

Operator-Splitting Methods Respecting Eigenvalue Problems for Nonlinear Equations and Applications for Burgers Equations.

Jürgen Geiser ^{*} Lena Noack [†]

July 28, 2008

Abstract

In this article we consider iterative operator-splitting methods for nonlinear differential equations with respect to their eigenvalues. The main feature of the proposed idea is the fixed-point iterative scheme that linearizes our underlying equations. Based on the approximated eigenvalues of such linearized systems we choose the order of the the operators for our iterative splitting scheme. The convergence properties of such a mixed method are studied and demonstrated. We confirm with numerical applications the effectiveness of the proposed scheme in comparison with the standard operator-splitting methods by providing improved results and convergence rates. We apply our results to deposition processes.

Keyword numerical analysis, operator-splitting method, initial value problems, iterative solver method, eigenvalue problem, convection-diffusion-reaction equation.

AMS subject classifications. 35J60, 35J65, 65M99, 65N12, 65Z05, 74S10, 76R50.

1 Introduction

Our study is motivated by complex models with coupled processes, e.g. transport and reaction equations with nonlinear parameters. These models arose from the simulation of a heat transport in an engineering apparatus, e.g. crystal growth, see [17], or the simulation of a chemical reaction and transport, e.g. in bioremediation or waste disposals, see [14].

^{*}Department of Mathematics, Humboldt Universität zu Berlin, Unter den Linden 6, D-10099 Berlin, Germany, E-mail: geiser@mathematik.hu-berlin.de

[†]Department of Mathematics, Humboldt Universität zu Berlin, Unter den Linden 6, D-10099 Berlin, Germany, E-mail: noackl@math.hu-berlin.de

We contribute an efficient decomposition method for nonlinear differential equations by applying the decomposition idea based on the eigenvalue problem, see [2], [8], and [15].

We propose an algorithm to compute such pre-eigenvalues for the scale separations. To apply the scale separations we have to discuss a Runge-Kutta method as a higher-order time-discretization to approximate the coarser scales and the finer scales. The efficiency of different scales due to each operator allows to optimize the iterative operator-splitting method.

The paper is organized as follows. A mathematical model based on the nonlinear convection-diffusion equation is introduced in Section 2. The iterative splitting method for the nonlinear equation is given in Section 3. The eigenvalue problem is discussed in Section 4. The error analysis is treated in Section 5. We introduce the numerical results in Section 6. Finally we discuss our future works in the area of splitting and decomposition methods.

2 Mathematical model

When gas or fluid transport is physically more complex because of combined flows in three dimensions, the fundamental equations of fluid dynamics become the starting points of the analysis.

Three basic equations describe the conservation of mass, momentum, and energy, that are sufficient to describe the gas transport in the reactors, see [34].

1. Continuity: The conservation of mass requires the net rate of the mass accumulation in a region to be equal to the difference between the inflow and outflow rate.
2. Navier-Stokes: Momentum conservation requires the net rate of momentum accumulation in a region to be equal to the difference between the in and out rate of the momentum, plus the sum of the forces acting on the system.
3. Energy: The rate of accumulation of internal and kinetic energy in a region is equal to the net rate of internal and kinetic energy by convection, plus the net rate of heat flow by conduction, minus the rate of work done by the fluid.

We will concentrate on the momentum equation, see [19], which can be modelled by a viscous Burgers equation:

$$\partial_t c + \nabla F - R_g = 0, \text{ in } \Omega \times [0, T] \quad (1)$$

$$F = \frac{1}{2}c^2 - D\nabla c,$$

$$c(x, 0) = c_0(x), \text{ on } \Omega, \quad (2)$$

$$c(x, t) = c_1(x, t), \text{ on } \partial\Omega \times [0, T], \quad (3)$$

where c is the molar concentration and F the flux of the species. D is the diffusivity matrix and R_g is the reaction term. The initial value is given as c_0

and we assume a Dirichlet boundary with the function $c_1(x, t)$ being sufficiently smooth.

3 The iterative splitting method

The previously defined sequential operator-splitting methods have several drawbacks besides their benefits. For instance, for non-commuting operators there might be a very large constant in the splitting error which requires the use of an unrealistically small time step. Also, splitting the original problem into the different subproblems with one operator, i.e. neglecting the other components, is physically questionable.

In order to avoid these problems, one can use the iterative operator-splitting method on an interval $[0, T]$. This algorithm is based on the iteration with fixed splitting discretization step size τ . On every time interval $[t^n, t^{n+1}]$ the method solves the following subproblems consecutively for $i = 1, 3, \dots, 2m + 1$.

$$\partial_t c_i(x, t) = A c_i(x, t) + B c_{i-1}(x, t), \text{ with } c_i(x, t^n) = c^n \quad (4)$$

$$\partial_t c_{i+1}(x, t) = A c_i(x, t) + B c_{i+1}(x, t), \text{ with } c_{i+1}(x, t^n) = c^n, \quad (5)$$

$$\text{and } c_{i+1}(x, t) = c_i(x, t) = c_1 \text{ on } \partial\Omega \times (0, T),$$

where c^n is the known split approximation at time level $t = t^n$ (see [12]). This algorithm constitutes an iterative method which involves in each step both operators A and B . Hence, there is no real separation of the different physical processes in these equations.

3.1 Iterative operator-splitting method as fixed-point scheme

The iterative operator-splitting method is used as a fixed-point scheme to linearize the nonlinear operators, see [16] and [25].

We concentrate again on nonlinear differential equations of the form

$$\partial_t c = A(c)c + B(c)c, \quad (6)$$

where $A(c), B(c)$ are matrices with nonlinear entries and densely defined, where we assume that the entries involve the spatial derivatives of c , see [41]. In the following we discuss the standard iterative operator-splitting method as a fixed-point iteration method to linearize the operators.

We split our nonlinear differential equation (6) by applying

$$\partial_t c_i = A(c_{i-1})c_i + B(c_{i-1})c_{i-1}, \text{ with } c_i(x, t^n) = c^n, \quad (7)$$

$$\partial_t c_{i+1} = A(c_{i-1})c_i + B(c_{i-1})c_{i+1}, \text{ with } c_{i+1}(x, t^n) = c^n, \quad (8)$$

where the time step is $\tau = t^{n+1} - t^n$. The iterations are $i = 1, 3, \dots, 2m + 1$. $c_0(x, t) = c^n$ is the initial solution, where we assume that the solution c^{n+1} is

near c^n , or $c_0(x, t) \equiv 0$. Thus we have to solve the local fixed-point problem. c^n is the known split approximation at time level $t = t^n$.

The split approximation at time level $t = t^{n+1}$ is defined as $c^{n+1} = c_{2m+2}(x, t^{n+1})$. We assume that the operators $A(c_{i-1}(x, t^{n+1}))$, $B(c_{i-1}(x, t^{n+1}))$ are constant for $i = 1, 3, \dots, 2m + 1$. Here the linearization is done with respect to the iterations, such that $A(c_{i-1})$, $B(c_{i-1})$ are at least non-dependent operators in the iterative equations, and we can apply the linear theory. For the linearization we assume at least in the first equation $A(c_{i-1}(x, t)) \approx A(c_i(x, t))$, and in the second equation $B(c_{i-1}(x, t)) \approx B(c_{i+1}(x, t))$, for small t .

We have

$$\|A(c_{i-1}(x, t^{n+1}))c_i(x, t^{n+1}) - A(c(x, t^{n+1}))c(x, t^{n+1})\| \leq \epsilon,$$

for sufficient iterations $i \in \{1, 3, \dots, 2m + 1\}$.

Remark 3.1 The linearization with the fixed-point scheme can be used for smooth or weak nonlinear operators, otherwise we lose the convergence behavior, while we did not converge to the local fixed point, see [25].

4 Decoupling ideas based on eigenvalue problems

We apply the linearized system of differential equations for stiff or non-stiff operators.

We deal with the approximated eigenvalues of the operators and use them as reciprocal time scales.

We assume the following eigenvalue problem:

$$\begin{aligned} \partial_t c_i &= A(c_{i-1})c_i + B(c_{i-1})c_i \\ &\approx (\lambda_{A_{i-1}} + \lambda_{B_{i-1}})c_i, \quad (x, t) \in \Omega \times [t^n, t^{n+1}], \\ c_i(x, t^n) &= c^n, \end{aligned} \quad (9)$$

where the operators $A(c_{i-1})$ and $B(c_{i-1})$ result from the spatial discretization and c_{i-1} is the solution of at the iteration step $i - 1$ and known.

We assume, that the fixed point $c_i \rightarrow c$ for $i \rightarrow \infty$.

The eigenvalues are detected in the decoupled equations:

$$\partial_t c_i = A(c_{i-1})c_i = \lambda_{A_{i-1}}c_i, \quad (x, t) \in \Omega \times [t^n, t^{n+1}], \quad c_i(x, t^n) = c^n, \quad (10)$$

$$\partial_t c_i = B(c_{i-1})c_i = \lambda_{B_{i-1}}c_i, \quad (x, t) \in \Omega \times [t^n, t^{n+1}], \quad c_i(x, t^n) = c^n, \quad (11)$$

where c_{i-1} is the known solution of the last iterative step.

Based on the eigenvalues $\lambda_{A_{i-1}}$ and $\lambda_{B_{i-1}}$ we can propose the time steps $\Delta t_A \approx 1/\lambda_A$ and $\Delta t_{A_{i-1}} \approx 1/\lambda_{B_{i-1}}$.

We propose the vector iteration based on the Rayleigh quotient for the computation of the eigenvalues of the operators A and B :

$$Ac_{i+1,k} = c_{i+1,k+1}, \quad (12)$$

$$Bc_{i+1,m} = c_{i+1,m+1}, \quad (13)$$

where $k, m = 0, 1, 2, \dots$ and the eigenvalues are given as

$$\frac{c_{i+1,k+1}}{c_{i+1,k}} = |\lambda_{A,1}| + \mathcal{O}(p^k), \quad (14)$$

$$\frac{c_{i+1,m+1}}{c_{i+1,m}} = |\lambda_{B,1}| + \mathcal{O}(q^m), \quad (15)$$

where $\lambda_{A,1}$ and $\lambda_{B,1}$ are the maximal eigenvalues. The values are given as

$$p = \frac{\lambda_{A,2}}{\lambda_{A,1}} \text{ with } \lambda_{A,1} \geq \lambda_{A,2} \dots \geq \lambda_{A,n},$$

$$q = \frac{\lambda_{B,2}}{\lambda_{B,1}} \text{ with } \lambda_{B,1} \geq \lambda_{B,2} \dots \geq \lambda_{B,n}.$$

The following algorithm is used for separating the different scales of the operators A and B .

Algorithm 4.1 *We have the operators A, B .*

1. *We compute pre-eigenvalues with a given norm $\|\cdot\|$:*

$$\|Au\|, \|Bu\|,$$

where u is a possible solution vector of the equations (4)-(5).

2. *We compare the pre-eigenvalues:*

$$\|Au\| \leq \|Bu\| : A \text{ is stiff, or}$$

$$\|Au\| \geq \|Bu\| : B \text{ is stiff.}$$

3. *We initialize our splitting method. In the first step the stiff operator is treated implicitly using a higher-order method, the non-stiff method is treated explicitly. In the second step, the operators are treated the other way around.*

Remark 4.2 The efficiency of the method is given with the correct decomposition, which means the correct ordering of the underlying operators. With respect to the local error, the starting operator B in the first iterative equation dominates the error. Therefore the pre-processing to obtain the underlying eigenvalues is important and accelerates the solver process. Here we propose the vector iterations to compute the eigenvalues as a method that is embedded in our iterative splitting method. The declaration of the operators to be stiff or non-stiff results in the use of the correct splitting operators.

5 Error analysis

Subsequently we demonstrate the error analysis for the linear and nonlinear decomposition methods. We concentrate on the bounded operators and the assumption to obtain maximal eigenvalues for the eigenvalue problems. For the nonlinear problem, we assume a linearization of the nonlinear operator and a formulation of a linearized eigenvalue problem.

5.1 Error analysis for the linear method

In this section, we taken into account the $A(0)$ -stability analysis, which can be used to derive the stability of methods for ordinary differential equations, see [20].

We only consider spatial discretized systems, where the boundary conditions of the partial differential equations are embedded in the operators, see [4].

We consider the linear problem

$$\partial_t c(t) = Ac(t) + Bc(t), \quad (16)$$

where the initial condition is $c^n = c(t^n)$. The operators A and B are spatially discretized operators, e.g. they correspond to the discretization in space of convection and diffusion operators (matrices). We assume, that they can be considered as bounded operators in moderate refined meshes, see [4].

In the following we discuss the improved and stable iterative method, given in (4)-(5).

Theorem 5.1 *Let us consider the iterative method with the starting solution $c_1 = c_1(t^{n+1})$, which is of m -th order exact. We assume to estimate our operators A and B with the maximum eigenvalues λ_1 and λ_2 . Further we define $z_1 = \lambda_1\tau$, $z_2 = \lambda_2\tau$, where τ is the local time step.*

Then we can prove, that all successive iterative solutions are stable, see proof idea [25]. It holds

$$c_{i+1}(z_1, z_2) = c_{i+1}(z, -\infty) = 0 \leq 1, i = 1, 2, \dots, \quad (17)$$

where $c_1(t) \in U$ and U is the solution space for the iterative solutions, with $c_i \rightarrow c$ for $i \rightarrow \infty$.

Proof. We can proof the stability of an analytical solution, that is exact or has at least order m and get the solution

$$c_1(t) = \exp((\lambda_1 + \lambda_2)t) c_n. \quad (18)$$

Further we have the stability, and we denote $z_1 = \lambda_1\tau$ and $z_2 = \lambda_2\tau$.

$$c_1(z_1, z_2) = \exp(z_1 + z_2) c_n. \quad (19)$$

For the stiff case, $z_2 \rightarrow -\infty$, we have

$$\lim_{z_2 \rightarrow -\infty} c_1(z_1, z_2) = 0, \quad (20)$$

and therefore we have the stability, since

$$\|c_1(z_1, -\infty)\| \leq 1 \quad (21)$$

is fulfilled.

The value c_1 is a start-value of the iterative method.

For the iterative method we have the following stability:

$$\frac{\partial c_{i+1}}{\partial t} = Ac_{i+1} + Bc_i, \quad c_{i+1}(0) = c_n, \quad (22)$$

$$\frac{\partial c_{i+2}}{\partial t} = Ac_{i+1} + Bc_{i+2}, \quad c_{i+2}(0) = c_n. \quad (23)$$

We insert the operators $A = \lambda_1$ and $B = \lambda_2$. We can derive the analytical solution for c_{i+1} and get the solution:

$$c_{i+1}(t) = \exp(\lambda_1(t - t_n)) \left(\int_{t_n}^t \exp(-\lambda_1(s - t_n)) \lambda_2 c_i ds + c_n \right), \quad (24)$$

$$c_{i+2}(t) = \exp(\lambda_2(t - t_n)) \left(\int_{t_n}^t \exp(-\lambda_2(s - t_n)) \lambda_1 c_{i+1} ds + c_n \right). \quad (25)$$

We compute c_2 by inserting c_1 and get:

$$c_2(t) = \exp(\lambda_1(t - t_n)) \left(\int_{t_n}^t \exp(-\lambda_1(s - t_n)) \lambda_2 \right. \quad (26)$$

$$\left. \exp((\lambda_1 + \lambda_2)(s - t_n)) c_n ds + c_n \right). \quad (27)$$

Now we compute the non-commutative case until order two and get

$$c_2(t) = (1 + \lambda_1 \tau + \lambda_1^2 \tau^2 / 2! + \mathcal{O}(\tau^3)) \quad (28)$$

$$(1 + \lambda_2 \tau - \lambda_1 \lambda_2 \tau^2 / 2! + \lambda_2 \lambda_1 \tau^2 / 2! + \lambda_2^2 \tau^2 / 2! + \mathcal{O}(\tau^3))$$

$$= (1 + \lambda_1 \tau + \lambda_2 \tau + \lambda_1^2 \tau^2 / 2! + \lambda_1 \lambda_2 \tau^2 / 2! + \lambda_2 \lambda_1 \tau^2 / 2! + \lambda_2^2 \tau^2 / 2! + \mathcal{O}(\tau^3))$$

$$\approx \exp((\lambda_1 + \lambda_2)(t - t_n)) \quad (29)$$

with $\tau = t - t_n$.

The stability result for the c_2 is given by

$$c_2(z_1, z_2) = \exp(z_1 + z_2) c_n. \quad (30)$$

For the stiff case, $z_2 \rightarrow -\infty$, we have

$$\lim_{z_2 \rightarrow -\infty} c_2(z_1, z_2) = 0, \quad (31)$$

and therefore we have the stability.

The same proof can be done recursively for a starting solution $c_1(t)$ of order m .

Remark 5.2 The iterative operator-splitting method is invariant to the analytical solution and therefore stable. So it is enough to guaranty that a prestepping method exists, that could have at least order m .

5.1.1 Stability analysis in the discretized formulation

The analysis for the noncommutative part is much more complicate and often the help of full-discretized formulations are important.

Here, we consider the time-discretization with a θ -method, that is a Crank-Nicolson method for $\theta = 0.5$. Time- and space-discretizations help to balance the errors.

To obtain an accurate starting solution for the iterative splitting method, we can assume A-B splitting, Strang-splitting methods or IMEX methods (implicit-explicit methods) of m -th order, see [1].

To model different methods for our equation (16), we may take different values of the method parameter θ in the stage. So consider $m = 1$, and let $\theta_1, \theta_2 \geq 0.5$.

Assume that the two stages for the iterative method in (16) are discretized as

$$\bar{c}_{i+1}^{n+1} = c_i^n + \tau(1 - \theta_1)(A(c_{i+1}^n) + B(c_i^n)) + \tau\theta_1(A(\bar{c}_{i+1}^{n+1}) + B(c_i^{n+1})), \quad (32)$$

$$c_{i+1}^{n+1} = c_{i+1}^n + \tau(1 - \theta_2)(A(c_{i+1}^n) + B(c_i^n)) + \tau\theta_2(A(\bar{c}_{i+1}^{n+1}) + B(c_{i+1}^{n+1})), \quad (33)$$

where $c_i^n = c_{i+1}^n = c^n$ with initialization $c_0^{n+1} = c^n$.

For the linear system we denote $Z_1 = \tau A$ and $Z_2 = \tau B$ and set $\theta_1 = \theta_2$.

We get the following stability equation, cf. [24], for $\theta = 1/2$. We compute the first iteration with $i = 1$ and get the equation

$$\begin{aligned} c_1^{n+1} &= (I + (I - 1/2Z_2)^{-1}(I - 1/2Z_1)^{-1}(Z_1 + Z_2))c^n, & (34) \\ &= (I - 1/2Z_2)^{-1}((I - 1/2Z_2) + (I - 1/2Z_1)^{-1}(Z_1 + Z_2))c^n \\ &= (I - 1/2Z_2)^{-1}(I - 1/2Z_1)^{-1}((I - 1/2Z_1)(I - 1/2Z_2) + Z_1 + Z_2)c^n \\ &= (I - 1/2Z_2)^{-1}(I - 1/2Z_1)^{-1}(I + 1/2Z_1)(I + 1/2Z_2)c^n \\ c_1^{n+1} &= R_1(Z_1, Z_2)c^n. \end{aligned}$$

The problem is that the stability function $R_1(Z_1, Z_2)$ is not stable for Z_2 , hence it is a combination of implicit Euler for Z_2 , CN for Z_1 , and explicit Euler for Z_2 .

We can only have the stability for Z_2 in the explicit case, i.e. we don't have an A-stable method.

To improve this method we suggest to do a prestepping for c_0^n , which means that we define c_0^n from the known value c^n with a suitably chosen stable method. Namely, we suggest the following algorithm.

- We apply the sequential splitting for the problem (16) on interval $[t_n, t_{n+1}]$, two times on the half interval, consecutively.
- To both sub-problems in the first splitting we apply the implicit Euler method.

- In the second splitting for the first sub-problem (with operator Z_1) we apply the explicit Euler and to the second sub-problem (with operator Z_2) we apply the implicit Euler method.

We get

$$c_0^n = (I - 0.5Z_2)^{-1}(I - 0.5Z_1)^{-1}(I + 0.5Z_1)(I - 0.5Z_2)^{-1}c^n, \quad (35)$$

$$c_0^n = R_2(Z_1, Z_2)c^n. \quad (36)$$

Hence,

$$c_1^{n+1} = R_1(Z_1, Z_2)R_2(Z_1, Z_2)c^n \quad (37)$$

$$= (I - 0.5Z_2)^{-1}(I - 0.5Z_1)^{-1}(I + 0.5Z_1)(I + 0.5Z_2) \quad (38)$$

$$(I - 0.5Z_2)^{-1}(I - 0.5Z_1)^{-1}(I + 0.5Z_1)(I - 0.5Z_2)^{-1}c^n$$

$$= R_{IE}(0.5Z_2)R_{CN}(Z_1)R_{CN}(Z_2)R_{CN}(Z_1)R_{IE}(0.5Z_2)c^n,$$

where R_{IE} and R_{CN} are the stability functions of implicit Euler and Crank-Nicolson method.

To improve this method we can do a prestepping for c^n with a stable method and complete Z_2 to a stable CN-method, we start with $c^{n-1/2}$, with starting point $1/2\tau$.

$$c^n = (I - 1/2Z_2)^{-1}(I - 1/2Z_1)^{-1}c^{n-1/2}, \quad (39)$$

$$c^n = R_2(Z_1, Z_2)c_0. \quad (40)$$

Hence, we get:

$$c_1^{n+1} = R_1(Z_1, Z_2)R_2(Z_2)c^{n-1/2} \quad (41)$$

$$= (I - 1/2Z_2)^{-1}(I - 1/2Z_1)^{-1}(I + 1/2Z_1)(I + 1/2Z_2) \quad (42)$$

$$(I - 1/2Z_2)^{-1}(I - 1/2Z_1)^{-1}c^{n-1/2}$$

$$= R_{impl.Euler}(1/2Z_2)R_{CN}(Z_1)R_{CN}(Z_2)R_{impl.Euler}(1/2Z_1)c^{n-1/2},$$

where $R_{impl.Euler}$ and R_{CN} are the stability functions of implicit Euler and Crank-Nicolson method.

For these prestepping we therefore have a stable method with implicit Euler and Crank-Nicolson methods.

5.2 Error analysis for the nonlinear method

Here we assume a linearization technique with iterative formulations. The transformation to a linear problem helps to consider the eigenvalue formulation for small time steps and weak nonlinear problems, see [25].

5.2.1 Linearization by iterative splitting method

Let us consider the following problem

$$\begin{aligned}\partial_t c &= A(c)c + B(c)c, \quad \text{for } (x, t) \in \Omega \times [0, T], \\ c(x, 0) &= c_0(x),\end{aligned}$$

where A, B are nonlinear differentiable bounded operators A, B in a Banach space \mathbf{X} .

We assume a convergent fixed-point-iterative scheme, which is used to linearize equation (43):

$$\partial_t c_i(x, t) = A(c_{i-1})c_i(x, t) + B(c_{i-1})c_i(x, t), \quad 0 < t \leq T, \quad (43)$$

$$c_0(x, t) = 0, c(x, 0) = c_0(x), \quad (44)$$

$$i = 1, 2, 3, \dots, \quad (45)$$

where we stop with $\|c_i - c_{i-1}\| \leq \text{err}$, $\text{err} \in \mathbb{R}^+$. We assume that $\tilde{A} = A(c_{i-1}), \tilde{B} = B(c_{i-1}) : \mathbf{X} \rightarrow \mathbf{X}$ are given, linear bounded operators for small time steps $\tau = t^{n+1} - t^n$, such that $c_{i-1}(t) \approx c_{i-1}(t^n)$ for all $t \in [t^n, t^{n+1}]$.

In the following we discuss the improved and stable iterative method, given in (7)-(8).

Theorem 5.3 *Let us consider the iterative method (43) with starting solution $c_1 = c_1(t^{n+1})$, which is of m -th order exact. We assume to estimate or operators \tilde{A}, \tilde{B} with the maximum eigenvalues λ_1 and λ_2 . Further we define $z_1 = \lambda_1\tau$, $z_2 = \lambda_2\tau$, where τ is the local time step.*

Then we can prove, that all successive iterative solutions of the iterative splitting method (7)-(8) are stable, see proof idea [25]. It holds

$$c_{i+1}(z_1, z_2) = c_{i+1}(z, -\infty) = 0 \leq 1, \quad i = 1, 2, \dots, \quad (46)$$

where $c_1(t) \in U$ and U is the solution space for the iterative solutions, with $c_i \rightarrow c$ for $i \rightarrow \infty$.

Proof. We can prove the stability of the linearized scheme following proof 5.1.

Remark 5.4 Here we have assumed moderate nonlinear operators and have taken into account the boundedness of the operators. For weaker assumptions, e.g. unbounded operators, we can apply the exponential integrators, see [21].

6 Numerical examples

6.1 Test example 1: Viscous Burgers equation

We deal with a 2D example where we can derive an analytical solution.

$$\partial_t c = -c\partial_x c - c\partial_y c + \mu(\partial_{xx}c + \partial_{yy}c) + f(x, y, t), \quad (47)$$

$$(x, y, t) \in \Omega \times [0, T]$$

$$c(x, y, 0) = c_{\text{ana}}(x, y, 0), \quad (x, y) \in \Omega \quad (48)$$

$$\text{with } c(x, y, t) = c_{\text{ana}}(x, y, t) \text{ on } \partial\Omega \times [0, T], \quad (49)$$

where $\Omega = [0, 1] \times [0, 1]$, $T = 1.25$, and μ is the viscosity.

The analytical solution is given as

$$c_{\text{ana}}(x, y, t) = (1 + \exp(\frac{x + y - t}{2\mu}))^{-1}, \quad (50)$$

where $f(x, y, t) = 0$.

The operators are given as:

$$\begin{aligned} A(c)c &= -c\partial_x c - c\partial_y c, \text{ hence } A(c) = -c\partial_x - c\partial_y \text{ (the nonlinear operator),} \\ Bc &= \mu(\partial_{xx}c + \partial_{yy}c) + f(x, y, t) \text{ (the linear operator).} \end{aligned}$$

We apply the nonlinear Algorithm 7 to the first equation and obtain

$$\begin{aligned} A(c_{i-1})c_i &= -c_{i-1}\partial_x c_i - c_{i-1}\partial_y c_i \text{ and} \\ Bc_{i-1} &= \mu(\partial_{xx} + \partial_{yy})c_{i-1} + f, \end{aligned}$$

and we obtain linear operators, because c_{i-1} is known from the previous time step.

In the second equation we obtain by using Algorithm 8:

$$\begin{aligned} A(c_{i-1})c_i &= -c_{i-1}\partial_x c_i - c_{i-1}\partial_y c_i \text{ and} \\ Bc_{i+1} &= \mu(\partial_{xx} + \partial_{yy})c_{i+1} + f, \end{aligned}$$

and we have also linear operators.

The maximal error at end time $t = T$ is given as

$$\text{err}_{\max} = |c_{\text{num}} - c_{\text{ana}}| = \max_{i=1}^p |c_{\text{num}}(x_i, y_i, t) - c_{\text{ana}}(x_i, y_i, t)|,$$

the numerical convergence rate is given as

$$\rho = \log(\text{err}_{h/2}/\text{err}_h)/\log(0.5).$$

$\Delta x = \Delta y$	Δt	err_{L_1}	err_{\max}	ρ_{L_1}	ρ_{\max}
0.125	0.125	0.13289	0.74383		
0.0625	0.125	0.089855	0.71891	0.56453	0.049156
0.03125	0.125	0.067359	0.61805	0.41574	0.2181
0.125	0.0625	0.17624	0.95448		
0.0625	0.0625	0.11871	0.95536	0.57009	-0.0013331
0.03125	0.0625	0.072207	0.85321	0.71724	0.16314
0.125	0.03125	0.2124	0.99166		
0.0625	0.03125	0.15298	0.99531	0.47349	-0.005293
0.03125	0.03125	0.09503	0.96871	0.68686	0.039073

Table 1: Numerical results for the Burgers equation with viscosity $\mu = 0.005$ using standard IOS method, initial condition $c_0(x, y, t) = c^n$, and four iterations per time step.

$\Delta x = \Delta y$	Δt	err_{L_1}	err_{\max}	ρ_{L_1}	ρ_{\max}
0.125	0.125	0.13261	0.7421		
0.0625	0.125	0.089137	0.71146	0.57308	0.060833
0.03125	0.125	0.066726	0.60535	0.41779	0.23302
0.125	0.0625	0.1759	0.95384		
0.0625	0.0625	0.11867	0.95522	0.56782	-0.0020951
0.03125	0.0625	0.072078	0.85195	0.71929	0.16506
0.125	0.03125	0.21236	0.99165		
0.0625	0.03125	0.15293	0.9953	0.47367	-0.0052973
0.03125	0.03125	0.095025	0.96869	0.68647	0.039086

Table 2: Numerical results for the Burgers equation with viscosity $\mu = 0.005$ using IOS method respecting eigenvalues, initial condition $c_0(x, y, t) = c^n$, and four iterations per time step.

We have the following results, see Tables 1 to 6, for different steps in time and space and different viscosities.

Figure 1 presents the profile of the 2D nonlinear Burgers equation.

Remark 6.1 In the examples, we have two different cases of μ , which smoothes our equation. In the first test we use a very small $\mu = 0.005$, such that we have a dominant hyperbolic behavior, due to this we have a loose in the regularity and sharp front. The iterative splitting method loses one order. In the second test, we have increased the smoothness with setting $\mu = 5$, we get a more parabolic behavior. We have shown that the results are improved to higher accuracy.

$\Delta x = \Delta y$	Δt	err $_{L_1}$	err $_{\max}$	ρ_{L_1}	ρ_{\max}
0.125	0.125	0.10446	0.65352		
0.0625	0.125	0.04442	0.46667	1.2336	0.48581
0.03125	0.125	0.032194	0.40191	0.46444	0.21553
0.125	0.0625	0.15974	0.94042		
0.0625	0.0625	0.084812	0.87705	0.91341	0.10064
0.03125	0.0625	0.02712	0.45108	1.6449	0.95929
0.125	0.03125	0.20487	0.99067		
0.0625	0.03125	0.13449	0.99256	0.60716	-0.0027481
0.03125	0.03125	0.061692	0.90457	1.1244	0.13393

Table 3: Numerical results for the Burgers equation with viscosity $\mu = 0.005$ using IOS and η -method respecting eigenvalues for $\eta = 0.25$, initial condition $c_0(x, y, t) = c^n$, and four iterations per time step.

$\Delta x = \Delta y$	Δt	err $_{L_1}$	err $_{\max}$	ρ_{L_1}	ρ_{\max}
0.125	0.125	$2.078 \cdot 10^{-8}$	$5.5031 \cdot 10^{-8}$		
0.0625	0.125	$1.1059 \cdot 10^{-8}$	$3.0482 \cdot 10^{-8}$	0.91001	0.85231
0.03125	0.125	$6.7341 \cdot 10^{-9}$	$2.4778 \cdot 10^{-8}$	0.71566	0.2989
0.125	0.0625	$2.0937 \cdot 10^{-8}$	$5.8173 \cdot 10^{-8}$		
0.0625	0.0625	$1.3356 \cdot 10^{-8}$	$3.7642 \cdot 10^{-8}$	0.64855	0.62802
0.03125	0.0625	$9.5076 \cdot 10^{-9}$	$2.8165 \cdot 10^{-8}$	0.49039	0.41841
0.125	0.03125	$1.8457 \cdot 10^{-8}$	$5.1464 \cdot 10^{-8}$		
0.0625	0.03125	$1.082 \cdot 10^{-8}$	$3.0148 \cdot 10^{-8}$	0.77038	0.77148
0.03125	0.03125	$7.0185 \cdot 10^{-9}$	$1.9705 \cdot 10^{-8}$	0.62452	0.61348

Table 4: Numerical results for the Burgers equation with viscosity $\mu = 5$ using standard IOS method, initial condition $c_0(x, y, t) = c^n$, and four iterations per time step.

6.2 Test example 2: mixed convection-diffusion and Burgers equation

We deal with a 2D example which is a mixture of a convection-diffusion and Burgers equation. We can derive an analytical solution.

$$\partial_t c = -1/2c\partial_x c - 1/2c\partial_y c - 1/2\partial_x c - 1/2\partial_y c + \mu(\partial_{xx} c + \partial_{yy} c) + f(x, y, t), \quad (x, y, t) \in \Omega \times [0, T] \quad (51)$$

$$c(x, y, 0) = c_{\text{anal}}(x, y, 0), \quad (x, y) \in \Omega \quad (52)$$

$$\text{with } c(x, y, t) = c_{\text{anal}}(x, y, t) \text{ on } \partial\Omega \times [0, T], \quad (53)$$

where $\Omega = [0, 1] \times [0, 1]$, $T = 1.25$, and μ is the viscosity.

$\Delta x = \Delta y$	Δt	err $_{L_1}$	err $_{\max}$	ρ_{L_1}	ρ_{\max}
0.125	0.125	2.078 $\cdot 10^{-8}$	5.5031 $\cdot 10^{-8}$		
0.0625	0.125	1.1059 $\cdot 10^{-8}$	3.0482 $\cdot 10^{-8}$	0.91001	0.85231
0.03125	0.125	6.7341 $\cdot 10^{-9}$	2.4778 $\cdot 10^{-8}$	0.71566	0.2989
0.125	0.0625	2.0937 $\cdot 10^{-8}$	5.8173 $\cdot 10^{-8}$		
0.0625	0.0625	1.3356 $\cdot 10^{-8}$	3.7642 $\cdot 10^{-8}$	0.64855	0.62802
0.03125	0.0625	9.5076 $\cdot 10^{-9}$	2.8165 $\cdot 10^{-8}$	0.49039	0.41841
0.125	0.03125	1.8457 $\cdot 10^{-8}$	5.1464 $\cdot 10^{-8}$		
0.0625	0.03125	1.082 $\cdot 10^{-8}$	3.0148 $\cdot 10^{-8}$	0.77038	0.77148
0.03125	0.03125	7.0185 $\cdot 10^{-9}$	1.9705 $\cdot 10^{-8}$	0.62452	0.61348

Table 5: Numerical results for the Burgers equation with viscosity $\mu = 5$ using IOS method respecting eigenvalues, initial condition $c_0(x, y, t) = c^n$, and four iterations per time step.

$\Delta x = \Delta y$	Δt	err $_{L_1}$	err $_{\max}$	ρ_{L_1}	ρ_{\max}
0.125	0.125	1.6861 $\cdot 10^{-8}$	4.3419 $\cdot 10^{-8}$		
0.0625	0.125	9.0139 $\cdot 10^{-9}$	2.6411 $\cdot 10^{-8}$	0.90349	0.71716
0.03125	0.125	5.4384 $\cdot 10^{-9}$	2.0332 $\cdot 10^{-8}$	0.72896	0.37739
0.125	0.0625	1.7447 $\cdot 10^{-8}$	4.8947 $\cdot 10^{-8}$		
0.0625	0.0625	1.0143 $\cdot 10^{-8}$	2.8656 $\cdot 10^{-8}$	0.78247	0.77236
0.03125	0.0625	6.4831 $\cdot 10^{-9}$	1.8994 $\cdot 10^{-8}$	0.64574	0.59333
0.125	0.03125	1.6513 $\cdot 10^{-8}$	4.6123 $\cdot 10^{-8}$		
0.0625	0.03125	8.9812 $\cdot 10^{-9}$	2.5026 $\cdot 10^{-8}$	0.87863	0.88204
0.03125	0.03125	5.2507 $\cdot 10^{-9}$	1.4699 $\cdot 10^{-8}$	0.77441	0.76773

Table 6: Numerical results for the Burgers equation with viscosity $\mu = 5$ using IOS and η -method respecting eigenvalues for $\eta = 0.25$, initial condition $c_0(x, y, t) = c^n$, and four iterations per time step.

The analytical solution is given as

$$c_{\text{anal}}(x, y, t) = (1 + \exp(\frac{x + y - t}{2\mu}))^{-1} + \exp(\frac{x + y - t}{2\mu}), \quad (54)$$

where we compute $f(x, y, t)$ accordingly.

We split the convection-diffusion and the Burgers equation. The operators are given as:

$$\begin{aligned} A(c)c &= -1/2c\partial_x c - 1/2c\partial_y c + 1/2\mu(\partial_{xx}c + \partial_{yy}c), \text{ hence} \\ A(c) &= -1/2c\partial_x - 1/2c\partial_y + 1/2\mu(\partial_{xx} + \partial_{yy}) \text{ (the Burgers term), and} \end{aligned}$$

$Bc = -1/2\partial_x c - 1/2\partial_y c + 1/2\mu(\partial_{xx}c + \partial_{yy}c) + f(x, y, t)$ (the convection-diffusion term).

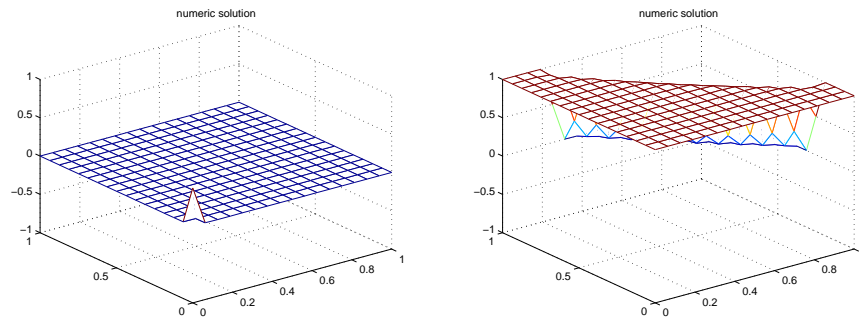


Figure 1: Burgers equation at initial time $t = 0.0$ (left figure) and end time $t = 1.25$ (right figure) for viscosity $\mu = 0.005$.

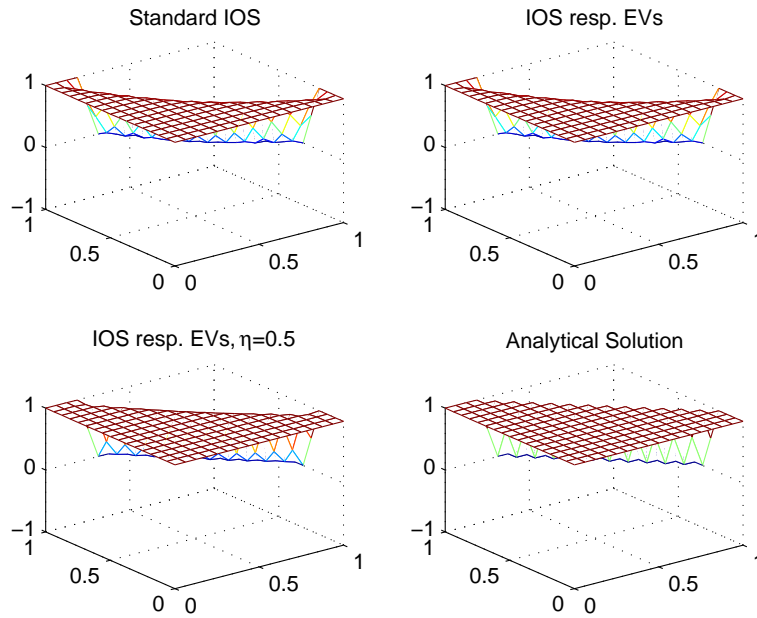


Figure 2: Comparison of the solutions of three different methods to the exact solution for viscous Burgers equation using viscosity $\mu = 0.005$. The three compared methods are the standard iterative operator-splitting (IOS), the IOS method respecting the stiffness (eigenvalues) of the operators A and B , as well as the modified last method using the η -method with $\eta = 0.5$.

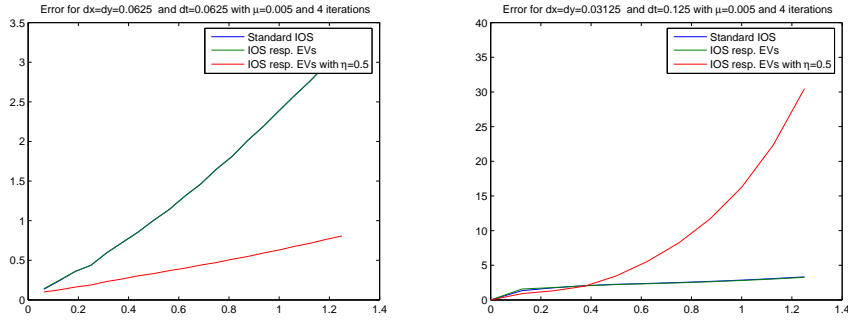


Figure 3: Errors of the solutions of the three different methods for viscous Burgers equation using viscosity $\mu = 0.005$. The third method, the IOS-method respecting eigenvalues using the η -method with $\eta = 0.5$, depends on the CFL-condition. The left figure shows a case, where the CFL-condition is satisfied, in the right figure the two other methods show much better results.

$\eta = 0.05$	$\Delta x = 1/4$	$\Delta x = 1/8$	$\Delta x = 1/16$	$\eta = 0.25$	$\Delta x = 1/4$	$\Delta x = 1/8$	$\Delta x = 1/16$
$\Delta t = 1/4$	2	1	1	$\Delta t = 1/4$	2	1	1
$\Delta t = 1/8$	2	1	1	$\Delta t = 1/8$	2	1	1
$\Delta t = 1/16$	2	2	1	$\Delta t = 1/16$	2	1	1
$\Delta t = 1/32$	2	2	1	$\Delta t = 1/32$	2	1	1
$\eta = 0.5$	$\Delta x = 1/4$	$\Delta x = 1/8$	$\Delta x = 1/16$	$\eta = 0.75$	$\Delta x = 1/4$	$\Delta x = 1/8$	$\Delta x = 1/16$
$\Delta t = 1/4$	1	3	3	$\Delta t = 1/4$	3	4	4
$\Delta t = 1/8$	2	1	3	$\Delta t = 1/8$	1	4	4
$\Delta t = 1/16$	2	1	1	$\Delta t = 1/16$	2	1	4
$\Delta t = 1/32$	2	1	1	$\Delta t = 1/32$	2	1	1

Table 7: Numerical results for the Burgers equation with viscosity $\mu = 0.005$. The third method using IOS method respecting eigenvalues coupled with the η -method is compared to the two other methods. The numbering is declared as following.

- 1: η -method yields much better results
- 2: η -method yields slightly better or the same results
- 3: η -method yields wrong results
- 4: η -method is instable

For the first equation we apply the nonlinear Algorithm 7 and obtain

$$A(c_{i-1})c_i = -1/2c_{i-1}\partial_x c_i - 1/2c_{i-1}\partial_y c_i + 1/2\mu(\partial_{xx}c_i + \partial_{yy}c_i) \text{ and}$$

$$Bc_{i-1} = 1/2(-\partial_x - \partial_y + \mu(\partial_{xx} + \partial_{yy}))c_{i-1},$$

and we obtain linear operators, because c_{i-1} is known from the previous time step.

In the second equation we obtain by using Algorithm 8:

$$A(c_{i-1})c_i = -1/2c_{i-1}\partial_x c_i - 1/2c_{i-1}\partial_y c_i + 1/2\mu(\partial_{xx}c_i + \partial_{yy}c_i) \text{ and}$$

$$Bc_{i+1} = 1/2(-\partial_x - \partial_y + \mu(\partial_{xx} + \partial_{yy}))c_{i+1},$$

and we have linear operators.

We deal with different viscosities μ as well as different step sizes in time and space. We have the following results, see Tables 8 to 13.

$\Delta x = \Delta y$	Δt	err_{L_1}	err_{\max}	ρ_{L_1}	ρ_{\max}
0.125	0.0625	0.0074702	0.018725		
0.0625	0.0625	0.0026137	0.0065412	1.5151	1.5173
0.03125	0.0625	0.00038015	0.00113	2.7814	2.5332
0.125	0.03125	0.0083789	0.020832		
0.0625	0.03125	0.0035067	0.008635	1.2567	1.2705
0.03125	0.03125	0.0012581	0.0031015	1.4788	1.4772

Table 8: Numerical results for the mixed convection-diffusion and Burgers equation with viscosity $\mu = 0.5$ using standard IOS method, initial condition $c_0(x, y, t) = c^n$, and four iterations per time step.

$\Delta x = \Delta y$	Δt	err_{L_1}	err_{\max}	ρ_{L_1}	ρ_{\max}
0.125	0.0625	0.0074702	0.018725		
0.0625	0.0625	0.0026137	0.0065412	1.5151	1.5173
0.03125	0.0625	0.00038015	0.00113	2.7814	2.5332
0.125	0.03125	0.0083789	0.020832		
0.0625	0.03125	0.0035067	0.008635	1.2567	1.2705
0.03125	0.03125	0.0012581	0.0031015	1.4788	1.4772

Table 9: Numerical results for the mixed convection-diffusion and Burgers equation with viscosity $\mu = 0.5$ using IOS method respecting eigenvalues, initial condition $c_0(x, y, t) = c^n$, and four iterations per time step.

Figure 4 presents the profile of the 2D mixed convection-diffusion and Burgers equation.

$\Delta x = \Delta y$	Δt	err $_{L_1}$	err $_{\max}$	ρ_{L_1}	ρ_{\max}
0.125	0.0625	0.0083005	0.021016		
0.0625	0.0625	0.0034255	0.0086649	1.2769	1.2783
0.03125	0.0625	0.0011728	0.0031685	1.5464	1.4514
0.125	0.03125	0.0087833	0.021955		
0.0625	0.03125	0.003901	0.0096743	1.1709	1.1823
0.03125	0.03125	0.0016446	0.0041048	1.2461	1.2369

Table 10: Numerical results for the mixed convection-diffusion and Burgers equation with viscosity $\mu = 0.5$ using IOS and η -method respecting eigenvalues for $\eta = 0.25$, initial condition $c_0(x, y, t) = c^n$, and four iterations per time step.

$\Delta x = \Delta y$	Δt	err $_{L_1}$	err $_{\max}$	ρ_{L_1}	ρ_{\max}
0.125	0.0625	$7.7362 \cdot 10^{-6}$	$1.8004 \cdot 10^{-5}$		
0.0625	0.0625	$2.5848 \cdot 10^{-6}$	$6.8215 \cdot 10^{-6}$	1.5816	1.4002
0.03125	0.0625	$7.341 \cdot 10^{-7}$	$1.8387 \cdot 10^{-6}$	1.816	1.8914
0.125	0.03125	$9.2036 \cdot 10^{-6}$	$2.0227 \cdot 10^{-5}$		
0.0625	0.03125	$4.1362 \cdot 10^{-6}$	$9.0405 \cdot 10^{-6}$	1.1539	1.1618
0.03125	0.03125	$1.4913 \cdot 10^{-6}$	$3.4521 \cdot 10^{-6}$	1.4717	1.3889

Table 11: Numerical results for the mixed convection-diffusion and Burgers equation with viscosity $\mu = 5$ using standard IOS method, initial condition $c_0(x, y, t) = c^n$, and four iterations per time step.

$\Delta x = \Delta y$	Δt	err $_{L_1}$	err $_{\max}$	ρ_{L_1}	ρ_{\max}
0.125	0.0625	$7.7362 \cdot 10^{-6}$	$1.8004 \cdot 10^{-5}$		
0.0625	0.0625	$2.5848 \cdot 10^{-6}$	$6.8215 \cdot 10^{-6}$	1.5816	1.4002
0.03125	0.0625	$7.341 \cdot 10^{-7}$	$1.8387 \cdot 10^{-6}$	1.816	1.8914
0.125	0.03125	$9.2036 \cdot 10^{-6}$	$2.0227 \cdot 10^{-5}$		
0.0625	0.03125	$4.1362 \cdot 10^{-6}$	$9.0405 \cdot 10^{-6}$	1.1539	1.1618
0.03125	0.03125	$1.4913 \cdot 10^{-6}$	$3.4521 \cdot 10^{-6}$	1.4717	1.3889

Table 12: Numerical results for the mixed convection-diffusion and Burgers equation with viscosity $\mu = 5$ using IOS method respecting eigenvalues, initial condition $c_0(x, y, t) = c^n$, and four iterations per time step.

Remark 6.2 In the examples, we deal with more iteration steps to obtain higher-order convergence results. In the first test we have four iterative steps but a smaller viscosity ($\mu = 0.5$), such that we can reach at least a second-order method. In the second test we use a high viscosity about $\mu = 5$ and get the second-order result with two iteration steps. Here we see the loose of differentiability. To obtain the same results, we have to increase the number of

$\Delta x = \Delta y$	Δt	err_{L_1}	err_{\max}	ρ_{L_1}	ρ_{\max}
0.125	0.0625	$8.8891 \cdot 10^{-6}$	$2.0159 \cdot 10^{-5}$		
0.0625	0.0625	$3.7397 \cdot 10^{-6}$	$8.9814 \cdot 10^{-6}$	1.2491	1.1664
0.03125	0.0625	$1.3079 \cdot 10^{-6}$	$3.3957 \cdot 10^{-6}$	1.5156	1.4032
0.125	0.03125	$9.7002 \cdot 10^{-6}$	$2.123 \cdot 10^{-5}$		
0.0625	0.03125	$4.6593 \cdot 10^{-6}$	$1.0043 \cdot 10^{-5}$	1.0579	1.08
0.03125	0.03125	$2.0105 \cdot 10^{-6}$	$4.4566 \cdot 10^{-6}$	1.2126	1.1721

Table 13: Numerical results for the mixed convection-diffusion and Burgers equation with viscosity $\mu = 5$ using IOS and η -method respecting eigenvalues for $\eta = 0.25$, initial condition $c_0(x, y, t) = c^n$, and four iterations per time step.

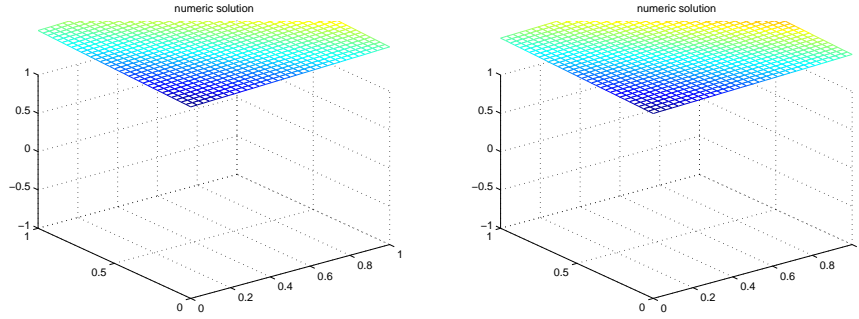


Figure 4: Mixed convection-diffusion and Burgers equation at initial time $t = 0.0$ (left figure) and end time $t = 1.25$ (right figure) for viscosity $\mu = 0.5$.

iteration steps. So we could show an improvement of the convergence order with respect to the iteration steps.

6.3 Test example 3: momentum equation (molecular flow)

We deal with an example of a momentum equation, that is used to model the viscous flow of a fluid.

$$\partial_t \mathbf{c} = -\mathbf{c} \cdot \nabla \mathbf{c} + 2\mu \nabla(D(\mathbf{c}) + 1/3 \nabla \mathbf{c}) + \mathbf{f}(x, y, t), \quad (x, y, t) \in \Omega \times [0, T] \quad (56)$$

$$\mathbf{c}(x, y, 0) = \mathbf{c}_0(x, y), \quad (x, y) \in \Omega$$

$$\text{with } \mathbf{c}(x, y, t) = \mathbf{c}_{\text{ana}}(x, y, t) \text{ on } \partial\Omega \times [0, T] \text{ (enclosed flow)}, \quad (57)$$

where $\mathbf{c} = (c_1, c_2)^t$ is the solution and $\Omega = [0, 1] \times [0, 1]$, $T = 1.25$, $\mu = 5$, and $\mathbf{v} = (0.001, 0.001)^t$ are the parameters and I is the unit matrix.

The nonlinear function $D(\mathbf{c}) = \mathbf{c} \cdot \mathbf{c} + \mathbf{v} \cdot \mathbf{c}$ is the viscosity flow, and \mathbf{v} is a constant velocity.

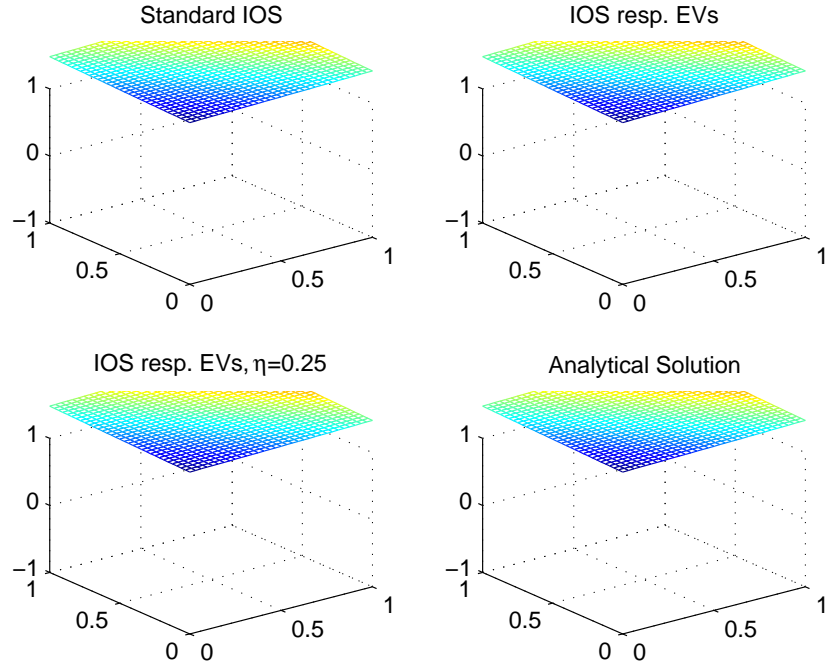


Figure 5: Comparison of the solutions of three different methods to the exact solution for mixed convection-diffusion and Burgers equation using viscosity $\mu = 0.5$. The three compared methods are the standard iterative operator-splitting (IOS), the IOS method respecting the stiffness (eigenvalues) of the operators A and B , as well as the modified last method using the η -method.

We can derive the analytical solution with respect to the first two test examples with the functions:

$$c_{1,\text{ana}}(x, y, t) = (1 + \exp(\frac{x+y-t}{2\mu}))^{-1} + \exp(\frac{x+y-t}{2\mu}), \quad (58)$$

$$c_{2,\text{ana}}(x, y, t) = (1 + \exp(\frac{x+y-t}{2\mu}))^{-1} + \exp(\frac{x+y-t}{2\mu}). \quad (59)$$

For the splitting method our operators are given as:

$$A(\mathbf{c})\mathbf{c} = -\mathbf{c}\nabla\mathbf{c} + 2\mu\nabla D(\mathbf{c}) \text{ (the nonlinear operator), and} \\ B\mathbf{c} = 2/3\mu\Delta\mathbf{c} \text{ (the linear operator).}$$

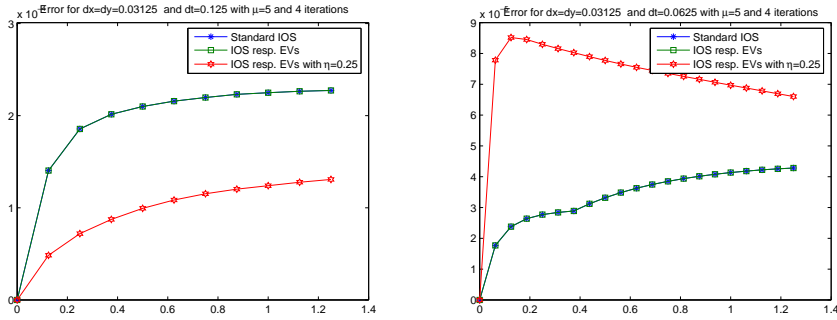


Figure 6: Errors of the solutions of the three different methods for mixed convection-diffusion and Burgers equation using viscosity $\mu = 0.5$. The third method, the IOS-method respecting eigenvalues using the η -method, depends on the CFL-condition. The left figure shows a case, where the CFL-condition is satisfied, in the right figure the two other methods show much better results.

We first deal with the one-dimensional case,

$$\partial_t c = -c \cdot \partial_x c + 2\mu \partial_x (D(c) + 1/3 \partial_x c) + f(x, t), \quad (x, t) \in \Omega \times [0, T] \quad (60)$$

$$c(x, 0) = c_0(x), \quad (x) \in \Omega \quad (61)$$

$$\text{with } c(x, t) = c_{\text{ana}}(x, t) \text{ on } \partial\Omega \times [0, T] \text{ (enclosed flow),} \quad (62)$$

where c is the solution and $\Omega = [0, 1]$, $T = 1.25$, $\mu = 5$, and $v = 0.001$ are the parameters.

Then the operators are given as:

$$A(c)c = -c \partial_x c + 2\mu \partial_x D(c) \text{ (the nonlinear operator), and}$$

$$Bc = 2/3 \mu \partial_{xx} c \text{ (the linear operator).}$$

We have the following results for our three different methods, see Tables 14 to 16.

Figure 7 presents the profile of the 1D momentum equation.

We have the following results for the 2D case, see Tables 17 to 20. The η -method showed best results for $\eta = 0$. Since this yields the iterative operator-splitting method respecting eigenvalues, only the first and second method are compared.

Figure 10 presents the profile of the 2D momentum equation.

Remark 6.3 In the more realistic examples of a 1D and 2D momentum equations, we can also observe the stiffness problem, which we obtain with a more hyperbolic behavior. In the 1D experiments we deal with a more hyperbolic

Δx	Δt	err_{L_1}	err_{\max}	ρ_{L_1}	ρ_{\max}
0.125	0.0625	$3.1411 \cdot 10^{-6}$	$7.5033 \cdot 10^{-6}$		
0.0625	0.0625	$2.6405 \cdot 10^{-6}$	$6.2463 \cdot 10^{-6}$	0.25046	0.26452
0.03125	0.0625	$2.3534 \cdot 10^{-6}$	$5.5714 \cdot 10^{-6}$	0.1661	0.16497
0.125	0.03125	$2.3422 \cdot 10^{-6}$	$5.6556 \cdot 10^{-6}$		
0.0625	0.03125	$1.8041 \cdot 10^{-6}$	$4.3984 \cdot 10^{-6}$	0.37662	0.36268
0.03125	0.03125	$1.4887 \cdot 10^{-6}$	$3.6586 \cdot 10^{-6}$	0.27721	0.26572

Table 14: Numerical results for the one-dimensional momentum equation with viscosity $\mu = 50$ and $v = 0.001$ using standard IOS method, initial condition $c_0(x, t) = c^n$, and four iterations per time step.

Δx	Δt	err_{L_1}	err_{\max}	ρ_{L_1}	ρ_{\max}
0.125	0.0625	$1.8713 \cdot 10^{-6}$	$5.0315 \cdot 10^{-6}$		
0.0625	0.0625	$8.1354 \cdot 10^{-7}$	$2.1334 \cdot 10^{-6}$	1.2017	1.2379
0.03125	0.0625	$2.9252 \cdot 10^{-7}$	$7.6796 \cdot 10^{-7}$	1.4757	1.474
0.125	0.03125	$2.0472 \cdot 10^{-6}$	$8.8496 \cdot 10^{-6}$		
0.0625	0.03125	$9.5555 \cdot 10^{-7}$	$3.3847 \cdot 10^{-6}$	1.0992	1.3866
0.03125	0.03125	$4.1632 \cdot 10^{-7}$	$1.3826 \cdot 10^{-6}$	1.1986	1.2917

Table 15: Numerical results for the one-dimensional momentum equation with viscosity $\mu = 50$ and $v = 0.001$ using IOS method respecting eigenvalues, initial condition $c_0(x, t) = c^n$, and four iterations per time step.

Δx	Δt	err_{L_1}	err_{\max}	ρ_{L_1}	ρ_{\max}
0.125	0.0625	$1.3151 \cdot 10^{-6}$	$3.8348 \cdot 10^{-6}$		
0.0625	0.0625	$2.9348 \cdot 10^{-7}$	$8.295 \cdot 10^{-7}$	2.1639	2.2088
0.03125	0.0625	$2.0942 \cdot 10^{-7}$	$5.7202 \cdot 10^{-7}$	0.48687	0.53618
0.125	0.03125	$1.8517 \cdot 10^{-6}$	$8.9997 \cdot 10^{-6}$		
0.0625	0.03125	$7.0537 \cdot 10^{-7}$	$2.641 \cdot 10^{-6}$	1.3924	1.7688
0.03125	0.03125	$1.5864 \cdot 10^{-7}$	$5.5162 \cdot 10^{-7}$	2.1526	2.2593

Table 16: Numerical results for the one-dimensional momentum equation with viscosity $\mu = 50$ and $v = 0.001$ using IOS and η -method respecting eigenvalues for $\eta = 0.25$, initial condition $c_0(x, t) = c^n$, and four iterations per time step.

behavior and obtain at least first-order convergence with 2 iteration steps. In the 2D experiments we obtain nearly second-order convergence results with 2 iteration steps, if we increase the parabolic behavior, e.g. use larger μ and \mathbf{v} values. For such methods, we have to balance the usage of the iteration steps, refinement in time and space with respect to the hyperbolicity of the equations. At least we can obtain a second-order method with more than 2 iteration steps.

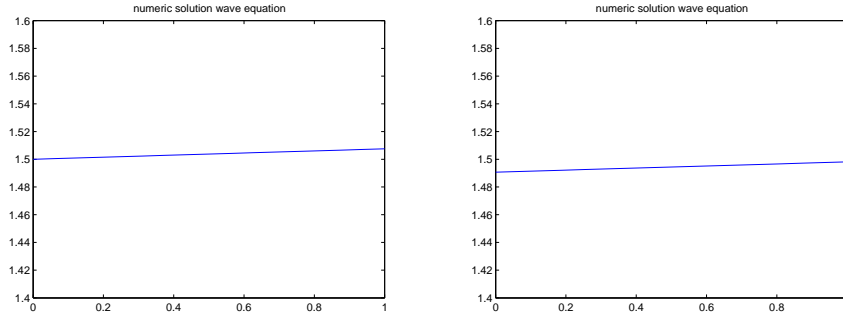


Figure 7: One-dimensional momentum equation at initial time $t = 0.0$ (left figure) and end time $t = 1.25$ (right figure) for viscosity $\mu = 50$ and $v = 0.001$.

$\Delta x = \Delta y$	Δt	err_{L_1}	err_{\max}	ρ_{L_1}	ρ_{\max}
0.2	0.0625	$2.0878 \cdot 10^{-5}$	$4.4078 \cdot 10^{-5}$		
0.1	0.0625	$6.4556 \cdot 10^{-6}$	$1.3052 \cdot 10^{-5}$	1.6934	1.7558
0.05	0.0625	$2.4956 \cdot 10^{-6}$	$4.4442 \cdot 10^{-6}$	1.3712	1.5542
0.2	0.03125	$2.9816 \cdot 10^{-5}$	$5.2392 \cdot 10^{-5}$		
0.1	0.03125	$1.4577 \cdot 10^{-5}$	$2.283 \cdot 10^{-5}$	1.0324	1.1984
0.05	0.03125	$4.0646 \cdot 10^{-6}$	$6.6101 \cdot 10^{-6}$	1.8425	1.7882
0.2	0.015625	$3.1298 \cdot 10^{-5}$	$5.4969 \cdot 10^{-5}$		
0.1	0.015625	$1.7869 \cdot 10^{-5}$	$2.6208 \cdot 10^{-5}$	0.80858	1.0686
0.05	0.015625	$7.7609 \cdot 10^{-6}$	$1.1286 \cdot 10^{-5}$	1.2032	1.2154

Table 17: Numerical results for the two-dimensional momentum equation for the first component with $\mu = 50$ and $v = (100, 0.01)^T$ using standard IOS method, initial condition $c_0(x, y, t) = c^n$, and four iterations per time step.

The stiffness influence the number of iterative steps.

7 Conclusions and Discussions

We present a new method to solve complicated mixed coupled partial differential equations. Based on a standard method we derive different new methods and reorder the operators for different scales. Such a reordering reduces the decomposition error. The more hyperbolic behavior of the equations leads to an increasement of the iteration steps of our method. At least we obtain a second-order method. Such iterative splitting method can balance the different behavior of the underlying operators. So the one operator smoothes the solution process, while the other operator decreases the smoothness. Further a balance between the implicit and explicit discretization with the iterative splitting method is a

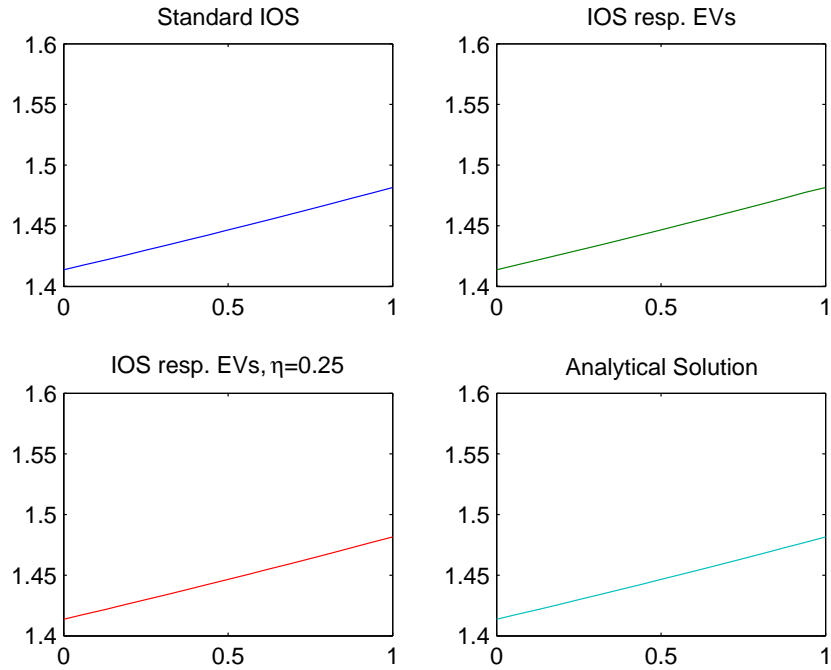


Figure 8: Comparison of the solutions of three different methods to the exact solution for one-dimensional momentum equation using viscosity $\mu = 50$ and $v = 0.001$. The three compared methods are the standard iterative operator-splitting (IOS), the IOS method respecting the stiffness (eigenvalues) of the operators A and B , as well as the modified last method using the η -method.

new method that overcomes to the mixed behavior in an unsplit method.

References

- [1] U.M. Ascher, St.J. Ruuth and B.T.R. Wetton. *Implicit-explicit methods for time-dependent partial differential equations*. SIAM Journal on Numerical Analysis, Volume 32, Issue 3, 797 - 823, 1995.
- [2] Z. S. Alterman and A. Rotenberg. *Seismic Waves in a Quarter Plane*. Bulletin of the Seismological Society of America, 59:347-368, 1969
- [3] W. Bangerth and R. Rannacher. *Adaptive finite Element Methods for differential equations*. Lectures in Mathematics, ETH Zürich, Birkhäuser Verlag, Basel-Boston-Berlin, 2003.

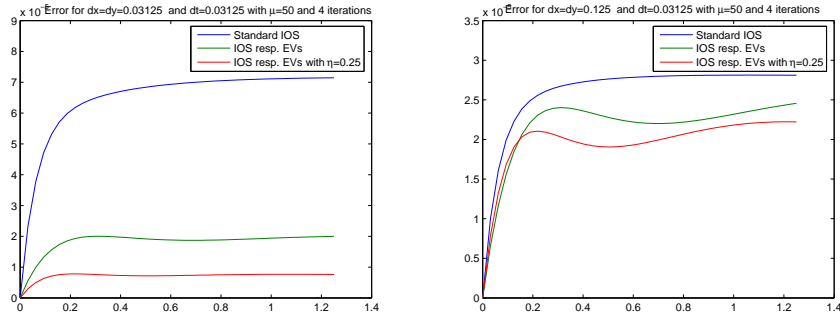


Figure 9: Errors of the solutions of the three different methods for one-dimensional momentum equation using viscosity $\mu = 50$ and $v = 0.001$. The third method, the IOS-method repeating eigenvalues using the η -method, depends on the CFL-condition. The left figure shows a case, where the CFL-condition is satisfied, in the right figure the methods show worse results.

$\Delta x = \Delta y$	Δt	err_{L_1}	err_{\max}	ρ_{L_1}	ρ_{\max}
0.2	0.0625	$4.3928 \cdot 10^{-5}$	$1.6352 \cdot 10^{-4}$		
0.1	0.0625	$3.34 \cdot 10^{-5}$	$7.8335 \cdot 10^{-5}$	0.39529	1.0617
0.05	0.0625	$2.6183 \cdot 10^{-5}$	$5.9027 \cdot 10^{-5}$	0.35124	0.40828
0.2	0.03125	$3.5311 \cdot 10^{-5}$	$2.3204 \cdot 10^{-4}$		
0.1	0.03125	$1.8705 \cdot 10^{-5}$	$9.7831 \cdot 10^{-5}$	0.91669	1.246
0.05	0.03125	$1.5796 \cdot 10^{-5}$	$3.7646 \cdot 10^{-5}$	0.24385	1.3778
0.2	0.015625	$1.0002 \cdot 10^{-4}$	$3.3894 \cdot 10^{-4}$		
0.1	0.015625	$3.6053 \cdot 10^{-5}$	$1.3972 \cdot 10^{-4}$	1.4721	1.2785
0.05	0.015625	$1.2837 \cdot 10^{-5}$	$4.7879 \cdot 10^{-5}$	1.4898	1.5451

Table 18: Numerical results for the two-dimensional momentum equation for the second component with $\mu = 50$ and $v = (100, 0.01)^T$ using standard IOS method, initial condition $c_0(x, y, t) = c^n$, and four iterations per time step.

- [4] V. Barbu. *Partial Differential Equations and Boundary value Problems*. Mathematics and its Applications, Kluwer Academic Publishers, Dordrecht, Boston, London, 1998.
- [5] X.C. Cai, Additive Schwarz algorithms for parabolic convection-diffusion equations, *Numer. Math.* 60 (1991) 41-61.
- [6] X.C. Cai, Multiplicative Schwarz methods for parabolic problems, *SIAM J. Sci Comput.* 15 (1994) 587-603.
- [7] W. Cheney, *Analysis for Applied Mathematics*, Graduate Texts in Mathematics., 208, Springer, New York, Berlin, Heidelberg, 2001.

$\Delta x = \Delta y$	Δt	err_{L_1}	err_{\max}	ρ_{L_1}	ρ_{\max}
0.2	0.0625	$3.9803 \cdot 10^{-6}$	$1.0521 \cdot 10^{-5}$		
0.1	0.0625	$2.5718 \cdot 10^{-6}$	$5.8651 \cdot 10^{-6}$	0.63014	0.84301
0.05	0.0625	$1.4105 \cdot 10^{-6}$	$3.0582 \cdot 10^{-6}$	0.86653	0.93946
0.2	0.03125	$4.5912 \cdot 10^{-6}$	$1.1424 \cdot 10^{-5}$		
0.1	0.03125	$3.0444 \cdot 10^{-6}$	$6.3743 \cdot 10^{-6}$	0.59269	0.84178
0.05	0.03125	$1.6933 \cdot 10^{-6}$	$3.3204 \cdot 10^{-6}$	0.84629	0.9409
0.2	0.015625	$4.6502 \cdot 10^{-6}$	$1.1527 \cdot 10^{-5}$		
0.1	0.015625	$3.1881 \cdot 10^{-6}$	$6.1458 \cdot 10^{-6}$	0.54462	0.90732
0.05	0.015625	$1.8166 \cdot 10^{-6}$	$3.1873 \cdot 10^{-6}$	0.81143	0.94727

Table 19: Numerical results for the two-dimensional momentum equation for the first component with $\mu = 50$ and $v = (100, 0.01)^T$ using IOS method respecting eigenvalues, initial condition $c_0(x, y, t) = c^n$, and four iterations per time step.

$\Delta x = \Delta y$	Δt	err_{L_1}	err_{\max}	ρ_{L_1}	ρ_{\max}
0.2	0.0625	$5.8145 \cdot 10^{-5}$	$1.5265 \cdot 10^{-4}$		
0.1	0.0625	$3.9711 \cdot 10^{-5}$	$9.4621 \cdot 10^{-5}$	0.55013	0.69003
0.05	0.0625	$2.2987 \cdot 10^{-5}$	$5.2152 \cdot 10^{-5}$	0.78869	0.85942
0.2	0.03125	$5.8159 \cdot 10^{-5}$	$1.5264 \cdot 10^{-4}$		
0.1	0.03125	$3.9561 \cdot 10^{-5}$	$9.4029 \cdot 10^{-5}$	0.55591	0.69895
0.05	0.03125	$2.2898 \cdot 10^{-5}$	$5.1961 \cdot 10^{-5}$	0.78888	0.85567
0.2	0.015625	$5.8165 \cdot 10^{-5}$	$1.5265 \cdot 10^{-4}$		
0.1	0.015625	$3.9354 \cdot 10^{-5}$	$9.3866 \cdot 10^{-5}$	0.56364	0.70152
0.05	0.015625	$2.274 \cdot 10^{-5}$	$5.1838 \cdot 10^{-5}$	0.79124	0.85659

Table 20: Numerical results for the two-dimensional momentum equation for the second component with $\mu = 50$ and $v = (100, 0.01)^T$ using IOS method respecting eigenvalues, initial condition $c_0(x, y, t) = c^n$, and four iterations per time step.

- [8] St.M. Day et al. *Test of 3D elastodynamic codes: Final report for lifelines project 1A02*. Technical report, Pacific Earthquake Engineering Center, 2003.
- [9] C. N. Dawson, Q. Du, and D. F. Dupont, A finite Difference Domain Decomposition Algorithm for Numerical solution of the Heat Equation, *Mathematics of Computation* 57 (1991) 63-71.
- [10] P. Deuffhard, *Newton Methods for Nonlinear Problems*, Springer-Verlag, New-York, Berlin, 2004.
- [11] K.-J. Engel and R. Nagel, *One-Parameter Semigroups for Linear Evolution Equations*, Springer, New York, 2000.

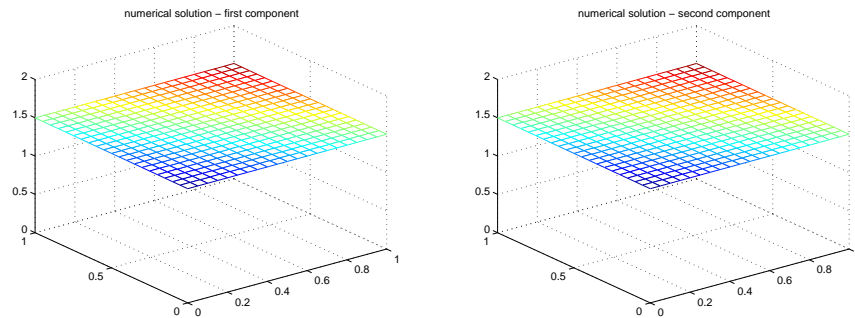


Figure 10: Two-dimensional momentum equation at end time $t = 1.25$ for viscosity $\mu = 50$ and $v = (100, 0.01)^T$.

- [12] I. Farago and J. Geiser, Iterative Operator-Splitting methods for Linear Problems, Preprint No. 1043 of Weierstrass Institute for Applied Analysis and Stochastics, Berlin, Germany. International Journal of Computational Science and Engineering, accepted September 2007.
- [13] M.J. Gander and H. Zhao, Overlapping Schwarz waveform relaxation for parabolic problems in higher dimension, In A. Handlovičová, Magda Komorníková, and Karol Mikula, editors, in: Proc. Algoritmy 14, Slovak Technical University, 1997, pp. 42-51.
- [14] J. Geiser, Discretisation Methods with embedded analytical solutions for convection dominated transport in porous media, in: Proc. NA&A '04, Lecture Notes in Computer Science, Vol.3401, Springer, Berlin, 2005, pp. 288-295.
- [15] J. Geiser, Operator-Splitting Methods Respecting Eigenvalue Problems for Convection-Diffusion and Wave Equations. International Journal of Applied Mathematics and Mechanics (IJAMM), Hong Kong, China, accepted April, 2008.
- [16] J. Geiser, Iterative Operator-Splitting Methods with higher order Time-Integration Methods and Applications for Parabolic Partial Differential Equations, J. Comput. Appl. Math., accepted, June 2007.
- [17] J. Geiser, O. Klein, and P. Philip, Influence of anisotropic thermal conductivity in the apparatus insulation for sublimation growth of SiC: Numerical investigation of heat transfer. Crystal Growth & Design 6, 2021 - 2028, 2006.
- [18] E. Giladi and H. Keller, Space time domain decomposition for parabolic problems. Technical Report 97-4, Center for research on parallel computation CRPC, Caltech, 1997.

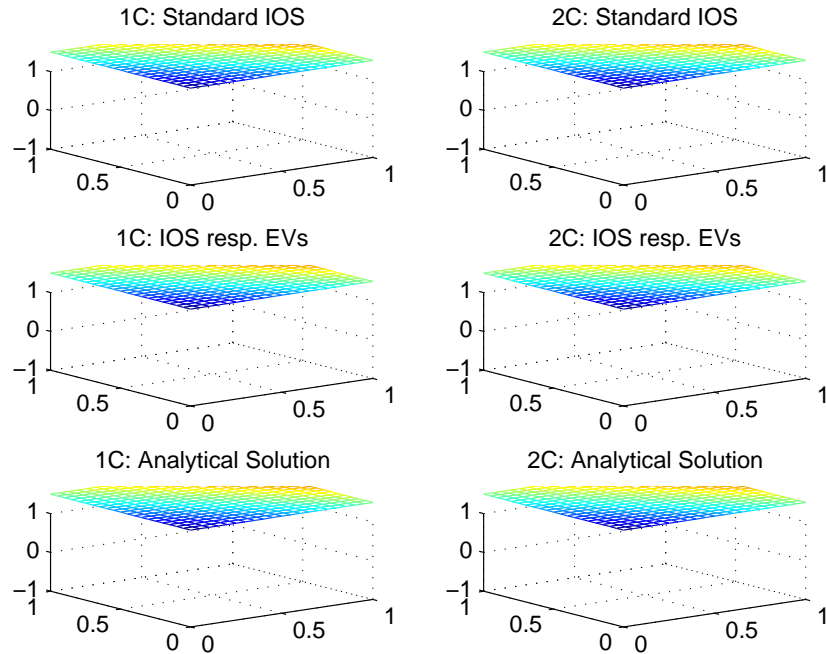


Figure 11: Comparison of the solutions of two different methods to the exact solution for two-dimensional momentum equation using viscosity $\mu = 50$ and $v = (100, 0.01)^T$. The two compared methods are the standard iterative operator-splitting (IOS) as well as the IOS method respecting the stiffness (eigenvalues) of the operators A and B .

- [19] M.K. Gobbert and C.A. Ringhofer. *An asymptotic analysis for a model of chemical vapor deposition on a microstructured surface*. SIAM Journal on Applied Mathematics, 58, 737–752, 1998.
- [20] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II*. SCM, Springer-Verlag Berlin-Heidelberg-New York, No. 14, 1996.
- [21] E. Hairer, C. Lubich, and G. Wanner. *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations*. SCM, Springer-Verlag Berlin-Heidelberg-New York, No. 31, 2002.
- [22] M. Holst, R. Kozack, F. Saied and S. Subramaniam, Treatment of Electrostatic Effects in Proteins: Multigrid-based Newton Iterative Method for Solution of the Full Nonlinear Poisson-Boltzmann Equation, Proteins: Structure, Function, and Genetics, vol. 18, 231–245, 1994.

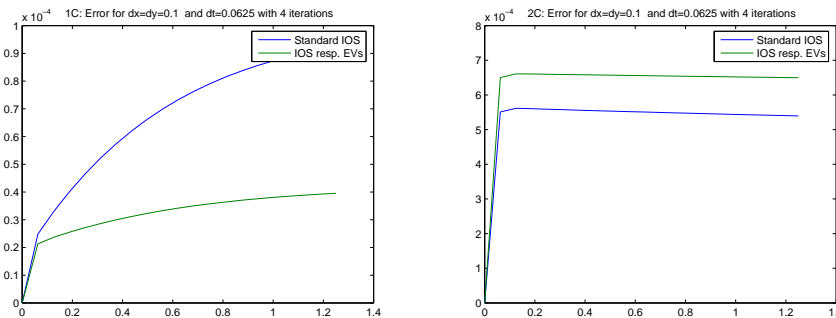


Figure 12: Errors of the solutions of the two different methods for two-dimensional momentum equation using viscosity $\mu = 50$ and $v = (100, 0.01)^T$.

- [23] W. Hundsdorfer and J.G. Verwer, Numerical Solution of Time-Dependent Advection-Diffusion-Reaction Equations, Springer Series in Computational Mathematics Vol. 33, Springer Verlag, 2003.
- [24] W. Hundsdorfer and L. Portero. A Note on Iterated Splitting Schemes. J. Comp. Appl. Math., 201, 146–152, 2007.
- [25] J. Kanney, C. Miller, and C.T. Kelley, Convergence of iterative split-operator approaches for approximating nonlinear reactive transport problems, Advances in Water Resources 26 (2003) 247-261.
- [26] K.H. Karlsen and N. Risebro. *An Operator Splitting method for nonlinear convection-diffusion equation*. Numer. Math., 77, 3 , 365–382, 1997.
- [27] K.H. Karlsen and N.H. Risebro, Corrected operator splitting for nonlinear parabolic equations, SIAM J. Numer. Anal. 37 (2000) 980-1003.
- [28] K.H. Karlsen, K.A. Lie, J.R. Natvig, H.F. Nordhaug and H.K. Dahle, Operator splitting methods for systems of convection-diffusion equations: nonlinear error mechanisms and correction strategies, J. Comput. Phys. 173 (2001) 636-663.
- [29] C.T. Kelly. *Iterative Methods for Linear and Nonlinear Equations*. Frontiers in Applied Mathematics, SIAM, Philadelphia, USA, 1995.
- [30] P. Knabner and L. Angermann, Numerical Methods for Elliptic and Parabolic Partial Differential Equations, Text in Applied Mathematics, Springer Verlag, New York, Berlin, vol. 44, 2003.
- [31] M.A. Lieberman and A.J. Lichtenberg. *Principle of Plasma Discharges and Materials Processing*. Wiley-Interscience, AA John Wiley & Sons, Inc Publication, Second edition, 2005.

- [32] G.I. Marchuk, Some applications of splitting-up methods to the solution of problems in mathematical physics, *Aplikace Matematiky* 1 (1968) 103-132.
- [33] G.A. Meurant, Numerical experiments with a domain decomposition method for parabolic problems on parallel computers, in: R. Glowinski, Y.A. Kuznetsov, G.A. Meurant, J. Périaux and O. Widlund, (Ed.), *Fourth International Symposium on Domain Decomposition Methods for Partial Differential Equations*, Philadelphia, PA, 1991. SIAM.
- [34] M. Ohring. *Materials Science of Thin Films*. Academic Press, San Diego, New York, Boston, London, Second edition, 2002.
- [35] H. Roos, M. Stynes and L. Tobiska. *Numerical Methods for Singular Perturbed Differential Equations*, Springer-Verlag, Berlin, Heidelberg, New York, 1996.
- [36] H.A. Schwarz, Über einige Abbildungsaufgaben, *Journal für Reine und Angewandte Mathematik* 70 (1869) 105-120.
- [37] T.K. Senega and R.P. Brinkmann. *A multi-component transport model for non-equilibrium low-temperature low-pressure plasmas*. *J. Phys. D: Appl. Phys.*, 39, 1606–1618, 2006.
- [38] G. Strang, On the construction and comparison of difference schemes, *SIAM J. Numer. Anal.* 5 (1968) 506-517.
- [39] H.A. van der Vorst, *Iterative Krylov Methods for Large Linear Systems*, Cambridge Monographs on Applied and Computational Mathematics, Cambridge University Press, New York, 2003.
- [40] H. Yoshida, Construction of higher order symplectic integrators, *Physics Letters A*, Vol. 150, no. 5,6,7, 1990.
- [41] E. Zeidler. *Nonlinear Functional Analysis and its Applications. II/B Nonlinear monotone operators* Springer-Verlag, Berlin-Heidelberg-New York, 1990.
- [42] Z. Zlatev. *Computer Treatment of Large Air Pollution Models*. Kluwer Academic Publishers, 1995.