A nonlinearity lagging method for non-steady diffusion equations with nonlinear convection terms

F. Mezzadri * [1] and E. Galligani †[1]

[1]Department of Engineering "Enzo Ferrari", University of Modena and Reggio Emilia, via P. Vivarelli 10/1, building 26, I-41125, Modena

Received: date / Accepted: date

Abstract

We analyze an iterative procedure for solving nonlinear algebraic systems arising from the discretization of nonlinear, non-steady reaction-convection-diffusion equations with non-constant (and, in general, nonlinear) velocity terms. The basic idea underlying the procedure consists in lagging the diffusion and the velocity terms of the discretized system, which is thus partly linearized. After analyzing the discretized system and proving some results on the monotonicity of the operators and on the uniqueness of the solution, we prove sufficient conditions that ensure the convergence of this lagged method. We also describe the inner iteration and show how the weakly nonlinear systems arising at each lagged iteration can be solved efficiently. Finally, we analyze numerically the entire solution process by several numerical experiments.

Keywords: Nonlinear diffusion equations, Lagged Diffusivity Method, Finite differences
MSC2010: 65H10,65M06,65M22

1 Introduction

In many important problems and applications, it is often necessary to solve systems of nonlinear algebraic equations

$$\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{0},$$

where $\boldsymbol{F} : \Omega \in \mathbb{R}^n \to \mathbb{R}^n$ is a continuously differentiable mapping and $\boldsymbol{u} \in \mathbb{R}^n$ is the unknown vector. These systems can arise, for instance, from the discretization of a differential operator by finite differences or by finite elements schemes. In these cases, the systems are usually large and the Jacobian matrix is often not available or hard to compute.

In this and in other situations where a strong nonlinearity is present, methods relying on a linearization of nonlinear terms can be of interest. For instance, in the context of denoising in image analysis (e.g. see [15] and [2]), a *lagged diffusivity fixed point iteration* was introduced in [21]. As the name suggests, in this study the diffusivity term was linearized by "lagging" its dependence on $\boldsymbol{u}$. The convergence of this procedure in this field has then been analyzed in [3].

While these papers focused on the specific field of image restoration (and on specific forms of diffusivity, like that of Perona-Malik [13]), more recent contributions have proposed the lagged diffusivity method for solving systems descending from entire classes of diffusion problems. Several cases have been analyzed, providing proofs of the convergence of the procedure under some smoothness assumptions. The main advantage of

*francesco.mezzadri@unimore.it
†emanuele.galligani@unimore.it

these techniques is that, at each lagging iteration, they linearize (al least partially) the nonlinear algebraic system, thus simplifying the computation of its Jacobian matrix. The most recent contribution [11] dealt with systems arising from general non-steady diffusion equations containing reaction and convection terms as well. The reader is referred to [11] also for a review of other papers on this topic.

In this paper, we proceed along this line of thought, but we consider a yet more general case. Indeed, none of the existing papers on this topic analyzed the case of convection with a non-constant velocity term. We, on the other hand, aim at solving systems arising from general non-steady diffusion equations with nonlinear velocity terms. This is actually a really interesting situation, which can occur in several practical problems. Moreover, this generalization is interesting also on a mathematical perspective, since it deeply modifies both the properties of the discretized system and the convergence of the algorithm. Indeed, in the general nonlinear case, the very idea underlying the lagged procedure must be changed since also the velocity term has to be lagged if $\tilde{\boldsymbol{v}}$ depends on $\boldsymbol{u}$. Finally, a non-constant velocity term also modifies the existence of the solution and the monotonicity of $\boldsymbol{F}(\boldsymbol{u})$, since terms that disappear in case of constant $\tilde{\boldsymbol{v}}$ cannot be eliminated anymore.

We take into account all these issues and analyze a lagged procedure applied to the system $\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{0}$, where $\boldsymbol{F}(\boldsymbol{u})$ arises from a finite difference discretization of a non-steady reaction-convection-diffusion equation with nonlinear velocity terms. As mentioned earlier, in this procedure both diffusivity and velocity terms are lagged. This means that, starting from an initial iterate $\boldsymbol{u}^{(0)}$, we compute the next iterate $\boldsymbol{u}^{(1)}$ as the solution of the weakly nonlinear algebraic system where diffusivity and velocity are evaluated at $\boldsymbol{u} = \boldsymbol{u}^{(0)}$. Then, in the following iteration, $\boldsymbol{u}^{(1)}$ is used to evaluate the diffusivity and the velocity terms, allowing us to find the new iterate $\boldsymbol{u}^{(2)}$, and so on until a stopping criterion is satisfied. Notwithstanding also the velocity term is lagged, we still refer to this procedure as the *Lagged Diffusivity Method* (LDM) for uniformity with previous works.

We also prove that, when some assumptions hold, $\boldsymbol{F}(\boldsymbol{u})$ is monotone, the solution of the nonlinear algebraic system $\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{0}$ is unique and the LDM converges. We focus our analysis on the more general non-steady case, but we provide a few remarks on the stationary case as well.

Then, on a more operational point of view, we illustrate how to choose starting vectors and tolerances of all the iterative procedures in an efficient way. This is of fundamental importance in order to achieve a fast implementation, since we have to carry out three nested iterative procedures at each time level. Indeed, the lagging procedure transforms the nonlinear system into a sequence of weakly nonlinear systems, each of which is solved, in this paper, by the simplified Newton's method, which, in turn, requires solving a linear system at each iteration, for example by an iterative linear solver. Finally, we introduce several numerical experiments to study the behavior of the algorithm as the velocity term, the inner linear solver or other parameters of the procedure are changed.

The paper is structured as follows. In Section 2 we introduce some assumptions on the smoothness of the involved functions and describe the discretization of the differential problem, illustrating how $\boldsymbol{F}(\boldsymbol{u})$ is made.

After some initial remarks, in Section 3 we then prove some lemmas needed in the successive analysis, devoting particular attention to the analysis of terms dependent on $\tilde{\boldsymbol{v}}$. We then characterize the existence of at least a solution to $\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{0}$. Lastly, we prove that, under some hypotheses, $\boldsymbol{F}(\boldsymbol{u})$ is monotone and that the solution of $\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{0}$ is unique.

Then, in Section 4 we describe the LDM and prove the convergence of the procedure. We also provide a few remarks on the application of the method to stationary problems.

In Section 5 we describe the solution process, describing how we solve the weakly nonlinear systems arising at each iteration of the LDM and how starting vectors and tolerances of all the iterative procedures can be chosen efficiently. The resulting algorithm is reported in Appendix.

Finally, Section 6 is devoted to the numerical experiments and Section 7 concludes this work.

2 The differential problem and finite difference discretization

Consider the general non-steady reaction-convection-diffusion equation. In a two-dimensional diffusion medium Ω with boundary $\partial\Omega$ and closure $\bar{\Omega}$, the differential problem reads

$$\begin{aligned} &\frac{\partial u}{\partial t} = \nabla \cdot (\sigma \nabla u) - \tilde{\boldsymbol{v}} \cdot \nabla u - \alpha u - g + s && (x,y) \in \Omega,\ t > t_0 \\ &u(x,y,t_0) = u_0(x,y) && (x,y) \in \bar{\Omega},\ t = t_0 \geq 0 \\ &u(x,y,t) = u_1(x,y,t) && (x,y) \in \partial\Omega,\ t > t_0 \end{aligned} \tag{1}$$

where t_0 is the initial time, $u = u(x,y,t)$ is the density function, $\sigma = \sigma(x,y,u) > 0$ is the diffusion coefficient, $\tilde{\boldsymbol{v}} = \tilde{\boldsymbol{v}}(x,y,u,t)$ is the velocity vector, $\alpha = \alpha(x,y) \geq 0$ is the absorption term, $-g = -g(x,y,u)$ is the rate of change of the reaction and $s = s(x,y)$ is the source term.

Let $(x,y) \in \Omega$ and let $\hat{u}$ be in a neighborhood of a solution. We assume that the following smoothness assumptions hold.

1. The functions α and s are continuous in their variables and σ and g are continuous in $\hat{u}$ and continuously differentiable in (x,y);
2. there exist two positive constants $\sigma_{\min}$ and $\sigma_{\max}$ such that $0 < \sigma_{\min} \leq \sigma(x,y,\hat{u}) \leq \sigma_{\max}$ (σ uniformly bounded in x,y and $\hat{u}$). Moreover, $\alpha(x,y) \geq \alpha_{\min} \geq 0$;
3. for fixed $(x,y) \in \Omega$, σ is locally Lipschitz continuous in $\hat{u}$ (uniformly in x,y) with constant $\Lambda > 0$;
4. for fixed $(x,y) \in \Omega$, the function g is continuously differentiable in $\hat{u}$ and uniformly monotone in $\hat{u}$ (uniformly in x,y) with constant $c > 0$ [14, p. 30].

These are the standard smoothness assumptions which are considered also when the velocity term is constant. However, if $\tilde{\boldsymbol{v}} = (\tilde{v}_1, \tilde{v}_2)^T$ is variable, we need to add some assumptions on the smoothness of $\tilde{\boldsymbol{v}}$ as well. These assumptions reflect those made on σ. Indeed, in the general, nonlinear case, $\tilde{\boldsymbol{v}}$ is lagged like the diffusivity term and similar assumptions are thus needed in the following analysis. We then require that:

5. the functions $\tilde{v}_1$ and $\tilde{v}_2$ are continuous in their variables;
6. there exist two positive constants $\tilde{v}_{1_{\max}}$ and $\tilde{v}_{2_{\max}}$ such that $|\tilde{v}_1| < \tilde{v}_{1_{\max}}$ and $|\tilde{v}_2| < \tilde{v}_{2_{\max}}$. We call $\tilde{v}_{\max} \coloneqq \max(\tilde{v}_{1_{\max}}, \tilde{v}_{2_{\max}})$;
7. for fixed $(x,y) \in \Omega$, $\tilde{v}_1$ and $\tilde{v}_2$ are locally Lipschitz continuous in $\hat{u}$ (uniformly in x,y) with constant $\Lambda_{\tilde{v}_1} > 0$ and $\Lambda_{\tilde{v}_2} > 0$ respectively. We call $\Lambda_{\tilde{v}} \coloneqq \max(\Lambda_{\tilde{v}_1}, \Lambda_{\tilde{v}_2})$.

2.1 Space discretization

For simplicity, let Ω be a 2D bounded rectangular domain. Let us superimpose to $\bar{\Omega}$ a grid $\bar{\Omega}_h = \Omega_h \cup \partial\Omega_h$ of mesh points (x_i, y_j), $i = 0, \ldots, N+1$, $j = 0, \ldots, M+1$, defined by

$$x_{i+1} = x_i + h, \quad i = 0, \ldots, N \qquad\qquad y_{j+1} = y_j + h \quad j = 0, \ldots, M,$$

where h is the mesh size along x and y (considered uniform).

Space discretization is performed by approximating the space derivatives in (1) by finite difference methods. Committing a discretization error $O(h^2)$ and denoting by $u_{i,j}(t)$ the grid function that approximates the solution $u(x_i, y_j, t)$ at the mesh points (x_i, y_j) of $\bar{\Omega}_h$, $i = 0, \ldots, N+1$, $j = 0, \ldots, M+1$, the right hand side of (1) can be written as

$$\begin{aligned} \Delta_x[\sigma(x_i, y_j, u_{i,j}(t))\nabla_x u_{i,j}(t)] + \Delta_y[\sigma(x_i, y_j, u_{i,j}(t))\nabla_y u_{i,j}(t)] - \tilde{v}_1(x_i, y_j, u_{i,j}(t), t)\delta_x u_{i,j}(t) - \\ - \tilde{v}_2(x_i, y_j, u_{i,j}(t), t)\delta_y u_{i,j}(t) - \alpha(x_i, y_j)u_{i,j}(t) - g(x_i, y_j, u_{i,j}(t)) + s(x_i, y_j, t), \end{aligned} \tag{2}$$

where we denote forward finite differences (in x and in y respectively) by Δ_x, Δ_y, backward finite differences by ∇_x, ∇_y and central finite differences by δ_x, δ_y. For compactness of notation, in the following we also denote $\sigma(x_i, y_j, u_{i,j}(t))$ by $\sigma(u_{i,j})$, $\tilde{v}_1(x_i, y_j, u_{i,j}(t), t)$ by $\tilde{v}_1(u_{i,j}, t)$ and $\tilde{v}_2(x_i, y_j, u_{i,j}(t), t)$ by $\tilde{v}_2(u_{i,j}, t)$.

Equation (2) can then be easily written as

$$\hat{B}_{i,j}u_{i,j-1}(t) + \hat{L}_{i,j}u_{i-1,j}(t) - \hat{D}_{i,j}u_{i,j}(t) + \hat{R}_{i,j}u_{i+1,j}(t) + \hat{T}_{i,j}u_{i,j+1}(t) - g(x_i, y_j, u_{i,j}(t)) + s(x_i, y_j, t), \quad (3)$$

where

$$\hat{L}_{i,j} = L_{i,j} + \tilde{L}_{i,j}, \quad \hat{B}_{i,j} = B_{i,j} + \tilde{B}_{i,j}, \quad \hat{R}_{i,j} = R_{i,j} + \tilde{R}_{i,j}, \quad \hat{T}_{i,j} = T_{i,j} + \tilde{T}_{i,j}, \quad \hat{D}_{i,j} = D_{i,j} + \tilde{D}_{i,j},$$

with

$$L_{i,j} = \frac{1}{h^2}\sigma(u_{i,j}), \quad B_{i,j} = \frac{1}{h^2}\sigma(u_{i,j}), \quad R_{i,j} = \frac{1}{h^2}\sigma(u_{i+1,j}), \quad T_{i,j} = \frac{1}{h^2}\sigma(u_{i,j+1}), \quad D_{i,j} = L_{i,j} + B_{i,j} + R_{i,j} + T_{i,j},$$

$$\tilde{L}_{i,j} = \frac{\tilde{v}_1(u_{i,j}, t)}{2h}, \quad \tilde{B}_{i,j} = \frac{\tilde{v}_2(u_{i,j}, t)}{2h}, \quad \tilde{R}_{i,j} = -\tilde{L}_{i,j} \qquad \tilde{T}_{i,j} = -\tilde{B}_{i,j}, \qquad \tilde{D}_{i,j} = \alpha(x_i, y_j).$$

We can write (3) more conveniently in matrix form. In this regard, we order the mesh points $P_k = (x_i, y_j)$, $i = 1, \ldots, N$, $j = 1, \ldots, M$, by row lexicographic ordering (i.e. $k = (j-1)N+i$, for $i = 1, \ldots, N$, $j = 1, \ldots, M$) and write the vector $\boldsymbol{u}(t)$ containing all the $u_{i,j}(t)$ at internal mesh points. For compactness, we also define $\mu \coloneqq NM$.

We then write the first five terms of (3) as a matrix-vector product $-A(\boldsymbol{u}(t))\boldsymbol{u}(t)$ plus a vector $\boldsymbol{b}(\boldsymbol{u}(t)) \in \mathbb{R}^\mu$ which contains terms coming from the Dirichlet boundary conditions. In particular, $A(\boldsymbol{u}(t)) \in \mathbb{R}^{\mu\times\mu}$ is a block tridiagonal matrix. The M diagonal blocks are tridiagonal matrices of order N of elements $\{-\hat{L}_{i,j}, \hat{D}_{i,j}, -\hat{R}_{i,j}\}$. Its lower- and upper-diagonal blocks are instead diagonal and are made of the elements $\{-\hat{B}_{i,j}\}$ and $\{-\hat{T}_{i,j}\}$, respectively. Moreover, it is irreducible and, if

$$h < \min\left\{\frac{2\sigma_{i,j}(u_{i,j})}{|\tilde{v}_1(u_{i,j}, t)|}, \frac{2\sigma_{i,j}(u_{i,j})}{|\tilde{v}_2(u_{i,j}, t)|}\right\}, \quad \forall (x_i, y_j) \in \Omega_h \quad (4)$$

holds, it is irreducibly diagonally dominant [20, p.23] with positive diagonal elements and non positive off-diagonal elements. In this case, $A(\boldsymbol{u}(t))$ is, thus, a non-singular M-matrix [20, p.91]. The vector $\boldsymbol{b}(\boldsymbol{u}(t))$ can instead be easily obtained by applying the boundary conditions to the terms depending on $\boldsymbol{u}(t)$ in (3).

Regarding the other terms, we collect reaction terms in the nonlinear mapping $\boldsymbol{G}(\boldsymbol{u}) \in \mathbb{R}^\mu$ of components $G_k(\boldsymbol{u}) = G_k(u_k) = g(x_i, y_j, u_k)$, with $i = 1, \ldots, N$, $j = 1, \ldots, M$ and $k = (j-1)N + i$. The mapping $\boldsymbol{G}(\boldsymbol{u})$ is diagonal [12, p. 11] since its k-th component, $k = 1, \ldots, \mu$, depends (in $\boldsymbol{u}$) only on the k-th component u_k of $\boldsymbol{u}$. Finally, we write source terms in the vector $\boldsymbol{s}(t) \in \mathbb{R}^\mu$ of components $s_k(t) = s(x_i, y_j, t)$, with $i = 1, \ldots, N$, $j = 1, \ldots, M$ and $k = (j-1)N + i$.

We thus write (3) for $i = 1, \ldots, N$, $j = 1, \ldots, M$ in the form

$$-A(\boldsymbol{u}(t))\boldsymbol{u}(t) + \boldsymbol{b}(\boldsymbol{u}(t)) - \boldsymbol{G}(\boldsymbol{u}(t)) + \boldsymbol{s}(t) \quad (5)$$

and (1) is replaced by the system of ordinary differential equations

$$\begin{aligned} &\frac{\mathrm{d}\boldsymbol{u}(t)}{\mathrm{d}t} = -A(\boldsymbol{u}(t))\boldsymbol{u}(t) + \boldsymbol{b}(\boldsymbol{u}(t)) - \boldsymbol{G}(\boldsymbol{u}(t)) + \boldsymbol{s}(t) \\ &u(x, y, t_0) = u_0(x, y), \quad (x, y) \in \bar{\Omega}. \end{aligned} \quad (6)$$

2.2 Time discretization

Passing to time discretization, let us introduce a time spacing Δt and define a series of time levels $t_n = n\Delta t + t_0$, $n = 0, 1, \ldots$. Calling $\boldsymbol{s}^n \coloneqq \boldsymbol{s}(t_n)$ and denoting by $\boldsymbol{u}^n$ the approximation of $\boldsymbol{u}(t_n)$ solution of (6) at $t = t_n$, we write the well-known θ-method (e.g. see [6])

$$\begin{aligned} \frac{\boldsymbol{u}^{n+1} - \boldsymbol{u}^n}{\Delta t} =& \theta\left[-A(\boldsymbol{u}^{n+1})\boldsymbol{u}^{n+1} + \boldsymbol{b}(\boldsymbol{u}^{n+1}) - \boldsymbol{G}(\boldsymbol{u}^{n+1}) + \boldsymbol{s}^{(n+1)}\right] + \\ &+(1-\theta)\left[-A(\boldsymbol{u}^n)\boldsymbol{u}^n + \boldsymbol{b}(\boldsymbol{u}^n) - \boldsymbol{G}(\boldsymbol{u}^n) + \boldsymbol{s}^n\right], \end{aligned} \quad (7)$$

for $n = 0, 1, \ldots$ and with $0 \leq \theta \leq 1$. In particular, in the following we consider an implicit time discretization (hence, we consider cases where $\theta \neq 0$). The use of an explicit scheme would, indeed, easily require an extremely short time step. For instance, it is easy to verify that the problems later analyzed in the numerical experiments would need a time-step as small as $\Delta t = 10^{-6}$ for the explicit method to converge. This, of course, greatly reduces efficiency. The advantages of an implicit time discretization come, however, at the cost of a nonlinearity in the discretized system, which we will later handle by the lagging procedure.

Let us then write more compactly the nonlinear system arising from the discretization. Calling I the $\mu \times \mu$ identity matrix, we collect the known terms in a vector $\boldsymbol{w} = \boldsymbol{w}^n \in \mathbb{R}^\mu$ defined as

$$\boldsymbol{w} = [I - \Delta t(1-\theta)A(\boldsymbol{u}^n)]\,\boldsymbol{u}^n + \Delta t(1-\theta)\,[\boldsymbol{b}(\boldsymbol{u}^n) - \boldsymbol{G}(\boldsymbol{u}^n)] + \Delta t\left[\theta \boldsymbol{s}^{n+1} + (1-\theta)\boldsymbol{s}^n\right].$$

At each time level $n = 0, 1, \ldots$, the vector $\boldsymbol{u}^{n+1}$ is thus given by the solution of the nonlinear algebraic system

$$\boldsymbol{F}(\boldsymbol{u}) = [I + \tau A(\boldsymbol{u})]\,\boldsymbol{u} - \tau\,[\boldsymbol{b}(\boldsymbol{u}) - \boldsymbol{G}(\boldsymbol{u})] - \boldsymbol{w} = \boldsymbol{0}, \tag{8}$$

where $\tau \coloneqq \theta \Delta t$.

3 Uniform monotonicity of $\boldsymbol{F}(\boldsymbol{u})$ and uniqueness of the solution of $\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{0}$

Thus, in (8) we obtained the nonlinear algebraic system $\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{0}$ that we aim to solve by the lagged diffusivity method. In the first part of this section, we assume that a solution exists and introduce some preliminary lemmas used in the following. We then better characterize the existence of at least one solution by exploiting the smoothness assumptions at the beginning of Section 2. Lastly, we study the monotonicity of the operator $\boldsymbol{F}(\boldsymbol{u})$ and the uniqueness of the solution of $\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{0}$.

3.1 Initial remarks

It is convenient to split the matrix $A(\boldsymbol{u})$ in

$$A(\boldsymbol{u}) = A_1(\boldsymbol{u}) + \tilde{A}(\boldsymbol{u}) + \tilde{D},$$

where $A_1(\boldsymbol{u})$ is a block tridiagonal matrix of row elements $\{-B_{i,j}, -L_{i,j}, D_{i,j}, -R_{i,j}, -T_{i,j}\}$ with the same structure as $A(\boldsymbol{u})$ and $\tilde{D}$ is the diagonal matrix of diagonal elements $\{\tilde{D}_{i,j}\}$. Finally, $\tilde{A}(\boldsymbol{u})$ is the block tridiagonal matrix whose lower- and upper-diagonal blocks are diagonal matrices of the elements $\{-\tilde{B}_{i,j}\}$ and $\{-\tilde{T}_{i,j}\}$, respectively. Its diagonal blocks are instead tridiagonal matrices of elements $\{-\tilde{L}_{i,j}, 0, -\tilde{R}_{i,j}\}$.

We then further split $A_1(\boldsymbol{u})$ and $\tilde{A}(\boldsymbol{u})$ in

$$A_1(\boldsymbol{u}) = A_1^x(\boldsymbol{u}) + A_1^y(\boldsymbol{u}) \qquad \tilde{A}(\boldsymbol{u}) = \tilde{A}^x(\boldsymbol{u}) + \tilde{A}^y(\boldsymbol{u})$$

where $A_1^x(\boldsymbol{u})$ and $\tilde{A}^x(\boldsymbol{u})$ are block diagonal matrices whose diagonal blocks are tridiagonal with row elements $\{-L_{i,j}, D_{i,j}^x, -R_{i,j}\}$ (with $D_{i,j}^x = L_{i,j} + R_{i,j}$) and $\{-\tilde{L}_{i,j}, 0, \tilde{R}_{i,j}\}$ respectively. $A_1^y(\boldsymbol{u})$ is the block tridiagonal matrix where the sub-, main and super-diagonal blocks are diagonal matrices of diagonal elements $-B_{i,j}$, $D_{i,j}^y$ and $-T_{i,j}$ respectively, with $D_{i,j}^y = B_{i,j} + T_{i,j}$. Finally, $\tilde{A}^y(\boldsymbol{u})$ has the same structure of $A_1^y(\boldsymbol{u})$ and the diagonal matrices have diagonal elements $-\tilde{B}_{i,j}$, 0 and $-\tilde{T}_{i,j}$ respectively.

It is useful to split also $\boldsymbol{b}(\boldsymbol{u})$ in a similar way. We thus write

$$\boldsymbol{b}(\boldsymbol{u}) = \boldsymbol{b}_1(\boldsymbol{u}) + \tilde{\boldsymbol{b}}(\boldsymbol{u}),$$

where $\boldsymbol{b}_1(\boldsymbol{u})$ and $\tilde{\boldsymbol{b}}(\boldsymbol{u})$ contain, respectively, terms dependent on the diffusivity and on the velocity term. We then further split these two vectors in

$$\boldsymbol{b}_1(\boldsymbol{u}) = \boldsymbol{b}_1^x(\boldsymbol{u}) + \boldsymbol{b}_1^y(\boldsymbol{u}), \qquad \tilde{\boldsymbol{b}}(\boldsymbol{u}) = \tilde{\boldsymbol{b}}^x(\boldsymbol{u}) + \tilde{\boldsymbol{b}}^y(\boldsymbol{u})$$

where $\boldsymbol{b}_1^x(\boldsymbol{u})$ and $\tilde{\boldsymbol{b}}^x(\boldsymbol{u})$ contain the contributions $u_1(x_0, y_j, t)$ and $u_1(x_{N+1}, y_j, t)$, $j = 1, \ldots, M$ and $\boldsymbol{b}_1^y(\boldsymbol{u})$ and $\tilde{\boldsymbol{b}}^y(\boldsymbol{u})$ contain the contributions $u_1(x_i, y_0, t)$ and $u_1(x_i, y_{M+1}, t)$, $i = 1, \ldots, N$.

Finally, given two grid functions $\boldsymbol{u}$, $\boldsymbol{v}$ in $\bar{\Omega}_h$, in the following we make use also of the $l_2(\Omega_h)$ discrete inner product

$$\langle \boldsymbol{u}, \boldsymbol{v} \rangle = h^2 \sum_{i=1}^{N} \sum_{j=1}^{M} u_{i,j} v_{i,j}$$

and of its associated norm $\| \cdot \|_h$.

3.2 Preliminary lemmas on $\tilde{A}(\boldsymbol{u})$

Lemma 1. *Let $\{u_{i,j}\}$, $\{v_{i,j}\}$ and $\{w_{i,j}\}$ be three grid functions defined at mesh points (x_i, y_j) of a grid $\bar{\Omega}_h$, $i = 0, ...N + 1$, $j = 0, ..., M + 1$, and satisfying the Dirichlet boundary conditions $u_{i,j} = u_1(x_i, y_j, t)$ $\forall (x_i, y_j) \in \partial\Omega_h$ and $t > 0$. Then:*

$$\left\langle \tilde{A}(\boldsymbol{u}) \cdot \boldsymbol{w} - \tilde{\boldsymbol{b}}(\boldsymbol{u}), \boldsymbol{v} \right\rangle = h^2 \sum_{i=1}^{N} \sum_{j=1}^{M} v_{i,j} \left[\tilde{v}_1(u_{i,j}) \delta_x w_{i,j} + \tilde{v}_2(u_{i,j}) \delta_y w_{i,j} \right]. \tag{9}$$

Proof. Let us split the inner product in $\langle \tilde{A}(\boldsymbol{u}) \cdot \boldsymbol{w} - \tilde{\boldsymbol{b}}(\boldsymbol{u}), \boldsymbol{v} \rangle = \langle \tilde{A}^x(\boldsymbol{u}) \cdot \boldsymbol{w} - \tilde{\boldsymbol{b}}^x(\boldsymbol{u}), \boldsymbol{v} \rangle + \langle \tilde{A}^y(\boldsymbol{u}) \cdot \boldsymbol{w} - \tilde{\boldsymbol{b}}^y(\boldsymbol{u}), \boldsymbol{v} \rangle$. By definition of discrete $l_2(\Omega_h)$ inner product and by the form of $\tilde{A}^x(\boldsymbol{u})$ and of $\tilde{\boldsymbol{b}}^x(\boldsymbol{u})$, with simple algebraic passages we find

$$\left\langle \tilde{A}^x(\boldsymbol{u}) \cdot \boldsymbol{w} - \tilde{\boldsymbol{b}}^x(\boldsymbol{u}), \boldsymbol{v} \right\rangle = h^2 \sum_{i=1}^{N} \sum_{j=1}^{M} \left(-\frac{\tilde{v}_1(u_{i,j})}{2h} w_{i-1,j} + \frac{\tilde{v}_1(u_{i,j})}{2h} w_{i+1,j} \right) v_{i,j}.$$

Then, collecting terms, by definition of central finite-difference quotients we get

$$\left\langle \tilde{A}^x(\boldsymbol{u}) \cdot \boldsymbol{w} - \tilde{\boldsymbol{b}}^x(\boldsymbol{u}), \boldsymbol{v} \right\rangle = h^2 \sum_{j=1}^{M} \sum_{i=1}^{N} \tilde{v}_1(u_{i,j}) \Big(\frac{w_{i+1,j} - w_{i-1,j}}{2h} \Big) v_{i,j} = h^2 \sum_{j=1}^{M} \sum_{i=1}^{N} \tilde{v}_1(u_{i,j}) \delta_x(w_{i,j}) v_{i,j}.$$

We find a similar relationship by proceeding analogously for the term containing $\tilde{A}^y$. Using these results in (9) and collecting terms, we finally prove the lemma. □

Corollary 1. *Expressions analogous to* (9) *can be obtained also when $v_{i,j}$ satisfies homogeneous Dirichlet boundary conditions for $\langle \tilde{A}(\boldsymbol{u}) \cdot \boldsymbol{w} - \tilde{\boldsymbol{b}}(\boldsymbol{u}), \boldsymbol{v} \rangle$ and for $\langle \tilde{A}(\boldsymbol{u}) \cdot \boldsymbol{v}, \boldsymbol{v} \rangle$.*

Proof. The proofs follow Lemma 1 without relevant modifications. Regarding $\langle \tilde{A}(\boldsymbol{u}) \cdot \boldsymbol{v}, \boldsymbol{v} \rangle$, however, we also need to consider that, since $\boldsymbol{v}$ is null on $\partial\Omega_h$, we have

$$\sum_{j=1}^{M} \sum_{i=1}^{N-1} \tilde{v}_1(u_{i,j}) v_{i+1,j} = \sum_{j=1}^{M} \sum_{i=1}^{N} \tilde{v}_1(u_{i,j}) v_{i+1,j}; \qquad \sum_{j=1}^{M} \sum_{i=2}^{N} \tilde{v}_1(u_{i,j}) v_{i-1,j} = \sum_{j=1}^{M} \sum_{i=1}^{N} \tilde{v}_1(u_{i,j}) v_{i-1,j}$$

and analogous expressions for terms in $\tilde{v}_2(u_{i,j}) v_{i,j\pm1}$. □

Lemma 2. *Let $\{u_{i,j}\}$ be a grid function defined at mesh points (x_i, y_j) of a grid $\bar{\Omega}_h$, $i = 0, \ldots, N + 1$, $j = 0, \ldots, M+1$, such that $\{u_{i,j}\}$ satisfies the Dirichlet boundary conditions $u_{i,j} = u_1(x_i, y_j, t)$ $\forall (x_i, y_j) \in \partial\Omega_h$ and $t > 0$. Moreover, let the backward difference quotients $\nabla_x u_{i,j}$ and $\nabla_y u_{i,j}$ be bounded. Then, denoting by $\tilde{v}_{\max}$ the bound on the velocity variable (see point 6, Section 2),*

$$\left\langle \tilde{A}(\boldsymbol{u}) \cdot \boldsymbol{u} - \tilde{\boldsymbol{b}}(\boldsymbol{u}), \boldsymbol{u} \right\rangle \geq -\frac{\tilde{v}_{\max}}{h} \|\boldsymbol{u}\|_h^2 - \frac{h^3 \tilde{v}_{\max}}{2} \left[\sum_{j=1}^{M} \sum_{i=1}^{N+1} (\nabla_x u_{i,j})^2 + \sum_{i=1}^{N} \sum_{j=1}^{M+1} (\nabla_y u_{i,j})^2 \right]. \tag{10}$$

Proof. By $\delta_x u_{i,j} = (\nabla_x u_{i+1,j} + \nabla_x u_{i,j})/2$ (and similarly for $\delta_y u_{i,j}$), we can write

$$\begin{aligned}\left\langle \tilde{A}(\boldsymbol{u})\cdot\boldsymbol{u} - \tilde{\boldsymbol{b}}(\boldsymbol{u}), \boldsymbol{u}\right\rangle =& h^2 \sum_{i=1}^{N}\sum_{j=1}^{M} u_{i,j}\left[\tilde{v}_1(u_{i,j})\delta_x u_{i,j} + \tilde{v}_2(u_{i,j})\delta_y u_{i,j}\right] \\ =& \frac{h^2}{2}\sum_{i=1}^{N}\sum_{j=1}^{M} u_{i,j}\Big[\tilde{v}_1(u_{i,j})\Big(\nabla_x u_{i+1,j} + \nabla_x u_{i,j}\Big) + \tilde{v}_2(u_{i,j})\Big(\nabla_y u_{i,j+1} + \nabla_y u_{i,j}\Big)\Big] \leq \\ \leq& \frac{h^2}{2}\sum_{i=1}^{N}\sum_{j=1}^{M}\Big[|u_{i,j}||\tilde{v}_1(u_{i,j})|\Big(\big|\nabla_x u_{i+1,j}\big| + \big|\nabla_x u_{i,j}\big|\Big) + |u_{i,j}||\tilde{v}_2(u_{i,j})|\Big(\big|\nabla_y u_{i,j+1}\big| + \big|\nabla_y u_{i,j}\big|\Big)\Big].\end{aligned}$$

By the boundedness of $\tilde{\boldsymbol{v}}$ and by the inequality $ab \leq (a^2+b^2)/2$ with a, b real numbers, we can further evaluate

$$\begin{aligned}\left\langle \tilde{A}(\boldsymbol{u})\cdot\boldsymbol{u} - \tilde{\boldsymbol{b}}(\boldsymbol{u}), \boldsymbol{u}\right\rangle \leq& \frac{h\tilde{v}_{\max}}{4}\sum_{i=1}^{N}\sum_{j=1}^{M}\Big[4|u_{i,j}|^2 + h^2\big|\nabla_x u_{i+1,j}\big|^2 + h^2\big|\nabla_x u_{i,j}\big|^2 + h^2\big|\nabla_y u_{i,j+1}\big|^2 + h^2\big|\nabla_y u_{i,j}\big|^2\Big] \leq \\ \leq& h\tilde{v}_{\max}\sum_{i=1}^{N}\sum_{j=1}^{M}|u_{i,j}|^2 + \frac{h^3\tilde{v}_{\max}}{2}\Big[\sum_{j=1}^{M}\sum_{i=1}^{N+1}(\nabla_x u_{i,j})^2 + \sum_{i=1}^{N}\sum_{j=1}^{M+1}(\nabla_y u_{i,j})^2\Big].\end{aligned}$$

Changing sign to the right-hand side of the equation and by definition of $\|\boldsymbol{u}\|_h$ norm, we finally prove the lemma. □

3.3 Existence of at least one solution to the discretized system

We now introduce a sufficient condition for the existence of at least one solution to the discretized system (8). We notice that we can put ourselves in this condition by a suitable choice of the discretization parameters h and τ.

Theorem 1. *Let $\{u_{i,j}\}$ be a grid function defined at mesh points (x_i, y_j) of a grid $\bar{\Omega}_h$, $i = 0, \ldots, N+1$, $j = 0, \ldots, M+1$, such that $\{u_{i,j}\}$ satisfies the Dirichlet boundary conditions $u_{i,j} = u_1(x_i, y_j, t) \forall (x_i, y_j) \in \partial\Omega_h$ and $t > 0$. Moreover, let the backward difference quotients $\nabla_x u_{i,j}$ and $\nabla_y u_{i,j}$ be bounded and let $\sigma_{\min}$, $\tilde{v}_{\max}$ and $\alpha_{\min}$ be the bounds on diffusivity, velocity and absorption terms defined at the beginning of Section 2. If*

$$h \leq 2\sigma_{\min}/\tilde{v}_{\max} \qquad \text{and} \qquad \frac{1}{\tau} + \alpha_{\min} + c > \frac{\tilde{v}_{\max}}{h},$$

then there exists at least one solution to system (8)*. Furthermore, all solutions belong to a ball of radius ρ with*

$$\rho = \frac{\tau\|\boldsymbol{G}(\boldsymbol{0})\|_h + \|\boldsymbol{w}\|_h}{1 - \frac{\tau\tilde{v}_{\max}}{h} + \tau\alpha_{\min} + \tau c}$$

Proof. Let us consider $\langle \boldsymbol{F}(\boldsymbol{u}), \boldsymbol{u}\rangle$. We can write

$$\begin{aligned}\langle \boldsymbol{F}(\boldsymbol{u}), \boldsymbol{u}\rangle &= \langle [I + \tau A(\boldsymbol{u})]\,\boldsymbol{u} - \tau\,[\boldsymbol{b}(\boldsymbol{u}) - \boldsymbol{G}(\boldsymbol{u})] - \boldsymbol{w}, \boldsymbol{u}\rangle = \\ &= \langle \boldsymbol{u}, \boldsymbol{u}\rangle + \tau\langle A(\boldsymbol{u})\boldsymbol{u} - \boldsymbol{b}(\boldsymbol{u}), \boldsymbol{u}\rangle + \tau\langle \boldsymbol{G}(\boldsymbol{u}) - \boldsymbol{G}(\boldsymbol{0}), \boldsymbol{u}\rangle + \langle \tau\boldsymbol{G}(\boldsymbol{0}) - \boldsymbol{w}, \boldsymbol{u}\rangle\end{aligned}$$

where

$$\langle A(\boldsymbol{u})\boldsymbol{u} - \boldsymbol{b}(\boldsymbol{u}), \boldsymbol{u}\rangle = \langle A_1(\boldsymbol{u})\boldsymbol{u} - \boldsymbol{b}_1(\boldsymbol{u}), \boldsymbol{u}\rangle + \langle \tilde{A}\boldsymbol{u} - \tilde{\boldsymbol{b}}(\boldsymbol{u}), \boldsymbol{u}\rangle + \langle \tilde{D}\boldsymbol{u}, \boldsymbol{u}\rangle.$$

Many of these terms can be easily evaluated. Indeed, $\langle \boldsymbol{u}, \boldsymbol{u}\rangle = \|\boldsymbol{u}\|_h^2$ by definition, and

$$\langle \tilde{D}\boldsymbol{u}, \boldsymbol{u}\rangle = h^2\sum_{i=1}^{N}\sum_{j=1}^{M}\alpha_{i,j}u_{i,j}^2 \geq \alpha_{\min}\|\boldsymbol{u}\|_h^2$$

by the lower bound and the non-negativity of α. Moreover,

$$\langle \boldsymbol{G}(\boldsymbol{u}) - \boldsymbol{G}(\boldsymbol{0}), \boldsymbol{u} \rangle \geq c \|\boldsymbol{u}\|_h^2$$

by the uniform monotonicity of g in u and

$$\langle \tau \boldsymbol{G}(\boldsymbol{0}) - \boldsymbol{w}, \boldsymbol{u} \rangle = \tau \langle \boldsymbol{G}(\boldsymbol{0}), \boldsymbol{u} \rangle - \langle \boldsymbol{w}, \boldsymbol{u} \rangle \geq -\tau \|\boldsymbol{G}(\boldsymbol{0})\|_h \|\boldsymbol{u}\|_h - \|\boldsymbol{w}\|_h \|\boldsymbol{u}\|_h$$

by the Cauchy-Schwarz inequality.
Regarding the other terms, we instead rely on other evaluations. Thus, by [11, Lemma 1], we easily find

$$\langle A_1(\boldsymbol{u}) \cdot \boldsymbol{u} - \boldsymbol{b}(\boldsymbol{u}), \boldsymbol{u} \rangle \geq h^2 \sigma_{\min} \left[\sum_{j=1}^{M} \sum_{i=1}^{N+1} (\nabla_x u_{i,j})^2 + \sum_{i=1}^{N} \sum_{j=1}^{M+1} (\nabla_y u_{i,j})^2 \right],$$

while Lemma 2 directly provides

$$\left\langle \tilde{A}(\boldsymbol{u}) \cdot \boldsymbol{u} - \tilde{\boldsymbol{b}}(\boldsymbol{u}), \boldsymbol{u} \right\rangle \geq -\frac{\tilde{v}_{\max}}{h} \|\boldsymbol{u}\|_h^2 - \frac{h^3 \tilde{v}_{\max}}{2} \left[\sum_{j=1}^{M} \sum_{i=1}^{N+1} (\nabla_x u_{i,j})^2 + \sum_{i=1}^{N} \sum_{j=1}^{M+1} (\nabla_y u_{i,j})^2 \right].$$

Therefore, combining all these evaluations,

$$\begin{aligned} \langle \boldsymbol{F}(\boldsymbol{u}), \boldsymbol{u} \rangle \geq & \|\boldsymbol{u}\|_h^2 - \frac{\tau \tilde{v}_{\max}}{h} \|\boldsymbol{u}\|_h^2 + \tau \left(h^2 \sigma_{\min} - \frac{h^3 \tilde{v}_{\max}}{2} \right) \left[\sum_{j=1}^{M} \sum_{i=1}^{N+1} (\nabla_x u_{i,j})^2 + \sum_{i=1}^{N} \sum_{j=1}^{M+1} (\nabla_y u_{i,j})^2 \right] + \\ & + \tau \alpha_{\min} \|\boldsymbol{u}\|_h^2 + c \|\boldsymbol{u}\|_h^2 - \tau \|\boldsymbol{G}(\boldsymbol{0})\|_h \|\boldsymbol{u}\|_h - \|\boldsymbol{w}\|_h \|\boldsymbol{u}\|_h. \end{aligned}$$

Since $h \leq 2\sigma_{\min}/\tilde{v}_{\max}$ by hypothesis, $\sigma_{\min} - h\tilde{v}_{\max}/2 \geq 0$ and we can further evaluate from above by canceling out the term with backward quotients. We also notice that this condition on the step h of the space discretization implies that condition (4) is satisfied as well and $A(\boldsymbol{u})$ is then an M-matrix.

Collecting $\|\boldsymbol{u}\|_h$, we finally have

$$\langle \boldsymbol{F}(\boldsymbol{u}), \boldsymbol{u} \rangle \geq \left(\|\boldsymbol{u}\|_h - \frac{\tau \tilde{v}_{\max}}{h} \|\boldsymbol{u}\|_h + \tau \alpha_{\min} \|\boldsymbol{u}\|_h + c \|\boldsymbol{u}\|_h - \tau \|\boldsymbol{G}(\boldsymbol{0})\|_h - \|\boldsymbol{w}\|_h \right) \|\boldsymbol{u}\|_h$$

and, since the second hypothesis implies $1 - \tau \tilde{v}_{\max}/h + \tau \alpha_{\min} + \tau c > 0$,

$$\|\boldsymbol{u}\|_h > \rho := \frac{\tau \|\boldsymbol{G}(\boldsymbol{0})\|_h + \|\boldsymbol{w}\|_h}{1 - \frac{\tau \tilde{v}_{\max}}{h} + \tau \alpha_{\min} + \tau c} \quad \text{implies} \quad \langle \boldsymbol{F}(\boldsymbol{u}), \boldsymbol{u} \rangle > 0.$$

This evidently means that no solution to $\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{0}$ can lie outside $\{\boldsymbol{u} \mid \|\boldsymbol{u}\|_h \leq \rho\}$.

The existence of at least one solution follows then by [8, Lemma 4.3]. Another proof, based on the invariance of the degree of a mapping under a homotopy [12, p. 156], is provided in [9, Theorem 2.1], which considers that $\langle \boldsymbol{H}(\boldsymbol{u}, \delta), \boldsymbol{u} \rangle > 0$ when $\|\boldsymbol{u}\|_h > \rho$ for all δ in $[0, 1]$, where $\boldsymbol{H}(\boldsymbol{u}, \delta) = \delta \boldsymbol{F}(\boldsymbol{u}) + (1 - \delta)\boldsymbol{u}$. □

The condition $h < 2\sigma_{\min}/\tilde{v}_{\max}$, often implying h small, is evidently in competition with the condition $1 - \tau \tilde{v}_{\max}/h + \tau \alpha_{\min} + \tau c > 0$, which is harder to satisfy as h gets smaller. It is however important to notice that a small step in the space grid can be compensated by an equally small step in time. Thus, in line of principle, both conditions can always be satisfied by acting on the size of space and time grids.

Finally, it is also possible to provide a bound to the backward difference quotients of a solution of (8) at each mesh point $(x_i, y_j) \in \bar{\Omega}$ by proceeding similarly to [9, Lemma 2.2]. We denote this bound by β. It is then convenient to define the ball $\mathcal{B}_{\rho,\beta}$ to which all solutions must belong.

Definition 1. Let $\boldsymbol{u} : \bar{\Omega}_h \to \mathbb{R}$ be a grid function satisfying the Dirichlet boundary conditions on $\partial\Omega_h$ for $t > t_0$ and let the hypotheses of Theorem 1 be satisfied. We say that $\boldsymbol{u}$ belongs to $\mathcal{B}_{\rho,\beta}$ if

$$\|\boldsymbol{u}\|_h \leq \rho \tag{11}$$

$$|\nabla_x u_{i,j}| \leq \beta, \quad |\nabla_y u_{i,j}| \leq \beta, \quad \text{for } i = 1, \ldots, N,\ j = 1, \ldots, M. \tag{12}$$

3.4 Monotonicity analysis

Before studying the monotonicity of $\boldsymbol{F}(\boldsymbol{u})$ we need a last lemma, which relies on the bound β to the finite difference quotients of the grid functions in $\mathcal{B}_{\rho,\beta}$.

Lemma 3. *Let $\boldsymbol{u}$, $\boldsymbol{v}$ and $\boldsymbol{w}$ be three grid functions belonging to $\mathcal{B}_{\rho,\beta}$. Then,*

$$\left|\left\langle \left(\tilde{A}(\boldsymbol{u})-\tilde{A}(\boldsymbol{w})\right)\cdot\boldsymbol{v}-\tilde{\boldsymbol{b}}(\boldsymbol{u})+\tilde{\boldsymbol{b}}(\boldsymbol{w}),\boldsymbol{u}-\boldsymbol{v}\right\rangle\right| \leq \beta\Lambda_{\tilde{v}}\Big[\|\boldsymbol{u}-\boldsymbol{v}\|^2+\|\boldsymbol{u}-\boldsymbol{w}\|^2\Big]. \tag{13}$$

Proof. Since $\boldsymbol{u}-\boldsymbol{v}$ is null on the boundary, we can apply Corollary 1 with $\boldsymbol{u}-\boldsymbol{v}$ instead of $\boldsymbol{v}$. We have

$$\left\langle \left(\tilde{A}(\boldsymbol{u})-\tilde{A}(\boldsymbol{w})\right)\cdot\boldsymbol{v}-\tilde{\boldsymbol{b}}(\boldsymbol{u})+\tilde{\boldsymbol{b}}(\boldsymbol{w}),\boldsymbol{u}-\boldsymbol{v}\right\rangle = \left\langle \tilde{A}(\boldsymbol{u})\cdot\boldsymbol{v}-\tilde{\boldsymbol{b}}(\boldsymbol{u}),\boldsymbol{u}-\boldsymbol{v}\right\rangle - \left\langle \tilde{A}(\boldsymbol{w})\cdot\boldsymbol{v}-\tilde{\boldsymbol{b}}(\boldsymbol{w}),\boldsymbol{u}-\boldsymbol{v}\right\rangle =$$
$$=h^2\sum_{i=1}^{N}\sum_{j=1}^{M}\Big\{(u_{i,j}-v_{i,j})\Big[\delta_x(v_{i,j})(\tilde{v}_1(u_{i,j})-\tilde{v}_1(w_{i,j}))+\delta_y(v_{i,j})(\tilde{v}_2(u_{i,j})-\tilde{v}_2(w_{i,j}))\Big]\Big\}.$$

Therefore, by the triangle inequality,

$$\left|\left\langle \left(\tilde{A}(\boldsymbol{u})-\tilde{A}(\boldsymbol{w})\right)\cdot\boldsymbol{v}-\tilde{\boldsymbol{b}}(\boldsymbol{u})+\tilde{\boldsymbol{b}}(\boldsymbol{w}),\boldsymbol{u}-\boldsymbol{v}\right\rangle\right| \leq$$
$$\leq h^2\sum_{i=1}^{N}\sum_{j=1}^{M}\Big\{|u_{i,j}-v_{i,j}|\Big[\,|\delta_x v_{i,j}|\,|\tilde{v}_1(u_{i,j})-\tilde{v}_1(w_{i,j})|+|\delta_y v_{i,j}|\,|\tilde{v}_2(u_{i,j})-\tilde{v}_2(w_{i,j})|\,\Big]\Big\}.$$

Next, by definition of backward and central finite-difference quotients and by boundedness of backward finite-difference quotients (12) in $\mathcal{B}_{\rho,\beta}$, we can write

$$|\delta_x v_{i,j}| = \left|\frac{1}{2}(\nabla_x v_{i+1,j}+\nabla_x v_{i,j})\right| \leq \frac{1}{2}(|\nabla_x v_{i+1,j}|+|\nabla_x v_{i,j}|) \leq \beta.$$

Furthermore, by Lipschitz continuity of every component of the velocity vector $\tilde{\boldsymbol{v}}$, we have

$$|\tilde{v}_1(u_{i,j})-\tilde{v}_1(w_{i,j})| \leq \Lambda_{\tilde{v}_1}|u_{i,j}-w_{i,j}| \leq \Lambda_{\tilde{v}}|u_{i,j}-w_{i,j}|.$$

Identical evaluations apply, respectively, to $|\delta_y v_{i,j}|$ and to $|\tilde{v}_2(u_{i,j})-\tilde{v}_2(w_{i,j})|$.

Therefore, combining these results,

$$\left|\left\langle \left(\tilde{A}(\boldsymbol{u})-\tilde{A}(\boldsymbol{w})\right)\cdot\boldsymbol{v}-\tilde{\boldsymbol{b}}(\boldsymbol{u})+\tilde{\boldsymbol{b}}(\boldsymbol{w}),\boldsymbol{u}-\boldsymbol{v}\right\rangle\right| \leq$$
$$\leq h^2\sum_{i=1}^{N}\sum_{j=1}^{M}\Big\{|u_{i,j}-v_{i,j}|\,\Big[\beta\Lambda_{\tilde{v}}|u_{i,j}-w_{i,j}|+\beta\Lambda_{\tilde{v}}|u_{i,j}-w_{i,j}|\Big]\Big\} = 2h^2\beta\Lambda_{\tilde{v}}\sum_{i=1}^{N}\sum_{j=1}^{M}|u_{i,j}-v_{i,j}|\,|u_{i,j}-w_{i,j}|$$

Since, for a,b real numbers, $ab \leq (a^2+b^2)/2$, we finally get

$$\left|\left\langle \left(\tilde{A}(\boldsymbol{u})-\tilde{A}(\boldsymbol{w})\right)\cdot\boldsymbol{v}-\tilde{\boldsymbol{b}}(\boldsymbol{u})+\tilde{\boldsymbol{b}}(\boldsymbol{w}),\boldsymbol{u}-\boldsymbol{v}\right\rangle\right| \leq h^2\beta\Lambda_{\tilde{v}}\sum_{i=1}^{N}\sum_{j=1}^{M}|u_{i,j}-v_{i,j}|^2+h^2\beta\Lambda_{\tilde{v}}\sum_{i=1}^{N}\sum_{j=1}^{M}|u_{i,j}-w_{i,j}|^2,$$

which, by definition of discrete $l_2(\Omega_h)$ norm, becomes:

$$\left|\left\langle \left(\tilde{A}(\boldsymbol{u})-\tilde{A}(\boldsymbol{w})\right)\cdot\boldsymbol{v}-\tilde{\boldsymbol{b}}(\boldsymbol{u})+\tilde{\boldsymbol{b}}(\boldsymbol{w}),\boldsymbol{u}-\boldsymbol{v}\right\rangle\right| \leq \beta\Lambda_{\tilde{v}}\Big[\|\boldsymbol{u}-\boldsymbol{v}\|^2+\|\boldsymbol{u}-\boldsymbol{w}\|^2\Big].$$

□

Theorem 2. *Let $\boldsymbol{u}^* \in \mathcal{B}_{\rho,\beta}$ be a solution of the nonlinear system $\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{0}$. If*

$$h < \frac{2\sigma_{\min}}{\tilde{v}_{\max}} \quad \textit{and} \quad \alpha_{\min} + \frac{1}{\tau} + c > \frac{\beta^2\Lambda^2}{2\sigma_{\min} - h\tilde{v}_{\max}} + \frac{\tilde{v}_{\max}}{h} + 2\beta\Lambda_{\tilde{v}}, \tag{14}$$

then $\boldsymbol{F}(\boldsymbol{u})$ is uniformly monotone in $\mathcal{B}_{\rho,\beta}$ and the solution $\boldsymbol{u}^*$ of (8) is unique.

Proof. To prove that $\boldsymbol{F}(\boldsymbol{u})$ is uniformly monotone in $\mathcal{B}_{\rho,\beta}$, we must show that there exists a positive scalar γ satisfying

$$\langle \boldsymbol{F}(\boldsymbol{u}) - \boldsymbol{F}(\boldsymbol{v}), \boldsymbol{u} - \boldsymbol{v} \rangle \geq \gamma \langle \boldsymbol{u} - \boldsymbol{v}, \boldsymbol{u} - \boldsymbol{v} \rangle \quad \forall \boldsymbol{u}, \boldsymbol{v} \in \mathcal{B}_{\rho,\beta}. \tag{15}$$

To this end, let us analyze $\langle \boldsymbol{F}(\boldsymbol{u}) - \boldsymbol{F}(\boldsymbol{v}), \boldsymbol{u} - \boldsymbol{v} \rangle$. By the definition of $\boldsymbol{F}(\boldsymbol{u})$ in (8), adding and subtracting $A(\boldsymbol{u})\boldsymbol{v}$ and rearranging terms, we can write

$$\begin{aligned}
&\frac{1}{\tau} \langle \boldsymbol{F}(\boldsymbol{u}) - \boldsymbol{F}(\boldsymbol{v}), \boldsymbol{u} - \boldsymbol{v} \rangle = \\
&= \left\langle \left(A(\boldsymbol{u}) + \frac{I}{\tau} \right) (\boldsymbol{u} - \boldsymbol{v}), \boldsymbol{u} - \boldsymbol{v} \right\rangle + \left\langle (A(\boldsymbol{u}) - A(\boldsymbol{v}))\boldsymbol{v} - \boldsymbol{b}(\boldsymbol{u}) + \boldsymbol{b}(\boldsymbol{v}) + \boldsymbol{G}(\boldsymbol{u}) - \boldsymbol{G}(\boldsymbol{v}), \boldsymbol{u} - \boldsymbol{v} \right\rangle.
\end{aligned}$$

By the splittings of $A(\boldsymbol{u})$ and of $A(\boldsymbol{v})$, the right-hand side of the previous equation becomes

$$\begin{aligned}
&\left\langle A_1(\boldsymbol{u})(\boldsymbol{u} - \boldsymbol{v}), \boldsymbol{u} - \boldsymbol{v} \right\rangle + \left\langle \left(\tilde{D} + \frac{I}{\tau} \right) (\boldsymbol{u} - \boldsymbol{v}), \boldsymbol{u} - \boldsymbol{v} \right\rangle + \left\langle (A_1(\boldsymbol{u}) - A_1(\boldsymbol{v}))\boldsymbol{v} - \boldsymbol{b}_1(\boldsymbol{u}) + \boldsymbol{b}_1(\boldsymbol{v}), \boldsymbol{u} - \boldsymbol{v} \right\rangle + \\
&+ \left\langle \tilde{A}(\boldsymbol{u})(\boldsymbol{u} - \boldsymbol{v}), \boldsymbol{u} - \boldsymbol{v} \right\rangle + \left\langle (\tilde{A}(\boldsymbol{u}) - \tilde{A}(\boldsymbol{v}))\boldsymbol{v} - \tilde{\boldsymbol{b}}(\boldsymbol{u}) + \tilde{\boldsymbol{b}}(\boldsymbol{v}), \boldsymbol{u} - \boldsymbol{v} \right\rangle + \left\langle \boldsymbol{G}(\boldsymbol{u}) - \boldsymbol{G}(\boldsymbol{v}), \boldsymbol{u} - \boldsymbol{v} \right\rangle.
\end{aligned}$$

We can make evaluations analogous to those in Theorem 1 for most of these terms, obtaining:

- $\langle A_1(\boldsymbol{u}) \cdot (\boldsymbol{u} - \boldsymbol{v}), \boldsymbol{u} - \boldsymbol{v} \rangle \geq h^2\sigma_{\min} \left[\sum_{j=1}^{M} \sum_{i=1}^{N+1} (\nabla_x(u_{i,j} - v_{i,j}))^2 + \sum_{i=1}^{N} \sum_{j=1}^{M+1} (\nabla_y(u_{i,j} - v_{i,j}))^2 \right]$;
- $\left\langle \left(\tilde{D} + \frac{I}{\tau} \right) (\boldsymbol{u} - \boldsymbol{v}), \boldsymbol{u} - \boldsymbol{v} \right\rangle \geq \left(\alpha_{\min} + \frac{1}{\tau} \right) \|\boldsymbol{u} - \boldsymbol{v}\|_h^2$;
- $\left\langle \boldsymbol{G}(\boldsymbol{u}) - \boldsymbol{G}(\boldsymbol{v}), \boldsymbol{u} - \boldsymbol{v} \right\rangle \geq c\|\boldsymbol{u} - \boldsymbol{v}\|_h^2$.

Proceeding as in Lemma 2 and by Lemma 3, we then have, respectively

$$\left\langle \tilde{A}(\boldsymbol{u})(\boldsymbol{u} - \boldsymbol{v}), \boldsymbol{u} - \boldsymbol{v} \right\rangle \geq -\frac{\tilde{v}_{\max}}{h} \|\boldsymbol{u} - \boldsymbol{v}\|_h^2 - \frac{h^3\tilde{v}_{\max}}{2} \left\{ \sum_{j=1}^{M} \sum_{i=1}^{N+1} |\nabla_x(u_{i,j} - v_{i,j})|^2 + \sum_{i=1}^{N} \sum_{j=1}^{M+1} |\nabla_y(u_{i,j} - v_{i,j})|^2 \right\};$$

$$\left\langle (\tilde{A}(\boldsymbol{u}) - \tilde{A}(\boldsymbol{v}))\boldsymbol{v} - \tilde{\boldsymbol{b}}(\boldsymbol{u}) + \tilde{\boldsymbol{b}}(\boldsymbol{v}), \boldsymbol{u} - \boldsymbol{v} \right\rangle \geq -\beta\Lambda_{\tilde{v}} \left[\|\boldsymbol{u} - \boldsymbol{v}\|^2 + \|\boldsymbol{u} - \boldsymbol{v}\|^2 \right] = -2\beta\Lambda_{\tilde{v}} \|\boldsymbol{u} - \boldsymbol{v}\|^2.$$

Lastly, proceeding as in [11, Theorem 1], we obtain

$$\begin{aligned}
\left\langle \big(A_1(\boldsymbol{u}) - A_1(\boldsymbol{v})\big) \cdot \boldsymbol{v} - \boldsymbol{b}(\boldsymbol{u}) + \boldsymbol{b}(\boldsymbol{v}), \boldsymbol{u} - \boldsymbol{v} \right\rangle \geq &- \frac{h^2\beta\Lambda\phi}{2} \left\{ \sum_{j=1}^{M} \sum_{i=1}^{N+1} \left| \nabla_x(u_{i,j} - v_{i,j}) \right|^2 + \sum_{i=1}^{N} \sum_{j=1}^{M+1} \left| \nabla_y(u_{i,j} - v_{i,j}) \right|^2 \right\} - \\
&- \frac{\beta\Lambda}{\phi} \|\boldsymbol{u} - \boldsymbol{w}\|_h^2,
\end{aligned}$$

where ϕ is an arbitrary positive parameter.

Thus, considering all previous inequalities and collecting terms, we can write

$$\begin{aligned}
\frac{1}{\tau} \langle \boldsymbol{F}(\boldsymbol{u}) - \boldsymbol{F}(\boldsymbol{v}), \boldsymbol{u} - \boldsymbol{v} \rangle \geq &\left(h^2\sigma_{\min} - \frac{h^2\beta\Lambda\phi}{2} - \frac{h^3\tilde{v}_{\max}}{2} \right) \left[\sum_{j=1}^{M} \sum_{i=1}^{N+1} (\nabla_x(u_{i,j} - v_{i,j}))^2 + \sum_{i=1}^{N} \sum_{j=1}^{M+1} (\nabla_y(u_{i,j} - v_{i,j}))^2 \right] \\
&+ \left(\alpha_{\min} + \frac{1}{\tau} - \frac{\beta\Lambda}{\phi} - \frac{\tilde{v}_{\max}}{h} - 2\beta\Lambda_{\tilde{v}} + c \right) \|\boldsymbol{u} - \boldsymbol{v}\|_h^2.
\end{aligned}$$

Since ϕ is an arbitrary, positive parameter, we now choose it in a suitable way. In particular, we choose

$$\phi = \frac{1}{\beta\Lambda}\left(2\sigma_{\min} - h\tilde{v}_{\max}\right),$$

which implies

$$h^2\sigma_{\min} - \frac{h^2\beta\Lambda\phi}{2} - \frac{h^3\tilde{v}_{\max}}{2} = 0.$$

We here exploit that ϕ is positive by the hypothesis $h < 2\sigma_{\min}/\tilde{v}_{\max}$, which implies $2\sigma_{\min} - h\tilde{v}_{\max} > 0$. With this choice of ϕ, we get

$$\frac{1}{\tau}\left\langle \boldsymbol{F}(\boldsymbol{u}) - \boldsymbol{F}(\boldsymbol{v}), \boldsymbol{u} - \boldsymbol{v}\right\rangle \geq \left(\alpha_{\min} + \frac{1}{\tau} - \frac{\beta^2\Lambda^2}{2\sigma_{\min} - h\tilde{v}_{\max}} - \frac{\tilde{v}_{\max}}{h} - 2\beta\Lambda_{\tilde{v}} + c\right)\|\boldsymbol{u} - \boldsymbol{v}\|_h^2.$$

Thus, we have obtained an equation in the form of (15) and monotonicity holds if

$$\gamma = \left(\alpha_{\min} + \frac{1}{\tau} - \frac{\beta^2\Lambda^2}{2\sigma_{\min} - h\tilde{v}_{\max}} - \frac{\tilde{v}_{\max}}{h} - 2\beta\Lambda_{\tilde{v}} + c\right)$$

is larger than zero. This happens when

$$\alpha_{\min} + \frac{1}{\tau} + c > \frac{\beta^2\Lambda^2}{2\sigma_{\min} - h\tilde{v}_{\max}} + \frac{\tilde{v}_{\max}}{h} + 2\beta\Lambda_{\tilde{v}},$$

proving the first part of the theorem.

The uniqueness of the solution follows then directly. Indeed, suppose that two distinct solutions $\boldsymbol{u}^*$ and $\hat{\boldsymbol{u}}$ to the system $\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{0}$ exist in $\mathcal{B}_{\rho,\beta}$. We would have

$$\langle \boldsymbol{F}(\boldsymbol{u}^*) - \boldsymbol{F}(\hat{\boldsymbol{u}}), \boldsymbol{u}^* - \hat{\boldsymbol{u}}\rangle \geq \gamma\|\boldsymbol{u}^* - \hat{\boldsymbol{u}}\|_h^2 > 0 \text{ for } \boldsymbol{u}^* \neq \hat{\boldsymbol{u}},$$

contradicting $\boldsymbol{F}(\boldsymbol{u}^*) = \boldsymbol{F}(\hat{\boldsymbol{u}}) = \boldsymbol{0}$. □

As a final remark, we also notice that the monotonicity conditions mirror those in Theorem 1. In particular, if the monotonicity conditions are satisfied, the conditions of Theorem 1 are satisfied as well. Then, Theorem 2 can also be seen as a sufficient condition for the existence and the uniqueness of the solution.

4 The lagged diffusivity method and convergence analysis

4.1 The lagged diffusivity method

The basic idea of the method consists in linearizing (8) by setting up an iterative procedure where diffusivity and velocity are lagged. In order to do this, chosen a starting vector $\boldsymbol{u}^{(0)}$, at the $(\nu+1)$-th iteration we consider the diffusivity and the velocity terms dependent on the solution $\boldsymbol{u}$ at the ν-th iteration, $\boldsymbol{u}^{(\nu)}$. This means that, at each lagged iteration, instead of having the nonlinear terms $A(\boldsymbol{u})$ and $\boldsymbol{b}(\boldsymbol{u})$, we have $A(\boldsymbol{u}^{(\nu)})$ and $\boldsymbol{b}(\boldsymbol{u}^{(\nu)})$. A weak nonlinearity is, however, still present due to the nonlinear mapping $\boldsymbol{G}(\boldsymbol{u})$.

Hence, we compute the new iterate $\boldsymbol{u}^{(\nu+1)}$ as the solution of the weakly nonlinear system

$$\boldsymbol{F}_\nu(\boldsymbol{u}) = \left[I + \tau A(\boldsymbol{u}^{(\nu)})\right]\boldsymbol{u} - \tau\left[\boldsymbol{b}(\boldsymbol{u}^{(\nu)}) - \boldsymbol{G}(\boldsymbol{u})\right] - \boldsymbol{w} = \boldsymbol{0}. \tag{16}$$

Here, $I + \tau A(\boldsymbol{u})$ is certainly nonsingular $\forall \boldsymbol{u} \in \mathbb{R}^\mu$ if condition (4) is satisfied, since it implies that $A(\boldsymbol{u})$ is a nonsingular M-matrix.

System (16) can be solved approximately by an iterative method. The lagged iterate is accepted when the residual

$$\boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu+1)}) = \left[I + \tau A(\boldsymbol{u}^{(\nu)})\right]\boldsymbol{u}^{(\nu+1)} - \tau\left[\boldsymbol{b}(\boldsymbol{u}^{(\nu)}) - \boldsymbol{G}(\boldsymbol{u}^{(\nu+1)})\right] - \boldsymbol{w} \tag{17}$$

satisfies a stopping criterion

$$\left\| \boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu+1)}) \right\| \leq \epsilon_{\nu+1}, \tag{18}$$

where $\|\cdot\|$ denotes the Euclidean norm and ϵ_ν is a given tolerance such that $\epsilon_\nu \to 0$ for $\nu \to \infty$. In the following, we refer to (18) as *lagged acceptability condition*. When it is satisfied, we find $\boldsymbol{u}^{(\nu+1)}$ and the outer iteration can be restarted: the computed $\boldsymbol{u}^{(\nu+1)}$ is used to evaluate diffusivity and velocity to find $\boldsymbol{u}^{(\nu+2)}$ by the new lagged iteration, and so on.

In the following, we choose the solution at the previous time level as starting vector of the lagged diffusivity method, which is we set $\boldsymbol{u}^{(0)} = \boldsymbol{u}^n$ to initialize the LDM iteration at the $(n+1)$-th time level. At the first time level, we instead use the initial condition. Tolerances are then defined starting from the norm of the initial residual, $\|\boldsymbol{F}(\boldsymbol{u}^{(0)})\|$. Indeed, we define ϵ_1 by multiplying the initial residual by a positive constant smaller than 1, e.g.

$$\epsilon_1 = 0.1\|\boldsymbol{F}(\boldsymbol{u}^{(0)})\|,$$

and we build a sequence of $\epsilon_{\nu+1}$, $\nu = 1, 2, \ldots$ such that $\epsilon_\nu \to 0$ for $\nu \to \infty$ simply by setting

$$\epsilon_{\nu+1} = \frac{\epsilon_\nu}{2}, \quad \nu = 1, 2, \ldots.$$

The lagged diffusivity procedure is then stopped when

$$\epsilon_{\nu+1} \leq \bar{\epsilon} \tag{19}$$

is satisfied, where $\bar{\epsilon}$ is a given tolerance. As a consequence of computing $\epsilon_{\nu+1}$ by halving ϵ_1 ν times, condition (19) is satisfied after

$$\nu^* = \left\lceil \log_2 \left(\frac{\epsilon_1}{\bar{\epsilon}} \right) \right\rceil$$

iterations, where $\lceil \cdot \rceil$ denotes the ceiling function.

4.2 Convergence analysis

We now analyze the convergence of the algorithm presented in the previous subsection. In this regard, let us first better characterize the solutions of the weakly nonlinear systems arising at each lagged iteration. In particular, if we solve (16) inexactly, the approximate solution $\boldsymbol{u}^{(\nu+1)}$ solves

$$\boldsymbol{F}_\nu(\boldsymbol{u}) = \boldsymbol{r}^{(\nu+1)} \qquad \text{with } \|\boldsymbol{r}^{(\nu+1)}\| \leq \epsilon_{\nu+1},$$

where $\boldsymbol{r}^{(\nu+1)} := \boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu+1)})$ is the residual at the $(\nu+1)$-th iteration.

Assuming that the hypotheses of Theorem 1 are satisfied, let us analyze $\langle \boldsymbol{F}_\nu(\boldsymbol{u}) - \boldsymbol{r}^{(\nu+1)}, \boldsymbol{u} \rangle$. We have

$$\langle \boldsymbol{F}_\nu(\boldsymbol{u}) - \boldsymbol{r}^{(\nu+1)}, \boldsymbol{u} \rangle = \langle \boldsymbol{u}, \boldsymbol{u} \rangle + \tau \langle A(\boldsymbol{u}^{(\nu)})\boldsymbol{u} - \boldsymbol{b}(\boldsymbol{u}^{(\nu)}), \boldsymbol{u} \rangle + \tau \langle \boldsymbol{G}(\boldsymbol{u}) - \boldsymbol{G}(\boldsymbol{0}), \boldsymbol{u} \rangle + \langle \tau \boldsymbol{G}(\boldsymbol{0}) - \boldsymbol{w} - \boldsymbol{r}^{(\nu+1)}, \boldsymbol{u} \rangle,$$

hence

$$\langle \boldsymbol{F}_\nu(\boldsymbol{u}) - \boldsymbol{r}^{(\nu+1)}, \boldsymbol{u} \rangle \geq \left(\|\boldsymbol{u}\|_h - \frac{\tau \tilde{v}_{\max}}{h}\|\boldsymbol{u}\|_h + \tau \alpha_{\min}\|\boldsymbol{u}\|_h + c\|\boldsymbol{u}\|_h - \tau\|\boldsymbol{G}(\boldsymbol{0})\|_h - \|\boldsymbol{w}\|_h - \|\boldsymbol{r}^{(\nu+1)}\|_h \right) \|\boldsymbol{u}\|_h,$$

where $\|\boldsymbol{r}^{(\nu+1)}\|_h = h\|\boldsymbol{r}^{(\nu+1)}\| \leq h\epsilon_{\nu+1}$. Then, if we define

$$\rho_{\nu+1} := \frac{\tau\|\boldsymbol{G}(\boldsymbol{0})\|_h + \|\boldsymbol{w}\|_h + h\epsilon_{\nu+1}}{1 - \frac{\tau \tilde{v}_{\max}}{h} + \tau \alpha_{\min} + \tau c},$$

we get $\langle \boldsymbol{F}_\nu(\boldsymbol{u}) - \boldsymbol{r}^{(\nu+1)}, \boldsymbol{u} \rangle > 0$ when $\|\boldsymbol{u}\|_h > \rho_{\nu+1}$. Therefore, for all $\nu = 0, 1, \ldots$, the solution $\boldsymbol{u}^{(\nu+1)}$ of $\boldsymbol{F}_\nu(\boldsymbol{u}) - \boldsymbol{r}^{(\nu+1)} = \boldsymbol{0}$ belongs to $\{\boldsymbol{u} | \|\boldsymbol{u}\|_h \leq \rho_{\nu+1}\}$. It is also interesting to notice that $\rho_{\nu+1}$ can be expressed as

$$\rho_{\nu+1} = \frac{\tau\|\boldsymbol{G}(\boldsymbol{0})\|_h + \|\boldsymbol{w}\|_h}{1 - \frac{\tau \tilde{v}_{\max}}{h} + \tau \alpha_{\min} + \tau c} + \frac{h\epsilon_{\nu+1}}{1 - \frac{\tau \tilde{v}_{\max}}{h} + \tau \alpha_{\min} + \tau c} = \rho + \epsilon_{\nu+1}\rho_0,$$

with $\rho_0 := \frac{h}{1-\frac{\tau\tilde{v}_{\max}}{h}+\tau\alpha_{\min}+\tau c}$. This will be useful in the analysis of convergence and it implies that $\rho \leq \rho_{\nu+1}$ for all $\nu = 0, 1, \ldots$.

Proceeding similarly with regard to the bound on backward difference quotients, it is possible to write $\beta_{\nu+1} = \beta + \epsilon_{\nu+1}\beta_0$. We then have $\boldsymbol{u}^{(\nu+1)} \in \mathcal{B}_{\rho_{\nu+1},\beta_{\nu+1}}$ for $\nu = 0, 1, \ldots$.

Theorem 3. *Let $\boldsymbol{u}^* \in \mathcal{B}_{\rho,\beta}$ be the solution of the nonlinear system $\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{0}$ defined in (8) with $A(\boldsymbol{u})$ non-singular and $\boldsymbol{G}(\boldsymbol{u})$ diagonal mapping. We assume that the smoothness conditions and the hypotheses of Theorem 2 are satisfied.*
Starting from an arbitrary $\boldsymbol{u}^{(0)}$, let $\boldsymbol{u}^{(\nu+1)}$ be the solution of system (16) with residual $\boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu+1)})$ satisfying (18), with $\epsilon_\nu \to 0$ for $\nu \to \infty$.
Then, the sequence $\{\boldsymbol{u}^{(\nu)}\}$ converges to $\boldsymbol{u}^*$.

Proof. Let us consider $\boldsymbol{F}(\boldsymbol{u}^*) = \boldsymbol{0}$ and the lagged acceptability condition (18), satisfied by $\boldsymbol{u}^{(\nu+1)}$:

$$\Big[I + \tau A(\boldsymbol{u}^*)\Big]\boldsymbol{u}^* - \tau\Big[\boldsymbol{b}(\boldsymbol{u}^*) - \boldsymbol{G}(\boldsymbol{u}^*)\Big] - \boldsymbol{w} = \boldsymbol{0};$$
$$\Big[I + \tau A(\boldsymbol{u}^{(\nu)})\Big]\boldsymbol{u}^{(\nu+1)} - \tau\Big[\boldsymbol{b}(\boldsymbol{u}^{(\nu)}) - \boldsymbol{G}(\boldsymbol{u}^{(\nu+1)})\Big] - \boldsymbol{w} = \boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu+1)}).$$

Then, let us subtract the second equation from the first one, divide by τ and cancel out constant terms, obtaining

$$\left[\frac{I}{\tau} + A(\boldsymbol{u}^*)\right]\boldsymbol{u}^* - \left[\frac{I}{\tau} + A(\boldsymbol{u}^{(\nu)})\right]\boldsymbol{u}^{(\nu+1)} - \boldsymbol{b}(\boldsymbol{u}^*) + \boldsymbol{b}(\boldsymbol{u}^{(\nu)}) + \boldsymbol{G}(\boldsymbol{u}^*) - \boldsymbol{G}(\boldsymbol{u}^{(\nu+1)}) = -\frac{\boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu+1)})}{\tau}.$$

At this point, we add and subtract the same term $A(\boldsymbol{u}^*)\boldsymbol{u}^{(\nu+1)}$ and rearrange terms, similarly to what done at the beginning of Theorem 2. If we then take the inner product of both sides with $\boldsymbol{u}^* - \boldsymbol{u}^{(\nu+1)}$, we get

$$\left\langle -\frac{1}{\tau}\boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu+1)}, \boldsymbol{u}^* - \boldsymbol{u}^{(\nu+1)}\right\rangle = \left\langle \left[\frac{I}{\tau} + A(\boldsymbol{u}^*)\right]\left(\boldsymbol{u}^* - \boldsymbol{u}^{(\nu+1)}\right) + \Big[A(\boldsymbol{u}^*) - A(\boldsymbol{u}^{(\nu)})\Big]\boldsymbol{u}^{(\nu+1)} - \boldsymbol{b}(\boldsymbol{u}^*) + \boldsymbol{b}(\boldsymbol{u}^{(\nu)}) + \right.$$
$$\left. + \boldsymbol{G}(\boldsymbol{u}^*) - \boldsymbol{G}(\boldsymbol{u}^{(\nu+1)}), \boldsymbol{u}^* - \boldsymbol{u}^{(\nu+1)}\right\rangle.$$

Next, we consider the splittings of $A(\boldsymbol{u})$ and of $A(\boldsymbol{u}^*)$. The right-hand side of the previous equation thus becomes

$$\left\langle A_1(\boldsymbol{u}^*)\left(\boldsymbol{u}^* - \boldsymbol{u}^{(\nu+1)}\right), \boldsymbol{u}^* - \boldsymbol{u}^{(\nu+1)}\right\rangle + \left\langle \tilde{A}(\boldsymbol{u}^*)\left(\boldsymbol{u}^* - \boldsymbol{u}^{(\nu+1)}\right), \boldsymbol{u}^* - \boldsymbol{u}^{(\nu+1)}\right\rangle +$$
$$+ \left\langle \left(\tilde{D} + \frac{I}{\tau}\right)\left(\boldsymbol{u}^* - \boldsymbol{u}^{(\nu+1)}\right), \boldsymbol{u}^* - \boldsymbol{u}^{(\nu+1)}\right\rangle + \left\langle \Big[A_1(\boldsymbol{u}^*) - A_1(\boldsymbol{u}^{(\nu)})\Big]\boldsymbol{u}^{(\nu+1)} - \boldsymbol{b}_1(\boldsymbol{u}^*) + \boldsymbol{b}_1(\boldsymbol{u}^{(\nu)}), \boldsymbol{u}^* - \boldsymbol{u}^{(\nu+1)}\right\rangle +$$
$$+ \left\langle \Big[\tilde{A}(\boldsymbol{u}^*) - \tilde{A}(\boldsymbol{u}^{(\nu)})\Big]\boldsymbol{u}^{(\nu+1)} - \tilde{\boldsymbol{b}}(\boldsymbol{u}^*) + \tilde{\boldsymbol{b}}(\boldsymbol{u}^{(\nu)}), \boldsymbol{u}^* - \boldsymbol{u}^{(\nu+1)}\right\rangle + \left\langle \boldsymbol{G}(\boldsymbol{u}^*) - \boldsymbol{G}(\boldsymbol{u}^{(\nu+1)}), \boldsymbol{u}^* - \boldsymbol{u}^{(\nu+1)}\right\rangle.$$

We can now use the previously introduced lemmas and theorems to evaluate all these terms. Thus, we find that the previous expression is larger than or equal to

$$\left(h^2\sigma_{\min} - \frac{h^2\beta_{\nu+1}\Lambda\phi}{2} - \frac{h^3\tilde{v}_{\max}}{2}\right)\sum_{i=1}^{N}\sum_{j=1}^{M}\left[\left|\nabla_x\left(u^*_{i,j} - u^{(\nu+1)}_{i,j}\right)\right|^2 + \left|\nabla_y\left(u^*_{i,j} - u^{(\nu+1)}_{i,j}\right)\right|^2\right] +$$
$$+ \left(\alpha_{\min} + \frac{1}{\tau} - \frac{\tilde{v}_{\max}}{h} - \beta_{\nu+1}\Lambda_{\tilde{v}} + c\right)\left\|\boldsymbol{u}^* - \boldsymbol{u}^{(\nu+1)}\right\|_h^2 - \left(\frac{\beta_{\nu+1}\Lambda}{\phi} + \beta_{\nu+1}\Lambda_{\tilde{v}}\right)\left\|\boldsymbol{u}^* - \boldsymbol{u}^{(\nu)}\right\|_h^2,$$

where ϕ is an arbitrary, positive parameter. Choosing $\phi = \frac{1}{\beta_{\nu+1}\Lambda}(2\sigma_{\min} - h\tilde{v}_{\max})$ (which is positive by the hypothesis $h < 2\sigma_{\min}/\tilde{v}_{\max}$), the first term of the previous expression vanishes and we get

$$-\frac{1}{\tau}\left\langle \boldsymbol{F}(\boldsymbol{u}^{(\nu+1)}), \boldsymbol{u}^* - \boldsymbol{u}^{(\nu+1)}\right\rangle \geq \left(\alpha_{\min} + \frac{1}{\tau} - \frac{\tilde{v}_{\max}}{h} - \beta_{\nu+1}\Lambda_{\tilde{v}} + c\right) -$$
$$- \left(\frac{\beta_{\nu+1}^2\Lambda^2}{2\sigma_{\min} - h\tilde{v}_{\max}} + \beta_{\nu+1}\Lambda_{\tilde{v}}\right)\|\boldsymbol{u}^* - \boldsymbol{u}^{(\nu)}\|_h^2.$$

We now rewrite the previous inequality by evaluating from above the term on the left-hand side. Since the iterate $\boldsymbol{u}^{(\nu+1)}$ must satisfy condition (18) and since it must belong to $\mathcal{B}_{\rho_{\nu+1},\beta_{\nu+1}}$ (implying $\left\|\boldsymbol{u}^* - \boldsymbol{u}^{(\nu+1)}\right\| \leq 2\rho_{\nu+1}$ since $\rho \leq \rho_{\nu+1}$), we have

$$\left\langle -\frac{1}{\tau}\boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu+1)}), \boldsymbol{u}^* - \boldsymbol{u}^{(\nu+1)} \right\rangle \leq \frac{1}{\tau}\left\|F_\nu(\boldsymbol{u}^{(\nu+1)})\right\|_h \left\|\boldsymbol{u}^* - \boldsymbol{u}^{(\nu+1)}\right\|_h \leq \frac{1}{\tau}\epsilon_{\nu+1}\left\|\boldsymbol{u}^* - \boldsymbol{u}^{(\nu+1)}\right\|_h \leq \frac{2\rho_{\nu+1}}{\tau}\epsilon_{\nu+1},$$

hence

$$\frac{2\rho_{\nu+1}}{\tau}\epsilon_{\nu+1} \geq \left(\alpha_{\min} + \frac{1}{\tau} - \frac{\tilde{v}_{\max}}{h} - \beta_{\nu+1}\Lambda_{\tilde{v}} + c\right)\|\boldsymbol{u}^* - \boldsymbol{u}^{(\nu+1)}\|_h^2 - \left(\frac{\beta_{\nu+1}^2\Lambda^2}{2\sigma_{\min} - h\tilde{v}_{\max}} + \beta_{\nu+1}\Lambda_{\tilde{v}}\right)\|\boldsymbol{u}^* - \boldsymbol{u}^{(\nu)}\|_h^2.$$

For compactness, let us then collect the coefficient of the first term of the right-hand side of the previous inequality in

$$\zeta = \alpha_{\min} + \frac{1}{\tau} - \frac{\tilde{v}_{\max}}{h} - \beta_{\nu+1}\Lambda_{\tilde{v}} + c.$$

Using the relation $\beta_{\nu+1} = \beta + \epsilon_{\nu+1}\beta_0$ and comparing this equation with the definition of γ in the monotonicity condition, it can be noticed that $\gamma > 0$ (which is satisfied by hypothesis) implies that also ζ is positive if

$$\beta_0 \leq \frac{1}{\epsilon_{\nu+1}}\left(\beta + \frac{1}{\Lambda_{\tilde{v}}}\frac{\beta^2\Lambda^2}{2\sigma_{\min} - h\tilde{v}_{\max}}\right) \tag{20}$$

holds true. Assumption (20) is nonetheless plausible since the tolerance $\epsilon_\nu \to 0$ for $\nu \to \infty$. Therefore, inequality (20) is certainly satisfied from a certain lagging iteration. Moreover, operatively we could grant that (20) is satisfied also at the first lagging iterations by setting ϵ_0 sufficiently small.
Thus assuming $\zeta > 0$, we can divide both sides by ζ without changing sign to the inequality. Rearranging terms, we get

$$\frac{2\rho}{\tau\zeta}\epsilon_{\nu+1} + \frac{1}{\zeta}\left(\frac{\beta_{\nu+1}^2\Lambda^2}{2\sigma_{\min} - h\tilde{v}_{\max}} - \beta_{\nu+1}\Lambda_{\tilde{v}}\right)\left\|\boldsymbol{u}^* + \boldsymbol{u}^{(\nu)}\right\|_h^2 \geq \left\|\boldsymbol{u}^* - \boldsymbol{u}^{(\nu+1)}\right\|_h^2. \tag{21}$$

We can then make further considerations on ζ. Indeed, again remembering assumption (20) and $\gamma > 0$, we have

$$\zeta = \alpha_{\min} + \frac{1}{\tau} - \frac{\tilde{v}_{\max}}{h} - (\beta + \epsilon_{\nu+1}\beta_0)\Lambda_{\tilde{v}} + c > \frac{\beta^2\Lambda^2}{2\sigma_{\min} - h\tilde{v}_{\max}} - \beta\Lambda_{\tilde{v}}. \tag{22}$$

Thus

$$\frac{1}{\zeta}\left(\frac{\beta_{\nu+1}^2\Lambda^2}{2\sigma_{\min} - h\tilde{v}_{\max}} - \beta_{\nu+1}\Lambda_{\tilde{v}}\right) < \left(\frac{\beta^2\Lambda^2}{2\sigma_{\min} - h\tilde{v}_{\max}} - \beta\Lambda_{\tilde{v}}\right)^{-1}\left(\frac{\beta_{\nu+1}^2\Lambda^2}{2\sigma_{\min} - h\tilde{v}_{\max}} - \beta_{\nu+1}\Lambda_{\tilde{v}}\right).$$

Since $\epsilon_\nu \to 0$ for $\nu \to \infty$, we also have

$$\lim_{\nu\to\infty}\frac{\beta_{\nu+1}^2\Lambda^2}{2\sigma_{\min} - h\tilde{v}_{\max}} - \beta_{\nu+1}\Lambda_{\tilde{v}} = \lim_{\nu\to\infty}\frac{(\beta + \epsilon_{\nu+1}\beta_0)^2\Lambda^2}{2\sigma_{\min} - h\tilde{v}_{\max}} - (\beta + \epsilon_{\nu+1}\beta_0)\Lambda_{\tilde{v}} = \frac{\beta^2\Lambda^2}{2\sigma_{\min} - h\tilde{v}_{\max}} - \beta\Lambda_{\tilde{v}}.$$

So, since the inequality in (22) holds in strict sense, we can assume that there exists an integer ν_0 such that

$$\frac{1}{\zeta}\left(\frac{\beta_{\nu+1}^2\Lambda^2}{2\sigma_{\min} - h\tilde{v}_{\max}} - \beta_{\nu+1}\Lambda_{\tilde{v}}\right) < 1 \qquad \forall\nu > \nu_0.$$

Thus, defining

$$\hat{\gamma} = \frac{1}{\zeta}\left(\frac{\beta_{\nu+1}^2\Lambda^2}{2\sigma_{\min} - h\tilde{v}_{\max}} - \beta_{\nu+1}\Lambda_{\tilde{v}}\right) \quad \text{and} \quad \hat{\zeta} = \frac{2\rho}{\tau\zeta},$$

proceeding iteratively from a ν_0 such that $\hat{\gamma} < 1$, by (21) we get

$$\left\|\boldsymbol{u}^* - \boldsymbol{u}^{(\nu+1)}\right\|_h^2 \leq \hat{\gamma}^r \left\|\boldsymbol{u}^* - \boldsymbol{u}^{(\nu)}\right\|_h^2 + \hat{\zeta} \sum_{j=1}^{r} \hat{\gamma}^{r-j} \epsilon_{\nu_0+j}, \quad r = 1, 2, \ldots. \tag{23}$$

Finally, since $\epsilon_\nu \to 0$ for $\nu \to \infty$, by the Toeplitz lemma [12, p. 399] we have

$$\lim_{\nu\to\infty} \left\|\boldsymbol{u}^* - \boldsymbol{u}^{(\nu)}\right\|_h^2 = 0.$$

Then, the sequence $\{\boldsymbol{u}^{(\nu)}\}$ converges to the solution $\boldsymbol{u}^*$ of the system $\boldsymbol{F}(\boldsymbol{u}) = \boldsymbol{0}$. □

Thus, the previous theorem proves the convergence of the method when it is applied to solving the nonlinear systems arising from a finite-difference discretization of the differential problem (1). It is nonetheless important to highlight that the lagged diffusivity method is applied to the discretized system, irrespectively of the used discretization. Thus, it can be applied, in principle, also when other discretization techniques (i.e. finite element or finite volume discretizations) are used. However, it is important to remark that the presented proofs of convergence rely on monotonicity properties which FD operators satisfy.

4.3 Remarks on the stationary case

In case we are studying a steady reaction-convection-diffusion equation, discretization has to be performed only along space. Using the finite difference schemes employed in the non-steady case, space discretization leads to an expression similar to (5). The only difference is that we do not have time dependence. Therefore, after space discretization, we do not have a system of ordinary differential equations like (6), but we get directly the nonlinear algebraic system

$$\boldsymbol{F}(\boldsymbol{u}) = A(\boldsymbol{u})\boldsymbol{u} - \boldsymbol{b}(\boldsymbol{u}) + \boldsymbol{G}(\boldsymbol{u}) - \boldsymbol{s} = \boldsymbol{0}. \tag{24}$$

We can then introduce the lagging iteration: with $\boldsymbol{F}(\boldsymbol{u})$ as in (24), the new iterate $\boldsymbol{u}^{(\nu+1)}$ of the lagged iteration is given by the solution of the weakly nonlinear system

$$\boldsymbol{F}_\nu(\boldsymbol{u}) = A(\boldsymbol{u}^{(\nu)})\boldsymbol{u} - \boldsymbol{b}(\boldsymbol{u}^{(\nu)}) + \boldsymbol{G}(\boldsymbol{u}) - \boldsymbol{s} = \boldsymbol{0}. \tag{25}$$

If $A(\boldsymbol{u})$ is nonsingular, system (25) can be solved approximately by a convergent iterative method and the lagged iterate is accepted when the residual

$$\boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu+1)}) = A(\boldsymbol{u}^{(\nu)})\boldsymbol{u}^{(\nu+1)} - \boldsymbol{b}(\boldsymbol{u}^{(\nu)}) + \boldsymbol{G}(u^{(\nu+1)}) - \boldsymbol{s} \tag{26}$$

satisfies a stopping criterion

$$\left\|\boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu+1)})\right\| \leq \epsilon_{\nu+1}, \tag{27}$$

where ϵ_ν is a given tolerance such that $\epsilon_\nu \to 0$ for $\nu \to \infty$.

Conditions of monotonicity and convergence in stationary case can be found proceeding as done in Theorems 2 and 3.

5 Solution process

In this section we analyze more in detail the solution process. In this regard, we describe inner and outer solvers and how to set their initialization and stopping criteria efficiently.

As described in Section 4.1, at each LDM iteration we must solve the weakly nonlinear system (16), which we do approximately by an iterative procedure that is stopped when the lagged acceptability condition (18) is satisfied. We choose to solve (16) by the simplified Newton's method [12, p. 182]. We are thus left with three solution levels at each time step:

1. we use the lagged diffusivity method to linearize the nonlinear algebraic systems (8);
2. we use the simplified Newton's method to solve the weakly nonlinear algebraic system (16) arising at each lagged iteration. The method is stopped when the lagged acceptability condition (18) is satisfied;
3. we use an iterative linear solver to solve the linear system arising at each simplified Newton's iteration. In the following, the tolerance of the linear solver is chosen so that (16) is solved by an inexact Newton's method [4, 5].

Having already described the lagged diffusivity method in Section 4.1, let us proceed to the analysis of the solution of the weakly nonlinear system (16).

5.1 Solution of the weakly nonlinear system

Let us introduce a second superscript k to account for the Newton iteration. Thus, $\boldsymbol{u}^{(\nu+1,k+1)}$ denotes $\boldsymbol{u}$ at the $(k+1)$-th simplified Newton iteration of the $(\nu+1)$-th lagged iteration. Choosing as starting vector the solution at the previous lagged iteration, $\boldsymbol{u}^{(\nu)}$, the $(k+1)$-th simplified Newton iteration, $k = 0, 1, \ldots$, of the $(\nu+1)$-th lagged iteration consists in finding $\Delta\boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(\nu+1,k+1)} - \boldsymbol{u}^{(\nu+1,k)}$ solution of the linear system

$$F'_\nu(\boldsymbol{u}^{(\nu)})\Delta\boldsymbol{u} = -\boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu+1,k)}), \tag{28}$$

with

$$F'_\nu(\boldsymbol{u}^{(\nu)}) = I + \tau A(\boldsymbol{u}^{(\nu)}) + \tau G'(\boldsymbol{u}^{(\nu)}) \tag{29}$$

Jacobian matrix of $\boldsymbol{F}_\nu(\boldsymbol{u})$ evaluated at $\boldsymbol{u}^{(\nu+1,0)} = \boldsymbol{u}^{(\nu)}$ and $G'(\boldsymbol{u}^{(\nu)})$ Jacobian matrix of $\boldsymbol{G}(\boldsymbol{u})$. Remembering that the criterion of acceptability of the lagged iteration is given by (18) with $\boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu+1)})$ as in (17), the simplified Newton iteration is stopped when $\|\boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu+1,k+1)})\| \leq \epsilon_{\nu+1}$ is satisfied. Denoting by k_{end} the number of Newton iterations needed for satisfying the acceptability criterion (18), we define $\boldsymbol{u}^{(\nu+1)} \equiv \boldsymbol{u}^{(\nu+1,k_{end})}$.

More conveniently, we can rewrite (28) by using the definition of $\Delta\boldsymbol{u}^{(k+1)}$ at lagging iteration $\nu+1$ and of $F'_\nu(\boldsymbol{u}^{(\nu)})$ in (29). Rearranging terms, we indeed find

$$F'_\nu(\boldsymbol{u}^{(\nu)})\boldsymbol{u}^{(\nu+1,k+1)} = \left[I + \tau A(\boldsymbol{u}^{(\nu)})\right]\boldsymbol{u}^{(\nu+1,k)} + \tau G'(\boldsymbol{u}^{(\nu)})\boldsymbol{u}^{(\nu+1,k)} - \boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu+1,k)}).$$

Then, replacing $\boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu+1,k)})$ by its definition in (16) and canceling out opposite terms, we find that at the $(k+1)$-th simplified Newton iteration, $k = 0, 1, \ldots$, $\boldsymbol{u}^{(\nu+1,k+1)}$ is the solution of the linear system

$$F'_\nu(\boldsymbol{u}^{(\nu)})\boldsymbol{u} = \tau G'(\boldsymbol{u}^{(\nu)})\boldsymbol{u}^{(\nu+1,k)} - \tau\boldsymbol{G}(\boldsymbol{u}^{(\nu+1,k)}) + \tau\boldsymbol{b}(\boldsymbol{u}^{(\nu)}) + \boldsymbol{w}, \quad k = 0, 1, \ldots. \tag{30}$$

Thus, now we have to solve a linear system at each Newton iteration. This can be done approximately by an iterative method. This helps increasing the computational efficiency, especially in case of systems of large dimensions. Let us then introduce a third superscript j. At the $(j+1)$-th iteration of the linear solver, $j = 0, 1, \ldots$, we have a residual $\boldsymbol{r}_{j+1}$ given by

$$\boldsymbol{r}_{j+1} = F'_\nu(\boldsymbol{u}^{(\nu)})\boldsymbol{u}^{(j+1)} - \tau G'(\boldsymbol{u}^{(\nu)})\boldsymbol{u}^{(\nu+1,k)} + \tau\boldsymbol{G}(\boldsymbol{u}^{(\nu+1,k)}) - \tau\boldsymbol{b}(\boldsymbol{u}^{(\nu)}) - \boldsymbol{w}, \tag{31}$$

where $\boldsymbol{u}^{(j+1)} \equiv \boldsymbol{u}^{(\nu+1,k+1,j+1)}$ denotes the solution of (30) at the $(j+1)$-th linear iteration of the $(k+1)$-th Newton iteration of the $(\nu+1)$-th lagged iteration. In the following, the starting vector of the linear iterative solver is given by the solution of the previous simplified Newton iteration, i.e. $\boldsymbol{u}^{(\nu+1,k+1,0)} = \boldsymbol{u}^{(\nu+1,k)}$.

Choosing a suitable stopping condition of the linear solver, we can set up an inexact Newton procedure. In this regard, we choose a forcing term $\hat{\sigma}$, $0 < \hat{\sigma} < 1$, and define the tolerance $\hat{\epsilon}^{(k+1)}$ of the linear solver at the $(k+1)$-th Newton iteration as

$$\hat{\epsilon}^{(k+1)} = \hat{\sigma}\|\boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu+1,k)})\|. \tag{32}$$

We then stop the linear solver when the norm of $\boldsymbol{r}_{j+1}$, $j = 0, 1, \ldots$, satisfies

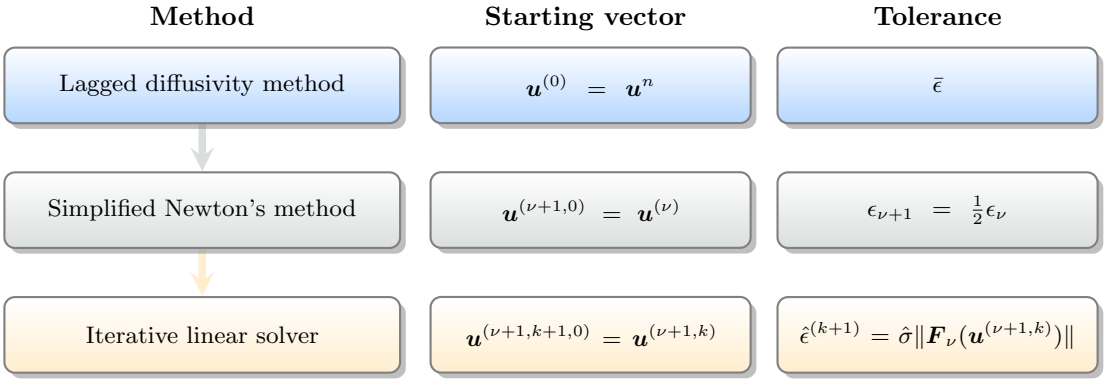
$$\|\boldsymbol{r}_{j+1}\| \leq \hat{\epsilon}^{(k+1)}. \tag{33}$$

Denoting by j_{end} the number of iterations needed to the linear solver for satisfying the stopping condition (33), we define $\boldsymbol{u}^{(\nu+1,k+1)} \equiv \boldsymbol{u}^{(\nu+1,k+1,j_{end})}$.

Finally, we provide a few operative remarks on the choice of the arbitrary parameters of the procedure. The choice of the prescribed tolerance $\bar{\epsilon}$ is connected to the desired accuracy, as we can observe that the value of the final residual is ultimately related to it. Regarding ϵ_0 and $\hat{\sigma}$, values excessively small or large may reduce efficiency, but their choice is, nonetheless, not problematic. In all our experiments, we fixed $\epsilon_0 = \hat{\sigma} = 0.1$.

5.2 A correction of the initialization of starting vectors

Starting vectors and stopping criteria of the used iterative procedures can be summarized as follows:



The starting vectors determine also the initialization of tolerances. For instance, at the first Newton iteration of the $(\nu+1)$-th lagged iteration, the tolerance of the linear solver is $\hat{\epsilon}^{(1)} = \hat{\sigma}\|\boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu+1,0)})\| = \hat{\sigma}\|\boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu)})\|$. We can also easily notice that all tolerances are initialized at $\hat{\sigma}\|\boldsymbol{F}_0(\boldsymbol{u}^{(0)})\| = \hat{\sigma}\|\boldsymbol{F}(\boldsymbol{u}^{(0)})\| = \epsilon_1$ at the first lagged iteration.

If we do not apply any correction, however, it can happen that the initial tolerance of the linear solver, $\hat{\epsilon}^{(1)}$, become smaller than $\hat{\sigma}\epsilon_{\nu+1}$ [11, Sec. 5.2]. Being $\epsilon_{\nu+1}$ the tolerance of the simplified Newton's method, $\hat{\epsilon}^{(1)}$ would thus be smaller than what could possibly be required by an inexact Newton's method, leading to an increase in computational cost with no foreseeable improvement to accuracy.

Then, we apply a correction on the initialization of the tolerance of the linear solver. In this way, we avoid solving too exactly the linear system and we reduce computational cost and numerical difficulties. This can be easily done by imposing $\hat{\sigma}\epsilon_{\nu+1}$ as the minimum tolerance of the linear solver. Following [11], we thus set

$$\hat{\epsilon}^{(1)} = \max\left(\hat{\sigma}\|\boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu)})\|, \hat{\sigma}\epsilon_{\nu+1}\right). \tag{34}$$

6 Numerical experiments

In this section, we solve various problems by a Fortran implementation of the LDM. We are concerned with how the method behaves with a nonlinear velocity term and we consider several possible choices of $\tilde{\boldsymbol{v}}$:

$$\tilde{\boldsymbol{v}}_1 = \begin{bmatrix} c_1 e^{-u^2} \\ c_2 u \end{bmatrix} \quad \tilde{\boldsymbol{v}}_2 = \begin{bmatrix} \sin((1-x+c_1 u)\pi) \\ \cos((1-y+c_2 u)\pi) \end{bmatrix} \quad \tilde{\boldsymbol{v}}_3 = \begin{bmatrix} (1+t)^2 \sin((1-x+c_1 u)\pi) \\ t^3 \cos((1-y+c_2 u)\pi) \end{bmatrix} \quad \tilde{\boldsymbol{v}}_4 = \begin{bmatrix} c_1 xy e^{-\frac{t^2}{u^2+1}} \\ \frac{c_2}{(\ln(tu+2))^2} \end{bmatrix}, \tag{35}$$

with c_1 and c_2 real constants. These choices represent different possible situations: indeed, $\tilde{\boldsymbol{v}}_1$ depends only on u, $\tilde{\boldsymbol{v}}_2$ depends on u, x and y but it is constant in time, while $\tilde{\boldsymbol{v}}_3$ and $\tilde{\boldsymbol{v}}_4$ depend on t as well. All these choices evidently satisfy the initial smoothness assumptions (for u finite).

We start our analysis by verifying the effectiveness of the lagged diffusivity method. In this regard, we choose, for example, the following test problems:

$$\begin{array}{lllll} \text{Problem 1} & u^* = (1+x-y)^3 t & \sigma = 0.4 + 0.5u & g = 100e^{0.5u} \\ \text{Problem 2} & u^* = 2(t+1)[(x-0.5)^2 + (y-0.5)^2] & \sigma = 0.01 + 0.5u^2 & g = 5u\ln(1+u). \end{array}$$

Problem 1 represents a situation where $\sigma_{\min}$ and g are quite large, while in *Problem 2* we have the opposite situation. Also the monotonicity constant of g is smaller in this latter case. Thus, *Problem 2* represents a situation which is potentially more critical for the monotonicity of $\boldsymbol{F}(\boldsymbol{u})$ and for convergence.

Finally, it appears suitable to introduce also a test problem motivated by situations which could occur in real-world applications. In this regard, consider, for instance, all those cases where velocity is produced by an external force F. This is, for example, the case of the Smoluchowski diffusion equation, which describes the flow of ions dissolved in a liquid in presence of an electric field that pulls the ions in a given direction. In this case, the velocity can be written as the quotient between the force of the field and a term ζ (called viscous drag, which accounts for the friction) and may be nonlinear (see, e.g., [1]). Conceptually similar situations may arise also in chemical-mechanical frameworks. For instance, consider all those processes where the material which is diffusing has a larger (or smaller) viscous drag and/or causes reactions that form substances characterized by a higher friction. The velocity produced by the same force, thus, may be nonlinear with concentration. Finally, we can find similar situations also in plastic compounding, which consists in mixing additives to a molten polymer. If we add a plasticizer, the viscosity of the polymer is reduced. Thus, if the polymer is mixed by applying a fixed force F, the velocity of the fluid is larger where the concentration of the plasticizer is higher. The opposite situation may arise, on the other hand, if additives that increase viscosity (e.g. fillers) are used.

Let us then build a test problem which can constitute a simplified model of these situations. Assume, that a specie b is diffusing in a medium a. The starting concentration of b in the domain is zero and, as b diffuses, the medium is mixed, with a velocity produced by an external force F. Supposing that b has a large viscosity, the parameter accounting for the viscous drag, η, will be larger where the concentration of b is higher. For instance, assume that η increases quadratically as the concentration of b increases (in any case, it is easy to modify the problem and consider different dependence laws). Then, calling η_0 the value of η where the concentration of b is zero, the velocity may be expressed as

$$\tilde{v}_1 = \frac{1}{\eta_0 + \kappa u^2} F_1; \qquad \tilde{v}_2 = \frac{1}{\eta_0 + \kappa u^2} F_2,$$

with $\kappa > 0$ and with F_1 and F_2 representing the components of the force along the axes x and y, respectively. Notice that we have here performed some simplifying assumptions, such as that F_1, F_2 are constant in the points of the domain. If this does not hold true (like, for instance, in case of a mechanical mixing, where we would likely have turbulences and we may stir the medium along a circumference) we can nonetheless adapt the formulation by considering F_1 and F_2 as dependent on x, y.

Then, we further assume that a reaction between b and the diffusion medium occurs and that its rate is proportional to the concentration of b itself. Finally, we may consider a linear diffusivity, in accordance with the formulation of common mass diffusivities and so to analyze also a case where only the velocity term is nonlinear.

Fixing a solution (for instance, the same as that of Problem 1) and choosing some parameters, let us then solve the test problem

$$\text{Problem 3} \qquad u^* = (1+x-y)^3 t \qquad \sigma = 1 \qquad g = 2u \qquad \tilde{v}_1 = \frac{c_1}{2+u^2} 4 \qquad \tilde{v}_2 = \frac{c_2}{2+u^2} 4.$$

For all problems, we choose the domain Ω to be the square $[0,1] \times [0,1]$, which we discretize by a uniform grid of $N \times N$ points. We also set the initial time $t_0 = 0$, the final time $t_f = 1$ and $\theta = 0.5$. When not otherwise specified, we set $N = 250$ and $\Delta t = 0.1$.

6.1 Verification of the method and analysis of the linear solvers

Let us solve the test problems considering all the forms of $\tilde{\boldsymbol{v}}$ introduced in (35) with $c_1 = c_2 = 1$, $\alpha = 0$ and different inner linear solvers: the Arithmetic Mean (AM) method [17] with $\omega = 1$, the BiConjugate Gradient-stabilized method with parameter l (BiCGstab(l)) [19] and the GMRES method [18]. The implementation of the GMRES follows [7][p. 45], which uses Givens rotations and re-orthogonalization. We choose to consider these solvers because, being $\tilde{\boldsymbol{v}} \neq \mathbf{0}$, the coefficient matrix of the linear systems arising at each Newton iteration cannot be symmetric, thus preventing the use of a conjugate gradient method. Depending on the problem, the non-symmetry can then be more or less significant, and it is thus interesting to compare the AM method, which is known to be well suited for solving strongly non-symmetric systems [16], and Krylov solvers such as BiCGstab(l) and GMRES. Moreover, since we here aim especially at analyzing the behavior of the algorithm, we do not apply any preconditioner. Preconditioning techniques can, nonetheless, be easily employed if higher efficiency is required.

We stop the lagging iteration when $\epsilon_{\nu+1} \leq \bar{\epsilon} = 10^{-4}$, while starting vectors and stopping criteria of all the iterative procedures are chosen as described in Section 5. We also set a maximum number of Newton iterations $k_{\max} = 500$ for each lagging iteration and a maximum number of linear iterations $j_{\max} = 10,000$ for each Newton iteration.

The results are reported in Table 1 and are referred to the last time level, except for t_{tot}, which denotes the whole time required to compute the solution from t_0 to t_f. By ν_{end}, k_{end} and j_{end} we denote, respectively, the total number of lagged, Newton and linear solver iterations at the last time level. Finally, res_0 denotes the Euclidean norm of the initial residual, res denotes the Euclidean norm of the final residual and err_h and err_2 denote the global error in $l_2(\Omega_h)$ and in relative Euclidean norm respectively.

$\tilde{\boldsymbol{v}}$	Lin. Solver	res_0	res	err_h	err_2	ν_{end}	k_{end}	j_{end}	t_{tot}
$\tilde{\boldsymbol{v}}_1$	AM	91,961	$1.42 \cdot 10^{-5}$	$2.19 \cdot 10^{-4}$	$1.04 \cdot 10^{-4}$	27	27	7,759	400.9
	BiCG(1)	91,961	$1.34 \cdot 10^{-5}$	$2.19 \cdot 10^{-4}$	$1.04 \cdot 10^{-4}$	27	27	1,119	61.6
	BiCG(2)	91,961	$1.36 \cdot 10^{-5}$	$2.19 \cdot 10^{-4}$	$1.04 \cdot 10^{-4}$	27	27	511	58.5
	BiCG(4)	91,961	$1.30 \cdot 10^{-5}$	$2.19 \cdot 10^{-4}$	$1.04 \cdot 10^{-4}$	27	27	250	65.8
	GMRES	91,961	$1.39 \cdot 10^{-4}$	$2.19 \cdot 10^{-4}$	$1.04 \cdot 10^{-4}$	27	27	1,762	1,460
$\tilde{\boldsymbol{v}}_2$	AM	91,810	$1.40 \cdot 10^{-5}$	$1.85 \cdot 10^{-4}$	$8.78 \cdot 10^{-5}$	27	27	7,604	375.0
	BiCG(1)	91,810	$1.37 \cdot 10^{-5}$	$1.85 \cdot 10^{-4}$	$8.78 \cdot 10^{-5}$	27	27	1,207	61.4
	BiCG(2)	91,810	$1.32 \cdot 10^{-5}$	$1.85 \cdot 10^{-4}$	$8.78 \cdot 10^{-5}$	27	27	531	62.0
	BiCG(4)	91,810	$1.29 \cdot 10^{-5}$	$1.85 \cdot 10^{-4}$	$8.78 \cdot 10^{-5}$	27	27	242	65.1
	GMRES	91,810	$1.38 \cdot 10^{-4}$	$1.85 \cdot 10^{-4}$	$8.78 \cdot 10^{-5}$	27	27	1,771	1,447
$\tilde{\boldsymbol{v}}_3$	AM	91,795	$1.60 \cdot 10^{-5}$	$6.29 \cdot 10^{-4}$	$2.98 \cdot 10^{-4}$	27	27	7,854	371.2
	BiCG(1)	91,795	$1.46 \cdot 10^{-5}$	$6.29 \cdot 10^{-4}$	$2.98 \cdot 10^{-4}$	27	27	1,244	68.5
	BiCG(2)	91,795	$1.35 \cdot 10^{-5}$	$6.29 \cdot 10^{-4}$	$2.98 \cdot 10^{-4}$	27	27	585	75.7
	BiCG(4)	91,795	$1.32 \cdot 10^{-5}$	$6.29 \cdot 10^{-4}$	$2.98 \cdot 10^{-4}$	27	27	289	72.8
	GMRES	91,795	$1.51 \cdot 10^{-4}$	$6.29 \cdot 10^{-4}$	$2.98 \cdot 10^{-4}$	27	27	1,893	1,575
$\tilde{\boldsymbol{v}}_4$	AM	91,814	$1.36 \cdot 10^{-5}$	$1.09 \cdot 10^{-4}$	$5.15 \cdot 10^{-5}$	27	27	7,544	364.4
	BiCG(1)	91,814	$1.36 \cdot 10^{-5}$	$1.09 \cdot 10^{-4}$	$5.15 \cdot 10^{-5}$	27	27	1,149	62.7
	BiCG(2)	91,814	$1.29 \cdot 10^{-5}$	$1.09 \cdot 10^{-4}$	$5.15 \cdot 10^{-5}$	27	27	516	60.7
	BiCG(4)	91,814	$1.27 \cdot 10^{-5}$	$1.09 \cdot 10^{-4}$	$5.15 \cdot 10^{-5}$	27	27	255	65.4
	GMRES	91,814	$1.36 \cdot 10^{-4}$	$1.09 \cdot 10^{-4}$	$5.15 \cdot 10^{-5}$	27	27	1,667	1,380

Table 1: Linear solver comparison for variable $\tilde{\boldsymbol{v}}$; results for different $\tilde{\boldsymbol{v}}$ for *Problem 1* with $\alpha = 0$.

We are able to compute the correct solution in all cases. Indeed, irrespective of the problem and of the used inner linear solver, the algorithm always converges and global errors never exceed 10^{-3}. The choice of

$\tilde{\boldsymbol{v}}$	Lin. Solver	res_0	res	err_h	err_2	ν_{end}	k_{end}	j_{end}	t_{tot}
$\tilde{\boldsymbol{v}}_1$	AM	7,510	$1.85 \cdot 10^{-5}$	$3.65 \cdot 10^{-3}$	$4.69 \cdot 10^{-3}$	23	23	4,720	192.3
	BiCG(1)	7,510	$1.77 \cdot 10^{-5}$	$3.65 \cdot 10^{-3}$	$4.69 \cdot 10^{-3}$	23	23	1,754	114.4
	BiCG(2)	7,510	$1.73 \cdot 10^{-5}$	$3.65 \cdot 10^{-3}$	$4.69 \cdot 10^{-3}$	23	23	891	111.8
	BiCG(4)	7,510	$1.78 \cdot 10^{-5}$	$3.65 \cdot 10^{-3}$	$4.69 \cdot 10^{-3}$	23	23	458	112.6
	GMRES	7,510	$1.92 \cdot 10^{-4}$	$3.65 \cdot 10^{-3}$	$4.69 \cdot 10^{-3}$	23	23	2,307	1,677
$\tilde{\boldsymbol{v}}_2$	AM	7,514	$2.44 \cdot 10^{-5}$	$5.21 \cdot 10^{-3}$	$6.69 \cdot 10^{-3}$	23	23	5,307	227.6
	BiCG(1)	7,514	$1.81 \cdot 10^{-5}$	$5.21 \cdot 10^{-3}$	$6.69 \cdot 10^{-3}$	23	23	1,842	100.1
	BiCG(2)	7,514	$1.82 \cdot 10^{-5}$	$5.21 \cdot 10^{-3}$	$6.69 \cdot 10^{-3}$	23	23	952	116.8
	BiCG(4)	7,514	$1.82 \cdot 10^{-5}$	$5.21 \cdot 10^{-3}$	$6.69 \cdot 10^{-3}$	23	23	461	120.0
	GMRES	7,514	$2.09 \cdot 10^{-4}$	$5.21 \cdot 10^{-3}$	$6.69 \cdot 10^{-3}$	23	23	2,345	1,672
$\tilde{\boldsymbol{v}}_3$	AM	7,527	$1.98 \cdot 10^{-4}$	$7.15 \cdot 10^{-3}$	$9.17 \cdot 10^{-3}$	23	23	8,595	297.9
	BiCG(1)	7,527	$5.24 \cdot 10^{-5}$	$7.15 \cdot 10^{-3}$	$9.17 \cdot 10^{-3}$	23	23	3,458	121.3
	BiCG(2)	7,527	$4.99 \cdot 10^{-5}$	$7.15 \cdot 10^{-3}$	$9.17 \cdot 10^{-3}$	23	23	1,717	133.5
	BiCG(4)	7,527	$4.61 \cdot 10^{-5}$	$7.15 \cdot 10^{-3}$	$9.17 \cdot 10^{-3}$	23	23	864	152.4
	GMRES	7,527	$7.75 \cdot 10^{-4}$	$7.15 \cdot 10^{-3}$	$9.17 \cdot 10^{-3}$	23	23	3,752	2,450
$\tilde{\boldsymbol{v}}_4$	AM	7,508	$1.96 \cdot 10^{-5}$	$3.49 \cdot 10^{-3}$	$4.48 \cdot 10^{-3}$	23	23	5,245	225.2
	BiCG(1)	7,508	$1.36 \cdot 10^{-5}$	$3.49 \cdot 10^{-3}$	$4.48 \cdot 10^{-3}$	23	23	2,435	112.0
	BiCG(2)	7,508	$1.71 \cdot 10^{-5}$	$3.49 \cdot 10^{-3}$	$4.48 \cdot 10^{-3}$	23	23	1,100	119.2
	BiCG(4)	7,508	$1.68 \cdot 10^{-5}$	$3.49 \cdot 10^{-3}$	$4.48 \cdot 10^{-3}$	23	23	614	161.9
	GMRES	7,508	$2.17 \cdot 10^{-4}$	$3.49 \cdot 10^{-3}$	$4.48 \cdot 10^{-3}$	23	23	2,958	2,113

Table 2: Linear solver comparison for variable $\tilde{\boldsymbol{v}}$; results for different $\tilde{\boldsymbol{v}}$ for *Problem 2* with $\alpha = 0$.

Lin. Solver	res_0	res	err_h	err_2	ν_{end}	k_{end}	j_{end}	t_{tot}
AM	30,295	$1.80 \cdot 10^{-5}$	$6.49 \cdot 10^{-4}$	$3.07 \cdot 10^{-4}$	25	25	22,186	1,078
BiCG(1)	30,295	$1.76 \cdot 10^{-5}$	$6.49 \cdot 10^{-4}$	$3.07 \cdot 10^{-4}$	25	25	611	47.6
BiCG(2)	30,295	$1.75 \cdot 10^{-5}$	$6.49 \cdot 10^{-4}$	$3.07 \cdot 10^{-4}$	25	25	317	48.5
BiCG(4)	30,295	$1.73 \cdot 10^{-5}$	$6.49 \cdot 10^{-4}$	$3.07 \cdot 10^{-4}$	25	25	148	52.2
GMRES	30,295	$1.80 \cdot 10^{-4}$	$6.49 \cdot 10^{-4}$	$3.07 \cdot 10^{-4}$	25	25	1,187	1,537

Table 3: Results for *Problem 3* with various inner solvers

the linear solver is nonetheless important, as we can see looking at j_{end} and at the total time t_{tot}. Indeed, we see that the AM method requires many more iterations than the BiCGstab(l). Time t_{tot} is higher as well: for instance, in *Problem 1* when we use the BiCGstab(1) as linear solver, the LDM computes the solution in less than one fifth of the time required when the AM is used. The difference is yet more remarkable for *Problem 3*. This had to be expected: indeed, setting $c_1 = c_2 = 1$, $\tilde{\boldsymbol{v}}$ remains quite small. Thus, the Jacobian is weakly non-symmetric and the AM method is thus slower than the other analyzed linear solvers.
In the BiCGstab(l) method, we also notice that large values of l lead to a reduction of the number of linear iterations but not to a reduction of the computational time, which rather tends to increase for $l > 2$. This is due to the increased computational cost of each iteration of the BiCG-stab(l) method. Since the BiCGstab(1) (which is equivalent to the Bi-CGSTAB in [22]) can efficiently solve all the analyzed problems, increasing l is thus not needed. In the following, we thus focus on the AM and on the BiCG-stab(1) methods.

Let us then see what happens as c_1 and c_2 are increased, leading to a more non-symmetric Jacobian matrix. We restrict our analysis to choices of c_1 and c_2 which satisfy the condition $h < 2\sigma_{\min}/\tilde{\boldsymbol{v}}_{\max}$ within the discretization used in the above experiments. This condition, moreover, ensures that the AM method

converges (see [16]) since it implies that $A(\boldsymbol{u})$ is an M-matrix.

For instance, let us consider *Problem 1* (which is more suitable for this analysis, since $\sigma_{\min}$ is larger) and $\tilde{\boldsymbol{v}} = \tilde{\boldsymbol{v}}_4$. Taking into account that both $\tilde{v}_1$ and $\tilde{v}_2$ attain larger values when t is small, it is easy to verify that $h < 2\sigma_{\min}/\tilde{\boldsymbol{v}}_{\max}$ is satisfied if $c_1 < 200$ and $c_2 < 96$ at $t = 0$. We thus consider these two values as thresholds for the choice of c_1 and c_2. In Table 4 we report the results computed for different choices of c_1 and c_2.

c_1	c_2	Lin. Solver	res_0	res	err_h	err_2	j_{end}	t_{tot}
10	10	AM	91,829	$1.33 \cdot 10^{-5}$	$1.00 \cdot 10^{-3}$	$4.74 \cdot 10^{-4}$	6,935	297.4
		BiCG(1)	91,829	$9.60 \cdot 10^{-6}$	$1.00 \cdot 10^{-3}$	$4.74 \cdot 10^{-4}$	1,075	74.2
50	50	AM	91,903	$1.32 \cdot 10^{-5}$	$2.93 \cdot 10^{-3}$	$1.39 \cdot 10^{-3}$	4,856	153.3
		BiCG(1)	91,903	$1.34 \cdot 10^{-5}$	$2.93 \cdot 10^{-3}$	$1.39 \cdot 10^{-3}$	2,041	126.0
190	90	AM	91,441	$1.32 \cdot 10^{-5}$	$4.45 \cdot 10^{-3}$	$2.10 \cdot 10^{-3}$	3,525	123.8
		BiCG(1)	91,441	$1.30 \cdot 10^{-5}$	$4.45 \cdot 10^{-3}$	$2.10 \cdot 10^{-3}$	2,957	128.1
50	95	AM	92,248	$1.21 \cdot 10^{-5}$	$3.17 \cdot 10^{-3}$	$1.50 \cdot 10^{-3}$	3,795	118.8
		BiCG(1)	92,248	$1.36 \cdot 10^{-5}$	$3.17 \cdot 10^{-3}$	$1.50 \cdot 10^{-3}$	2,625	147.4
195	95	AM	91,454	$1.31 \cdot 10^{-5}$	$4.49 \cdot 10^{-3}$	$2.12 \cdot 10^{-3}$	3,438	111.2
		BiCG(1)	91,454	$1.09 \cdot 10^{-5}$	$4.49 \cdot 10^{-3}$	$2.12 \cdot 10^{-3}$	2,897	133.6

Table 4: Results for different choices of c_1 and c_2 for *Problem 1* with $\tilde{\boldsymbol{v}} = \tilde{\boldsymbol{v}}_4$ and $\alpha = 0$.

We notice that, as expected, the AM method gets faster and faster as c_1 and c_2 increase, which is when the Jacobian gets more non-symmetric. Even more, being $\tilde{\boldsymbol{v}}$ variable and dependent on t, the non-symmetry of the Jacobian vary at different time levels. As mentioned above, with $\tilde{\boldsymbol{v}} = \tilde{\boldsymbol{v}}_4$ the Jacobian gets less non-symmetric at t increases, since t damps the values of $\tilde{\boldsymbol{v}}$. So, we expect the AM method to become less efficient as t increases. Eventually, when t is large, the BiCGstab(1) should become competitive also when c_1 and c_2 are large. In order to confirm this, let us then see what happens for different choices of c_1 and c_2. The results are reported in Table 5, 6 and 7 and represented in Figures 1, 2 and 3.

	Computational time at different time levels									
Lin. Solver	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
AM	29.3	32.3	35.1	36.4	35.6	38.2	38.2	37.8	40.2	41.1
BiCG(1)	4.2	5.1	5.3	6.0	6.1	6.3	7.3	7.5	7.2	7.7

Table 5: Computational time at different time levels. *Problem 1* with $c_1 = c_2 = 1$, $\tilde{\boldsymbol{v}} = \tilde{\boldsymbol{v}}_4$ and $\alpha = 0$.

	Computational time at different time levels									
Lin. Solver	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
AM	4.5	5.2	7.1	9.0	10.1	12.4	14.5	17.3	18.5	20.1
BiCG(1)	8.5	10.1	11.5	15.8	15.3	17.2	16.9	18.3	16.4	17.2

Table 6: Computational time at different time levels. *Problem 1* with $c_1 = 50$, $c_2 = 95$, $\tilde{\boldsymbol{v}} = \tilde{\boldsymbol{v}}_4$ and $\alpha = 0$.

In Table 5 and in Figure 1 we consider a weakly non-symmetric case, where we set $c_1 = c_2 = 1$. In this case, the Jacobian is always almost symmetric and the BiCG-stab is evidently faster at any time level.

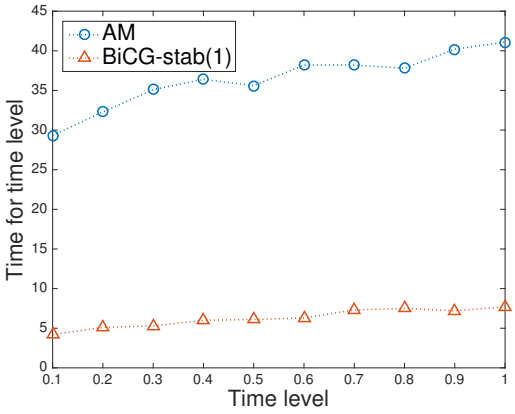
In Table 6 and in Figure 2 we instead set $c_1 = 50$ and $c_2 = 95$. Non-symmetry is thus much more relevant, and we see that at the beginning the algorithm using the AM as linear solver is twice as fast as the one using

	Computational time at different time levels									
Lin. Solver	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
AM	4.3	4.9	7.1	9.6	11.3	11.5	12.8	14.8	16.3	18.5
BiCG(1)	9.6	8.9	9.8	11.8	13.4	13.1	15.9	16.1	15.8	18.9

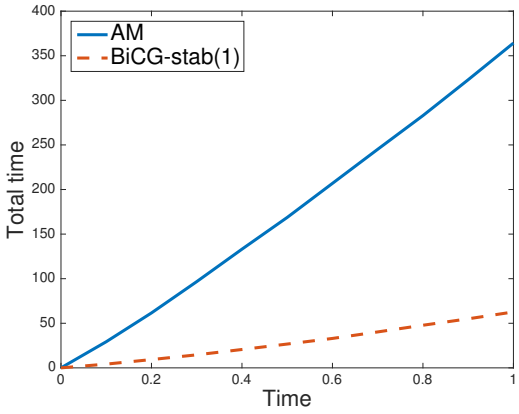
Table 7: Computational time at different time levels. *Problem 1* with $c_1 = 195$, $c_2 = 95$, $\tilde{\boldsymbol{v}} = \tilde{\boldsymbol{v}}_4$ and $\alpha = 0$.

the BiCG-stab(1). However, the difference gets smaller as t increases, until, for $t > 0.8$, the BiCG-stab(1) prevails. The total time required for computing the solution from $t = 0$ to $t = 1$ is however smaller when using the AM method, as already seen in Table 4. Choosing $t = 1$ as final time, the AM thus prevails.

This is even more true for the case $c_1 = 195$ and $c_2 = 95$, whose results are reported in Table 7 and in Figure 3. Here the AM method tends to be faster also at terminal time levels.

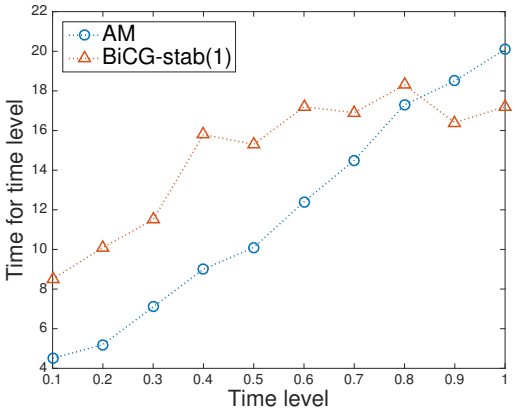


(a) Domain

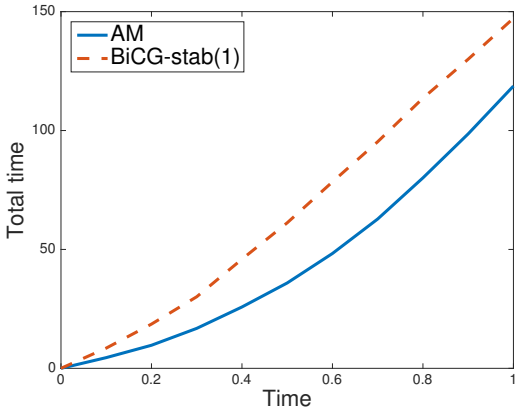


(b) Plot of u_1^* at final time $t = 1$

Figure 1: Computational time at different time levels and trend of total computational. *Problem 1* with $c_1 = c_2 = 1$, $\tilde{\boldsymbol{v}} = \tilde{\boldsymbol{v}}_4$ and $\alpha = 0$.

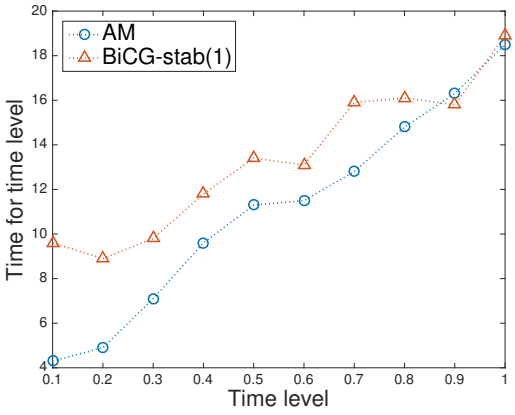


(a) Domain

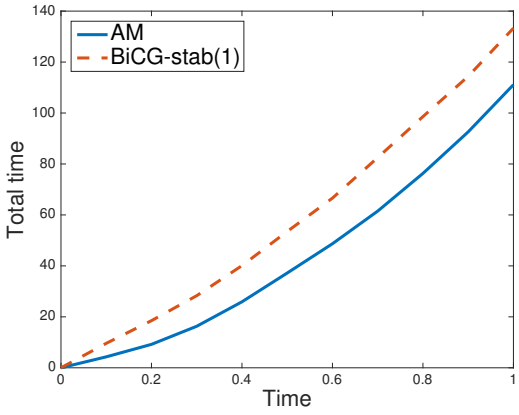


(b) Plot of u_1^* at final time $t = 1$

Figure 2: Computational time at different time levels and trend of total computational. *Problem 1* with $c_1 = 50$, $c_2 = 95$, $\tilde{\boldsymbol{v}} = \tilde{\boldsymbol{v}}_4$ and $\alpha = 0$.



(a) Domain



(b) Plot of u_1^* at final time $t = 1$

Figure 3: Computational time at different time levels and trend of total computational. *Problem 1* with $c_1 = 195$, $c_2 = 95$, $\tilde{\boldsymbol{v}} = \tilde{\boldsymbol{v}}_4$ and $\alpha = 0$.

6.2 Effect of discretization, α and initialization of linear solver and tolerances

We complete the analysis of the algorithm by summarizing some results that show what happens varying the discretization, changing the initialization and/or stopping criteria of the linear solvers and when α is different from zero.

In the following, when not otherwise specified, we consider *Problem 1* with $\tilde{\boldsymbol{v}} = \tilde{\boldsymbol{v}}_4$, $c_1 = c_2 = 50$, $\alpha = 0$ and we again set $N = 250$ and $\Delta t = 0.1$. With this choice, the Jacobian presents a significant non-symmetry, which, however, is not so big as to make the AM method more efficient than the BiCGstab(l), as we can see in Table 4. The following results have thus been obtained using the BiCGstab(1) as inner linear solver.

Let us start by considering the liner solver itself and see what happens if we modify initializations and stopping criteria. The results are reported in Table 8. In the second column of the table, we declare which initialization of the stopping criterion of the linear solver is used. *No bound on* $\hat{\epsilon}^{(k+1)}$ denotes that we are using (32) to define the tolerance of the linear solver at the first Newton iteration, while $\hat{\epsilon}^{(k+1)} \geq \hat{\sigma}\epsilon_{\nu+1}$ indicates that we are using the tolerance in (34). Finally, $\hat{\epsilon}^{(k+1)} \geq \hat{\sigma}\bar{\epsilon}$ denotes an intermediate case, where the bound on the tolerance of the linear solver at the first Newton iteration is given in a non-dynamical way by using $\bar{\epsilon}$.

$\boldsymbol{u}^{(\nu+1,k+1,0)}$	Bound on stop	j_{end}	res	err_h	err_2 rel.
$\mathbf{0}$	No bound on $\hat{\epsilon}^{(k+1)}$	$36,842$	$9.67 \cdot 10^{-8}$	$2.93 \cdot 10^{-3}$	$1.39 \cdot 10^{-3}$
	$\hat{\epsilon}^{(k+1)} \geq \sigma\bar{\epsilon}/2$	$35,127$	$4.63 \cdot 10^{-6}$	$2.93 \cdot 10^{-3}$	$1.39 \cdot 10^{-3}$
	$\hat{\epsilon}^{(k+1)} \geq \sigma\epsilon_{\nu+1}$	$32,230$	$1.33 \cdot 10^{-5}$	$2.93 \cdot 10^{-3}$	$1.39 \cdot 10^{-3}$
$\boldsymbol{u}^{(\nu+1,k)}$	No bound on $\hat{\epsilon}^{(k+1)}$	$3,840$	$3.14 \cdot 10^{-9}$	$2.93 \cdot 10^{-3}$	$1.39 \cdot 10^{-3}$
	$\hat{\epsilon}^{(k+1)} \geq \sigma\bar{\epsilon}/2$	$3,393$	$2.03 \cdot 10^{-6}$	$2.93 \cdot 10^{-3}$	$1.39 \cdot 10^{-3}$
	$\hat{\epsilon}^{(k+1)} \geq \sigma\epsilon_{\nu+1}$	$2,041$	$1.34 \cdot 10^{-5}$	$2.93 \cdot 10^{-3}$	$1.39 \cdot 10^{-3}$

Table 8: Effect of starting vectors and stopping criterion of the linear solver

Global errors do not change in all analyzed cases: indeed, some differences could be spotted only if we considered two more decimal digits. However, different choices of initializations and stopping criteria deeply influence the efficiency of the algorithm, here represented by the total number on linear iterations required to compute the solution at the last time level, j_{end}. Indeed, since we use always the same linear solver, j_{end} is a better indicator than, e.g., t_{tot}, which is less reproducible.

We notice that j_{end} is ten times larger when the starting vector of the linear solver is initialized by zero instead of by the solution at the previous Newton iteration, $\boldsymbol{u}^{(\nu+1,k)}$. This is consistent with what happens for constant $\tilde{\boldsymbol{v}}$ and also with what we expected: indeed, the choice $\boldsymbol{u}^{(\nu+1,k+1,0)} = \boldsymbol{u}^{(\nu+1,k)}$ was made in order to choose a starting vector closer to the solution.

Analogously, we notice that we achieve a consistent reduction of the computational cost with no loss in accuracy if we apply the initialization of the tolerance of the linear solver described in Section 5.2. Indeed, j_{end} is reduced by about 20% when $\boldsymbol{u}^{(\nu+1,k+1,0)} = \mathbf{0}$ and by almost 50% when $\boldsymbol{u}^{(\nu+1,k+1,0)} = \boldsymbol{u}^{(\nu+1,k)}$, with no increase of global errors.

Passing to the analysis of the discretization, let us set $\boldsymbol{u}^{(\nu+1,k+1,0)} = \boldsymbol{u}^{(\nu+1,k)}$ and $\epsilon^{(k+1)} \geq \sigma\epsilon_{\nu+1}$ and let us change the number N of points in which the space domain is discretized. In this regard, we also note that, since $c_1 = c_2 = 50$, the condition $h < 2\sigma_{\min}/\tilde{v}_{\max}$ holds only for $N > 104$. We get the results in Table 9, where EOC denotes the experimental order of convergence.

We notice that global errors decrease as N increases. On the other hand, also the initial residual res_0 increases, leading to higher number of lagged iterations and, thus, of linear iterations, with j_{end} that roughly doubles when N is doubled. The increase in computational times is, then, steeper, since increasing N means increasing also the order of all the matrices involved, thus making the entire solution process more complicated.

Passing to time discretization, Table 10 reports the results obtained as Δt is varied.

N	res_0	res	err_h	err_2 rel.	EOC	ν_{end}	k_{end}	j_{end}	t_{tot}
125	16,668	$1.58 \cdot 10^{-5}$	$5.80 \cdot 10^{-3}$	$2.77 \cdot 10^{-3}$	0.98	24	24	1,019	27.1
250	91,903	$1.34 \cdot 10^{-5}$	$2.93 \cdot 10^{-3}$	$1.39 \cdot 10^{-3}$	0.99	27	27	2,041	126.0
500	514,995	$1.67 \cdot 10^{-5}$	$1.47 \cdot 10^{-3}$	$6.93 \cdot 10^{-4}$	-	29	29	4,717	840.6

Table 9: Effect of space discretization

Δt	res_0	res	err_h	err_2 rel.	ν_{end}	k_{end}	j_{end}	t_{last}	t_{tot}
$2 \cdot 10^{-1}$	332,033	$1.18 \cdot 10^{-5}$	$2.94 \cdot 10^{-3}$	$1.39 \cdot 10^{-3}$	29	29	2,325	15.37	65.78
10^{-1}	91,903	$1.34 \cdot 10^{-5}$	$2.93 \cdot 10^{-3}$	$1.39 \cdot 10^{-3}$	27	27	2,041	13.17	126.0
10^{-2}	1,000	$1.55 \cdot 10^{-5}$	$1.97 \cdot 10^{-3}$	$9.33 \cdot 10^{-4}$	20	20	1,207	8.84	715.4
10^{-3}	10.18	$9.40 \cdot 10^{-6}$	$4.22 \cdot 10^{-4}$	$2.00 \cdot 10^{-4}$	14	14	160	1.90	1,686.0

Table 10: Effect of time discretization

Also in this case, global errors decrease as Δt is reduced. Moreover also res_0 decreases with Δt, implying that also the times required for computing the solution at each time level get smaller (e.g. see t_{last}, which refers to the time needed for computing the solution at the last time level). However, a smaller Δt implies that we also need more time levels in order to compute the solution from t_0 to t_f. Thus, the total time t_{tot} increases as Δt is reduced. However, we cannot arbitrarily increase the size of the time step, since the conditions in Theorems 1 and 2 are eventually not satisfied for excessively large time steps (if h is not increased as well).

For completeness, we finally see what happens when α is not zero. The results are reported in Table 11.

α	res_0	res	err_h	err_2 rel.	ν_{end}	k_{end}	j_{end}
0	332,033	$1.18 \cdot 10^{-5}$	$2.94 \cdot 10^{-3}$	$1.39 \cdot 10^{-3}$	29	29	2,325
10	91,908	$1.34 \cdot 10^{-5}$	$2.78 \cdot 10^{-3}$	$1.31 \cdot 10^{-3}$	27	27	2,030
100	91,952	$1.15 \cdot 10^{-5}$	$1.87 \cdot 10^{-3}$	$8.86 \cdot 10^{-4}$	27	27	1,860
$x^2 + y^2$	91,904	$1.06 \cdot 10^{-5}$	$2.92 \cdot 10^{-3}$	$1.38 \cdot 10^{-3}$	27	27	2,133
$10/(10^{-3} + x + y)^2$	92,079	$1.32 \cdot 10^{-5}$	$2.34 \cdot 10^{-3}$	$1.11 \cdot 10^{-3}$	27	27	1,438
$\log^2(x^2 + y)$	91,904	$1.34 \cdot 10^{-5}$	$2.89 \cdot 10^{-3}$	$1.37 \cdot 10^{-3}$	27	27	2,105

Table 11: Results for $\alpha \neq 0$

We see that α does not have a significant effect on the results, as we expected: indeed, it only acts on the diagonal elements, strengthening (due to its sign) the diagonal dominance of $A(\boldsymbol{u})$. The first three rows of Table 11 show that, however, a larger α can have the effect of reducing the complexity of the problem (since j_{end} decreases as α increases) and, possibly, also the global errors. This happens also when α is variable: e.g. $\alpha = 10/(10^{-3} + x + y)^2$ leads to $\alpha \in [10^4, 4.998]$ in the domain we are considering and j_{end} is smaller than in the cases where α is smaller. However, when α is variable it is harder to evaluate the effect of α itself, since its value can vary consistently in different parts of the domain.

6.3 Discretization of the convective term and convection-dominated problems

Up to now, we have considered the solution of systems arising from space discretizations where the convective term is discretized by central finite differences. This choice is convenient in the analysis of the algorithm and it also makes so that we have the discretization of the convective term has the same order of accuracy as the discretization of the diffusion term (i.e., $O(h^2)$). However, as remarked in Section 2.1, it also has the shortcoming that $A(\boldsymbol{u})$ is an M-matrix only if h satisfies (4). Evidently, satisfying this condition requires

smaller and smaller values of h as the problem gets more convection-dominated, with important consequences on the convergence theorems. Moreover, in convection-dominated problems, it is especially desirable that $A(\boldsymbol{u})$ be an M-matrix so to be able to use the AM method, which, as seen in the previous subsections, is particularly well-suited to solve systems with non-symmetric matrices.

Considering other discretizations of the convective term, it is nonetheless easy to notice that $A(\boldsymbol{u})$ is an M-matrix for any choice of h if we instead employ the upwind discretization, at the cost, however, of reducing the order of accuracy to $O(h)$. Thus, we here consider this alternative discretization in order to show numerically its behavior and to provide an useful framework for convection-dominated problems. Further information on this case (referred to the case of constant velocity term) can be also found in [10].

Let us then consider, for instance, Problem 1 with $\tilde{\boldsymbol{v}} = \tilde{\boldsymbol{v}}_4$. In Table 12, we report the results obtained with different discretizations as the value of c_1 and c_2 increases. Observe that, in all analyzed cases, condition (4) is not respected, as c_1 and c_2 exceed the threshold values reported in Section 6.1 (which are 200 and 96, respectively).

c_1, c_2	Discretization	Lin. Solver	res_0	res	err_h	err_2 rel.	j_{end}	t_{tot}
$500, 500$	Central FD	AM	-	-	-	-	-	-
		BiCG(1)	$93,250$	$1.32 \cdot 10^{-5}$	$5.06 \cdot 10^{-3}$	$2.40 \cdot 10^{-3}$	$6,213$	435.2
		BiCG(2)	$93,250$	$1.26 \cdot 10^{-5}$	$5.06 \cdot 10^{-3}$	$2.40 \cdot 10^{-3}$	$1,751$	184.3
		BiCG(4)	$93,250$	$1.36 \cdot 10^{-5}$	$5.06 \cdot 10^{-3}$	$2.40 \cdot 10^{-3}$	762	181.6
	Upwind	AM	$100,015$	$1.42 \cdot 10^{-5}$	$6.17 \cdot 10^{-3}$	$2.92 \cdot 10^{-3}$	$1,546$	82.6
		BiCG(1)	$100,015$	$1.37 \cdot 10^{-5}$	$6.17 \cdot 10^{-3}$	$2.92 \cdot 10^{-3}$	$2,322$	110.4
		BiCG(2)	$100,015$	$1.43 \cdot 10^{-5}$	$6.17 \cdot 10^{-3}$	$2.92 \cdot 10^{-3}$	$1,012$	110.4
		BiCG(4)	$100,015$	$1.27 \cdot 10^{-5}$	$6.17 \cdot 10^{-3}$	$2.92 \cdot 10^{-3}$	481	115.5
$2000, 2000$	Central FD	AM	-	-	-	-	-	-
		BiCG(1)	-	-	-	-	-	-
		BiCG(2)	$103,953$	$1.24 \cdot 10^{-5}$	$5.47 \cdot 10^{-3}$	$2.59 \cdot 10^{-3}$	$1,761$	179.1
		BiCG(4)	$103,953$	$1.16 \cdot 10^{-5}$	$5.47 \cdot 10^{-3}$	$2.59 \cdot 10^{-3}$	773	178.2
	Upwind	AM	$129,319$	$1.63 \cdot 10^{-5}$	$6.56 \cdot 10^{-3}$	$3.11 \cdot 10^{-3}$	914	72.1
		BiCG(1)	$129,319$	$7.63 \cdot 10^{-6}$	$6.56 \cdot 10^{-3}$	$3.11 \cdot 10^{-3}$	$2,131$	117.7
		BiCG(2)	$129,319$	$1.39 \cdot 10^{-5}$	$6.56 \cdot 10^{-3}$	$3.11 \cdot 10^{-3}$	874	103.2
		BiCG(4)	$129,319$	$7.03 \cdot 10^{-6}$	$6.56 \cdot 10^{-3}$	$3.11 \cdot 10^{-3}$	463	116.7
$50000, 50000$	Central FD	AM	-	-	-	-	-	-
		BiCG(1)	-	-	-	-	-	-
		BiCG(2)	-	-	-	-	-	-
		BiCG(4)	-	-	-	-	-	-
	Upwind	AM	$1.35 \cdot 10^{6}$	$1.12 \cdot 10^{-5}$	$6.71 \cdot 10^{-3}$	$3.18 \cdot 10^{-3}$	$1,122$	78.6
		BiCG(1)	$1.35 \cdot 10^{6}$	$1.07 \cdot 10^{-5}$	$6.71 \cdot 10^{-3}$	$3.18 \cdot 10^{-3}$	$2,558$	131.9
		BiCG(2)	$1.35 \cdot 10^{6}$	$1.19 \cdot 10^{-5}$	$6.71 \cdot 10^{-3}$	$3.18 \cdot 10^{-3}$	$1,142$	111.7
		BiCG(4)	$1.35 \cdot 10^{6}$	$8.60 \cdot 10^{-6}$	$6.71 \cdot 10^{-3}$	$3.18 \cdot 10^{-3}$	548	133.5

Table 12: Comparison of discretizations by finite differences and upwind scheme for Problem 1 with $\tilde{\boldsymbol{v}} = \tilde{\boldsymbol{v}}_4$.

As we expected, the algorithms employing central finite differences on the convective term incur in difficulties as convection becomes more dominant. The AM method, in particular, immediately fails. Algorithms using BiCGstab(l) method work better (especially when l is quite large), but they ultimately fail as convection becomes more dominant. Moreover, also when they work, they are less efficient than the

corresponding algorithm employing the upwind discretization. On the other hand, if the discretization is performed by an upwind scheme, the AM method always converges without presenting any problem (on the contrary, the algorithm gets faster with larger values of $\tilde{\boldsymbol{v}}$). This was to be expected, since the Jacobian is now an M-matrix, as required by the convergence of the AM method.

7 Conclusions

We have presented an iterative procedure which solves nonlinear steady and non-steady reaction-convection-diffusion equations with a variable velocity term. We have discretized the partial differential problems and proved that the LDM applied to the resulting nonlinear algebraic systems converges when some assumptions are satisfied. In this context, we have also studied the uniform monotonicity of the finite-difference operator, which is crucial for the uniqueness of the solution and for the convergence of the LDM itself.

We have then described the solution of the weakly-nonlinear algebraic systems generated by the LDM and provided some details over the implementation of the entire procedure.

Finally, we have provided several numerical experiments showing the behavior of the LDM in a variety of situations. We have showed that the LDM can successfully compute the solution of all the considered test problems. We have also pointed out that the choice of the linear inner solver affects the efficiency of the method: indeed, the fastest linear solver depends on the non-symmetry of the coefficient matrix of the linear system, which, in turn, is determined by the velocity term $\tilde{\boldsymbol{v}}$. We have then showed that efficiency is affected also by other factors, including the refinement of the discretization and the choice of starting vectors and stopping criteria.

A The algorithm

Algorithm 1 Lagged diffusivity procedure

Require: initial condition $\boldsymbol{u}|_0$ in $t = 0$; a tolerance $\bar{\epsilon}$

for $n = 1, 2, \ldots$ **do** *Time step*
  Initialize solution vector for lagged iteration: $\boldsymbol{u}^{(0)} = \boldsymbol{u}|_{n-1}$
  Initialize lagged tol.: $\epsilon_1 = \epsilon_0 \|\boldsymbol{F}(\boldsymbol{u}^{(0)})\|$
  for $\nu = 0, 1, \ldots$ **do** *Lagged iteration*
    Initialize linear solver tol.: $\hat{\epsilon}^{(1)} = \max\left(\hat{\sigma}\left\|\boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu)})\right\|, \hat{\sigma}\epsilon_{\nu+1}\right)$
    for $k = 0, 1, \ldots$ **do** *Simpl. Newton iteration*
      for $j = 0, 1, \ldots$ **do** *Linear solver iteration*
        Compute $(j+1)$-th iterate $\boldsymbol{u}^{(\nu+1,k+1,j+1)}$ for solving (30)
        Compute residual $\boldsymbol{r}_{j+1}$ as in (31)
        if $\|\boldsymbol{r}_{j+1}\| \leq \hat{\epsilon}^{(k+1)}$ **then return**
        j = j+1
      end for
      Compute Newton residual $\boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu+1,k+1)})$
      if $\left\|\boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu+1,k+1)})\right\| \leq \epsilon_{\nu+1}$ **then return**
      Update linear solver tol.: $\hat{\epsilon}^{(k+1)} = \hat{\sigma}\left\|\boldsymbol{F}_\nu(\boldsymbol{u}^{(\nu+1,k+1)})\right\|$
      k = k+1
    end for
    Update vectors and matrices: find $\boldsymbol{F}_{\nu+1}(\boldsymbol{u}^{(\nu+1)})$
    $\nu = \nu + 1$
    $\epsilon_{\nu+1} = 0.5\epsilon_\nu$
    if $\epsilon_{\nu+1} \leq \bar{\epsilon}$ **then return**
  end for
  $n = n + 1$
end for

References

[1] Ben, Y., Chang, H.: Nonlinear Smoluchowski slip velocity and micro-vortex generation. J. Fluid Mech. **461**, 229–238 (2002)

[2] Catté, F., Lions, P.L., Morel, J.M., Coll, T.: Image selective smoothing and edge detection by nonlinear diffusion. SIAM J. Numer. Anal. **29**(1), 182–193 (1992)

[3] Chan, T.F., Mulet, P.: On the convergence of the lagged diffusivity fixed point method in total variation image restoration. SIAM J. Numer. Anal. **36**(2), 354–367 (1999)

[4] Dembo, R., Eisenstat, S., Steihaug, T.: Inexact Newton methods. SIAM J. Numer. Anal. **19**, 400–408 (1982)

[5] Eisenstat, S., Walker, H.: Globally convergent inexact Newton methods. SIAM J. Optimiz. **4**, 393–422 (1994)

[6] Isaacson, E., Keller, H.B.: Analysis of Numerical Methods. John Wiley & and Sons, New York (1966)

[7] Kelley, C.: Iterative Methods for Linear and Nonlinear Equations. Frontiers in Applied Mathematics. SIAM, Philadelphia (1995)

[8] Lions, J.L.: Quelques Méthodes De Résolution Des Problémes Aux Limites Non Linéaires. Collection Études Mathématiques. Dunod, Paris (1969)

[9] Meyer, G.H.: The numerical solution of quasilinear ellipting equations. In: G. Byrne, C. Hall (eds.) Numerical solution of systems of nonlinear algebraic equations. Academic Press (1973)

[10] Mezzadri, F., Galligani, E.: On the lagged diffusivity method for the solution of nonlinear finite difference systems. Algoritms **10**(88) (2017)

[11] Mezzadri, F., Galligani, E.: A lagged diffusivity method for reaction–convection–diffusion equations with Dirichlet boundary conditions. Appl. Numer. Math. **123**, 300–319 (2018)

[12] Ortega, J., Rheinboldt, W.: Iterative Solution of Nonlinear Equations in Several Variables. Computer Science and Applied Mathematics. Academic Press inc. (1970)

[13] Perona, P., Malik, J.: Scale-space and edge detection using anisotropic diffusion. IEEE T. Pattern. Anal. **12**(7), 629–639 (1990)

[14] Roubíček, T.: Nonlinear Partial Differential Equations with Applications. ISNM International Series of Numerical Mathematics. Birkhauser Verlag, Basel-Boston-Berlin (2005)

[15] Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Phys. D **60**, 259–268 (1992)

[16] Ruggiero, V., Galligani, E.: An iterative method for large sparse linear systems on a vector computer. Comput. Math. Appl. **20**(1), 25–28 (1990)

[17] Ruggiero, V., Galligani, E.: A parallel algorithm for solving block tridiagonal linear systems. Comput. Math. Appl. **24**, 15–21 (1992)

[18] Saad, Y., Schultz, M.: GMRES a generalized minimal residual algorithm for solving nonsymmetrical systems. SIAM J. Sci. Statist. Comput. **7**, 856–869 (1986)

[19] Sleijpen, G.L.G., Fokkema, D.R.: Bicgstab(l) for linear equations involving unsymmetric matrices with complex spectrum. Electron. T. Numer. Ana. **1**, 11–32 (1993)

[20] Varga, R.: Matrix Iterative Analysis, 2nd edn. Springer, Berlin (2000)

[21] Vogel, C., Oman, M.: Iterative methods for total variation denoising. SIAM J. Sci. comput. **17**(1), 227–238 (1996)

[22] van der Vorst, H.: Bi-CGSTAB: A fast and smoothly convergent variant to the Bi-CG for the solution of nonlinear systems. SIAM J. Statist. Comput. **13**, 631–644 (1992)