, \qquad i = 1, \ldots, m, \tag{5.34}$$

$$u_i^{(\nu+1)} = u_i^b, \qquad i = m+1, \ldots, n. \tag{5.35}$$

For each active row i, the definition of θ_i gives

$$\theta_i(\underline{v})n_i(\underline{v}) \leq p_i.$$

Moreover,

$$\min\{\theta_i(\underline{v}), \theta_j(\underline{v})\} \leq \theta_i(\underline{v}), \qquad j = 1, \ldots, n.$$

Hence the total damped negative flux-correction contribution in the i-th active equation is bounded in absolute value by p_i. Therefore, the active right-hand side satisfies the component-wise nonnegativity condition (5.17). By Remark 4.1, the active part of the next iterate is nonnegative.

Compared with right-hand-side clipping, limiter damping modifies the limiter coefficients before assembly of the nonlinear right-hand side. Thus, the correction acts at the level of the flux limiting procedure and preserves the symmetry of the limiters, while still keeping the fixed low-order matrix $\bar{A}$ of the fixed-point right-hand-side iteration.

5.3. Projection-Based Bound Corrections

5.3.1. Algorithm 3: Projection at Every Iterate

The third modification is a projection-based correction of the nonlinear iterates. In contrast to Algorithms 1 and 2, it neither modifies the right-hand side of the fixed-point right-hand-side iteration (5.11)–(5.12) nor changes the limiter coefficients. Instead, after each nonlinear iteration step, the active degrees of freedom are projected onto a prescribed interval.

Let $u_{\min}$ and $u_{\max}$ be prescribed bounds satisfying

$$u_{\min} < u_{\max}. \tag{5.36}$$

Let $\underline{u}^{(k)}$ denote the iterate obtained after one step of the fixed-point right-hand-side iteration. Algorithm 3 replaces only the active components by their component-wise projection onto $[u_{\min}, u_{\max}]$, namely

$$u_i^{(k)} := \min\left\{u_{\max}, \max\left\{u_{\min}, u_i^{(k)}\right\}\right\}, \qquad i = 1, \dots, m, \tag{5.37}$$

while the prescribed Dirichlet values are kept unchanged,

$$u_i^{(k)} = u_i^b, \qquad i = m+1, \dots, n. \tag{5.38}$$

Consequently, the corrected active components satisfy

$$u_{\min} \le u_i^{(k)} \le u_{\max}, \qquad i = 1, \dots, m. \tag{5.39}$$

This modification enforces the prescribed bounds at every nonlinear iterate. In particular, choosing $u_{\min} = 0$ yields component-wise nonnegative active iterates. Since the projection is applied only after the linear system has been solved, it is not a modification of the AFC fluxes and does not preserve the nonlinear algebraic equations exactly at intermediate iterates. Therefore, Algorithm 3 should be regarded as a practical bound correction whose influence on accuracy and nonlinear convergence has to be assessed numerically.

5.3.2. Algorithm 4: Projection of the Final Iterate

The fourth modification applies the projection only after the nonlinear iteration has been terminated. In contrast to Algorithm 3, the intermediate iterates are not modified. Thus, the fixed-point right-hand-side iteration (5.11)–(5.12) is performed in its original form until the stopping criterion is satisfied.

Let $\underline{u}^{(N)}$ denote the final iterate obtained from the nonlinear iteration. Using the same prescribed bounds $u_{\min}$ and $u_{\max}$ as in (5.36), the active components are corrected by the component-wise projection

$$\widehat{u}_i := \min\left\{u_{\max}, \max\left\{u_{\min}, u_i^{(N)}\right\}\right\}, \qquad i = 1, \dots, m. \tag{5.40}$$

The prescribed Dirichlet values are kept unchanged,

$$\widehat{u}_i := u_i^b, \qquad i = m+1, \dots, n. \tag{5.41}$$

Consequently, the active components of the corrected final solution satisfy

$$u_{\min} \le \widehat{u}_i \le u_{\max}, \qquad i = 1, \dots, m. \tag{5.42}$$

This modification is a purely final postprocessing step. It does not affect the nonlinear iteration itself and therefore does not change the convergence behavior of the fixed-point iteration before termination. If the final iterate already satisfies the prescribed bounds, then $\underline{\widehat{u}} = \underline{u}^{(N)}$. Otherwise, the projection

enforces the bounds only for the reported final solution. As in Algorithm 3, the projected vector does not, in general, satisfy the original nonlinear algebraic system exactly after the projection has been applied.

In summary, the four algorithms introduced in this chapter act at different stages of the nonlinear iteration. Right-hand-side clipping modifies the assembled right-hand side, limiter damping acts on the limiter coefficients before assembly, and the two projection-based strategies correct either every nonlinear iterate or only the final iterate. Their effect on positivity preservation, bound preservation, and nonlinear convergence will be investigated numerically in the next chapter.

6. Numerical Experiments

This chapter investigates the numerical behavior of the four modified fixed-point right-hand-side iterations introduced in Chapter 5. For each test problem, the finite element discretization and the algebraic stabilization method are fixed, while the nonlinear iteration strategy is varied. The aim is to study how early stopping and the proposed correction strategies affect accuracy, nonlinear convergence, and the preservation of prescribed bounds.

The numerical study considers three benchmark problems. The smooth and layer problems are based on Examples 10.108 and 10.109, respectively, in [BJK25, Section 10.12]. The first permits the direct computation of FEM errors, while the second is assessed using independent reference cross-sections. The Hemker problem, introduced in [Hem96], is studied in the bounded obstacle configuration from [JKP23, Example 3.2]. Its downstream layer structure is assessed using cross-section profiles and layer widths.

6.1. General Numerical Framework

All computations in this chapter are performed with the finite element code ParMooN [WBAea17]. The spatial discretization uses conforming P_1 finite elements on triangular meshes. The nonlinear MUAS discretization from Section 4.3 is used in all experiments. The resulting nonlinear algebraic system is solved by the fixed-point right-hand-side iteration (5.11)–(5.12). The Tabata upwind solution is used as the initial iterate, as described in (5.13). The unmodified fixed-point iteration is referred to as the standard approach.

The standard approach is compared with the four modified iterations introduced in Chapter 5. Algorithms 1 and 2 modify the nonlinear right-hand side or the limiter coefficients in order to support nonnegativity. Algorithms 3 and 4 are projection-based corrections onto prescribed lower and upper bounds.

For the nonlinear iteration, two stopping procedures are used. For Algorithms 1–4, the early-stopping runs use $N_{\max} \in \{10, 25, 50, 100\}$. In addition, a tolerance run is performed with $N_{\max} = 10000$. The standard approach is computed only as a tolerance run with $N_{\max} = 10000$.

In a tolerance run, the iteration is terminated earlier if the nonlinear stopping criterion is satisfied. Thus, the actual number of nonlinear iterations may be smaller than $N_{\max}$. The prescribed nonlinear tolerance is 10^{-12}, using the normalization implemented in the code. This tolerance concerns only the fixed-point iteration for the discrete nonlinear problem and is not a tolerance for the FEM errors with respect to the exact solution.

In the tables, "actual iter." denotes the number of nonlinear iterations actually performed. The label "tol." indicates that the nonlinear stopping criterion was satisfied, whereas "max. iter." indicates that the prescribed maximum number of nonlinear iterations was reached.

The precise numerical setup depends on the test problem. The meshes, stopping criteria, prescribed bounds, reference data, and diagnostic quantities are therefore specified separately for each example below.

6.2. Smooth Solution Example

The first experiment is based on Example 10.108 in [BJK25, Section 10.12]. The underlying problem without layers is the convection-diffusion-reaction boundary value problem from [BJK25, Example 8.63]. The domain is $\Omega = (0,1)^2$, the diffusion coefficient is $\varepsilon = 10^{-8}$, and the convection field is

$$\mathbf{b}(x,y) = (x,-y)^{\mathrm{T}}.$$

The reaction coefficient is

$$c(x,y) = \big(2+\cos(-4x+3y)\big)e^{(1-x)(1-y)^2}.$$

The prescribed exact solution is

$$u(x,y) = 16\sin(2\pi x)y^2(1-y)^2.$$

The right-hand side is obtained by inserting this function into the differential equation, and homogeneous Dirichlet boundary conditions are imposed on $\partial\Omega$. The exact solution is illustrated in Figure 6.1.

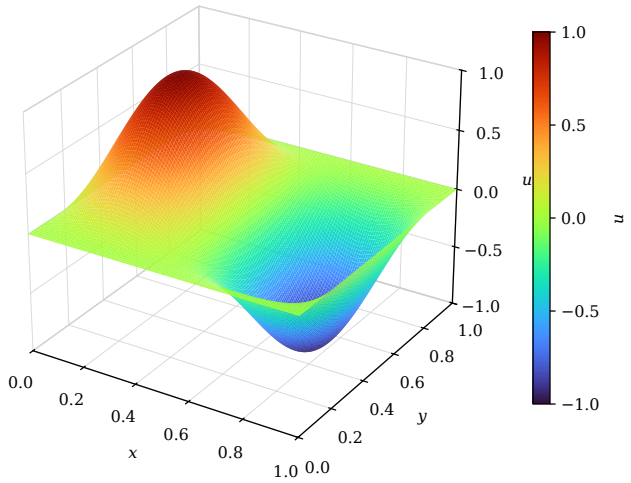


Figure 6.1.: Smooth solution example: prescribed exact solution $u(x,y) = 16\sin(2\pi x)y^2(1-y)^2$, based on [BJK25, Example 8.63].

Since the exact solution is known explicitly, the accuracy of the finite element solution u_h can be measured directly. The reported quantities are the $L^2(\Omega)$ error, the $H^1(\Omega)$ seminorm error, and the $L^\infty(\Omega)$ error, given respectively by

$$\|u-u_h\|_{L^2(\Omega)}, \qquad |u-u_h|_{H^1(\Omega)}, \qquad \|u-u_h\|_{L^\infty(\Omega)}.$$

The solution is smooth and does not contain boundary or interior layers. Therefore, this problem is suitable for checking whether the modified nonlinear iterations change the accuracy of the MUAS solution when no layer-driven oscillations are present. The experiment is performed on three mesh families used in [BJK25, Example 10.108]: a Friedrichs–Keller mesh, a Chevron grid, and a Delaunay mesh. In the present computations, one representative mesh from each family is used:

Friedrichs–Keller mesh (16×16), Chevron grid (64×64), Delaunay mesh (64×64).

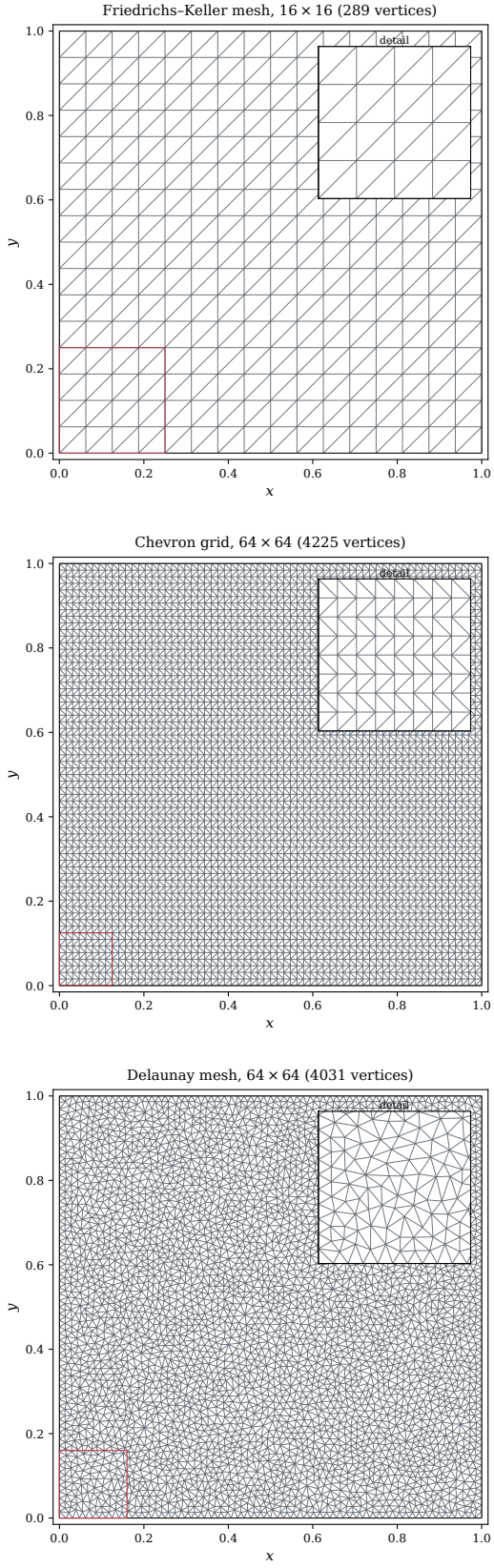


Figure 6.2.: Smooth solution example: meshes used in the numerical experiments. From top to bottom: Friedrichs–Keller mesh of size 16×16, Chevron grid of size 64×64, and Delaunay mesh of size 64×64.

The corresponding total numbers of degrees of freedom are 289, 4225 and 4031, respectively. After imposing the homogeneous Dirichlet boundary conditions, the corresponding numbers of active degrees of freedom are 225, 3969, and 3775, respectively. The three meshes used in this experiment are shown in Figure 6.2.

Since the exact solution of this smooth example lies in $[-1, 1]$, this test is not intended to assess the positivity-preserving Algorithms 1 and 2. Instead, we use it to study the bound-preserving projection strategies in Algorithms 3 and 4. In particular, we compare whether projection onto prescribed bounds changes the nonlinear convergence behavior or the final errors.

Two choices of bounds are considered: the fixed interval $[-1, 1]$, which contains the range of the exact solution, and bounds derived from the Tabata upwind solution.

6.2.1. Prescribed Bounds $[-1, 1]$

In the first set of computations, the bounds are prescribed manually as

$$u_{\min} = -1, \qquad u_{\max} = 1.$$

The corresponding error curves are shown in Figures 6.3–6.5, and the numerical values are listed in Tables A.1, A.2, and A.3. Since Algorithms 3 and 4 give identical errors in this test, they are listed together as Alg. 3/4.

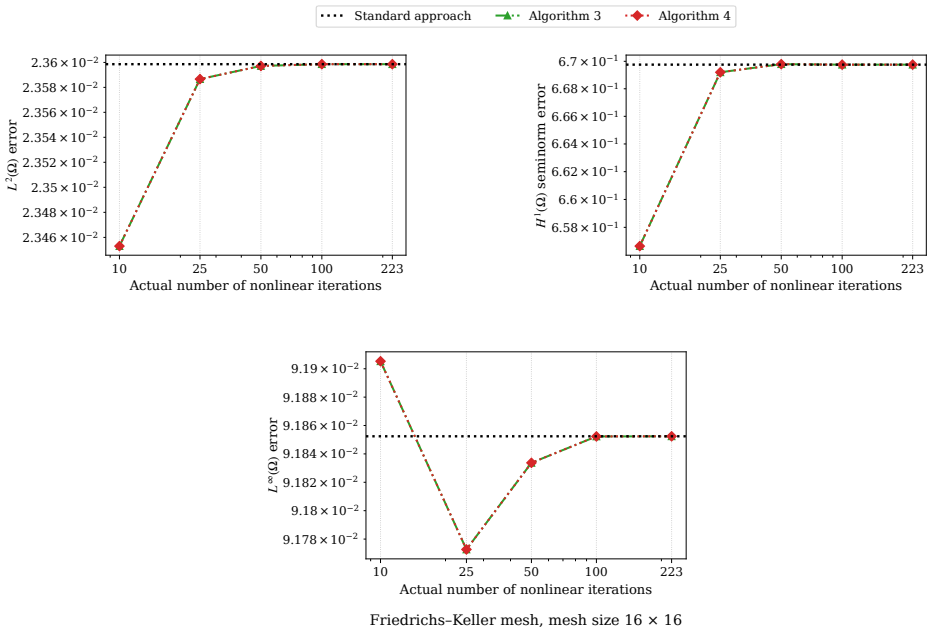


Figure 6.3.: Smooth solution problem with prescribed bounds $[-1, 1]$: errors on the Friedrichs–Keller mesh of size 16×16.

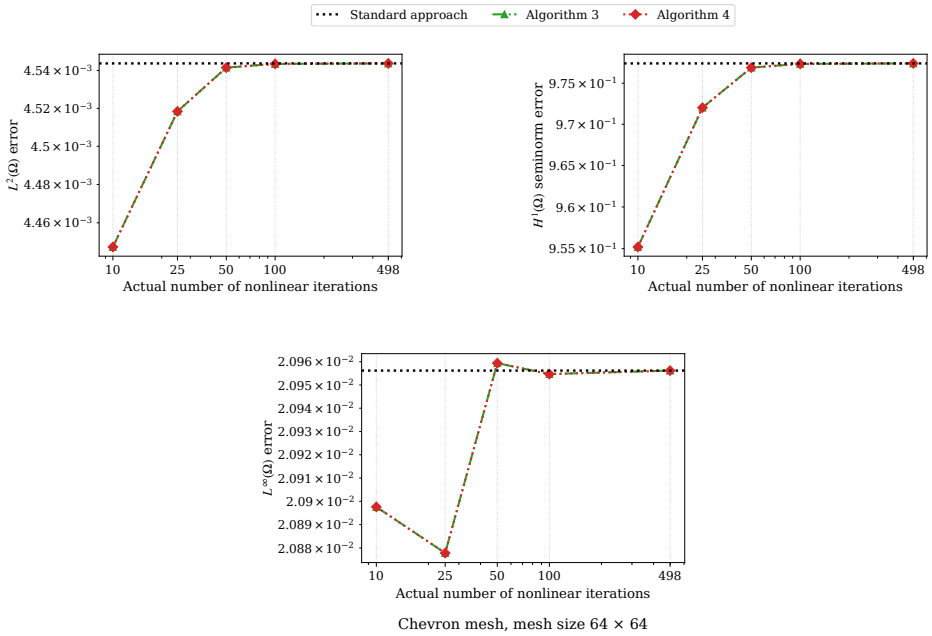


Figure 6.4.: Smooth solution problem with prescribed bounds $[-1, 1]$: errors on the Chevron grid of size 64×64.

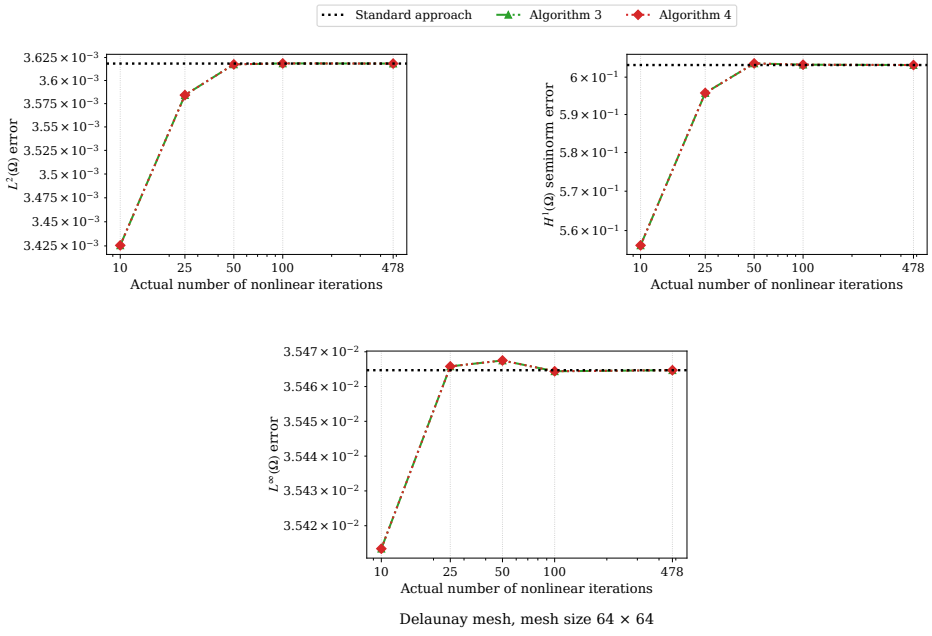


Figure 6.5.: Smooth solution problem with prescribed bounds $[-1, 1]$: errors on the Delaunay mesh of size 64×64.

The observations from the error plots in Figures 6.3–6.5 and from the numerical values in Tables A.1, A.2, and A.3 can be summarized as follows.

1. For every tested stopping condition and for all three meshes, Algorithms 3 and 4 give identical error values in the $L^2(\Omega)$ norm, the $H^1(\Omega)$ seminorm, and the $L^\infty(\Omega)$ norm.

2. At the tolerance stopping point, Algorithms 3 and 4 have the same actual iteration counts as the

standard approach, namely 223, 498, and 478 on the Friedrichs–Keller, Chevron, and Delaunay meshes, respectively. They also give the same errors as the standard approach. Thus, for the prescribed interval $[-1, 1]$, the projection is inactive and does not change the final discrete solution in this example.

3. The errors do not decrease monotonically with the prescribed maximum number of nonlinear iterations. For instance, on the Chevron grid, the $L^\infty(\Omega)$ error at 25 iterations is slightly smaller than the corresponding errors at 50, 100, and at the tolerance stopping point. This shows that nonlinear residual reduction and discretization error with respect to the exact solution should be interpreted separately.

4. The error curves become nearly flat after about 50 to 100 nonlinear iterations. Hence, for this smooth test problem, increasing the number of nonlinear iterations beyond this range has only a small effect on the final error.

6.2.2. Bounds Derived from the Tabata Solution

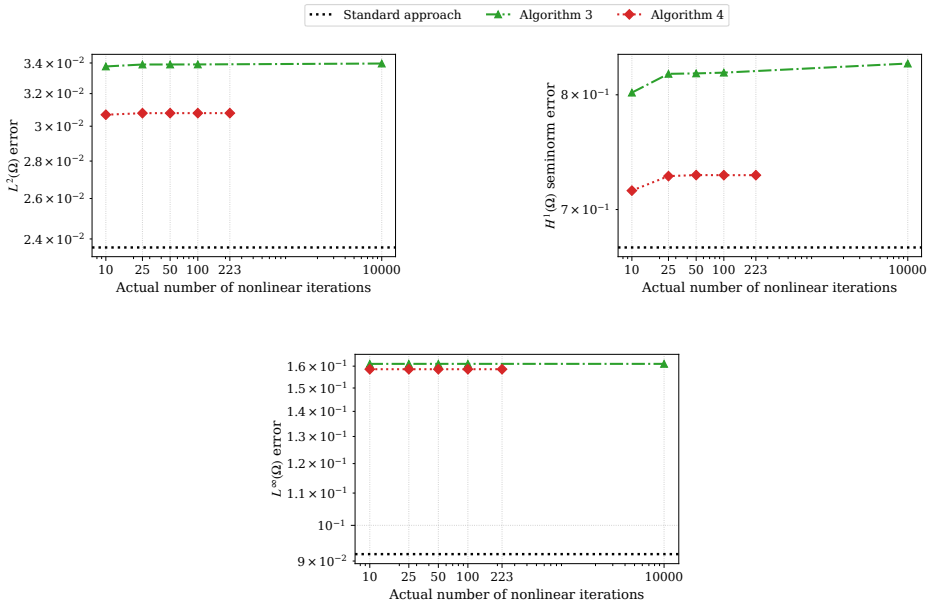
In the second set of computations, the bounds are not prescribed explicitly. Instead, the lower and upper bounds are obtained from the Tabata upwind solution on each mesh. Hence the projection interval is mesh-dependent and is determined by the initial upwind solution, rather than by the known exact solution. The resulting bounds are listed in Table 6.1.

Table 6.1.: Smooth solution problem: bounds derived from the Tabata solution.

Mesh	Lower bound	Upper bound
Friedrichs–Keller 16×16	-0.840334058935	0.912423965797
Chevron 64×64	-0.965123335430	0.977943099809
Delaunay 64×64	-0.968555659759	0.979985688484

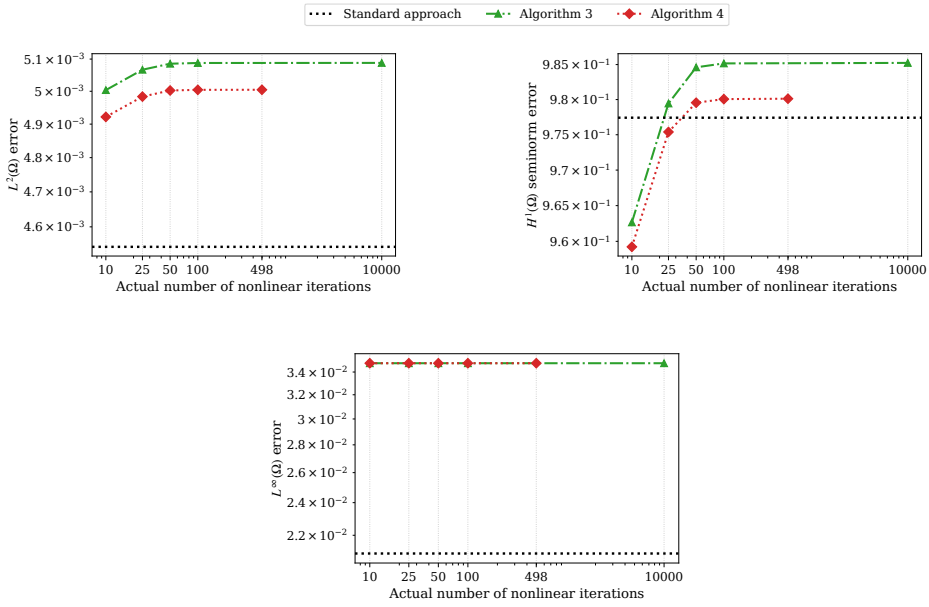
The Tabata-derived intervals are narrower than the prescribed interval $[-1, 1]$. This is particularly visible on the Friedrichs–Keller mesh. Since the exact solution takes values close to the endpoints of $[-1, 1]$, these narrower bounds may cut off part of the MUAS solution.

The corresponding error curves are shown in Figures 6.6–6.8, and the numerical values are listed in Tables A.4, A.5, and A.6. In contrast to the prescribed-bound case, Algorithms 3 and 4 no longer give identical results and are therefore listed separately.



Friedrichs–Keller mesh, mesh size 16 × 16, Tabata-derived bounds

Figure 6.6.: Smooth solution problem with Tabata-derived bounds: errors on the Friedrichs–Keller mesh of size 16×16.



Chevron mesh, mesh size 64 × 64, Tabata-derived bounds

Figure 6.7.: Smooth solution problem with Tabata-derived bounds: errors on the Chevron grid of size 64×64.

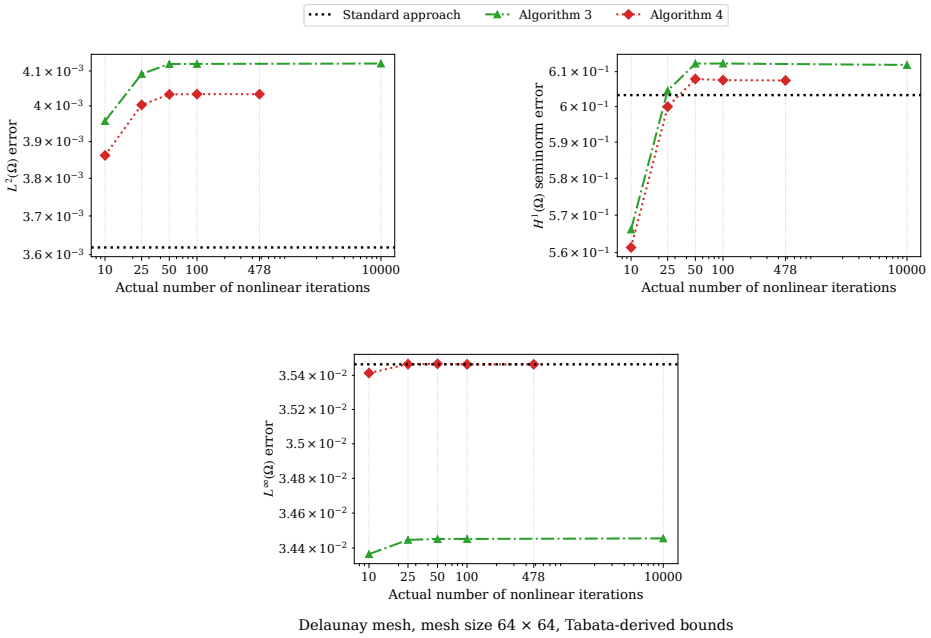


Figure 6.8.: Smooth solution problem with Tabata-derived bounds: errors on the Delaunay mesh of size 64×64.

The observations from the error plots in Figures 6.6–6.8 and from the numerical values in Tables A.4–A.6 can be summarized as follows.

1. The Tabata-derived bounds are fixed for each mesh, but they are narrower than the interval $[-1, 1]$. Hence the projection is active in this test, unlike in the prescribed-bound case.

2. Algorithm 3 does not reach the nonlinear stopping criterion on any of the three meshes. Even for $N_{\max} = 10000$, the stopping status is "max. iter.".

3. Algorithm 4 reaches the nonlinear stopping criterion on all three meshes. The actual stopping iteration numbers are 223 for the Friedrichs–Keller mesh, 498 for the Chevron grid, and 478 for the Delaunay mesh. These are the same iteration numbers as for the standard approach.

4. Although Algorithm 4 reaches the nonlinear stopping criterion, its final errors are larger than those of the standard approach. This is visible in the $L^2(\Omega)$ error on all three meshes and in the $H^1(\Omega)$ seminorm error especially on the Friedrichs–Keller mesh.

5. The $L^\infty(\Omega)$ error behaves differently from the other two norms. On the Delaunay mesh, Algorithm 4 and the standard approach have the same $L^\infty(\Omega)$ error at the tolerance stopping point. On the Chevron grid, the $L^\infty(\Omega)$ errors of Algorithms 3 and 4 coincide in the reported values.

6. Compared with the prescribed-bound case, the Tabata-derived bounds constrain the solution more strongly. Thus the experiment shows that bounds obtained automatically from the initial upwind solution can affect both nonlinear convergence and final accuracy.

Overall, the two sets of computations lead to two main conclusions. First, when appropriate bounds such as $[-1, 1]$ are prescribed, the FEM errors obtained with early termination of the nonlinear iteration are already very close to those obtained with the converged solution. Thus, in this case, further nonlinear iterations have only a minor influence on the numerical accuracy.

Second, deriving the bounds from the Tabata upwind solution is not successful for this example. The resulting intervals are too restrictive. Consequently, Algorithm 3 does not satisfy the nonlinear stopping criterion, even when $N_{\max} = 10000$, while the final projection in Algorithm 4 increases the FEM errors compared with the standard approach. Therefore, the bounds used in the projection-based algorithms must be chosen carefully.

6.3. Layer Solution Example

The second experiment is based on Example 10.109 in [BJK25, Section 10.12]. The underlying model problem is Example 2.25 from [BJK25]. The domain is $\Omega = (0,1)^2$, and the problem is given by

$$-\varepsilon \Delta u + (1,0)^{\mathrm{T}} \cdot \nabla u = f \quad \text{in } \Omega, \qquad u = 0 \quad \text{on } \partial\Omega,$$

where $\varepsilon = 10^{-5}$ and

$$f(x,y) = \begin{cases} 1, & y \in [0, 0.5), \\ 0.5, & y = 0.5, \\ 0, & y \in (0.5, 1]. \end{cases}$$

For small values of ε, the solution changes rapidly in several parts of the domain. An outflow layer occurs near $x = 1$. Further layers start at $y = 0$ and at the discontinuity line $y = 0.5$ and spread in the direction of convection. Where these layers meet, an additional corner layer is formed; see [BJK25, Example 2.25].

For a clearer visualization of the layer structure, a representative numerical solution obtained with the standard approach on a finer Chevron grid of size 128×128 is shown in Figure 6.9. This finer grid is used only for visualization; the numerical comparisons below are performed on the two meshes specified subsequently.

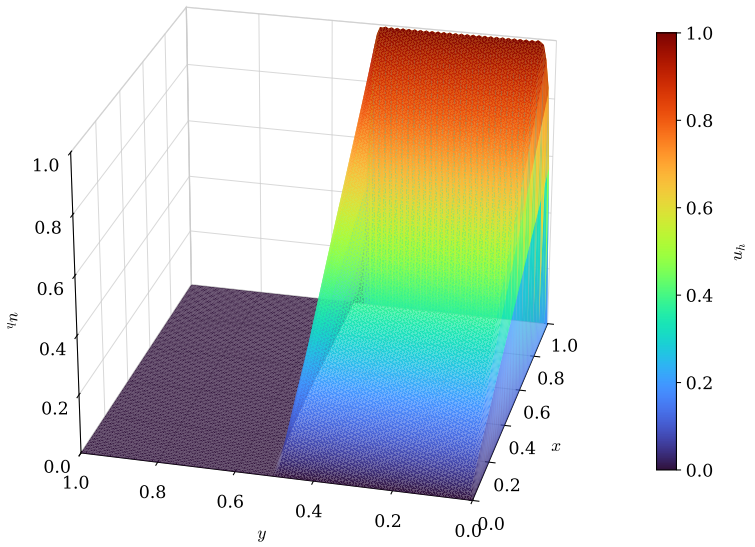


Figure 6.9.: Representative numerical solution for the layer solution example on a 128×128 Chevron grid.

For the numerical comparisons, two meshes considered in [BJK25, Example 10.109] are used:

Chevron grid (64×64), acute grid at refinement level 4.

The Chevron grid has 4225 total degrees of freedom and 3969 active degrees of freedom. The acute grid has 1857 total degrees of freedom and 1729 active degrees of freedom.

The Chevron grid has already been shown in Figure 6.2. The acute grid used in the present experiment is shown in Figure 6.10.

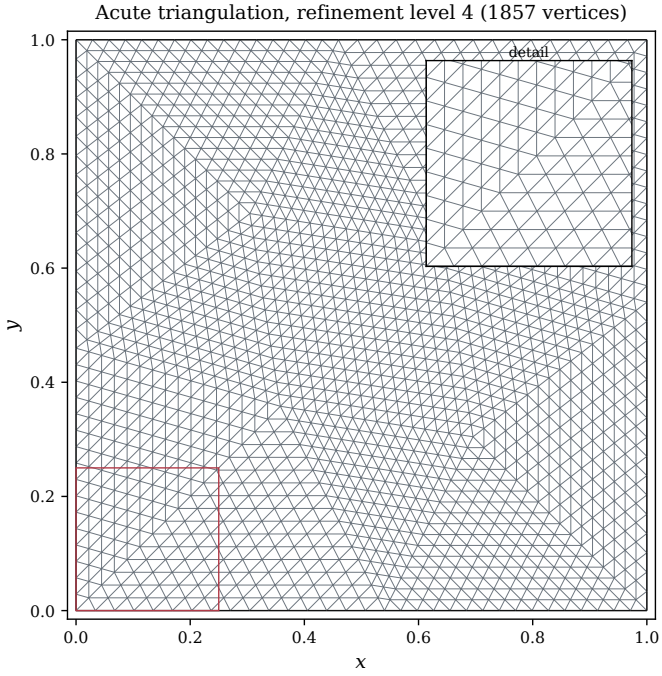


Figure 6.10.: Layer solution example: acute grid at refinement level 4.

The Chevron grid shown in Figure 6.2 contains a grid line aligned with the interior layer at $y = 0.5$. By contrast, the acute grid shown in Figure 6.10 does not contain edges aligned with this layer. The two meshes therefore represent different interactions between the triangulation and the layer structure.

For Algorithms 3 and 4, the prescribed projection interval is

$$0 \leq u_h \leq 1.$$

Since the exact solution is not known analytically, the assessment of accuracy is based on the reference profiles for $\varepsilon = 10^{-5}$ given in [BJK25, Fig. 2.8, p. 42]. The numerical solution is evaluated along the two cross-sections

$$u_h(x, 0.25), \qquad 0 \leq x \leq 1, \qquad \text{and} \qquad u_h(0.75, y), \qquad 0 \leq y \leq 1.$$

The first cross-section displays the outflow layer near $x = 1$, whereas the second intersects the characteristic layers near $y = 0$ and $y = 0.5$.

Let Γ_{cs} denote one of these cross-sections and let u_{ref} denote the corresponding reference profile. The cross-section errors are measured by

$$\|u_h - u_{\text{ref}}\|_{L^1(\Gamma_{\text{cs}})}, \qquad \|u_h - u_{\text{ref}}\|_{L^2(\Gamma_{\text{cs}})}, \qquad \|u_h - u_{\text{ref}}\|_{L^\infty(\Gamma_{\text{cs}})}.$$

Since the exact solution is not known analytically, no global FEM errors are considered. The numerical assessment therefore focuses on the computed profiles, their errors with respect to the reference profiles, the nonlinear iteration counts, and the preservation of the expected solution bounds.

6.3.1. Experiments on the Chevron Grid

Figures 6.11 and 6.12 compare the computed profiles with the reference cross-sections. Figure 6.13 shows the corresponding L^1, L^2, and L^∞ errors as functions of the actual number of nonlinear iterations. The numerical values are listed in Tables A.7 and A.8.

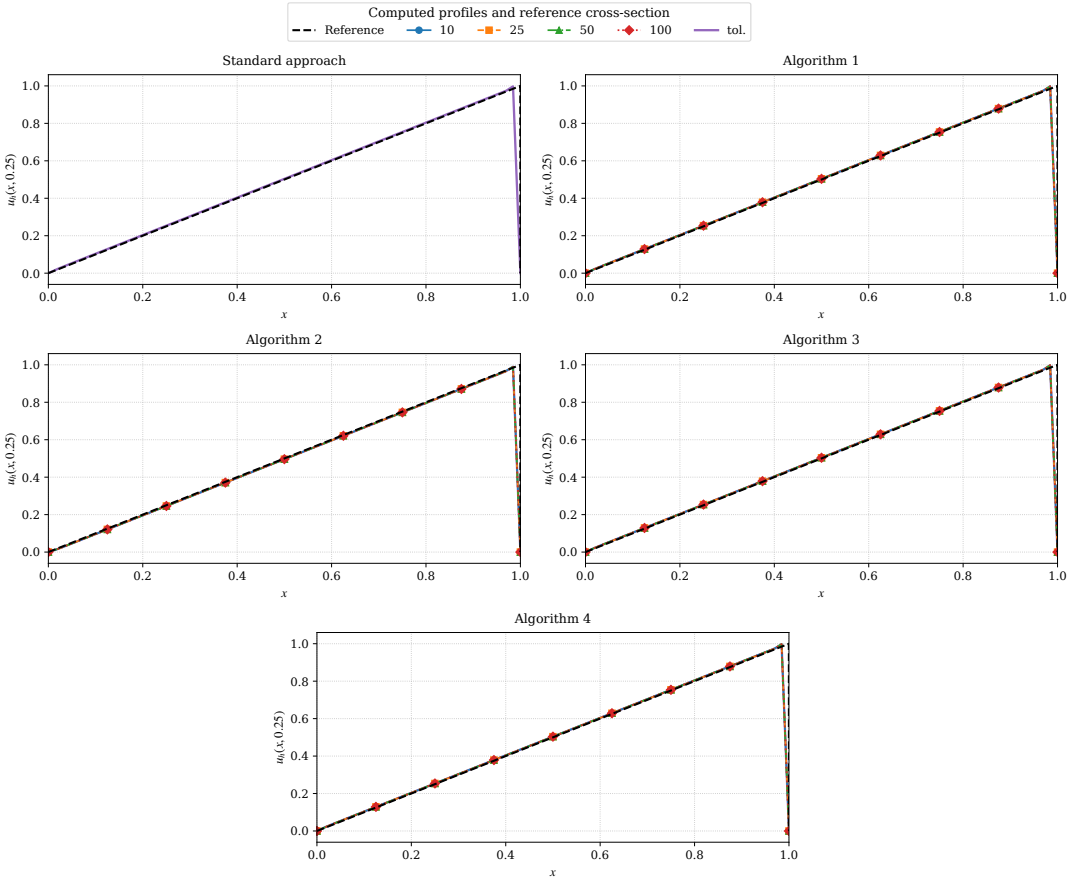


Figure 6.11.: Layer problem on the Chevron grid: computed profiles and reference cross-section at $y = 0.25$.

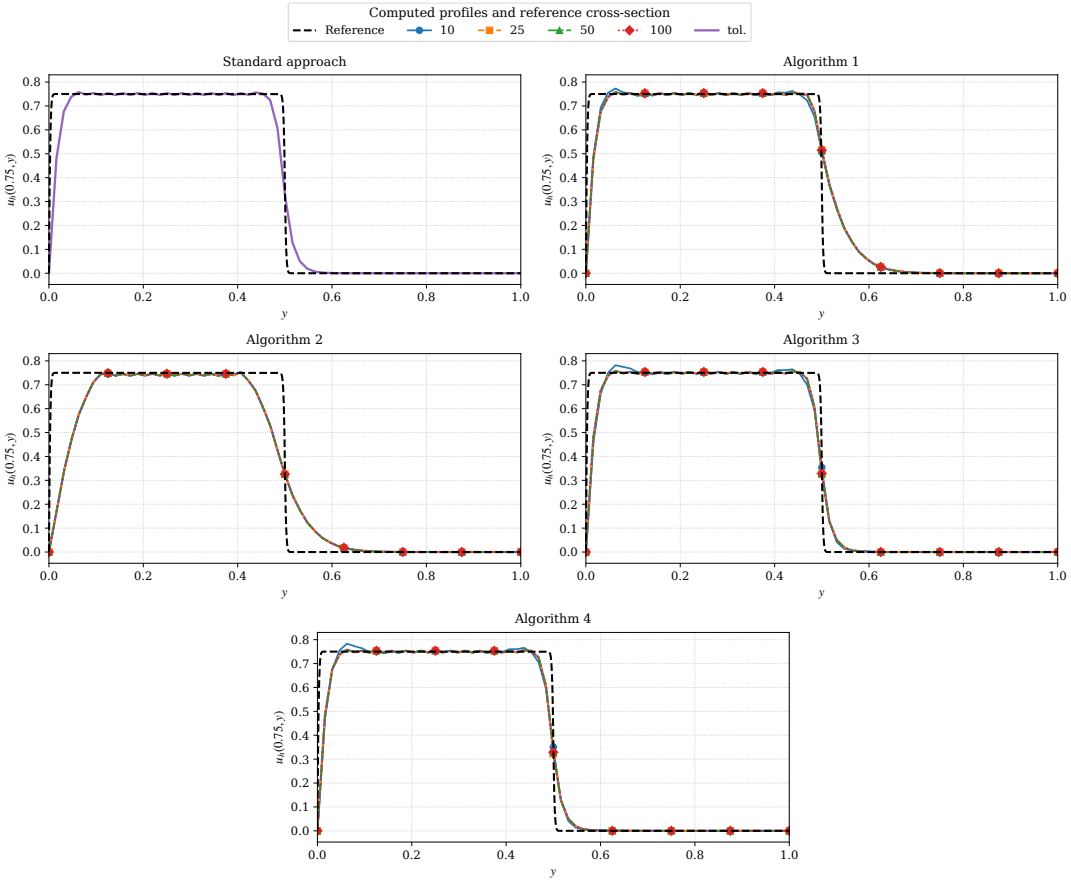


Figure 6.12.: Layer problem on the Chevron grid: computed profiles and reference cross-section at $x = 0.75$.

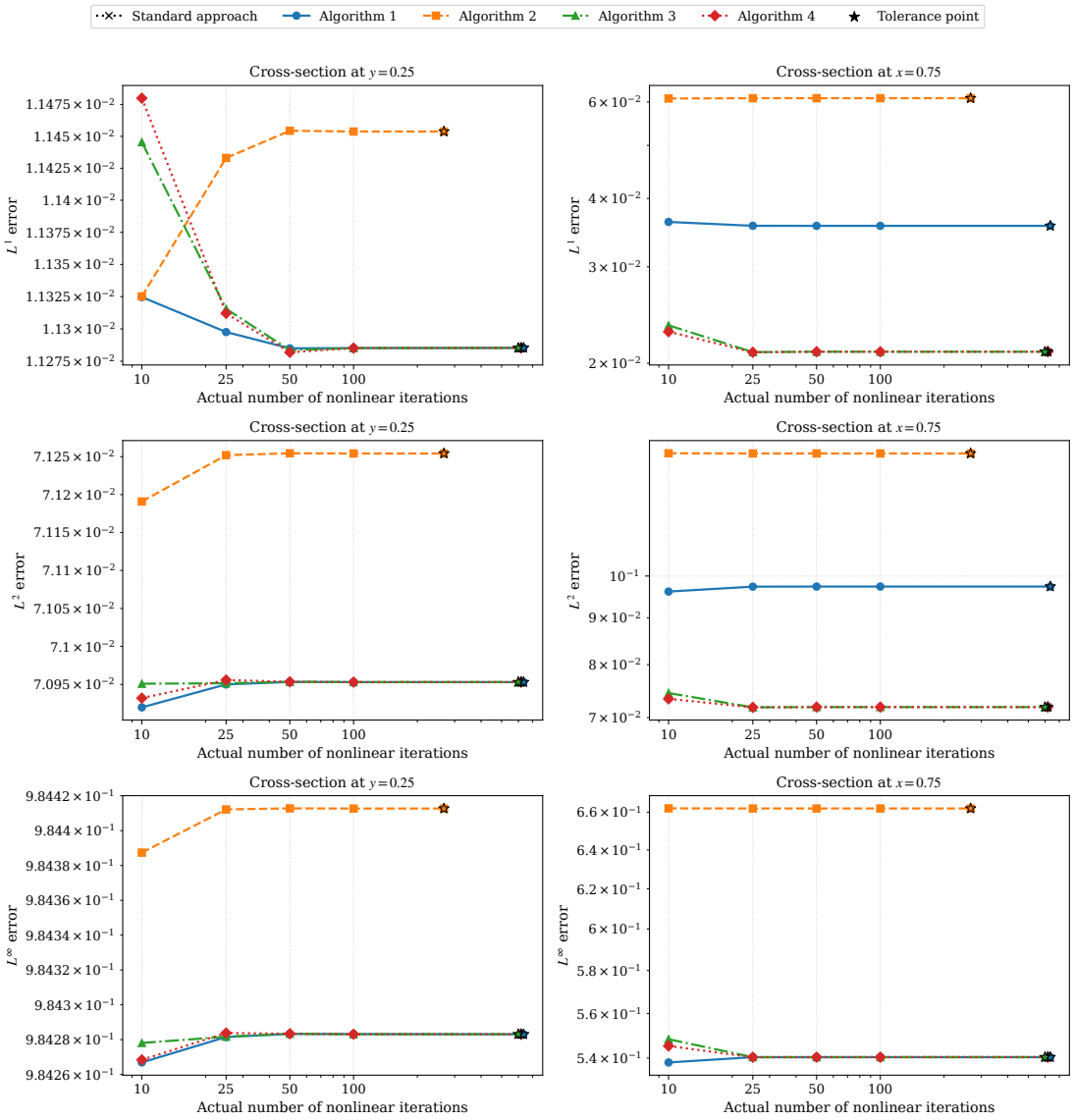


Figure 6.13.: Layer problem on the Chevron grid: cross-section errors with respect to the reference profiles. The horizontal axis gives the actual number of nonlinear iterations, and a star marks a tolerance run.

The observations for the Chevron grid, based on Figures 6.11–6.13 and Tables A.7–A.8, can be summarized as follows.

1. On the cross-section $y = 0.25$, the computed solutions reproduce the approximately linear part of the reference profile. However, the narrow outflow layer near $x = 1$ is not resolved, since its width is smaller than the local mesh size. Consequently, the discrepancy from the reference profile is dominated by the unresolved layer region.

2. The cross-section $x = 0.75$ distinguishes the algorithms more clearly. The reference profile contains narrow transitions near $y = 0$ and $y = 0.5$, whereas the computed profiles spread these transitions over wider intervals. These layers cannot be represented more sharply than permitted by the local mesh width.

3. After approximately 25 nonlinear iterations, the cross-section errors no longer decrease sub-

stantially. At this stage, the errors produced by Algorithms 3 and 4 are already very close to those of the standard approach. At their tolerance stopping points, the three approaches give the same reported errors to the displayed precision.

4. Algorithm 1 remains close to the standard approach on $y = 0.25$, but produces a more visible deviation on $x = 0.75$.

5. Algorithm 2 produces the strongest smearing. On $x = 0.75$, its tolerance L^2 error is approximately 1.89 times the corresponding error of the standard approach. Since the error has already reached a plateau after approximately 25 iterations, this difference is not caused by insufficient nonlinear convergence. Instead, Algorithm 2 converges to a more diffusive discrete solution. This is an important recurring observation in the numerical examples with layers.

6.3.2. Experiments on the Acute Grid

Figures 6.14 and 6.15 show the computed and reference profiles on the acute grid. The corresponding cross-section errors are shown in Figure 6.16 and listed in the subsequent tables.

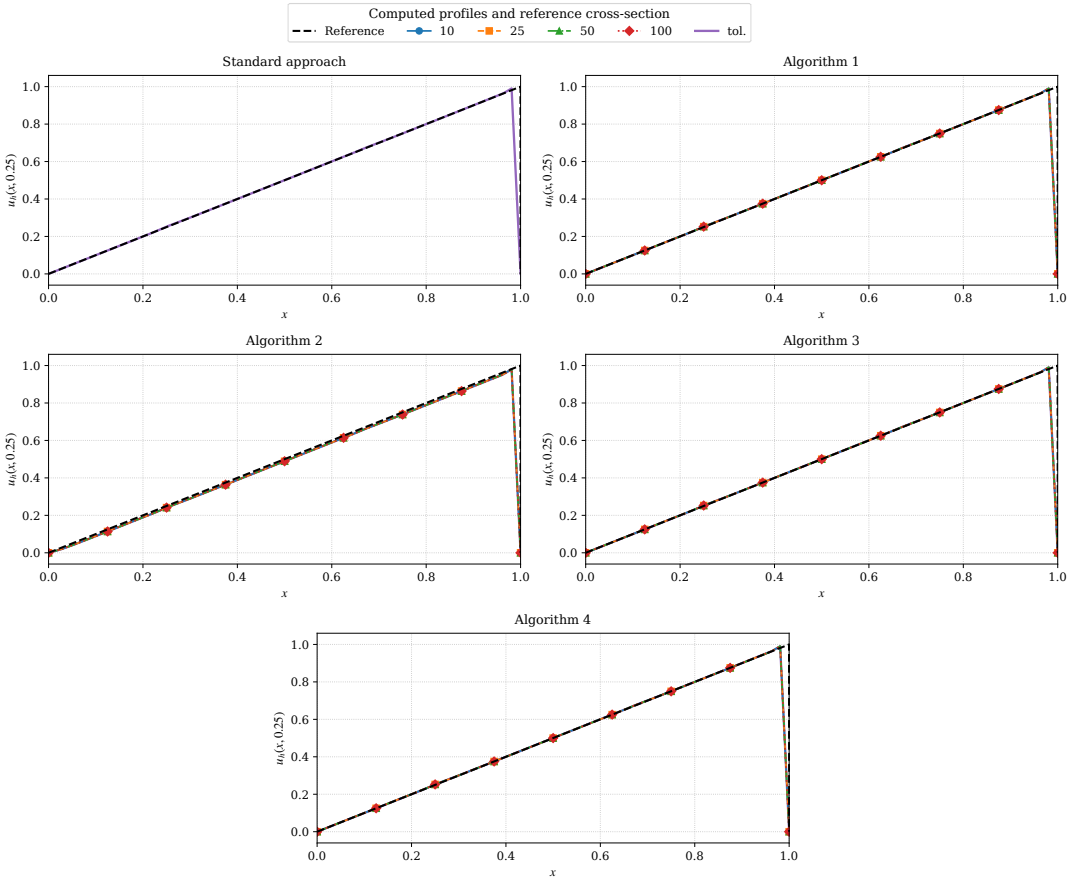


Figure 6.14.: Layer problem on the acute grid: computed profiles and reference cross-section at $y = 0.25$.

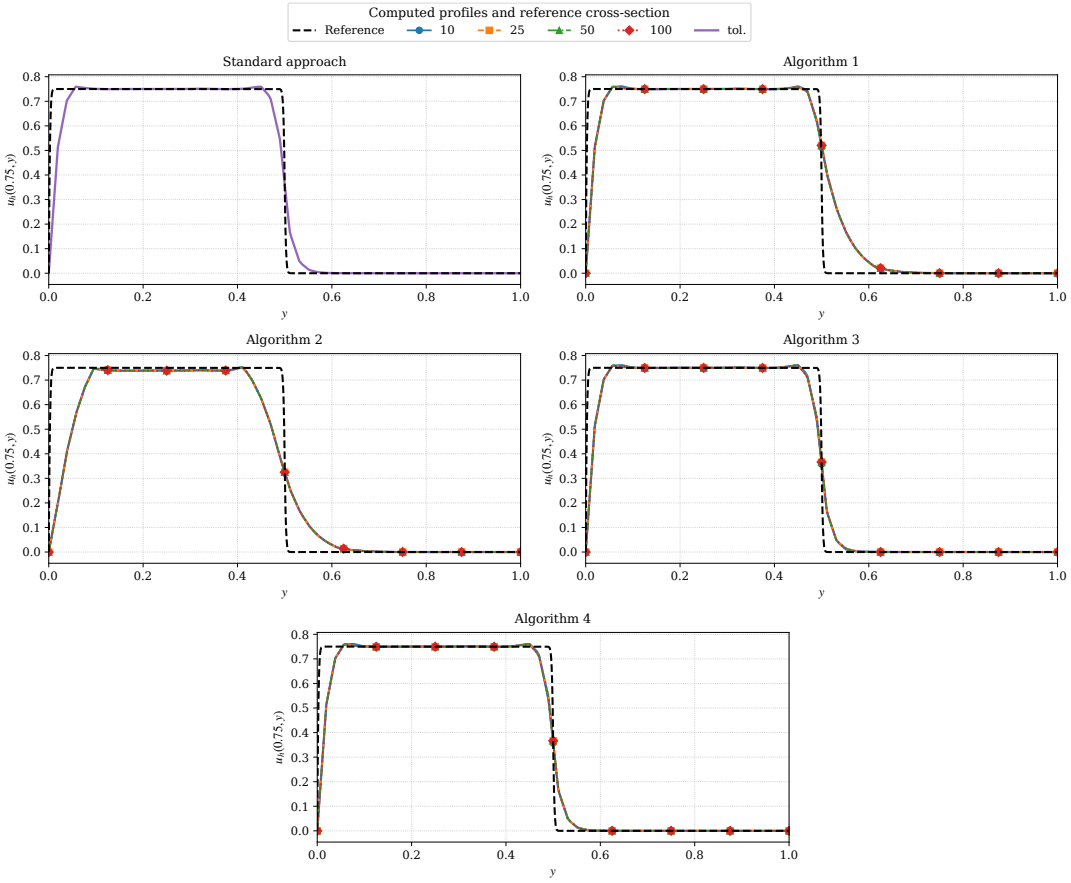


Figure 6.15.: Layer problem on the acute grid: computed profiles and reference cross-section at $x = 0.75$.

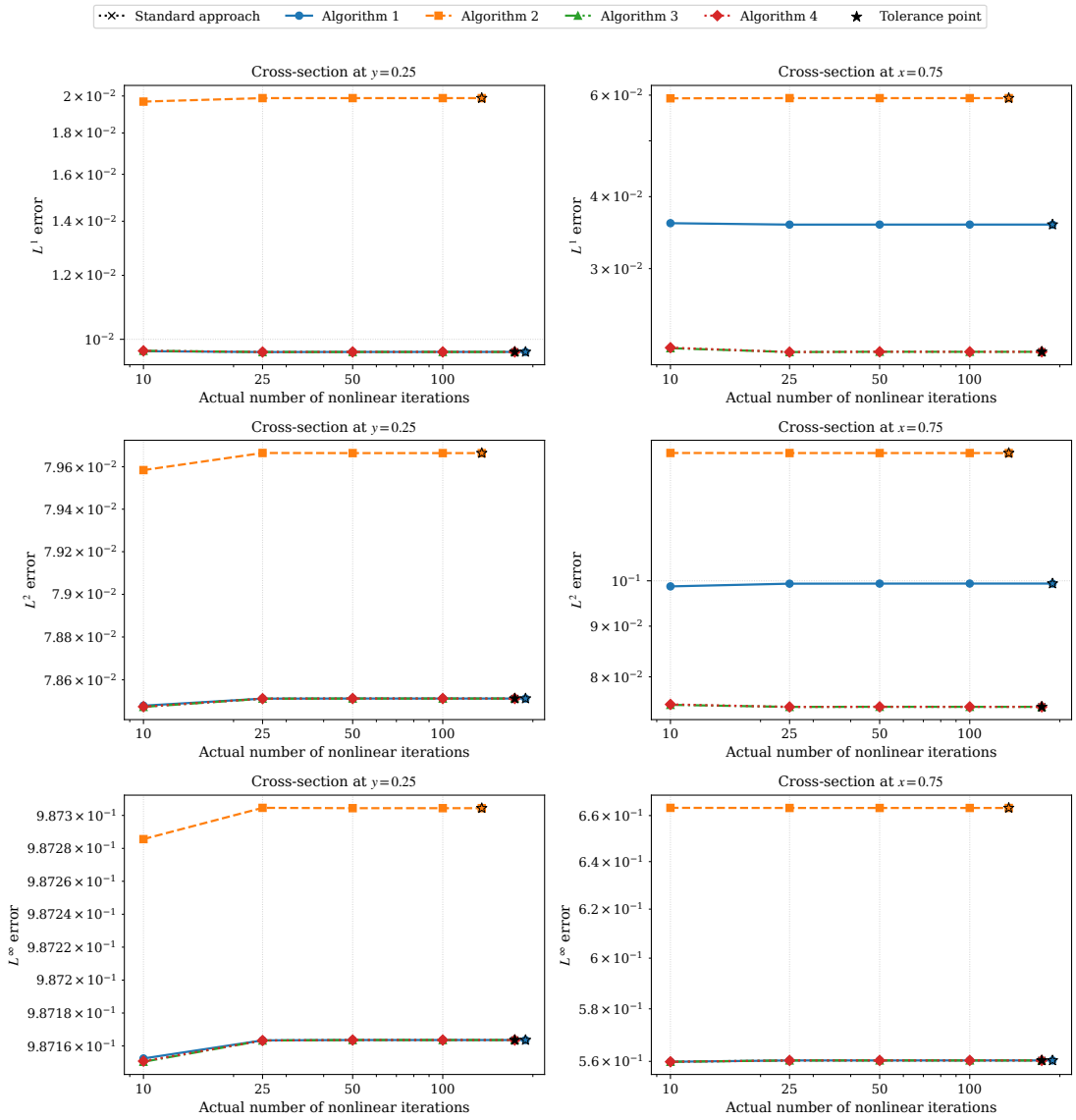


Figure 6.16.: Layer problem on the acute grid: cross-section errors with respect to the reference profiles. The horizontal axis gives the actual number of nonlinear iterations, and a star marks a tolerance run.

Based on Figures 6.14–6.16, and Tables A.9–A.10, the observations for the acute grid can be summarized as follows.

1. The qualitative behavior is similar to that on the Chevron grid. The outflow layer on $y = 0.25$ is not resolved, and the transitions on $x = 0.75$ are spread over wider intervals than in the reference profile.

2. At tolerance, Algorithms 3 and 4 give the same reported cross-section errors as the standard approach to the displayed precision. On this mesh, all three methods also terminate after 174 nonlinear iterations.

3. Algorithm 1 is again close to the standard approach on $y = 0.25$, but its L^1 and L^2 errors are larger on $x = 0.75$. This confirms that the second cross-section is the more sensitive diagnostic.

4. Algorithm 2 gives the largest deviations from the reference profiles. On $x = 0.75$, its tolerance L^1 and L^2 errors are approximately 2.76 and 1.80 times the corresponding errors of the standard approach.

5. The same behavior is observed on both geometrically different meshes. Hence, the stronger smearing produced by Algorithms 1 and 2 is not specific to one triangulation.

6.3.3. Nonlinear Convergence and Bound Behavior

This subsection summarizes the nonlinear convergence behavior and the preservation of the expected solution bounds for the layer problem. The data reported in the following tables are selected from the complete minimum and maximum value tables in Appendix A.2.3, in particular from Tables A.11 and A.12.

Table 6.2 gives the actual nonlinear iteration numbers in the tolerance runs.

Table 6.2.: Layer problem: actual nonlinear iteration numbers in the tolerance runs.

Mesh	Standard	Alg. 1	Alg. 2	Alg. 3	Alg. 4
Chevron grid 64×64	618	635	267	599	618
Acute grid, refinement level 4	174	189	135	174	174

The minimum and maximum values of the tolerance solutions are listed in Table 6.3. Negative values of order 10^{-16} are interpreted as round-off errors.

Table 6.3.: Layer problem: minimum and maximum values of the tolerance solutions.

Mesh	Method	Minimum value	Maximum value
Chevron	Standard	$-1.1098 \cdot 10^{-16}$	0.999624
Chevron	Algorithm 1	$-1.1096 \cdot 10^{-16}$	0.999418
Chevron	Algorithm 2	$-1.0999 \cdot 10^{-16}$	0.990679
Chevron	Algorithm 3	$-1.1098 \cdot 10^{-16}$	0.999624
Chevron	Algorithm 4	$-1.1098 \cdot 10^{-16}$	0.999624
Acute	Standard	0	0.999337
Acute	Algorithm 1	0	0.997144
Acute	Algorithm 2	$-1.0673 \cdot 10^{-17}$	0.983073
Acute	Algorithm 3	0	0.999337
Acute	Algorithm 4	0	0.999337

All tolerance solutions satisfy the expected bounds up to round-off. To distinguish the tolerance behavior from the behavior under early stopping, Table 6.4 lists the only relevant upper-bound violations found in the complete data set.

Table 6.4.: Layer problem: relevant upper-bound violations under early stopping. Negative values of order 10^{-16} are interpreted as round-off errors.

Mesh	Method	$N_{\max}$	Minimum value	Maximum value
Chevron grid 64×64	Algorithm 1	10	$-1.1362 \cdot 10^{-16}$	1.023356
Chevron grid 64×64	Algorithm 1	25	$-1.1130 \cdot 10^{-16}$	1.002502
Acute grid, refinement level 4	Algorithm 1	10	0	1.002951

Tables 6.2–6.4 summarize the main convergence and bound data, while the complete minimum and maximum value data are given in Appendix A.2.3. The convergence and bound results can be summarized as follows.

1. No negative undershoot beyond round-off is observed. All tolerance solutions satisfy the interval $[0, 1]$ up to round-off.
2. Algorithm 1 is the only method with a relevant upper-bound violation. These overshoots occur only under early stopping and disappear after 50 iterations on the Chevron grid and after 25 iterations on the acute grid.
3. Algorithms 3 and 4 reproduce the minimum and maximum values of the standard approach at tolerance to the displayed precision. Algorithm 4 also has the same tolerance iteration count as the standard approach. Algorithm 3 needs fewer nonlinear iterations on the Chevron grid, but the same number of iterations on the acute grid.
4. Algorithm 2 gives the largest reduction in the number of nonlinear iterations and preserves the interval $[0, 1]$ in every reported run. However, the smaller maximum values in Table 6.3 and the larger cross-section errors reported above show that this faster convergence is accompanied by stronger damping and a less accurate representation of the layer profiles.
5. Therefore, preservation of the bounds and fast nonlinear convergence do not by themselves imply an accurate resolution of the internal layer. The cross-section errors have to be considered together with the nonlinear iteration counts and the minimum–maximum values.

Overall, the results on both meshes lead to the same qualitative conclusions. Algorithms 3 and 4 preserve the expected bounds without noticeably changing the accuracy of the standard approach, and their cross-section errors are already close to the final values after relatively few nonlinear iterations. Algorithm 1 produces only moderate additional smearing, whereas Algorithm 2 reduces the nonlinear iteration count most strongly but converges to a more diffusive discrete solution. Thus, for problems with layers, fast nonlinear convergence and bound preservation must be assessed together with the accuracy of the computed layer profiles.

6.4. Hemker Problem

The final experiment considers the Hemker problem, a standard benchmark for steady convection-diffusion equations introduced in [Hem96]. The bounded obstacle configuration used here follows [JKP23, Example 3.2]. The computational domain is

$$\Omega = \big((-3, 9) \times (-3, 3)\big) \setminus \big\{(x, y) : x^2 + y^2 \leq 1\big\}.$$

The problem is given by

$$-\varepsilon \Delta u + \mathbf{b} \cdot \nabla u = 0 \quad \text{in } \Omega, \qquad \varepsilon = 10^{-4}, \qquad \mathbf{b} = (1, 0)^{\mathrm{T}}.$$

The direction of convection is therefore from left to right. Dirichlet boundary conditions are imposed on the inflow boundary and on the circular obstacle:

$$u = 0 \quad \text{on } \{x = -3\}, \qquad u = 1 \quad \text{on } \left\{(x, y) : x^2 + y^2 = 1\right\}.$$

On the remaining outer boundary, homogeneous Neumann boundary conditions are prescribed,

$$\varepsilon \nabla u \cdot \mathbf{n} = 0.$$

A boundary layer is formed on the upstream side of the circular obstacle. Moreover, two interior layers originate near the upper and lower points of the circle and extend downstream in the direction of convection. This characteristic layer structure is qualitatively visible in the numerical solution shown in Figure 6.18. Since the downstream layers remain thin over a large part of the domain, the Hemker problem provides a demanding test of whether the modified iterations preserve the resolution of the layer structure.

All solution-field plots use the same displayed color range $0 \le u_h \le 1$. This restriction applies only to the color mapping; the actual minimum and maximum values are considered separately below.

The general settings are those described in Section 6.1. For Algorithms 3 and 4, the prescribed bounds are

$$u_{\min} = 0, \qquad u_{\max} = 1.$$

Since $c = 0$, $f = 0$, the Neumann boundary data are homogeneous, and $\mathbf{b} \cdot \mathbf{n} \ge 0$ on the Neumann part of the boundary, the weak solution satisfies

$$a(u, v) = 0 \qquad \text{for all } v \in V_0.$$

Hence, both implications in Theorem 2.11 are applicable. Since the prescribed Dirichlet values are 0 and 1, it follows that

$$0 \le u \le 1 \qquad \text{almost everywhere in } \Omega.$$

The initial grid at refinement level 0 is the Delaunay grid shown in [JKP23, Fig. 6]. It was generated using Gmsh [GR09] and satisfies the Delaunay property. In the present experiment, only refinement level 4 is considered. The resulting mesh consists of 190 208 triangles and has 95 784 total degrees of freedom, of which 95 383 are active degrees of freedom. For this mesh, the normalized stopping tolerance implemented in the code is $3.094899 \cdot 10^{-10}$. The refined triangulation used in the computations is shown in Figure 6.17.

Following [ACF+11] and [JKP23, Example 3.2], the downstream layer is assessed by its width at $x = 4$. For $\varepsilon = 10^{-4}$, the reported reference value is

$$w_{\text{ref}}(4) = 0.0723.$$

The same layer-width diagnostic is applied to the computed cross-section $u_h(4, y)$. If $w_h(4)$ denotes the resulting numerical layer width, its absolute difference from the reference value is

$$\left|w_h(4) - w_{\text{ref}}(4)\right|.$$

No exact solution is used as a reference for computing global finite element errors for the bounded-domain problem considered here. Instead, the numerical assessment is based on the computed solution fields, cross-sectional profiles at $x = 4$, the corresponding layer widths, nonlinear convergence behavior, and preservation of the prescribed discrete bounds.

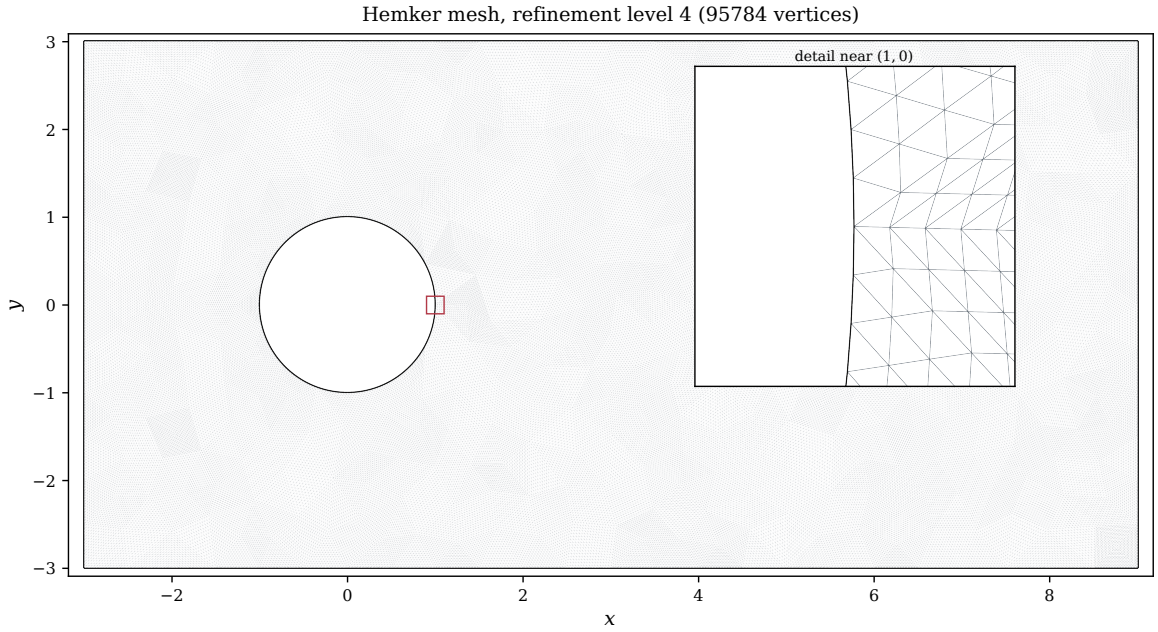


Figure 6.17.: Hemker problem: triangulation at refinement level 4.

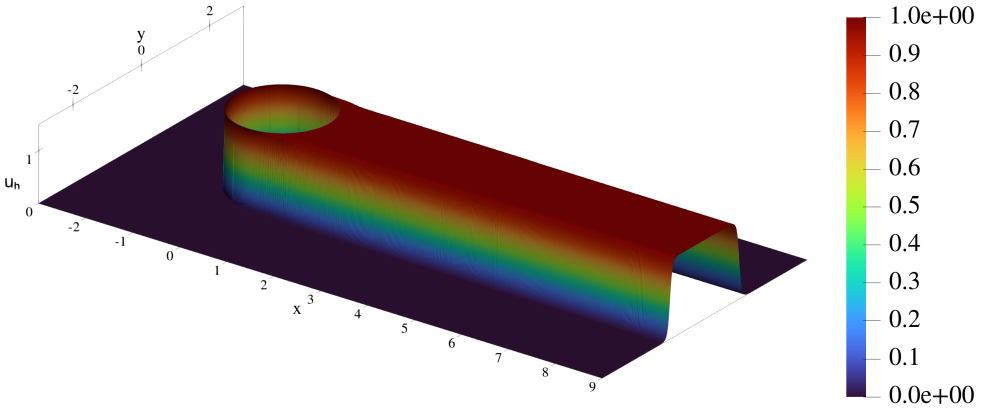


Figure 6.18.: Hemker problem at refinement level 4: three-dimensional visualization of the standard-approach solution after the prescribed maximum of 10000 nonlinear iterations.

6.4.1. Downstream Layer Resolution

The downstream layers are examined at three complementary levels. First, the two-dimensional solution fields show the overall structure of the computed solutions. Second, the cross-section at $x = 4$ gives a direct comparison of the transition profiles. Finally, the measured layer widths provide a scalar quantity for assessing the sharpness of these transitions.

For Algorithms 1 and 2, the final available solutions are the tolerance solutions obtained after 574 and 49 nonlinear iterations, respectively. The standard approach and Algorithms 3 and 4 do not satisfy the nonlinear stopping criterion within 10000 iterations. Their final available solutions are therefore the solutions at the prescribed maximum iteration number and must not be interpreted as converged nonlinear solutions.

Figure 6.19 compares the corresponding two-dimensional solution fields. All methods reproduce the expected qualitative flow pattern around the circular obstacle. Since the main visible differences concern the widths of the two interior layers, Figure 6.20 shows the cross-section $u_h(4, y)$ for a more direct comparison of the transition profiles.

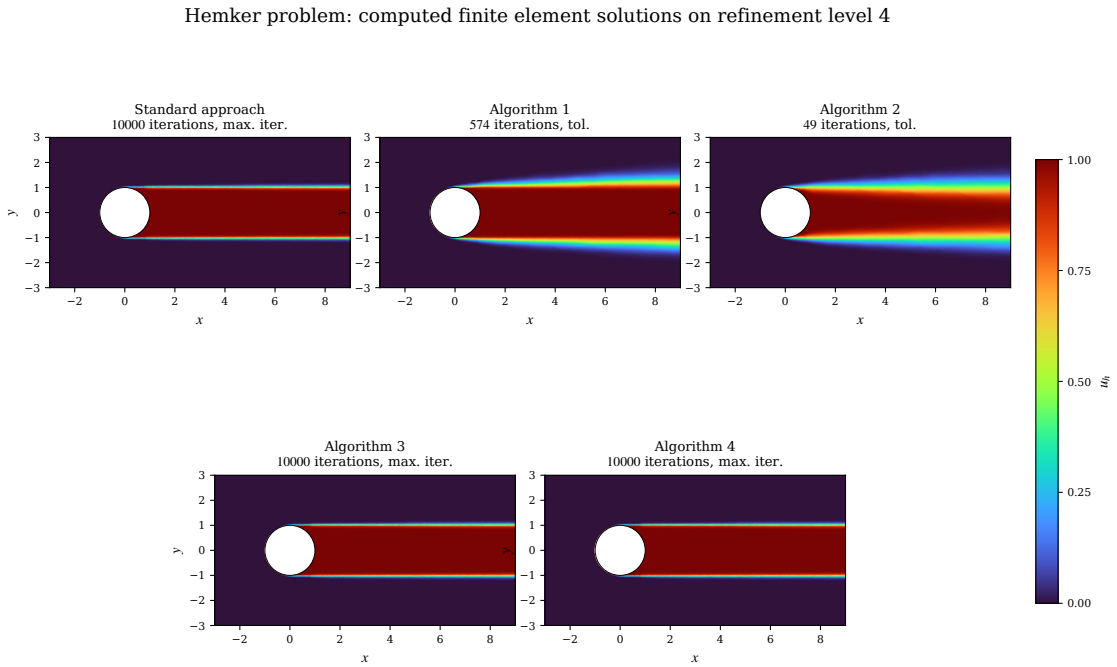


Figure 6.19.: Hemker problem at refinement level 4: final available solution fields. Algorithms 1 and 2 reach the nonlinear tolerance, whereas the standard approach and Algorithms 3 and 4 stop at $N_{\max} = 10000$.

The cross-section confirms the differences suggested by the solution fields and provides the data from which the upper layer width is measured. Figure 6.21 shows this width as a function of the actual number of nonlinear iterations. The black dotted line gives the value obtained with the standard approach after 10000 iterations, and the grey dashed line gives the reference value 0.0723 reported in [JKP23, Example 3.2].

Figure 6.22 shows the corresponding absolute differences from the reference value. The results for the final available solutions are summarized in Table 6.5. The complete layer-width data for all stopping conditions are given in Table A.13 in Appendix A.3.

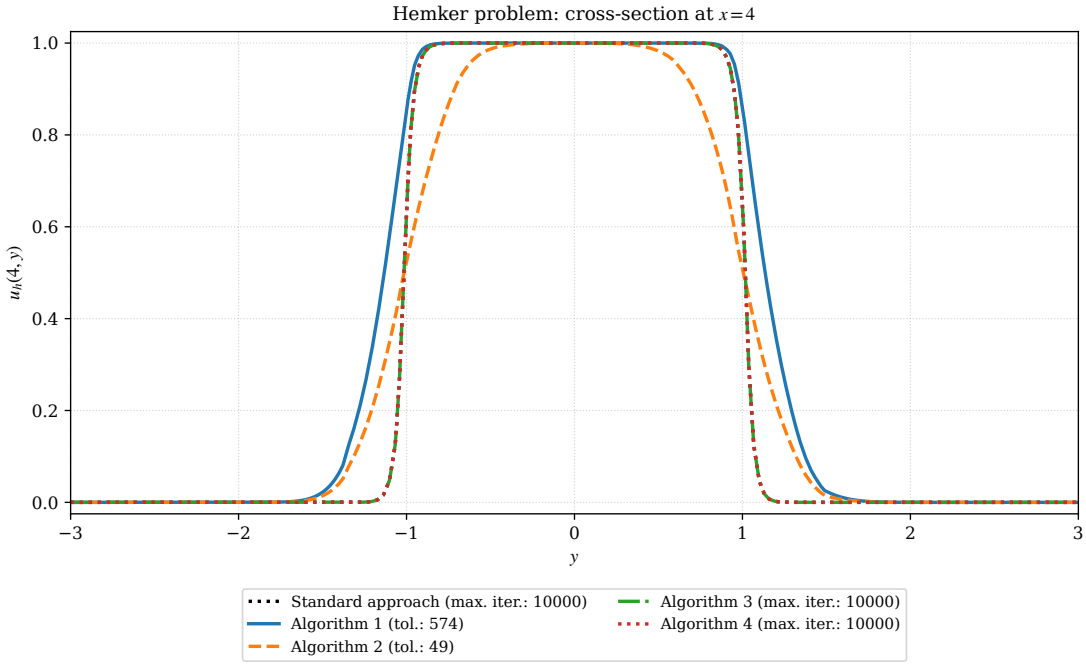


Figure 6.20.: Hemker problem at refinement level 4: cross-section profiles $u_h(4, y)$ for the final available solutions. Algorithms 1 and 2 reach the nonlinear tolerance, whereas the standard approach and Algorithms 3 and 4 stop at $N_{\max} = 10000$.

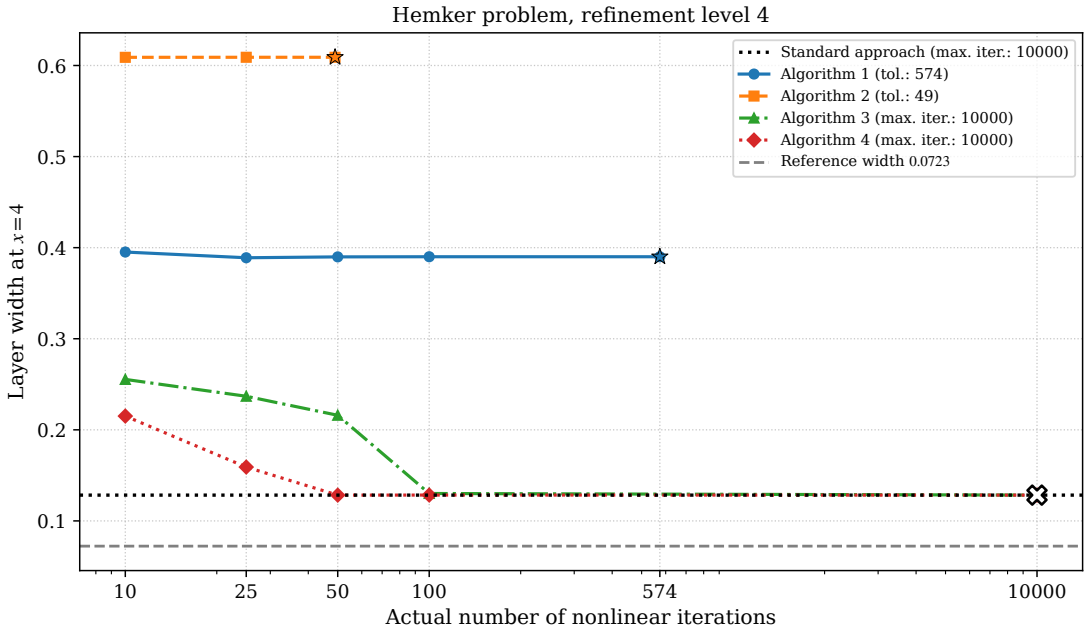


Figure 6.21.: Hemker problem at refinement level 4: layer width at $x = 4$. A star marks a run that reaches the nonlinear tolerance, whereas a cross marks a run stopped at the prescribed maximum iteration number.

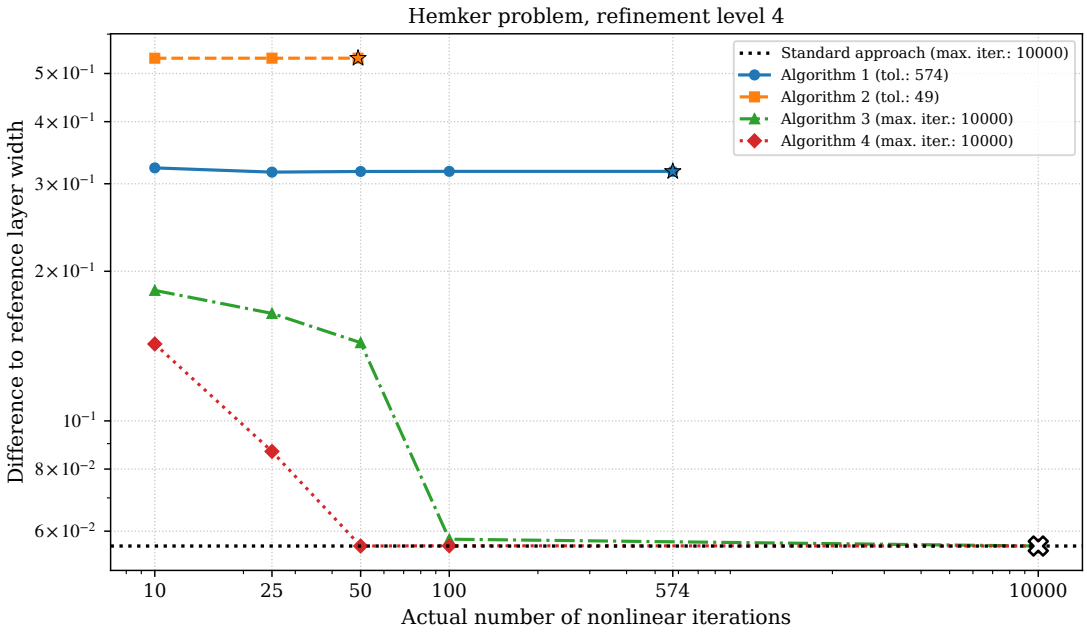


Figure 6.22.: Hemker problem at refinement level 4: absolute difference between the computed layer width and the reference value 0.0723.

Table 6.5.: Hemker problem at refinement level 4: layer widths of the final available solutions and their absolute differences from the reference value 0.0723.

Method	Status	Actual iter.	$w_h(4)$	$\lvert w_h(4) - w_{\text{ref}}(4)\rvert$
Standard	max. iter.	10000	0.12831	0.05601
Algorithm 1	tol.	574	0.39003	0.31773
Algorithm 2	tol.	49	0.60903	0.53673
Algorithm 3	max. iter.	10000	0.12831	0.05601
Algorithm 4	max. iter.	10000	0.12831	0.05601

The observations from Figures 6.19–6.22 and Table 6.5 can be summarized as follows.

1. The standard approach and Algorithm 4 produce indistinguishable two-dimensional solution fields and cross-section profiles at 10000 nonlinear iterations. Algorithm 3 is also visually very close to them. All three methods give the layer width 0.12831, whose absolute difference from the reference value is 0.05601. Since none of these runs satisfies the nonlinear stopping criterion, these values belong to the final available iterates rather than to converged nonlinear solutions.

2. Algorithms 3 and 4 have different early-stopping behavior. For Algorithm 3, the layer width decreases from 0.25536 after 10 iterations to 0.13011 after 100 iterations. Algorithm 4 reaches the value 0.12831 already after 50 iterations.

3. Algorithm 1 preserves the plateau value close to one, but its transition regions are substantially wider than those of the standard approach. Its tolerance solution has the layer width 0.39003, and this value changes only slightly after 25 iterations.

4. Algorithm 2 reaches the tolerance after only 49 iterations, but its profile has the most gradual transitions. Its layer width is 0.60903, with an absolute difference 0.53673 from the reference

value. Thus, it produces the strongest smearing among the tested methods.

6.4.2. Nonlinear Convergence and Bound Behavior

Figure 6.23 shows the final normalized nonlinear residual for each run. The horizontal dashed line denotes the normalized stopping tolerance. The final stopping data are summarized together with the minimum and maximum values in Table 6.6. The complete nonlinear stopping data for all runs are listed in Table A.14 in Appendix A.3.

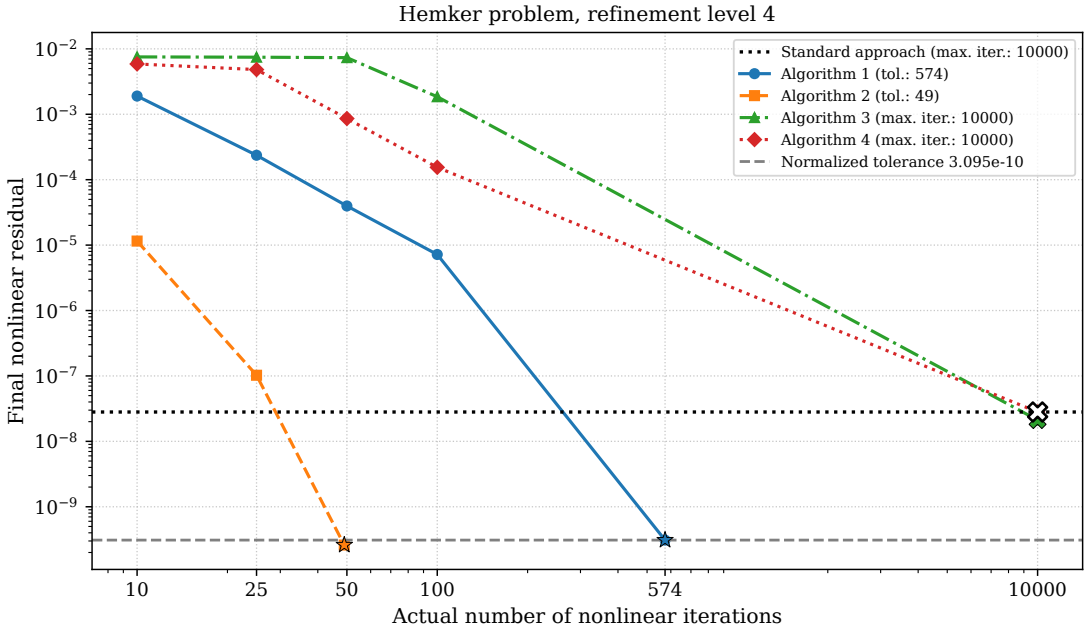


Figure 6.23.: Hemker problem at refinement level 4: final normalized nonlinear residual as a function of the actual number of nonlinear iterations.

The nonlinear convergence results can be summarized as follows.

1. Algorithm 2 reaches the prescribed stopping tolerance after 49 nonlinear iterations. It is the fastest method in this experiment.

2. Algorithm 1 reaches the stopping tolerance after 574 iterations. Its convergence is slower than that of Algorithm 2, but it still converges within the prescribed maximum number of iterations.

3. The standard approach and Algorithms 3 and 4 do not satisfy the stopping criterion within 10000 iterations. The final residual is $2.809370 \cdot 10^{-8}$ for the standard approach and Algorithm 4 and $2.117150 \cdot 10^{-8}$ for Algorithm 3. All three values remain above the normalized stopping tolerance $3.094899 \cdot 10^{-10}$.

4. Algorithm 4 has the same final residual and the same number of rejected nonlinear steps as the standard approach. Algorithm 3 has a slightly smaller final residual and one additional rejected step. Although all three methods give the same reported layer width, the final solution of Algorithm 3 is only numerically close to, and not identical with, the standard approach.

Figures 6.24 and 6.25 show the violations of the upper and lower bounds, respectively. The complete minimum and maximum values for all stopping conditions are listed in Table A.15 in Appendix A.3. Values of order 10^{-16} are interpreted as round-off errors.

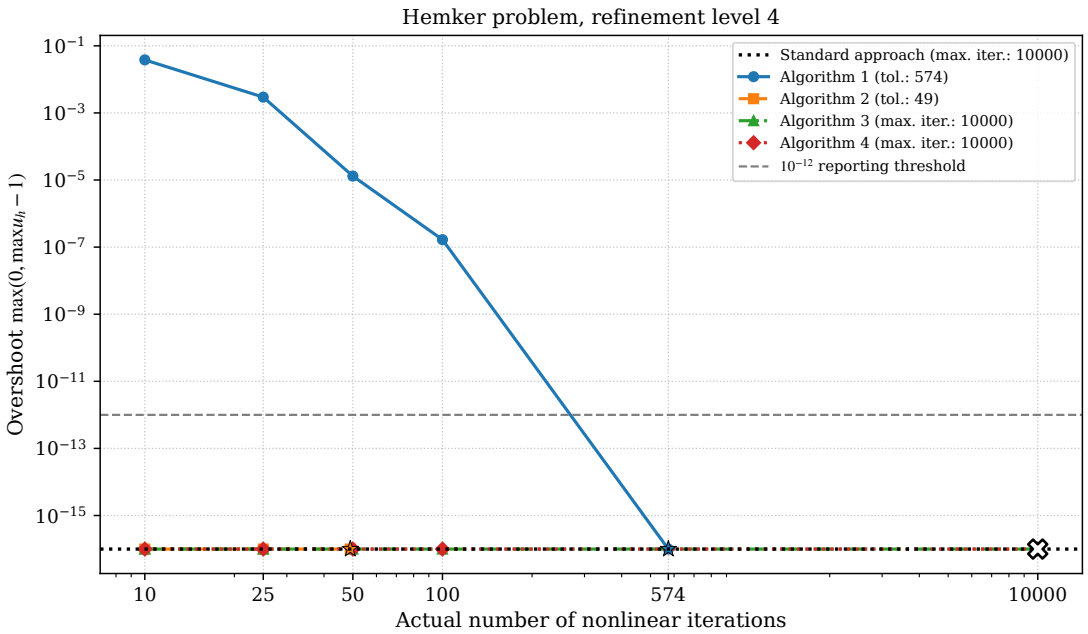


Figure 6.24.: Hemker problem at refinement level 4: violation of the expected upper bound $u_h \leq 1$.

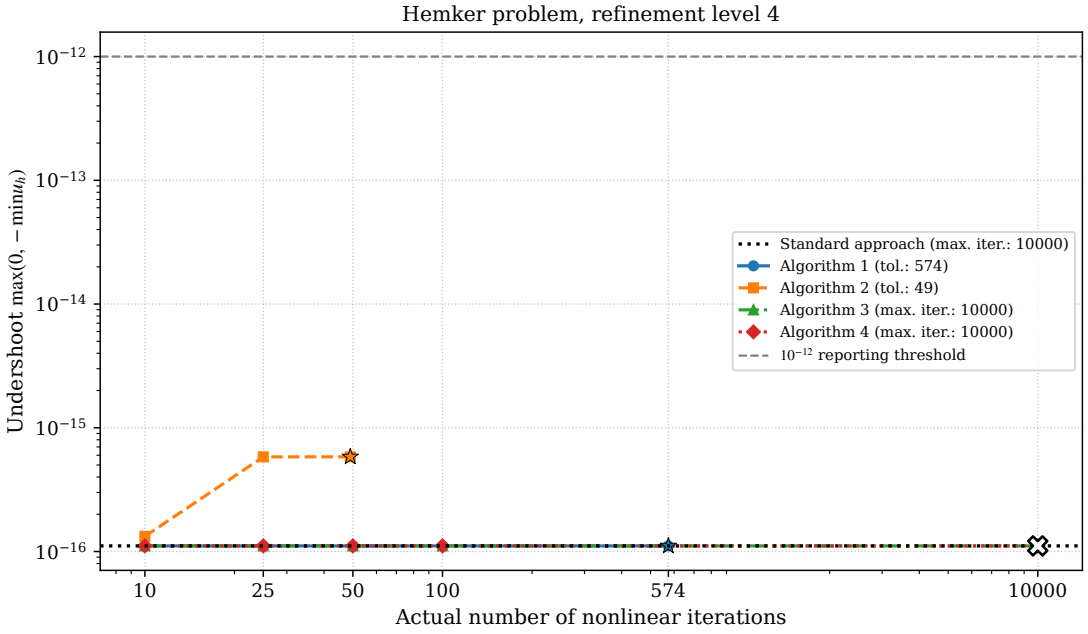


Figure 6.25.: Hemker problem at refinement level 4: violation of the expected lower bound $u_h \geq 0$.

Table 6.6.: Hemker problem at refinement level 4: nonlinear stopping data and bounds of the final available solutions.

Method	Status	Actual iter.	Final residual	Rejections	Minimum	Maximum
Standard	max. iter.	10000	$2.809370 \cdot 10^{-8}$	47	$-1.1102 \cdot 10^{-16}$	1
Algorithm 1	tol.	574	$3.093490 \cdot 10^{-10}$	0	$-1.1102 \cdot 10^{-16}$	1
Algorithm 2	tol.	49	$2.600760 \cdot 10^{-10}$	0	$-5.8244 \cdot 10^{-16}$	1
Algorithm 3	max. iter.	10000	$2.117150 \cdot 10^{-8}$	48	$-1.1102 \cdot 10^{-16}$	1
Algorithm 4	max. iter.	10000	$2.809370 \cdot 10^{-8}$	47	$-1.1102 \cdot 10^{-16}$	1

The bound data can be summarized as follows.

1. Algorithm 1 preserves nonnegativity up to round-off. It has an upper overshoot under early stopping, but this overshoot decreases from about $3.81 \cdot 10^{-2}$ after 10 iterations to about $1.67 \cdot 10^{-7}$ after 100 iterations. No relevant upper-bound violation remains at the tolerance stopping point.

2. Algorithm 2 preserves both bounds up to round-off for all reported stopping conditions.

3. For Algorithms 3 and 4, the projection interval is prescribed as $[0, 1]$. Both algorithms satisfy the lower and upper bounds for every reported stopping condition, up to round-off errors of order 10^{-16}.

Overall, the present computations reveal a clear distinction between nonlinear convergence and the spatial resolution of the downstream layers. Algorithm 2 reaches the nonlinear stopping tolerance after only 49 iterations and preserves the expected bounds, but it produces the strongest smearing and the largest deviation from the reference layer width. Algorithm 1 also reaches the nonlinear tolerance and produces a smaller layer width than Algorithm 2, although its downstream layers remain substantially wider than the reference layer.

Among the tested methods, the standard approach and Algorithms 3 and 4 produce the smallest measured layer widths. After 10000 iterations, their final available iterates give the same reported layer width. Algorithm 4 reproduces the standard-approach solution up to round-off, whereas the solution obtained with Algorithm 3 is numerically close but not identical. Since none of these three runs satisfies the nonlinear stopping criterion within 10000 iterations, their results must be interpreted as maximum-iteration results rather than converged tolerance solutions.

7. Summary and Outlook

7.1. Summary

This thesis investigated fixed-point right-hand-side iterations for the nonlinear MUAS discretization of steady-state convection-diffusion-reaction problems. Four modifications of the standard fixed-point iteration were introduced and studied. The main objective was to determine whether intermediate iterates and solutions obtained by early termination preserve the expected bounds without significantly reducing numerical accuracy.

The numerical experiments showed that bound preservation, nonlinear convergence, and spatial accuracy have to be assessed separately. For the smooth prescribed-solution problem, Algorithms 3 and 4 coincided with the standard approach when the appropriate bounds $[-1, 1]$ were used. Moreover, the errors after approximately 25 to 50 nonlinear iterations were already close to those obtained after satisfying the nonlinear stopping criterion. By contrast, bounds derived from the Tabata upwind solution were too restrictive. Algorithm 3 did not reach the stopping tolerance within 10000 iterations, while the final projection in Algorithm 4 increased the errors.

For the layer problem, Algorithms 3 and 4 remained close to the standard approach. Algorithm 1 caused moderate additional smearing and small upper-bound violations under early termination. Algorithm 2 required the fewest nonlinear iterations and preserved the expected bounds up to round-off, but it also produced the largest deviations from the independent reference cross-sections.

A similar compromise was observed in the Hemker computations. Algorithm 2 reached the stopping tolerance after only 49 iterations but produced the widest downstream layers and the largest deviation from the reference layer width. Algorithm 1 converged after 574 iterations and caused less smearing than Algorithm 2. The standard approach and Algorithms 3 and 4 produced narrower layers, but they did not satisfy the nonlinear stopping criterion within 10000 iterations.

The principal conclusions can be summarized as follows.

1. None of the four modifications is uniformly superior. Algorithm 1 provides a simple positivity correction but does not control an upper bound. Algorithm 2 may substantially accelerate nonlinear convergence, but this improvement can be accompanied by increased numerical diffusion. Algorithms 3 and 4 enforce prescribed bounds, but their effectiveness depends on the choice of these bounds.

2. Bounds derived from a low-order initial solution are not necessarily suitable for the final nonlinear solution. If the prescribed interval is too restrictive, projection may interfere with nonlinear convergence or reduce numerical accuracy.

3. Early termination can be computationally reasonable. In several experiments, the relevant numerical quantities changed only slightly after 25 to 50 iterations. Reducing the nonlinear residual further did not necessarily improve the spatial discretization error or the resolution of layers.

Overall, the experiments demonstrate that satisfying a nonlinear stopping criterion or preserving prescribed bounds does not by itself guarantee an accurate discrete solution. A useful iteration strategy must balance nonlinear convergence, bound preservation, and the resolution of the relevant solution features.

7.2. Outlook

Several questions remain open. First, a more complete convergence analysis of the modified fixed-point iterations would be valuable. In particular, it should be investigated under which assumptions the modified iterations converge and whether their limits coincide with the solution of the original nonlinear MUAS system. This question is especially important for Algorithms 1 and 2, since their corrections act directly on the nonlinear algebraic equations.

For Algorithm 2, the sufficient positivity condition used in this thesis is deliberately restrictive. Positive flux-correction contributions are not used to compensate negative contributions. This construction provides a transparent positivity argument, but it can also suppress a substantial part of the anti-diffusive correction. A less restrictive damping strategy could distribute the available positivity budget between neighboring degrees of freedom and may reduce the additional smearing observed in the layer problem.

The automatic determination of reliable projection bounds is another important topic. The experiments show that the minimum and maximum values of the Tabata solution do not necessarily provide suitable bounds for the nonlinear MUAS solution. Future approaches could instead use bounds derived from the continuous maximum principle, local boundary and source data, or a local discrete maximum principle. Spatially varying bounds may also be less restrictive than a single global interval.

The stopping criterion for the nonlinear iteration deserves further study. The experiments indicate that the discretization error can become nearly stationary long before the nonlinear residual reaches the prescribed tolerance. An effective stopping criterion should therefore balance nonlinear and spatial errors. Establishing an estimator that relates the nonlinear residual to the change in relevant solution quantities could provide a mathematically justified early-stopping rule.

Finally, the numerical study could be extended to further mesh families, different diffusion parameters, other algebraic limiters, and three-dimensional problems. It would also be useful to compare the fixed-point right-hand-side iteration with accelerated fixed-point methods or suitable non-smooth Newton-type methods. Such investigations could clarify whether the observed balance between nonlinear convergence, bound preservation, and numerical diffusion persists in larger and more complex applications.

A. Detailed Numerical Tables

This appendix provides detailed numerical values for the experiments presented in Chapter 6. The main text contains the corresponding figures, summary tables, and discussion of the numerical behavior. The tables below give the complete values for the tested stopping levels.

A.1. Smooth Solution Example

This section contains the detailed error values for the smooth solution example discussed in Section 6.2. The errors are computed with respect to the prescribed exact solution.

A.1.1. Smooth Solution Example with Prescribed Bounds

This section contains the detailed error values for the smooth solution example with prescribed bounds $u_{\min} = -1$ and $u_{\max} = 1$, discussed in Section 6.2.1. The errors are computed with respect to the prescribed exact solution.

Table A.1.: Smooth solution problem with prescribed bounds $[-1, 1]$: numerical errors on the Friedrichs–Keller mesh of size 16×16.

Method	Stopping	Actual iter.	$L^2(\Omega)$ error	$H^1(\Omega)$ seminorm error	$L^\infty(\Omega)$ error
Alg. 3/4	10	10	$2.345305 \cdot 10^{-2}$	$6.566572 \cdot 10^{-1}$	$9.190530 \cdot 10^{-2}$
Alg. 3/4	25	25	$2.358664 \cdot 10^{-2}$	$6.692011 \cdot 10^{-1}$	$9.177270 \cdot 10^{-2}$
Alg. 3/4	50	50	$2.359719 \cdot 10^{-2}$	$6.697914 \cdot 10^{-1}$	$9.183370 \cdot 10^{-2}$
Alg. 3/4	100	100	$2.359861 \cdot 10^{-2}$	$6.697551 \cdot 10^{-1}$	$9.185230 \cdot 10^{-2}$
Alg. 3/4	tol.	223	$2.359860 \cdot 10^{-2}$	$6.697566 \cdot 10^{-1}$	$9.185240 \cdot 10^{-2}$
Standard	tol.	223	$2.359860 \cdot 10^{-2}$	$6.697566 \cdot 10^{-1}$	$9.185240 \cdot 10^{-2}$

Table A.2.: Smooth solution problem with prescribed bounds $[-1, 1]$: numerical errors on the Chevron grid of size 64×64.

Method	Stopping	Actual iter.	$L^2(\Omega)$ error	$H^1(\Omega)$ seminorm error	$L^\infty(\Omega)$ error
Alg. 3/4	10	10	$4.447428 \cdot 10^{-3}$	$9.551990 \cdot 10^{-1}$	$2.089760 \cdot 10^{-2}$
Alg. 3/4	25	25	$4.518408 \cdot 10^{-3}$	$9.720182 \cdot 10^{-1}$	$2.087780 \cdot 10^{-2}$
Alg. 3/4	50	50	$4.541542 \cdot 10^{-3}$	$9.769021 \cdot 10^{-1}$	$2.095940 \cdot 10^{-2}$
Alg. 3/4	100	100	$4.543527 \cdot 10^{-3}$	$9.773604 \cdot 10^{-1}$	$2.095470 \cdot 10^{-2}$
Alg. 3/4	tol.	498	$4.543787 \cdot 10^{-3}$	$9.774186 \cdot 10^{-1}$	$2.095620 \cdot 10^{-2}$
Standard	tol.	498	$4.543787 \cdot 10^{-3}$	$9.774186 \cdot 10^{-1}$	$2.095620 \cdot 10^{-2}$

Table A.3.: Smooth solution problem with prescribed bounds $[-1, 1]$: numerical errors on the Delaunay mesh of size 64×64.

Method	Stopping	Actual iter.	$L^2(\Omega)$ error	$H^1(\Omega)$ seminorm error	$L^\infty(\Omega)$ error
Alg. 3/4	10	10	$3.425586 \cdot 10^{-3}$	$5.562878 \cdot 10^{-1}$	$3.541340 \cdot 10^{-2}$
Alg. 3/4	25	25	$3.584212 \cdot 10^{-3}$	$5.957497 \cdot 10^{-1}$	$3.546580 \cdot 10^{-2}$
Alg. 3/4	50	50	$3.617539 \cdot 10^{-3}$	$6.037193 \cdot 10^{-1}$	$3.546750 \cdot 10^{-2}$
Alg. 3/4	100	100	$3.618579 \cdot 10^{-3}$	$6.033719 \cdot 10^{-1}$	$3.546440 \cdot 10^{-2}$
Alg. 3/4	tol.	478	$3.618405 \cdot 10^{-3}$	$6.032802 \cdot 10^{-1}$	$3.546470 \cdot 10^{-2}$
Standard	tol.	478	$3.618405 \cdot 10^{-3}$	$6.032802 \cdot 10^{-1}$	$3.546470 \cdot 10^{-2}$

A.1.2. Smooth Solution Example with Tabata-Derived Bounds

This section contains the detailed error values for the smooth solution example with bounds derived from the Tabata upwind solution, discussed in Section 6.2.2. The errors are computed with respect to the prescribed exact solution.

Table A.4.: Smooth solution problem with Tabata-derived bounds: numerical errors on the Friedrichs–Keller mesh of size 16×16.

Method	Max. iter.	Actual iter.	Status	$L^2(\Omega)$ error	$H^1(\Omega)$ seminorm error	$L^\infty(\Omega)$ error
Alg. 3	10	10	max. iter.	$3.376732 \cdot 10^{-2}$	$8.020279 \cdot 10^{-1}$	$1.610700 \cdot 10^{-1}$
Alg. 3	25	25	max. iter.	$3.389744 \cdot 10^{-2}$	$8.197656 \cdot 10^{-1}$	$1.610830 \cdot 10^{-1}$
Alg. 3	50	50	max. iter.	$3.389906 \cdot 10^{-2}$	$8.201766 \cdot 10^{-1}$	$1.610830 \cdot 10^{-1}$
Alg. 3	100	100	max. iter.	$3.390263 \cdot 10^{-2}$	$8.209118 \cdot 10^{-1}$	$1.610820 \cdot 10^{-1}$
Alg. 3	10000	10000	max. iter.	$3.396283 \cdot 10^{-2}$	$8.296543 \cdot 10^{-1}$	$1.610790 \cdot 10^{-1}$
Alg. 4	10	10	max. iter.	$3.068946 \cdot 10^{-2}$	$7.155679 \cdot 10^{-1}$	$1.585550 \cdot 10^{-1}$
Alg. 4	25	25	max. iter.	$3.078490 \cdot 10^{-2}$	$7.277521 \cdot 10^{-1}$	$1.585550 \cdot 10^{-1}$
Alg. 4	50	50	max. iter.	$3.078624 \cdot 10^{-2}$	$7.286435 \cdot 10^{-1}$	$1.585550 \cdot 10^{-1}$
Alg. 4	100	100	max. iter.	$3.078697 \cdot 10^{-2}$	$7.285636 \cdot 10^{-1}$	$1.585550 \cdot 10^{-1}$
Alg. 4	10000	223	tol.	$3.078696 \cdot 10^{-2}$	$7.285653 \cdot 10^{-1}$	$1.585550 \cdot 10^{-1}$
Standard	10000	223	tol.	$2.359860 \cdot 10^{-2}$	$6.697566 \cdot 10^{-1}$	$9.185240 \cdot 10^{-2}$

Table A.5.: Smooth solution problem with Tabata-derived bounds: numerical errors on the Chevron grid of size 64×64.

Method	Max. iter.	Actual iter.	Status	$L^2(\Omega)$ error	$H^1(\Omega)$ seminorm error	$L^\infty(\Omega)$ error
Alg. 3	10	10	max. iter.	$5.003642 \cdot 10^{-3}$	$9.626530 \cdot 10^{-1}$	$3.480720 \cdot 10^{-2}$
Alg. 3	25	25	max. iter.	$5.066687 \cdot 10^{-3}$	$9.794482 \cdot 10^{-1}$	$3.480720 \cdot 10^{-2}$
Alg. 3	50	50	max. iter.	$5.085472 \cdot 10^{-3}$	$9.846084 \cdot 10^{-1}$	$3.480720 \cdot 10^{-2}$
Alg. 3	100	100	max. iter.	$5.087729 \cdot 10^{-3}$	$9.851685 \cdot 10^{-1}$	$3.480720 \cdot 10^{-2}$
Alg. 3	10000	10000	max. iter.	$5.088006 \cdot 10^{-3}$	$9.852405 \cdot 10^{-1}$	$3.480720 \cdot 10^{-2}$
Alg. 4	10	10	max. iter.	$4.921776 \cdot 10^{-3}$	$9.592330 \cdot 10^{-1}$	$3.480720 \cdot 10^{-2}$
Alg. 4	25	25	max. iter.	$4.983602 \cdot 10^{-3}$	$9.753933 \cdot 10^{-1}$	$3.480720 \cdot 10^{-2}$
Alg. 4	50	50	max. iter.	$5.002406 \cdot 10^{-3}$	$9.795536 \cdot 10^{-1}$	$3.480720 \cdot 10^{-2}$
Alg. 4	100	100	max. iter.	$5.004403 \cdot 10^{-3}$	$9.800655 \cdot 10^{-1}$	$3.480720 \cdot 10^{-2}$
Alg. 4	10000	498	tol.	$5.004629 \cdot 10^{-3}$	$9.801222 \cdot 10^{-1}$	$3.480720 \cdot 10^{-2}$
Standard	10000	498	tol.	$4.543787 \cdot 10^{-3}$	$9.774186 \cdot 10^{-1}$	$2.095620 \cdot 10^{-2}$

Table A.6.: Smooth solution problem with Tabata-derived bounds: numerical errors on the Delaunay mesh of size 64×64.

Method	Max. iter.	Actual iter.	Status	$L^2(\Omega)$ error	$H^1(\Omega)$ seminorm error	$L^\infty(\Omega)$ error
Alg. 3	10	10	max. iter.	$3.957198 \cdot 10^{-3}$	$5.661506 \cdot 10^{-1}$	$3.436500 \cdot 10^{-2}$
Alg. 3	25	25	max. iter.	$4.091339 \cdot 10^{-3}$	$6.045529 \cdot 10^{-1}$	$3.444660 \cdot 10^{-2}$
Alg. 3	50	50	max. iter.	$4.120386 \cdot 10^{-3}$	$6.123271 \cdot 10^{-1}$	$3.445140 \cdot 10^{-2}$
Alg. 3	100	100	max. iter.	$4.120548 \cdot 10^{-3}$	$6.123490 \cdot 10^{-1}$	$3.445170 \cdot 10^{-2}$
Alg. 3	10000	10000	max. iter.	$4.121674 \cdot 10^{-3}$	$6.119389 \cdot 10^{-1}$	$3.445490 \cdot 10^{-2}$
Alg. 4	10	10	max. iter.	$3.862058 \cdot 10^{-3}$	$5.613559 \cdot 10^{-1}$	$3.541340 \cdot 10^{-2}$
Alg. 4	25	25	max. iter.	$4.002673 \cdot 10^{-3}$	$5.999892 \cdot 10^{-1}$	$3.546580 \cdot 10^{-2}$
Alg. 4	50	50	max. iter.	$4.032582 \cdot 10^{-3}$	$6.078893 \cdot 10^{-1}$	$3.546750 \cdot 10^{-2}$
Alg. 4	100	100	max. iter.	$4.033381 \cdot 10^{-3}$	$6.075606 \cdot 10^{-1}$	$3.546440 \cdot 10^{-2}$
Alg. 4	10000	478	tol.	$4.033218 \cdot 10^{-3}$	$6.074710 \cdot 10^{-1}$	$3.546470 \cdot 10^{-2}$
Standard	10000	478	tol.	$3.618405 \cdot 10^{-3}$	$6.032802 \cdot 10^{-1}$	$3.546470 \cdot 10^{-2}$

A.2. Layer Solution Example

This section contains the detailed cross-section error values for the layer solution example discussed in Section 6.3. Since the exact solution is not known, the errors are computed with respect to the independent reference profiles on the cross sections $y = 0.25$ and $x = 0.75$.

A.2.1. The Chevron Grid

The following tables correspond to the computations on the Chevron grid discussed in Section 6.3.1.

Table A.7.: Layer problem on the Chevron grid: cross-section errors with respect to the independent reference profile at $y = 0.25$.

Method	Stopping	Actual iter.	L^1 error	L^2 error	L^∞ error
Standard	tol.	618	$1.128527 \cdot 10^{-2}$	$7.095306 \cdot 10^{-2}$	$9.842831 \cdot 10^{-1}$
Algorithm 1	10	10	$1.132464 \cdot 10^{-2}$	$7.091988 \cdot 10^{-2}$	$9.842669 \cdot 10^{-1}$
Algorithm 1	25	25	$1.129756 \cdot 10^{-2}$	$7.095004 \cdot 10^{-2}$	$9.842815 \cdot 10^{-1}$
Algorithm 1	50	50	$1.128497 \cdot 10^{-2}$	$7.095331 \cdot 10^{-2}$	$9.842833 \cdot 10^{-1}$
Algorithm 1	100	100	$1.128517 \cdot 10^{-2}$	$7.095306 \cdot 10^{-2}$	$9.842832 \cdot 10^{-1}$
Algorithm 1	tol.	635	$1.128535 \cdot 10^{-2}$	$7.095306 \cdot 10^{-2}$	$9.842831 \cdot 10^{-1}$
Algorithm 2	10	10	$1.132514 \cdot 10^{-2}$	$7.119087 \cdot 10^{-2}$	$9.843874 \cdot 10^{-1}$
Algorithm 2	25	25	$1.143303 \cdot 10^{-2}$	$7.125201 \cdot 10^{-2}$	$9.844122 \cdot 10^{-1}$
Algorithm 2	50	50	$1.145438 \cdot 10^{-2}$	$7.125450 \cdot 10^{-2}$	$9.844128 \cdot 10^{-1}$
Algorithm 2	100	100	$1.145376 \cdot 10^{-2}$	$7.125425 \cdot 10^{-2}$	$9.844127 \cdot 10^{-1}$
Algorithm 2	tol.	267	$1.145377 \cdot 10^{-2}$	$7.125425 \cdot 10^{-2}$	$9.844127 \cdot 10^{-1}$
Algorithm 3	10	10	$1.144542 \cdot 10^{-2}$	$7.095095 \cdot 10^{-2}$	$9.842781 \cdot 10^{-1}$
Algorithm 3	25	25	$1.131553 \cdot 10^{-2}$	$7.095156 \cdot 10^{-2}$	$9.842817 \cdot 10^{-1}$
Algorithm 3	50	50	$1.128354 \cdot 10^{-2}$	$7.095354 \cdot 10^{-2}$	$9.842834 \cdot 10^{-1}$
Algorithm 3	100	100	$1.128513 \cdot 10^{-2}$	$7.095305 \cdot 10^{-2}$	$9.842831 \cdot 10^{-1}$
Algorithm 3	tol.	599	$1.128527 \cdot 10^{-2}$	$7.095306 \cdot 10^{-2}$	$9.842831 \cdot 10^{-1}$
Algorithm 4	10	10	$1.147994 \cdot 10^{-2}$	$7.093203 \cdot 10^{-2}$	$9.842684 \cdot 10^{-1}$
Algorithm 4	25	25	$1.131202 \cdot 10^{-2}$	$7.095582 \cdot 10^{-2}$	$9.842838 \cdot 10^{-1}$
Algorithm 4	50	50	$1.128191 \cdot 10^{-2}$	$7.095345 \cdot 10^{-2}$	$9.842834 \cdot 10^{-1}$
Algorithm 4	100	100	$1.128514 \cdot 10^{-2}$	$7.095305 \cdot 10^{-2}$	$9.842831 \cdot 10^{-1}$
Algorithm 4	tol.	618	$1.128527 \cdot 10^{-2}$	$7.095306 \cdot 10^{-2}$	$9.842831 \cdot 10^{-1}$

Table A.8.: Layer problem on the Chevron grid: cross-section errors with respect to the independent reference profile at $x = 0.75$.

Method	Stopping	Actual iter.	L^1 error	L^2 error	L^∞ error
Standard	tol.	618	$2.098796 \cdot 10^{-2}$	$7.188184 \cdot 10^{-2}$	$5.404075 \cdot 10^{-1}$
Algorithm 1	10	10	$3.623658 \cdot 10^{-2}$	$9.615927 \cdot 10^{-2}$	$5.380725 \cdot 10^{-1}$
Algorithm 1	25	25	$3.565098 \cdot 10^{-2}$	$9.735571 \cdot 10^{-2}$	$5.404015 \cdot 10^{-1}$
Algorithm 1	50	50	$3.564585 \cdot 10^{-2}$	$9.737889 \cdot 10^{-2}$	$5.403975 \cdot 10^{-1}$
Algorithm 1	100	100	$3.564495 \cdot 10^{-2}$	$9.737981 \cdot 10^{-2}$	$5.404075 \cdot 10^{-1}$
Algorithm 1	tol.	635	$3.564496 \cdot 10^{-2}$	$9.737979 \cdot 10^{-2}$	$5.404075 \cdot 10^{-1}$
Algorithm 2	10	10	$6.090296 \cdot 10^{-2}$	$1.362321 \cdot 10^{-1}$	$6.621128 \cdot 10^{-1}$
Algorithm 2	25	25	$6.098479 \cdot 10^{-2}$	$1.361749 \cdot 10^{-1}$	$6.620573 \cdot 10^{-1}$
Algorithm 2	50	50	$6.098556 \cdot 10^{-2}$	$1.361743 \cdot 10^{-1}$	$6.620584 \cdot 10^{-1}$
Algorithm 2	100	100	$6.098579 \cdot 10^{-2}$	$1.361742 \cdot 10^{-1}$	$6.620584 \cdot 10^{-1}$
Algorithm 2	tol.	267	$6.098579 \cdot 10^{-2}$	$1.361742 \cdot 10^{-1}$	$6.620584 \cdot 10^{-1}$
Algorithm 3	10	10	$2.345572 \cdot 10^{-2}$	$7.446584 \cdot 10^{-2}$	$5.483915 \cdot 10^{-1}$
Algorithm 3	25	25	$2.095431 \cdot 10^{-2}$	$7.183978 \cdot 10^{-2}$	$5.403995 \cdot 10^{-1}$
Algorithm 3	50	50	$2.098930 \cdot 10^{-2}$	$7.188071 \cdot 10^{-2}$	$5.403975 \cdot 10^{-1}$
Algorithm 3	100	100	$2.098796 \cdot 10^{-2}$	$7.188184 \cdot 10^{-2}$	$5.404075 \cdot 10^{-1}$
Algorithm 3	tol.	599	$2.098796 \cdot 10^{-2}$	$7.188184 \cdot 10^{-2}$	$5.404075 \cdot 10^{-1}$
Algorithm 4	10	10	$2.283426 \cdot 10^{-2}$	$7.338162 \cdot 10^{-2}$	$5.453805 \cdot 10^{-1}$
Algorithm 4	25	25	$2.094012 \cdot 10^{-2}$	$7.182692 \cdot 10^{-2}$	$5.403795 \cdot 10^{-1}$
Algorithm 4	50	50	$2.098969 \cdot 10^{-2}$	$7.188036 \cdot 10^{-2}$	$5.403945 \cdot 10^{-1}$
Algorithm 4	100	100	$2.098796 \cdot 10^{-2}$	$7.188184 \cdot 10^{-2}$	$5.404075 \cdot 10^{-1}$
Algorithm 4	tol.	618	$2.098796 \cdot 10^{-2}$	$7.188184 \cdot 10^{-2}$	$5.404075 \cdot 10^{-1}$

A.2.2. The Acute Grid

The following tables correspond to the computations on the acute grid discussed in Section 6.3.2.

Table A.9.: Layer problem on the acute grid: cross-section errors with respect to the independent reference profile at $y = 0.25$.

Method	Stopping	Actual iter.	L^1 error	L^2 error	L^∞ error
Standard	tol.	174	$9.647166 \cdot 10^{-3}$	$7.851295 \cdot 10^{-2}$	$9.871636 \cdot 10^{-1}$
Algorithm 1	10	10	$9.667353 \cdot 10^{-3}$	$7.847964 \cdot 10^{-2}$	$9.871524 \cdot 10^{-1}$
Algorithm 1	25	25	$9.643484 \cdot 10^{-3}$	$7.851243 \cdot 10^{-2}$	$9.871634 \cdot 10^{-1}$
Algorithm 1	50	50	$9.646400 \cdot 10^{-3}$	$7.851294 \cdot 10^{-2}$	$9.871636 \cdot 10^{-1}$
Algorithm 1	100	100	$9.647167 \cdot 10^{-3}$	$7.851295 \cdot 10^{-2}$	$9.871636 \cdot 10^{-1}$
Algorithm 1	tol.	189	$9.647173 \cdot 10^{-3}$	$7.851295 \cdot 10^{-2}$	$9.871636 \cdot 10^{-1}$
Algorithm 2	10	10	$1.968315 \cdot 10^{-2}$	$7.958435 \cdot 10^{-2}$	$9.872856 \cdot 10^{-1}$
Algorithm 2	25	25	$1.987901 \cdot 10^{-2}$	$7.966526 \cdot 10^{-2}$	$9.873046 \cdot 10^{-1}$
Algorithm 2	50	50	$1.987925 \cdot 10^{-2}$	$7.966471 \cdot 10^{-2}$	$9.873044 \cdot 10^{-1}$
Algorithm 2	100	100	$1.987929 \cdot 10^{-2}$	$7.966471 \cdot 10^{-2}$	$9.873044 \cdot 10^{-1}$
Algorithm 2	tol.	135	$1.987929 \cdot 10^{-2}$	$7.966471 \cdot 10^{-2}$	$9.873044 \cdot 10^{-1}$
Algorithm 3	10	10	$9.678205 \cdot 10^{-3}$	$7.847390 \cdot 10^{-2}$	$9.871504 \cdot 10^{-1}$
Algorithm 3	25	25	$9.644531 \cdot 10^{-3}$	$7.851223 \cdot 10^{-2}$	$9.871633 \cdot 10^{-1}$
Algorithm 3	50	50	$9.646485 \cdot 10^{-3}$	$7.851299 \cdot 10^{-2}$	$9.871636 \cdot 10^{-1}$
Algorithm 3	100	100	$9.647167 \cdot 10^{-3}$	$7.851295 \cdot 10^{-2}$	$9.871636 \cdot 10^{-1}$

Method	Stopping	Actual iter.	L^1 error	L^2 error	L^∞ error
Algorithm 3	tol.	174	$9.647166 \cdot 10^{-3}$	$7.851295 \cdot 10^{-2}$	$9.871636 \cdot 10^{-1}$
Algorithm 4	10	10	$9.676800 \cdot 10^{-3}$	$7.847501 \cdot 10^{-2}$	$9.871508 \cdot 10^{-1}$
Algorithm 4	25	25	$9.644507 \cdot 10^{-3}$	$7.851225 \cdot 10^{-2}$	$9.871633 \cdot 10^{-1}$
Algorithm 4	50	50	$9.646490 \cdot 10^{-3}$	$7.851299 \cdot 10^{-2}$	$9.871636 \cdot 10^{-1}$
Algorithm 4	100	100	$9.647166 \cdot 10^{-3}$	$7.851295 \cdot 10^{-2}$	$9.871636 \cdot 10^{-1}$
Algorithm 4	tol.	174	$9.647166 \cdot 10^{-3}$	$7.851295 \cdot 10^{-2}$	$9.871636 \cdot 10^{-1}$

Table A.10.: Layer problem on the acute grid: cross-section errors with respect to the independent reference profile at $x = 0.75$.

Method	Stopping	Actual iter.	L^1 error	L^2 error	L^∞ error
Standard	tol.	174	$2.151891 \cdot 10^{-2}$	$7.461439 \cdot 10^{-2}$	$5.603387 \cdot 10^{-1}$
Algorithm 1	10	10	$3.597301 \cdot 10^{-2}$	$9.871513 \cdot 10^{-2}$	$5.598427 \cdot 10^{-1}$
Algorithm 1	25	25	$3.576989 \cdot 10^{-2}$	$9.933791 \cdot 10^{-2}$	$5.603357 \cdot 10^{-1}$
Algorithm 1	50	50	$3.577389 \cdot 10^{-2}$	$9.935627 \cdot 10^{-2}$	$5.603397 \cdot 10^{-1}$
Algorithm 1	100	100	$3.577574 \cdot 10^{-2}$	$9.936459 \cdot 10^{-2}$	$5.603387 \cdot 10^{-1}$
Algorithm 1	tol.	189	$3.577580 \cdot 10^{-2}$	$9.936487 \cdot 10^{-2}$	$5.603387 \cdot 10^{-1}$
Algorithm 2	10	10	$5.924349 \cdot 10^{-2}$	$1.345930 \cdot 10^{-1}$	$6.634193 \cdot 10^{-1}$
Algorithm 2	25	25	$5.929383 \cdot 10^{-2}$	$1.345745 \cdot 10^{-1}$	$6.633672 \cdot 10^{-1}$
Algorithm 2	50	50	$5.929099 \cdot 10^{-2}$	$1.345716 \cdot 10^{-1}$	$6.633674 \cdot 10^{-1}$
Algorithm 2	100	100	$5.929104 \cdot 10^{-2}$	$1.345717 \cdot 10^{-1}$	$6.633676 \cdot 10^{-1}$
Algorithm 2	tol.	135	$5.929104 \cdot 10^{-2}$	$1.345717 \cdot 10^{-1}$	$6.633676 \cdot 10^{-1}$
Algorithm 3	10	10	$2.183064 \cdot 10^{-2}$	$7.495884 \cdot 10^{-2}$	$5.598427 \cdot 10^{-1}$
Algorithm 3	25	25	$2.149758 \cdot 10^{-2}$	$7.460149 \cdot 10^{-2}$	$5.603317 \cdot 10^{-1}$
Algorithm 3	50	50	$2.152424 \cdot 10^{-2}$	$7.462052 \cdot 10^{-2}$	$5.603397 \cdot 10^{-1}$
Algorithm 3	100	100	$2.151881 \cdot 10^{-2}$	$7.461431 \cdot 10^{-2}$	$5.603387 \cdot 10^{-1}$
Algorithm 3	tol.	174	$2.151891 \cdot 10^{-2}$	$7.461439 \cdot 10^{-2}$	$5.603387 \cdot 10^{-1}$
Algorithm 4	10	10	$2.187624 \cdot 10^{-2}$	$7.503627 \cdot 10^{-2}$	$5.598427 \cdot 10^{-1}$
Algorithm 4	25	25	$2.149796 \cdot 10^{-2}$	$7.460166 \cdot 10^{-2}$	$5.603357 \cdot 10^{-1}$
Algorithm 4	50	50	$2.152425 \cdot 10^{-2}$	$7.462051 \cdot 10^{-2}$	$5.603397 \cdot 10^{-1}$
Algorithm 4	100	100	$2.151881 \cdot 10^{-2}$	$7.461431 \cdot 10^{-2}$	$5.603387 \cdot 10^{-1}$
Algorithm 4	tol.	174	$2.151891 \cdot 10^{-2}$	$7.461439 \cdot 10^{-2}$	$5.603387 \cdot 10^{-1}$

A.2.3. Additional Bound and Convergence Data

The following tables give the minimum and maximum values of the computed solutions for all stopping conditions in 6.3. Values of order 10^{-16} or smaller are interpreted as round-off errors and are treated as numerical zero.

Table A.11.: Layer problem on the Chevron grid of size 64×64: minimum and maximum values of the computed solutions for all stopping conditions.

Method	Stopping	Actual iter.	Minimum value	Maximum value
Standard	tol.	618	$-1.109806 \cdot 10^{-16}$	0.999624
Algorithm 1	10	10	$-1.136153 \cdot 10^{-16}$	1.023356
Algorithm 1	25	25	$-1.113000 \cdot 10^{-16}$	1.002502

Method	Stopping	Actual iter.	Minimum value	Maximum value
Algorithm 1	50	50	$-1.109544 \cdot 10^{-16}$	0.999388
Algorithm 1	100	100	$-1.109577 \cdot 10^{-16}$	0.999418
Algorithm 1	tol.	635	$-1.109577 \cdot 10^{-16}$	0.999418
Algorithm 2	10	10	$-1.104450 \cdot 10^{-16}$	0.994800
Algorithm 2	25	25	$-1.099974 \cdot 10^{-16}$	0.990768
Algorithm 2	50	50	$-1.099879 \cdot 10^{-16}$	0.990683
Algorithm 2	100	100	$-1.099875 \cdot 10^{-16}$	0.990679
Algorithm 2	tol.	267	$-1.099875 \cdot 10^{-16}$	0.990679
Algorithm 3	10	10	$-1.110223 \cdot 10^{-16}$	1.000000
Algorithm 3	25	25	$-1.110223 \cdot 10^{-16}$	1.000000
Algorithm 3	50	50	$-1.109804 \cdot 10^{-16}$	0.999622
Algorithm 3	100	100	$-1.109806 \cdot 10^{-16}$	0.999624
Algorithm 3	tol.	599	$-1.109806 \cdot 10^{-16}$	0.999624
Algorithm 4	10	10	$-1.110223 \cdot 10^{-16}$	1.000000
Algorithm 4	25	25	$-1.110223 \cdot 10^{-16}$	1.000000
Algorithm 4	50	50	$-1.109797 \cdot 10^{-16}$	0.999617
Algorithm 4	100	100	$-1.109806 \cdot 10^{-16}$	0.999624
Algorithm 4	tol.	618	$-1.109806 \cdot 10^{-16}$	0.999624

Table A.12.: Layer problem on the acute grid at refinement level 4: minimum and maximum values of the computed solutions for all stopping conditions.

Method	Stopping	Actual iter.	Minimum value	Maximum value
Standard	tol.	174	0	0.999337
Algorithm 1	10	10	0	1.002951
Algorithm 1	25	25	0	0.997221
Algorithm 1	50	50	0	0.997138
Algorithm 1	100	100	0	0.997144
Algorithm 1	tol.	189	0	0.997144
Algorithm 2	10	10	$-7.420955 \cdot 10^{-19}$	0.983832
Algorithm 2	25	25	$-1.067154 \cdot 10^{-17}$	0.983099
Algorithm 2	50	50	0	0.983073
Algorithm 2	100	100	0	0.983073
Algorithm 2	tol.	135	$-1.067332 \cdot 10^{-17}$	0.983073
Algorithm 3	10	10	$-7.179732 \cdot 10^{-19}$	1.000000
Algorithm 3	25	25	$-2.385751 \cdot 10^{-21}$	0.999241
Algorithm 3	50	50	$-6.554812 \cdot 10^{-26}$	0.999341
Algorithm 3	100	100	0	0.999336
Algorithm 3	tol.	174	0	0.999337
Algorithm 4	10	10	$-5.004911 \cdot 10^{-19}$	1.000000
Algorithm 4	25	25	$-6.568336 \cdot 10^{-22}$	0.999246
Algorithm 4	50	50	$-2.966804 \cdot 10^{-25}$	0.999341
Algorithm 4	100	100	0	0.999336
Algorithm 4	tol.	174	0	0.999337

A.3. The Hemker Problem

This section provides the detailed numerical data for the Hemker problem discussed in Section 6.4. The layer-width data, nonlinear stopping data, and minimum and maximum values for all tested stopping conditions are reported below.

Table A.13.: Hemker problem at refinement level 4: layer widths at $x = 4$ and their absolute differences from the reference value 0.0723.

Method	Stopping	Max. iter.	Actual iter.	Status	Layer width	Absolute difference
Standard	10000	10000	10000	max. iter.	0.12831	0.05601
Algorithm 1	10	10	10	max. iter.	0.39519	0.32289
Algorithm 1	25	25	25	max. iter.	0.38892	0.31662
Algorithm 1	50	50	50	max. iter.	0.38988	0.31758
Algorithm 1	100	100	100	max. iter.	0.39003	0.31773
Algorithm 1	tol.	10000	574	tol.	0.39003	0.31773
Algorithm 2	10	10	10	max. iter.	0.60897	0.53667
Algorithm 2	25	25	25	max. iter.	0.60903	0.53673
Algorithm 2	tol.	50	49	tol.	0.60903	0.53673
Algorithm 2	tol.	100	49	tol.	0.60903	0.53673
Algorithm 2	tol.	10000	49	tol.	0.60903	0.53673
Algorithm 3	10	10	10	max. iter.	0.25536	0.18306
Algorithm 3	25	25	25	max. iter.	0.23685	0.16455
Algorithm 3	50	50	50	max. iter.	0.21597	0.14367
Algorithm 3	100	100	100	max. iter.	0.13011	0.05781
Algorithm 3	10000	10000	10000	max. iter.	0.12831	0.05601
Algorithm 4	10	10	10	max. iter.	0.21510	0.14280
Algorithm 4	25	25	25	max. iter.	0.15912	0.08682
Algorithm 4	50	50	50	max. iter.	0.12831	0.05601
Algorithm 4	100	100	100	max. iter.	0.12837	0.05607
Algorithm 4	10000	10000	10000	max. iter.	0.12831	0.05601

Table A.14.: Hemker problem at refinement level 4: nonlinear stopping data. The normalized stopping tolerance is $3.094899 \cdot 10^{-10}$.

Method	Stopping	Max. iter.	Actual iter.	Status	Final residual	Rejections
Standard	10000	10000	10000	max. iter.	$2.809370 \cdot 10^{-8}$	47
Algorithm 1	10	10	10	max. iter.	$1.898390 \cdot 10^{-3}$	0
Algorithm 1	25	25	25	max. iter.	$2.362400 \cdot 10^{-4}$	0
Algorithm 1	50	50	50	max. iter.	$3.964350 \cdot 10^{-5}$	0
Algorithm 1	100	100	100	max. iter.	$7.186430 \cdot 10^{-6}$	0
Algorithm 1	tol.	10000	574	tol.	$3.093490 \cdot 10^{-10}$	0
Algorithm 2	10	10	10	max. iter.	$1.152310 \cdot 10^{-5}$	0
Algorithm 2	25	25	25	max. iter.	$1.024480 \cdot 10^{-7}$	0
Algorithm 2	tol.	50	49	tol.	$2.600760 \cdot 10^{-10}$	0
Algorithm 2	tol.	100	49	tol.	$2.600760 \cdot 10^{-10}$	0
Algorithm 2	tol.	10000	49	tol.	$2.600760 \cdot 10^{-10}$	0
Algorithm 3	10	10	10	max. iter.	$7.531300 \cdot 10^{-3}$	4
Algorithm 3	25	25	25	max. iter.	$7.453080 \cdot 10^{-3}$	8
Algorithm 3	50	50	50	max. iter.	$7.338890 \cdot 10^{-3}$	10
Algorithm 3	100	100	100	max. iter.	$1.852930 \cdot 10^{-3}$	10
Algorithm 3	10000	10000	10000	max. iter.	$2.117150 \cdot 10^{-8}$	48
Algorithm 4	10	10	10	max. iter.	$5.874480 \cdot 10^{-3}$	3
Algorithm 4	25	25	25	max. iter.	$4.821790 \cdot 10^{-3}$	3
Algorithm 4	50	50	50	max. iter.	$8.604090 \cdot 10^{-4}$	3

Continued on the next page.

Table A.14: Nonlinear stopping data (continued).

Method	Stopping	Max. iter.	Actual iter.	Status	Final residual	Rejections
Algorithm 4	100	100	100	max. iter.	$1.543980 \cdot 10^{-4}$	3
Algorithm 4	10000	10000	10000	max. iter.	$2.809370 \cdot 10^{-8}$	47

Table A.15.: Hemker problem at refinement level 4: minimum and maximum values of the computed solutions.

Method	Stopping	Max. iter.	Actual iter.	Status	Minimum value	Maximum value
Standard	10000	10000	10000	max. iter.	$-1.110223 \cdot 10^{-16}$	1
Algorithm 1	10	10	10	max. iter.	$-1.110223 \cdot 10^{-16}$	1.03808276238
Algorithm 1	25	25	25	max. iter.	$-1.110223 \cdot 10^{-16}$	1.00295577203
Algorithm 1	50	50	50	max. iter.	$-1.110223 \cdot 10^{-16}$	1.00001309084
Algorithm 1	100	100	100	max. iter.	$-1.110223 \cdot 10^{-16}$	1.000000167
Algorithm 1	tol.	10000	574	tol.	$-1.110223 \cdot 10^{-16}$	1
Algorithm 2	10	10	10	max. iter.	$-1.325755 \cdot 10^{-16}$	1
Algorithm 2	25	25	25	max. iter.	$-5.821400 \cdot 10^{-16}$	1
Algorithm 2	tol.	50	49	tol.	$-5.824384 \cdot 10^{-16}$	1
Algorithm 2	tol.	100	49	tol.	$-5.824384 \cdot 10^{-16}$	1
Algorithm 2	tol.	10000	49	tol.	$-5.824384 \cdot 10^{-16}$	1
Algorithm 3	10	10	10	max. iter.	$-1.110223 \cdot 10^{-16}$	1
Algorithm 3	25	25	25	max. iter.	$-1.110223 \cdot 10^{-16}$	1
Algorithm 3	50	50	50	max. iter.	$-1.110223 \cdot 10^{-16}$	1
Algorithm 3	100	100	100	max. iter.	$-1.110223 \cdot 10^{-16}$	1
Algorithm 3	10000	10000	10000	max. iter.	$-1.110223 \cdot 10^{-16}$	1
Algorithm 4	10	10	10	max. iter.	$-1.110223 \cdot 10^{-16}$	1
Algorithm 4	25	25	25	max. iter.	$-1.110223 \cdot 10^{-16}$	1
Algorithm 4	50	50	50	max. iter.	$-1.110223 \cdot 10^{-16}$	1
Algorithm 4	100	100	100	max. iter.	$-1.110223 \cdot 10^{-16}$	1
Algorithm 4	10000	10000	10000	max. iter.	$-1.110223 \cdot 10^{-16}$	1

Bibliography

[ACF+11] Matthias Augustin, Alfonso Caiazzo, André Fiebach, Jürgen Fuhrmann, Volker John, Alexander Linke, and Rudolf Umla. An assessment of discretizations for convection-dominated convection-diffusion equations. *Comput. Methods Appl. Mech. Engrg.*, 200(47-48):3395–3409, 2011.

[BH82] Alexander N. Brooks and Thomas J. R. Hughes. Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Comput. Methods Appl. Mech. Engrg.*, 32(1-3):199–259, 1982. FENOMECH "81, Part I (Stuttgart, 1981).

[BJK15] Gabriel R. Barrenechea, Volker John, and Petr Knobloch. Some analytical results for an algebraic flux correction scheme for a steady convection-diffusion equation in one dimension. *IMA J. Numer. Anal.*, 35(4):1729–1756, 2015.

[BJK16] Gabriel R. Barrenechea, Volker John, and Petr Knobloch. Analysis of algebraic flux correction schemes. *SIAM J. Numer. Anal.*, 54(4):2427–2451, 2016.

[BJK17] Gabriel R. Barrenechea, Volker John, and Petr Knobloch. An algebraic flux correction scheme satisfying the discrete maximum principle and linearity preservation on general meshes. *Math. Models Methods Appl. Sci.*, 27(3):525–548, 2017.

[BJK25] Gabriel R. Barrenechea, Volker John, and Petr Knobloch. *Monotone discretizations for elliptic second order partial differential equations*, volume 61 of *Springer Series in Computational Mathematics*. Springer, Cham, [2025] ©2025.

[BT81] Kinji Baba and Masahisa Tabata. On a conservative upwind finite element scheme for convective diffusion equations. *RAIRO Anal. Numér.*, 15(1):3–25, 1981.

[Cia70] Philippe G. Ciarlet. Discrete maximum principle for finite-difference operators. *Aequationes Math.*, 4:338–352, 1970.

[Eva10] Lawrence C. Evans. *Partial differential equations*, volume 19 of *Graduate Studies in Mathematics*. American Mathematical Society, Providence, RI, second edition, 2010.

[GR09] Christophe Geuzaine and Jean-Fran¸cois Remacle. Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities. *Internat. J. Numer. Methods Engrg.*, 79(11):1309–1331, 2009.

[Hem96] P. W. Hemker. A singularly perturbed model problem for numerical computation. *J. Comput. Appl. Math.*, 76(1-2):277–285, 1996.

[JJ19] Abhinav Jha and Volker John. A study of solvers for nonlinear AFC discretizations of convection-diffusion equations. *Comput. Math. Appl.*, 78(9):3117–3138, 2019.

[JJ20] Abhinav Jha and Volker John. On basic iteration schemes for nonlinear AFC discretizations. In *Boundary and interior layers, computational and asymptotic methods—BAIL 2018*, volume 135 of *Lect. Notes Comput. Sci. Eng.*, pages 113–128. Springer, Cham, [2020] ©2020.

[JK07] Volker John and Petr Knobloch. On spurious oscillations at layers diminishing (SOLD) methods for convection-diffusion equations. I. A review. *Comput. Methods Appl. Mech. Engrg.*, 196(17-20):2197–2215, 2007.

[JK08] Volker John and Petr Knobloch. On spurious oscillations at layers diminishing (SOLD) methods for convection-diffusion equations. II. Analysis for P_1 and Q_1 finite elements. *Comput. Methods Appl. Mech. Engrg.*, 197(21-24):1997–2014, 2008.

[JK22] Volker John and Petr Knobloch. On algebraically stabilized schemes for convection-diffusion-reaction problems. *Numer. Math.*, 152(3):553–585, 2022.

[JKP23] Volker John, Petr Knobloch, and Ondˇrej Pártl. A numerical assessment of finite element discretizations for convection-diffusion-reaction equations satisfying discrete maximum principles. *Comput. Methods Appl. Math.*, 23(4):969–988, 2023.

[Kno21] Petr Knobloch. A new algebraically stabilized method for convection-diffusion-reaction equations. In *Numerical mathematics and advanced applications—ENUMATH 2019*, volume 139 of *Lect. Notes Comput. Sci. Eng.*, pages 605–613. Springer, Cham, [2021] ©2021.

[KS80] David Kinderlehrer and Guido Stampacchia. *An introduction to variational inequalities and their applications*, volume 88 of *Pure and Applied Mathematics*. Academic Press, Inc. [Harcourt Brace Jovanovich, Publishers], New York-London, 1980.

[Kuz07] Dmitri Kuzmin. Algebraic flux correction for finite element discretizations of coupled systems. In Manolis Papadrakakis, Eugenio Oñate, and Bernhard Schrefler, editors, *Proceedings of the International Conference on Computational Methods for Coupled Problems in Science and Engineering*, pages 653–656, Barcelona, 2007. CIMNE.

[Ost37] Alexander Ostrowski. über die determinanten mit überwiegender Hauptdiagonale. *Comment. Math. Helv.*, 10(1):69–96, 1937.

[Tab77] Masahisa Tabata. A finite element approximation corresponding to the upwind finite differencing. *Mem. Numer. Math.*, (4):47–63, 1977.

[VS18] F. J. Vermolen and A. Segal. On an integration rule for products of barycentric coordinates over simplexes in $\mathbb{R}^n$. *J. Comput. Appl. Math.*, 330:289–294, 2018.

[WBAea17] Ulrich Wilbrandt, Clemens Bartsch, Naveed Ahmed, and et al. ParMooN—a modernized program package based on mapped finite elements. *Comput. Math. Appl.*, 74(1):74–88, 2017.

[XZ99] Jinchao Xu and Ludmil Zikatanov. A monotone finite element scheme for convection-diffusion equations. *Math. Comp.*, 68(228):1429–1446, 1999.

[Zal79] Steven T. Zalesak. Fully multidimensional flux-corrected transport algorithms for fluids. *J. Comput. Phys.*, 31(3):335–362, 1979.

Freie Universität Berlin

Fachbereich Mathematik, Informatik und Physik

SELBSTSTÄNDIGKEITSERKLÄRUNG

Name: Xu	(BITTE nur Block- oder Maschinenschrift verwenden.)
Vorname(n): Shi	
Studiengang: Mathematik, M.Sc.	
Matr. Nr.: 5592941	

Ich erkläre gegenüber der Freien Universität Berlin, dass ich die vorliegende Masterarbeit selbstständig und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel angefertigt habe.

Die vorliegende Arbeit ist frei von Plagiaten. Alle Ausführungen, die wörtlich oder inhaltlich aus anderen Schriften entnommen sind, habe ich als solche kenntlich gemacht.

Diese Arbeit wurde in gleicher oder ähnlicher Form noch bei keiner anderen Universität als Prüfungsleistung eingereicht.

Datum: 26.06.2026 Unterschrift: Shi Xu