Iterative operator-splitting methods for nonlinear differential equations and applications of deposition processes.

Jürgen Geiser [*] Lena Noack [†]

July 7, 2008

Abstract

In this article we consider iterative operator-splitting methods for nonlinear differential equations with bounded and unbounded operators. The main feature of the proposed idea is the embedding of Newton's method for solving the split parts of the nonlinear equation at each step. 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, stability analysis, 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. The ideas for these models came from the simulation of heat transport in an engineering apparatus, e.g. crystal growth, cf. [13], or the simulation of chemical reaction and transport, e.g. in bio-remediation or waste disposals, cf. [11]. In the past many software tools have been developed for multi-dimensional and multi-physical problems, e.g.

[*]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@mathematik.hu-berlin.de

for the multi-dimensional transport reaction based on different PDE and ODE solvers. In the future a coupling between various software tools with different solver methods will be of interest and could be done using the fractional 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 error analysis is discussed in 4. We introduce the numerical results in Section 5. Finally we discuss our future works in the area of splitting and decomposition methods.

2 Mathematical model

When gas transport is physically more complex because of combined flows in three dimensions, the fundamental equations of fluid dynamics become starting points of the analysis. For our models with small Knudsen numbers, we can assume a continuum flow, and the fluid equations can be treated with a Navier-Stokes or especially with a convection-diffusion equation.

Three basic equations describe the conservation of mass, momentum, and energy, that are sufficient to describe the gas transport in the reactors, see [26].

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 conservation of mass and assume that the energy and momentum is conserved, see [14]. Therefore the continuum flow can be described as a convection-diffusion equation given as:

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

$$F = -D \nabla c, \tag{2}$$

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

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

where c is the molar concentration and F the flux of the species. D is the diffusion 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 $i = 1, 3, \ldots 2m + 1$.

$$\partial_t c_i(x,t) = A c_i(x,t) \ + \ B c_{i-1}(x,t), \text{ with } \ c_i(x,t^n) = c^n \tag{3}$$

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

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

where c^n is the known split approximation at time level $t = t^n$ (see [8]). The approximation at time step $t = t^{n+1}$ is now given as $c^{n+1} = c_{2m+2}(x, t^{n+1})$. $c_0(x,t)$ is given by an initialization process, e.g. $c_0(x,t) = c^n$ or $c_0(x,t) \equiv 0$. This algorithm constitutes an iterative method which in each step involves 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 [12] and [17].

We concentrate again on nonlinear differential equations of the form

$$\partial_t c = A(c)c + B(c)c, \tag{5}$$

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 [33]. 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 (5) by applying

$$\partial_t c_i = A(c_{i-1})c_i \ + \ B(c_{i-1})c_{i-1}, \text{ with } \ c_i(x,t^n) = c^n, \tag{6}$$

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

where the time step is $\tau = t^{n+1} - t^n$. The iterations are $i = 1, 3, \ldots, 2m + 1$. $c_0(x,t) = c^n$ 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 $i = 1, 3, \ldots, 2m+1$. 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 $i \in \{1, 3, \ldots, 2m+1\}$ and exact solution c.

Remark 3.1 The linearization with the fixed-point scheme can be used for smooth or weak nonlinear operators, otherwise we loose the convergence behavior, while we did not converge to the local fixed point, see [17].

3.2 Operator-splitting method with embedded iterative Jacobian-Newton's method

The Newton's method is used to solve the nonlinear parts of the iterative operator-splitting method, see the linearization techniques in [17], [18]. We apply the iterative operator-splitting method and obtain:

$$\begin{aligned}
&F_1(c_i) = \partial_t c_i - A(c_i)c_i - B(c_{i-1})c_{i-1} = 0, \\
&\text{with} \quad c_i(x, t^n) = c^n, \\
&F_2(c_{i+1}) = \partial_t c_{i+1} - A(c_i)c_i - B(c_{i+1})c_{i+1} = 0, \\
&\text{with} \quad c_{i+1}(x, t^n) = c^n,
\end{aligned}$$

where the time step is $\tau = t^{n+1} - t^n$. The iterations are $i = 1, 3, \ldots, 2m+1$. $c_0(x,t) \equiv 0$ or $c_0(x,t) = c^n$ is the starting solution and c^n is the known split approximation at time level $t = t^n$. The results of the methods are $c^{n+1} = c_{2m+2}(x, t^{n+1})$. The splitting method with embedded Newton's method is given as

$$\begin{aligned}
&c_i^{(k+1)} = c_i^{(k)} - D(F_1(c_i^{(k)}))^{-1}(\partial_t c_i^{(k)} - A(c_i^{(k)})c_i^{(k)} - B(c_{i-1}^{(k)})c_{i-1}^{(k)}), \\
&\text{with } D(F_1(c_i^{(k)})) = -(A(c_i^{(k)}) + \frac{\partial A(c_i^{(k)})}{\partial c_i^{(k)}} c_i^{(k)}), \\
&\text{and } k = 0, 1, 2, \ldots, K, \text{ with } \quad c_i^{(k+1)}(x, t^n) = c^n,
\end{aligned}$$

$$\begin{aligned}
&c_{i+1}^{(l+1)} = c_{i+1}^{(l)} - D(F_2(c_{i+1}^{(l)}))^{-1}(\partial_t c_{i+1}^{(l)} - A(c_i^{(l)})c_i^{(l)} - B(c_{i+1}^{(l)})c_{i+1}^{(l)}), \\
&\text{with } D(F_2(c_{i+1}^{(l)})) = -(B(c_{i+1}^{(l)}) + \frac{\partial B(c_{i+1}^{(l)})}{\partial c_{i+1}^{(l)}} c_{i+1}^{(l)}), \\
&\text{and } l = 0, 1, 2, \ldots, L, \text{ with } \quad c_{i+1}^{(l+1)}(x, t^n) = c^n.
\end{aligned}$$

Remark 3.2 For the iterative operator-splitting method with Newton's method we have two iteration procedures. The first iteration is Newton's method for computing the solution of the nonlinear equations, the second iteration is the iterative splitting method, which computes the resulting solution of the coupled equation systems. The embedded method is used for strong nonlinearities.

3.3 Stabilization of the initial values of the iterative Newton's method

To stabilize the initial conditions for the Newton's method we can apply the following ideas:

1) Apply the implicit value for B to stabilize the diagonal of the matrix.

2) Apply first the linear operator, if one operator is linear.

3) Apply an iterated pre-step for the first value.

1) For the stabilization, we use the B operator and balance the diagonal entries of the matrices.
For $i \geq 0$ we have:

$$c_i^{(k+1)} = c_i^{(k)} - D(F_1(c_i^{(k)}))^{-1}(\partial_t c_i^{(k)} - A(c_{i-1}^{(k)})c_i^{(k)} - B(c_i^{(k+1)})c_i^{(k+1)}),$$

$$\text{with } D(F_1(c_i^{(k)})) = -(A(c_i^{(k)}) + \frac{\partial A(c_{i-1}^{(k)})}{\partial c_i^{(k)}} c_i^{(k)}),$$

$$\text{and } k = 0, 1, 2, \ldots, K, \text{ with } \quad c_i^{(k+1)}(x, t^n) = c^n,$$

$$c_{i+1}^{(l+1)} = c_{i+1}^{(l)} - D(F_2(c_{i+1}^{(l)}))^{-1}(\partial_t c_{i+1}^{(l)} - A(c_i^{(l+1)})c_{i+1}^{(l+1)} - B(c_i^{(l)})c_{i+1}^{(l)}),$$

$$\text{with } D(F_2(c_{i+1}^{(l)})) = -(B(c_{i+1}^{(l)}) + \frac{\partial B(c_i^{(l)})}{\partial c_{i+1}^{(l)}} c_{i+1}^{(l)}),$$

$$\text{and } l = 0, 1, 2, \ldots, L, \text{ with } \quad c_{i+1}^{(l+1)}(x, t^n) = c^n.$$

Here we stabilize Newton's method with further entries in the diagonals.

2) If B is linear, then use

$$c_i^{(k+1)} = c_i^{(k)} - D(F_1(c_i^{(k)}))^{-1}(\partial_t c_i^{(k)} - A(c_i^{(k)})c_i^{(k)} - B(c_{i-1}^{(k+1)})c_{i-1}^{(k)}),$$

$$\text{with } D(F_1(c_i^{(k)})) = -(A(c_i^{(k)}) + \frac{\partial A(c_i^{(k)})}{\partial c_i^{(k)}} c_i^{(k)}),$$

$$\text{and } k = 0, 1, 2, \ldots, K, \text{ with } \quad c_i^{(k+1)}(x, t^n) = c^n,$$

$$
\begin{aligned}
&c_{i+1}^{(l+1)} = c_{i+1}^{(l)} - D(F_2(c_{i+1}^{(l)}))^{-1}(\partial_t c_{i+1}^{(l)} - A(c_i^{(l)})c_i^{(l)} - B(c_{i+1}^{(l)})c_{i+1}^{(l)}), \\
&\text{with } D(F_2(c_{i+1}^{(l)})) = -(B(c_{i+1}^{(l)}) + \frac{\partial B(c_{i+1}^{(l)})}{\partial c_{i+1}^{(l)}} c_{i+1}^{(l)}), \\
&\text{and } l = 0, 1, 2, \ldots, L, \text{ with } \quad c_{i+1}^{(l+1)}(x, t^n) = c^n.
\end{aligned}
$$

3) As pre-step, we use

$$
\begin{aligned}
\partial_t c_{i-1} &= A(c_{i-2})c_{i-2} + Bc_{i-1} \\
&\text{with} \quad c_{i-1}(x, t^n) = c^n.
\end{aligned}
$$

Remark 3.3 For the iterative operator-splitting method with Newton's method we have two iteration procedures. The first iteration is Newton's method for computing the solution of the nonlinear equations, the second iteration is the iterative splitting method, which computes the resulting solution of the coupled equation systems. The embedded method is used for strong nonlinearities.

4 Error analysis

Subsequently we demonstrate the error analysis for the linear and nonlinear decomposition methods. In this section we designate $e_i(x,t) := c(x,t) - c_i(x,t)$ as error between the exact solution and the approximated solution after i iterations.

4.1 Error analysis for the linear method

We present the convergence and the rate of convergence of method (3)–(4), where m tends to infinity.

Theorem 4.1 *Let us consider the abstract Cauchy problem in a Banach space* $\mathbf{X}$

$$
\begin{aligned}
\partial_t c(x,t) &= Ac(x,t) + Bc(x,t), \quad 0 < t \leq T, \\
c(x,0) &= c_0(x),
\end{aligned}
$$

where $A, B, A+B : \mathbf{X} \to \mathbf{X}$ are given linear bounded operators being generators of a C_0-semigroup and $c_0 \in \mathbf{X}$ is a given element.
Then the iteration process (3)–(4) for $i = 1, 3, \ldots, 2m+1$ is consistent with order $\mathcal{O}(\tau_n^{2m+1})$.
The estimate is given as:

$$
\|e_{i+1}\| = K_1 \tau_n^2 \|e_{i-1}\| + \mathcal{O}(\tau_n^3). \tag{8}
$$

A proof can be found in [8].
The a priori error expression is given in the following result (see [17]).

Corollary 4.2 *Equation (8) shows that after one more iteration step ($i = 2m+1$) we have the estimate*

$$\|e_{2m+1}\| = K_m \|e_0\| \tau_n^{2m} + \mathcal{O}(\tau_n^{2m+1}), \tag{9}$$

where $c_0(x,t)$ is the initial guess, e.g. $c_0(x,t) \equiv 0$ or $c_0(x,t) = c^n$.

The global error is given in the next theorem (see [17]).

Theorem 4.3 *We assume the local error of the estimate (8) and a kth-order discretization method for time. After $i = 2m+1$ iteration steps there holds*

$$\|c(x,t_n) - c_{2m+1}(x,t_n)\| = t_n^k K_m \|e_0\| \tau_n^{2m} + t^n \mathcal{O}(\tau_n^{2m+1}), \tag{10}$$

where $c_0(x,t)$ is the initial guess.

The proof uses classical operator-splitting methods (see [30]).

Remark 4.4 If A and B are matrices, we obtain a system of ordinary differential equations. To estimate the growth of the matrices, we can use the concept of the logarithmic norm and obtain more detailed results, see [16].

Remark 4.5 *We note that a huge class of important differential operators generate a contractive semigroup. This means that for such problems – assuming the exact solvability of the split subproblems – the iterative splitting method is convergent in second order to the exact solution.*

4.2 Error analysis for the nonlinear method

Here we discuss the linearization techniques and their approximations.

4.2.1 Linearization by iterative splitting method

Theorem 4.6 *Let us consider the following problem*

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

where A, B are nonlinear differentiable bounded operators A, B in a Banach space $\mathbf{X}$.
Linearizing the nonlinear operators yields the linearized equation

$$
\begin{aligned}
\partial_t c(x,t) &= \tilde{A} c(x,t) + \tilde{B} c(x,t) + R(c_{\tilde{i}}) c_{\tilde{i}}, \quad 0 < t \leq T, \\
\tilde{A} &= A(c_{\tilde{i}}) + \frac{\partial A(c_{\tilde{i}})}{\partial c} c_{\tilde{i}}, \\
\tilde{B} &= B(c_{\tilde{i}}) + \frac{\partial B(c_{\tilde{i}})}{\partial c} c_{\tilde{i}}, \\
R(c_{\tilde{i}}) &= \frac{\partial A(c_{\tilde{i}})}{\partial c} c_{\tilde{i}} + \frac{\partial B(c_{\tilde{i}})}{\partial c} c_{\tilde{i}}, \\
c(x,0) &= c_0(x),
\end{aligned} \tag{11}
$$

where $\tilde{A}, \tilde{B}, \tilde{A}+\tilde{B} : \mathbf{X} \to \mathbf{X}$ are given, linear bounded operators being generators of the C_0-semigroup and $c_0 \in \mathbf{X}$ is a given element. The linearization is of the form $A(c)c \approx A(c_{\tilde{i}})c_{\tilde{i}} + (\frac{\partial A(c_{\tilde{i}})}{\partial c} c_{\tilde{i}})(c - c_{\tilde{i}})$ where $c_{\tilde{i}} \in \mathbf{X}$ is a linearized solution, we further assume that $(\frac{\partial A(c_{\tilde{i}})}{\partial c})c_{\tilde{i}}$ is a constant Jacobian matrix.
We assume that the iteration process (3)–(4) is convergent and the convergence is of second order.
There holds

$$
\|e_i\| = K \tau_n \|e_{i-1}\| + \mathcal{O}(\tau_n^2), \tag{12}
$$

where K is an estimation of the residual $||R(\tilde{c})|| \leq R_{\max} \in I\!R^+$ for all $\tilde{c} \in \mathbf{X}$ and $||\tilde{B}|| \leq \tilde{K}$.

We can also obtain the result with Lipschitz constants.
We now prove the argument using the semigroup theory.
Proof.
Let us consider the iteration (3)–(4) in the subinterval $[t^n, t^{n+1}]$.
The linearized splitting method is given as

$$
\begin{aligned}
&\partial_t c_i = \tilde{A} c_i + \tilde{B} c_{i-1} + R(c_{i-1}) c_{i-1}, && (13) \\
&\text{with } \ c_i(x, t^n) = c^n, \\
&\partial_t c_{i+1} = \ \tilde{A} c_i \ + \ \tilde{B} c_{i+1} + R(c_{i-1}) c_{i-1}, && (14) \\
&\text{with } \ c_{i+1}(x, t^n) = c^n,
\end{aligned}
$$

where c^n is the known split approximation at time level $t = t^n$. We solve the subproblems consecutively for $i = 1, 3, \ldots, 2m+1$ and obtain $c^{n+1} = c_{2m+2}(x,t)$.

For the error function $e_i(x,t) = c(x,t) - c_i(x,t)$ we have the relations

$$
\begin{aligned}
\partial_t e_i = \tilde{A}(e_i) + \tilde{B}(e_{i-1}) + R(e_{i-1}) e_{i-1}, \quad x \in \Omega, \ t \in (t^n, t^{n+1}], \\
e_i(x, t^n) = c(x, t^n) - c^n,
\end{aligned} \tag{15}
$$

and

$$
\begin{aligned}
\partial_t e_{i+1} = \tilde{A}(e_i) + \tilde{B}(e_{i+1}) + R(e_{i-1}) e_{i-1}, \quad x \in \Omega, \ t \in (t^n, t^{n+1}], \\
e_{i+1}(x, t^n) = c(x, t^n) - c^n,
\end{aligned} \tag{16}
$$

for $i = 1, 3, 5, \ldots$, with $e_0(x, 0) = 0$ and

$$\begin{aligned}
\tilde{A} &= A(e_{i-1}) + \frac{\partial A(e_{i-1})}{\partial c} e_{i-1}, \\
\tilde{B} &= B(e_{i-1}) + \frac{\partial B(e_{i-1})}{\partial c} e_{i-1}, \\
R(e_{i-1}) &= \frac{\partial A(e_{i-1})}{\partial c} e_{i-1} + \frac{\partial B(e_{i-1})}{\partial c} e_{i-1}.
\end{aligned}$$

In the following we derive the linearized equations. We use the notation $\mathbf{X}^2$ for the product space $\mathbf{X} \times \mathbf{X}$ enabled with the norm $\|(u, v)\| = \max\{\|u\|, \|v\|\}$ $(u, v \in \mathbf{X})$. The elements $\mathcal{E}_i(x, t)$, $\mathcal{F}_i(x, t) \in \mathbf{X}^2$, and the linear operator $\mathcal{A} : \mathbf{X}^2 \to \mathbf{X}^2$ are defined as follows

$$\mathcal{E}_i(x, t) = \begin{bmatrix} e_i(x, t) \\ e_{i+1}(x, t) \end{bmatrix}; \quad \mathcal{A} = \begin{bmatrix} \tilde{A} & 0 \\ \tilde{A} & \tilde{B} \end{bmatrix}, \tag{17}$$

$$\mathcal{F}_i(x, t) = \begin{bmatrix} R(e_{i-1})e_{i-1} + \tilde{B}e_{i-1} \\ R(e_{i-1})e_{i-1} \end{bmatrix}, \tag{18}$$

where we have the bounded and linearized operators $\tilde{A}$, $\tilde{B}$, and R.

Using notations (17) and (18), the relations (15)–(16) can be written in the form

$$\begin{aligned}
\partial_t \mathcal{E}_i(x, t) = \mathcal{A}\mathcal{E}_i(x, t) + \mathcal{F}_i(x, t), \quad x \in \Omega, \ t \in (t^n, t^{n+1}], \\
\mathcal{E}_i(x, t^n) = 0.
\end{aligned} \tag{19}$$

Due to our assumptions that A and B are bounded and differentiable and that we have a Lipschitz domain, $\mathcal{A}$ is a generator of the one-parameter C_0-semigroup $(\mathcal{A}(t))_{t \geq 0}$. We also assume the estimate of our term $\mathcal{F}_i(x, t)$ with the growth conditions.

We can estimate the right-hand side $\mathcal{F}_i(x, t)$ with help of Lemma 4.7 presented after this proof. Hence, using the variations of constants formula, the solution of the abstract Cauchy problem (19) with homogeneous initial condition can be written as (cf. e.g. [7])

$$\mathcal{E}_i(x, t) = \int_{t^n}^{x,t} \exp(\mathcal{A}(t - s))\mathcal{F}_i(x, s) ds, \quad x \in \Omega, \ t \in [t^n, t^{n+1}]. \tag{20}$$

Hence, using the notation

$$\|\mathcal{E}_i\|_\infty = \sup_{t \in [t^n, t^{n+1}]} \|\mathcal{E}_i(x, t)\| \ , \tag{21}$$

and taking into account Lemma 4.7, we have

$$\begin{aligned}
\|\mathcal{E}_i(x, t)\|_\infty &\leq \|\mathcal{F}_i\|_\infty \int_{t^n}^{t} \|\exp(\mathcal{A}(t - s))\| ds \\
&\leq C \, \|e_{i-1}(t)\| \int_{t^n}^{t} \|\exp(\mathcal{A}(t - s))\| ds, \quad x \in \Omega, \ t \in [t^n, t^{n+1}].
\end{aligned} \tag{22}$$

Since $(\mathcal{A}(t))_{t\geq 0}$ is a semigroup, the so-called *growth estimate* is

$$\|\exp(\mathcal{A}t)\| \leq K\exp(\omega t), \quad t \geq 0, \tag{23}$$

with some numbers $K \geq 0$ and $\omega \in I\!R$ (see [7]).

- Assume that $(\mathcal{A}(t))_{t\geq 0}$ is a bounded or exponentially stable semigroup, i.e. that (23) holds with some $\omega \leq 0$. Then obviously the inequality

$$\|\exp(\mathcal{A}t)\| \leq K; \quad t \geq 0 \tag{24}$$

holds, and hence from (22) we have

$$\|\mathcal{E}_i(x,t)\|_\infty \leq K\tau_n\|e_{i-1}(x,t)\|, \quad x \in \Omega,\ t \in (0,\tau_n). \tag{25}$$

- Assume that $(\mathcal{A}(t))_{t\geq 0}$ has exponential growth with some $\omega > 0$. From (23) we have

$$\int_{t^n}^{t^{n+1}} \|\exp(\mathcal{A}(t-s))\| ds \leq K_\omega(t), \quad t \in [t^n, t^{n+1}], \tag{26}$$

where

$$K_\omega(t) = \frac{K}{\omega}\left(\exp(\omega(t-t^n)) - 1\right), \quad t \in [t^n, t^{n+1}], \tag{27}$$

and hence

$$K_\omega(t) \leq \frac{K}{\omega}\left(\exp(\omega\tau_n) - 1\right) = K\tau_n + \mathcal{O}(\tau_n^2), \tag{28}$$

where $\tau_n = t^{n+1} - t^n$. The estimations (25) and (28) result in

$$\|\mathcal{E}_i\|_\infty = K\tau_n\|e_{i-1}\| + \mathcal{O}(\tau_n^2). \tag{29}$$

Taking into account the definition of $\mathcal{E}_i$ and the norm $\|\cdot\|_\infty$, that result in the estimation $||e_{i+1}|| \leq ||e_i||$, we obtain

$$\|e_i\| = K\tau_n\|e_{i-1}\| + \mathcal{O}(\tau_n^2),$$

which proves our statement. □

Lemma 4.7 *The term $\mathcal{F}_i(x,t)$ given by (18) can be estimated as*

$$||\mathcal{F}_i(x,t)|| \leq C||e_{i-1}||, \tag{30}$$

where we assume the boundedness of $R(e_{i-1})$ and $\tilde{B}$, see Theorem 4.6.

Proof. We have the norm $||\mathcal{F}_i(x,t)|| = \max\{\mathcal{F}_{i_1}(x,t), \mathcal{F}_{i_2}(x,t)\}$.
Each term can be bounded as follows.

$$\begin{aligned} ||\mathcal{F}_{i_1}(x,t)|| &\leq ||(R(e_{i-1}(x,t)) + \tilde{B})e_{i-1}(x,t)|| \\ &\leq (R_{max} + \tilde{K})||e_{i-1}(x,t)||, \qquad (31) \\ ||\mathcal{F}_{i_2}(x,t)|| &\leq ||R(e_{i-1}(x,t))e_{i-1}(x,t)|| \\ &\leq R_{max}||e_{i-1}(x,t)||, \qquad (32) \end{aligned}$$

where R_{max} and $\tilde{K}$ are constants and defined in Theorem 4.6.

Thus we obtain the estimate

$$||\mathcal{F}_i(x,t)|| \leq C||e_{i-1}(x,t)||,$$

where $C = R_{max} + \tilde{K}$. □

4.2.2 Linearization by Newton's Method

In this approach we use Newton's method for a linearization. Here we have two steps in the proof of the error analysis.

1) Error of Newton's method;
2) Error of the iterative or non-iterative operator-splitting method.

Theorem 4.8 *Consider the problem*

$$\begin{aligned} \partial_t c(x,t) = A(c(x,t)) + B(c(x,t)), \quad x \in \Omega,\ 0 < t \leq T, \\ c(x,0) = c_0(x), \end{aligned} \qquad (33)$$

where A, B are nonlinear differentiable bounded operators in a Banach space $\mathbf{X}$. *We apply Newton's method to solve the nonlinear equations and obtain*

$$\begin{aligned} & c_i^{(k+1)} = c_i^{(k)} - D(F_1(c_i^{(k)}))^{-1}(\partial_t c_i^{(k)} - A(c_i^{(k)})c_i^{(k)} - B(c_{i-1}^{(k)})c_{i-1}^{(k)}), \\ & \textit{with } D(F_1(c_i^{(k)})) = -(A(c_i^{(k)}) + \frac{\partial A(c_i^{(k)})}{\partial c_i^{(k)}} c_i^{(k)}), \\ & \textit{with} \quad c_i(x,t^n) = c^n, \\ & \\ & c_{i+1}^{(k+1)} = c_{i+1}^{(k)} - D(F_2(c_{i+1}^{(k)}))^{-1}(\partial_t c_{i+1}^{(k)} - A(c_i^{(k)})c_i^{(k)} - B(c_{i+1}^{(k)})c_{i+1}^{(k)}), \\ & \textit{with } D(F_2(c_{i+1}^{(k)})) = -(B(c_{i+1}^{(k)}) + \frac{\partial B(c_{i+1}^{(k)})}{\partial c_{i+1}^{(k)}} c_{i+1}^{(k)}), \\ & \textit{and } k = 0, 1, 2, \ldots, \\ & \textit{with} \quad c_{i+1}(x,t^n) = c^n. \end{aligned}$$

The iterations are $i = 1, 3, \ldots, 2m+1$. $c_0(x,t) \equiv 0$ or $c_0(x,t) = c^n$ is the starting solution and c^n is the known split approximation at time level $t = t^n$.

The result of the schemes is $c^{n+1} = c_{2m+2}(x, t^{n+1})$.

The following inequality holds,

$$||e_i(x,t)^{(k+1)}|| \leq K\tau_n^2 ||e_{i-1}^{(k)}(x,t)||^2, \tag{34}$$

where $\tau_n = t^{n+1} - t^n$, K *is a constant, and* k *the index for the Newton iteration.*

Proof.
The sketch of the proof is outlined in two parts. The first part gives the approximation error of Newton's method and the second part the approximation error of the iterative operator-splitting method.

First Part:
The error for Newton's method can be derived as

$$||e_i^{(k+1)}|| \leq K_1 ||e_i^{(k)}||^2, \tag{35}$$

where $e_i^{(k+1)} = c_i^{(k+1)} - c$, c is the exact solution of the nonlinear problem, and K_1 is a constant, see [21].

Second Part:
For the iterative operator-splitting method, we obtain the approximation error

$$||e_i^{(k)}|| = K_2 \tau_n ||e_{i-1}^{(k)}|| + \mathcal{O}(\tau_n^2), \tag{36}$$

where K_2 is an estimation of the residual, see Theorem 4.6, and $\tau_n = t^{n+1} - t^n$.

We insert the result of 35 into 36 and obtain the error of the nonlinear splitting scheme, which is given as:

$$||e_i^{(k+1)}(x,t)|| \leq K\tau_n^2 ||e_{i-1}^{(k)}(x,t)||^2,$$

where K is a combination of the constants K_1 and K_2.

□

5 Numerical examples

In the next experiments we deal with nonlinear differential equations. Because of the regularity assumptions to our splitting method we apply 2-4 iteration steps.

In the numerical examples, operator B is linear. Therefore the iterative Newton's method is given by

$$\begin{aligned} c_i^{(k+1)} &= c_i^{(k)} - D(F(c_{i-1}^{(k)}))^{-1}(\partial_t c_i^{(k)} - A(c_{i-1}^{(k)})c_i^{(k)} - Bc_i^{(k+1)}), \quad (37) \\ & \quad \text{with } D(F(c_{i-1}^{(k)})) = -\frac{\partial A(c_{i-1}^{(k)})}{\partial c_{i-1}^{(k)}} c_i^{(k)}, \quad c_i(x,t^n) = c^n, \\ & \quad \text{and } k = 0, 1, 2, \ldots, K, \\ \partial_t c_{i+1} &= A(c_{i-1})c_i \ + \ Bc_{i+1}, \text{ with } \ c_{i+1}(x,t^n) = c^n. \quad (38) \end{aligned}$$

5.1 Test example 1: Burgers equation

We deal with a 2D example where we can derive an analytical solution and compare the classical iterative operator-splitting method with the iterative Newton's method.

$$\begin{aligned} &\partial_t c = -c\partial_x c - c\partial_y c + \mu(\partial_{xx} c + \partial_{yy} c) + f(x,y,t), \\ &\text{for } (x,y,t) \in \Omega \times [0,T] \\ &c(x,y,0) = c_{\text{ana}}(x,y,0),\ (x,y) \in \Omega \\ &\text{with } c(x,y,t) = c_{\text{ana}}(x,y,t) \text{ on} \partial\Omega \times [0,T], \end{aligned} \tag{39}$$

where $\Omega = [0,1] \times [0,1]$, $T = 1.25$, and μ is the viscosity.
The analytical solution is given as

$$c_{\text{ana}}(x,y,t) = (1 + \exp(\frac{x+y-t}{2\mu}))^{-1}, \tag{40}$$

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

The operators are given as:

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

We apply the nonlinear Algorithm 6 to the first equation and obtain

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

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 7:

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

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}	$\mathrm{err}_{\max}$	ρ_{L_1}	$\rho_{\max}$
1/10	1/10	0.0549	0.1867		
1/20	1/10	0.0468	0.1599	0.2303	0.2234
1/40	1/10	0.0418	0.1431	0.1630	0.1608
1/10	1/20	0.0447	0.1626		
1/20	1/20	0.0331	0.1215	0.4353	0.4210
1/40	1/20	0.0262	0.0943	0.3352	0.3645
1/10	1/40	0.0405	0.1551		
1/20	1/40	0.0265	0.1040	0.6108	0.5768
1/40	1/40	0.0181	0.0695	0.5517	0.5804

Table 1: Numerical results for the Burgers equation with viscosity $\mu = 0.05$, initial condition $c_0(x, y, t) = c^n$, and two iterations per time step.

$\Delta x = \Delta y$	Δt	err_{L_1}	$\mathrm{err}_{\max}$	ρ_{L_1}	$\rho_{\max}$
1/10	1/10	$1.1168 \cdot 10^{-7}$	$2.4390 \cdot 10^{-7}$		
1/20	1/10	$8.2098 \cdot 10^{-8}$	$1.7163 \cdot 10^{-7}$	0.4439	0.5070
1/40	1/10	$6.4506 \cdot 10^{-8}$	$1.3360 \cdot 10^{-7}$	0.3479	0.3614
1/10	1/20	$3.8260 \cdot 10^{-8}$	$9.0093 \cdot 10^{-8}$		
1/20	1/20	$2.5713 \cdot 10^{-8}$	$5.6943 \cdot 10^{-8}$	0.5733	0.6619
1/40	1/20	$1.8738 \cdot 10^{-8}$	$4.0020 \cdot 10^{-8}$	0.4565	0.5088
1/10	1/40	$1.9609 \cdot 10^{-8}$	$4.9688 \cdot 10^{-8}$		
1/20	1/40	$1.1863 \cdot 10^{-8}$	$2.8510 \cdot 10^{-8}$	0.7250	0.8014
1/40	1/40	$7.8625 \cdot 10^{-9}$	$1.8191 \cdot 10^{-8}$	0.5934	0.6482

Table 2: Numerical results for the Burgers equation with viscosity $\mu = 5$, initial condition $c_0(x, y, t) = c^n$, and two iterations per time step.

We have the following results, see Tables 1 and 2, for different steps in time and space and different viscosities.

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

Remark 5.1 In the examples, we have two different cases of μ, which smoothes our equation. In the first test we use a very small $\mu = 0.05$, such that we have a dominant hyperbolic behavior, due to this we have a loss in regularity and sharp front. The iterative splitting method looses 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 and we achieve higher accuracy.

$\Delta x = \Delta y$	Δt	err_{L_1}	$\text{err}_{\max}$	ρ_{L_1}	$\rho_{\max}$
1/10	1/10	$2.5038 \cdot 10^{-8}$	$7.3067 \cdot 10^{-8}$		
1/20	1/10	$1.9001 \cdot 10^{-8}$	$5.5507 \cdot 10^{-8}$	0.39804	0.39655
1/40	1/10	$1.5992 \cdot 10^{-8}$	$4.7145 \cdot 10^{-8}$	0.24873	0.23557
1/10	1/20	$1.9503 \cdot 10^{-8}$	$5.6176 \cdot 10^{-8}$		
1/20	1/20	$1.3250 \cdot 10^{-8}$	$3.8448 \cdot 10^{-8}$	0.55767	0.54705
1/40	1/20	$1.0177 \cdot 10^{-8}$	$3.0008 \cdot 10^{-8}$	0.38063	0.35755
1/10	1/40	$1.6329 \cdot 10^{-8}$	$4.7092 \cdot 10^{-8}$		
1/20	1/40	$9.9375 \cdot 10^{-9}$	$2.9072 \cdot 10^{-8}$	0.71645	0.69587
1/40	1/40	$6.8369 \cdot 10^{-9}$	$2.0423 \cdot 10^{-8}$	0.53955	0.50945

Table 3: Numerical results for the Burgers equation with viscosity $\mu = 5$, initial condition $c_0(x, y, t) = c^n$, two iterations per time step and $K = 2$ using iterative Newton's method.

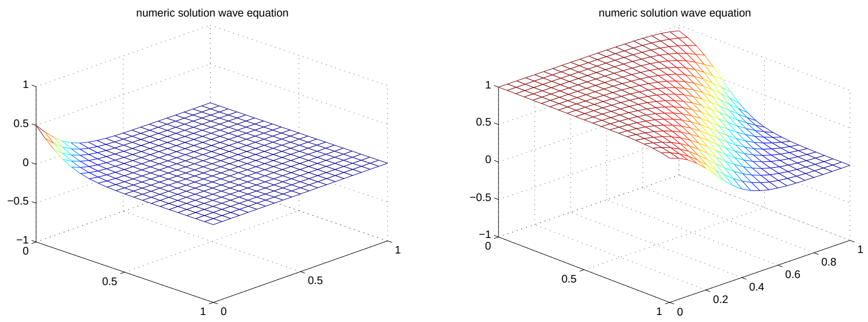

Figure 1: Burgers equation at initial time $t = 0.0$ (left figure) and end time $t = 1.25$ (right figure) for viscosity $\mu = 0.05$.

5.2 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.

$$\begin{aligned}
&\partial_t c = -1/2 c \partial_x c - 1/2 c \partial_y c - 1/2 \partial_x c - 1/2 \partial_y c \\
&+\mu(\partial_{xx} c + \partial_{yy} c) + f(x, y, t),\ (x, y, t) \in \Omega \times [0, T] \\
&c(x, y, 0) = c_{\text{ana}}(x, y, 0),\ (x, y) \in \Omega \\
&\text{with } c(x, y, t) = c_{\text{ana}}(x, y, t) \text{ on} \partial\Omega \times [0, T],
\end{aligned} \tag{41}$$

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

The analytical solution is given as

$$c_{\text{ana}}(x, y, t) = (1 + \exp(\frac{x + y - t}{2\mu}))^{-1} + \exp(\frac{x + y - t}{2\mu}), \tag{42}$$

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

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

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

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

For the first equation we apply the nonlinear Algorithm 6 and obtain

$A(c_{i-1})c_i = -1/2c_{i-1}\partial_x c_i - 1/2c_{i-1}\partial_y c_i + 1/2\mu(\partial_{xx}c_i + \partial_{yy}c_i)$ 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 7:

$A(c_{i-1})c_i = -1/2c_{i-1}\partial_x c_i - 1/2c_{i-1}\partial_y c_i + 1/2\mu(\partial_{xx}c_i + \partial_{yy}c_i)$ 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 4 and 5.
Figure 2 presents the profile of the 2D linear and nonlinear convection-diffusion equation.

Remark 5.2 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 higher viscosity about $\mu = 5$ and get the second-order result with two iteration steps. Here we see the loss of differentiability due to the stiff equation parts. To obtain the same results, we have to increase the number of iteration steps. Thus we could show an improvement of the convergence order with respect to the iteration steps.

$\Delta x = \Delta y$	Δt	err_{L_1}	$\mathrm{err}_{\max}$	ρ_{L_1}	$\rho_{\max}$
1/5	1/20	0.0137	0.0354		
1/10	1/20	0.0055	0.0139	1.3264	1.3499
1/20	1/20	0.0017	0.0043	1.6868	1.6900
1/40	1/20	$8.8839 \cdot 10^{-5}$	$3.8893 \cdot 10^{-4}$	4.2588	3.4663
1/5	1/40	0.0146	0.0377		
1/10	1/40	0.0064	0.0160	1.1984	1.2315
1/20	1/40	0.0026	0.0063	1.3004	1.3375
1/40	1/40	$8.2653 \cdot 10^{-4}$	0.0021	1.6478	1.6236

Table 4: Numerical results for the mixed convection-diffusion and Burgers equation with viscosity $\mu = 0.5$, initial condition $c_0(x, y, t) = c^n$, and four iterations per time step.

$\Delta x = \Delta y$	Δt	err_{L_1}	$\mathrm{err}_{\max}$	ρ_{L_1}	$\rho_{\max}$
1/5	1/20	$1.3166 \cdot 10^{-5}$	$2.9819 \cdot 10^{-5}$		
1/10	1/20	$5.6944 \cdot 10^{-6}$	$1.3541 \cdot 10^{-5}$	1.2092	1.1389
1/20	1/20	$1.6986 \cdot 10^{-6}$	$4.5816 \cdot 10^{-6}$	1.7452	1.5634
1/40	1/20	$7.8145 \cdot 10^{-7}$	$2.0413 \cdot 10^{-6}$	1.1201	1.1663
1/5	1/40	$1.4425 \cdot 10^{-5}$	$3.2036 \cdot 10^{-5}$		
1/10	1/40	$7.2343 \cdot 10^{-6}$	$1.5762 \cdot 10^{-5}$	0.9957	1.0233
1/20	1/40	$3.0776 \cdot 10^{-6}$	$6.7999 \cdot 10^{-6}$	1.2330	1.2129
1/40	1/40	$9.8650 \cdot 10^{-7}$	$2.3352 \cdot 10^{-6}$	1.6414	1.5420

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

5.3 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.

$$\begin{aligned} &\partial_t \mathbf{c} = -\mathbf{c} \cdot \nabla \mathbf{c} + 2\mu \nabla (D(\mathbf{c}) + 1/3 \nabla \mathbf{c}) + \mathbf{f}(x, y, t), \qquad (43) \\ &(x, y, t) \in \Omega \times [0, T], \ \mathbf{c}(x, y, 0) = \mathbf{c}_0(x, y), \ (x, y) \in \Omega \\ &\text{with } \mathbf{c}(x, y, t) = \mathbf{c}_{\text{ana}}(x, y, t) \text{ on } \partial\Omega \times [0, T] \text{ (enclosed flow)}, \end{aligned}$$

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.

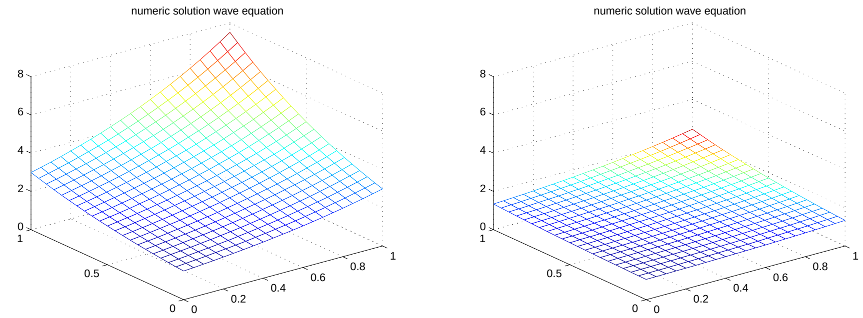

Figure 2: Mixed convection-diffusion and Burgers equation at initial time $t = 0.0$ (left figure) and end time $t = 1.25$ (right figure) for viscosity $\mu = 0.5$.

We can derive the analytical solution with the functions:

$$c_{1,\text{ana}}(x, y, t) = (1 + \exp(\frac{x + y - t}{2\mu}))^{-1} + \exp(\frac{x + y - t}{2\mu}), \quad (44)$$

$$c_{2,\text{ana}}(x, y, t) = (1 + \exp(\frac{x + y - t}{2\mu}))^{-1} + \exp(\frac{x + y - t}{2\mu}). \quad (45)$$

For the splitting method our operators are given as:

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

We first deal with the one-dimensional case,

$$\partial_t c = -c \cdot \partial_x c + 2\mu\partial_x(D(c) + 1/3\partial_x c) + f(x, t), \ (x, t) \in \Omega \times [0, T] \quad (46)$$

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

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

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:

$A(c)c = -c\partial_x c + 2\mu\partial_x D(c)$ (the nonlinear operator), and
$Bc = 2/3\mu\partial_{xx}c$ (the linear operator).

For the iterative operator-splitting method as fixed-point scheme, we have the following results, see Tables 6 and 7.

Figure 3 presents the profile of the 1D momentum equation.

Δx	Δt	err_{L_1}	$\mathrm{err}_{\max}$	ρ_{L_1}	$\rho_{\max}$
1/10	1/20	0.0213	0.0495		
1/20	1/20	0.0203	0.0470	0.0689	0.0746
1/40	1/20	0.0198	0.0457	0.0401	0.0402
1/80	1/20	0.0195	0.0450	0.0216	0.0209
1/10	1/40	0.0134	0.0312		
1/20	1/40	0.0117	0.0271	0.1957	0.2009
1/40	1/40	0.0108	0.0249	0.1213	0.1211
1/80	1/40	0.0103	0.0238	0.0682	0.0674
1/10	1/80	0.0094	0.0217		
1/20	1/80	0.0073	0.0169	0.3591	0.3641
1/40	1/80	0.0062	0.0143	0.2451	0.2448
1/80	1/80	0.0056	0.0129	0.1478	0.1469

Table 6: Numerical results for the 1D momentum equation with $\mu = 5$, $v = 0.001$, initial condition $c_0(x,t) = c^n$, and two iterations per time step.

Δx	Δt	err_{L_1}	$\mathrm{err}_{\max}$	ρ_{L_1}	$\rho_{\max}$
1/10	1/20	$2.7352 \cdot 10^{-6}$	$6.4129 \cdot 10^{-6}$		
1/20	1/20	$2.3320 \cdot 10^{-6}$	$5.4284 \cdot 10^{-6}$	0.2301	0.2404
1/40	1/20	$2.1144 \cdot 10^{-6}$	$4.9247 \cdot 10^{-6}$	0.1413	0.1405
1/80	1/20	$2.0021 \cdot 10^{-6}$	$4.6614 \cdot 10^{-6}$	0.0787	0.0793
1/10	1/40	$2.1711 \cdot 10^{-6}$	$5.2875 \cdot 10^{-6}$		
1/20	1/40	$1.7001 \cdot 10^{-6}$	$4.1292 \cdot 10^{-6}$	0.3528	0.3567
1/40	1/40	$1.4388 \cdot 10^{-6}$	$3.4979 \cdot 10^{-6}$	0.2408	0.2394
1/80	1/40	$1.3023 \cdot 10^{-6}$	$3.1694 \cdot 10^{-6}$	0.1438	0.1423
1/10	1/80	$1.6788 \cdot 10^{-6}$	$4.1163 \cdot 10^{-6}$		
1/20	1/80	$1.1870 \cdot 10^{-6}$	$2.9138 \cdot 10^{-6}$	0.5001	0.4984
1/40	1/80	$9.1123 \cdot 10^{-7}$	$2.2535 \cdot 10^{-6}$	0.3814	0.3707
1/80	1/80	$7.6585 \cdot 10^{-7}$	$1.9025 \cdot 10^{-6}$	0.2507	0.2443

Table 7: Numerical results for the 1D momentum equation with $\mu = 50$, $v = 0.1$, initial condition $c_0(x,t) = c^n$, and two iterations per time step.

We have the following results for the 2D case, see Tables 8, 9, and 10.

Figure 4 presents the profile of the 2D momentum equation.

Remark 5.3 In the more realistic examples of a 1D and 2D momentum equation, we also observe the stiff problem, which we obtain with a more hyperbolic behavior. In the 1D experiments we deal with a more hyperbolic 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

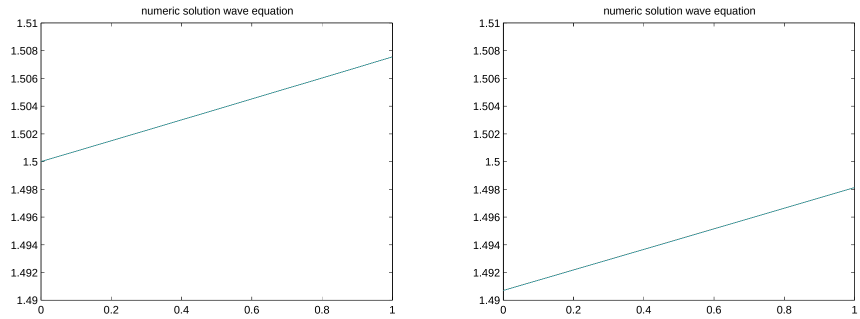

Figure 3: 1D momentum equation at initial time $t = 0.0$ (left figure) and end time $t = 1.25$ (right figure) for $\mu = 5$ and $v = 0.001$.

$\Delta x = \Delta y$	Δt	err_{L_1} 1st c.	$\mathrm{err}_{\max}$ 1st c.	ρ_{L_1} 1st c.	$\rho_{\max}$ 1st c.	err_{L_1} 2nd c.	$\mathrm{err}_{\max}$ 2nd c.	ρ_{L_1} 2nd c.	$\rho_{\max}$ 2nd c.
1/5	1/20	0.0027	0.0112			0.0145	0.0321		
1/10	1/20	0.0016	0.0039	0.7425	1.5230	0.0033	0.0072	2.1526	2.1519
1/20	1/20	0.0007	0.0022	1.2712	0.8597	0.0021	0.0042	0.6391	0.7967
1/5	1/40	0.0045	0.0148			0.0288	0.0601		
1/10	1/40	0.0032	0.0088	0.5124	0.7497	0.0125	0.0239	1.2012	1.3341
1/20	1/40	0.0014	0.0034	1.1693	1.3764	0.0029	0.0054	2.1263	2.1325
1/5	1/80	0.0136	0.0425			0.0493	0.1111		
1/10	1/80	0.0080	0.0241	0.7679	0.8197	0.0278	0.0572	0.8285	0.9579
1/20	1/80	0.0039	0.0113	1.0166	1.0872	0.0115	0.0231	1.2746	1.3058

Table 8: Numerical results for the 2D momentum equation with $\mu = 2$, $v = (1,1)^t$, initial condition $c_0(x, y, t) = c^n$, and two iterations per time step.

steps, if we increase the parabolic behavior, e.g. 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. Hence the stiffness influences the number of iteration steps.

6 Conclusions and Discussions

We present a new method to solve complicate 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 number of iteration steps of our method. At least we

Δx $= \Delta y$	Δt	err_{L_1} 1st c.	$\mathrm{err}_{\max}$ 1st c.	ρ_{L_1} 1st c.	$\rho_{\max}$ 1st c.
1/5	1/20	$1.5438 \cdot 10^{-5}$	$3.4309 \cdot 10^{-5}$		
1/10	1/20	$4.9141 \cdot 10^{-6}$	$1.0522 \cdot 10^{-5}$	1.6515	1.7052
1/20	1/20	$1.5506 \cdot 10^{-6}$	$2.9160 \cdot 10^{-6}$	1.6641	1.8513
1/5	1/40	$2.8839 \cdot 10^{-5}$	$5.5444 \cdot 10^{-5}$		
1/10	1/40	$1.3790 \cdot 10^{-5}$	$2.3806 \cdot 10^{-5}$	1.0645	1.2197
1/20	1/40	$3.8495 \cdot 10^{-6}$	$6.8075 \cdot 10^{-6}$	1.8408	1.8061
1/5	1/80	$3.1295 \cdot 10^{-5}$	$5.5073 \cdot 10^{-5}$		
1/10	1/80	$1.7722 \cdot 10^{-5}$	$2.6822 \cdot 10^{-5}$	0.8204	1.0379
1/20	1/80	$7.6640 \cdot 10^{-6}$	$1.1356 \cdot 10^{-5}$	1.2094	1.2400

Table 9: Numerical results for the 2D momentum equation for the first component with $\mu = 50$, $v = (100, 0.01)^t$, initial condition $c_0(x, y, t) = c^n$, and two iterations per time step.

Δx $= \Delta y$	Δt	err_{L_1} 2nd c.	$\mathrm{err}_{\max}$ 2nd c.	ρ_{L_1} 2nd c.	$\rho_{\max}$ 2nd c.
1/5	1/20	$4.3543 \cdot 10^{-5}$	$1.4944 \cdot 10^{-4}$		
1/10	1/20	$3.3673 \cdot 10^{-5}$	$7.9483 \cdot 10^{-5}$	0.3708	0.9109
1/20	1/20	$2.6026 \cdot 10^{-5}$	$5.8697 \cdot 10^{-5}$	0.3717	0.4374
1/5	1/40	$3.4961 \cdot 10^{-5}$	$2.2384 \cdot 10^{-4}$		
1/10	1/40	$1.7944 \cdot 10^{-5}$	$8.9509 \cdot 10^{-5}$	0.9622	1.3224
1/20	1/40	$1.5956 \cdot 10^{-5}$	$3.6902 \cdot 10^{-5}$	0.1695	1.2783
1/5	1/80	$9.9887 \cdot 10^{-5}$	$3.3905 \cdot 10^{-4}$		
1/10	1/80	$3.5572 \cdot 10^{-5}$	$1.3625 \cdot 10^{-4}$	1.4896	1.3153
1/20	1/80	$1.0557 \cdot 10^{-5}$	$4.4096 \cdot 10^{-5}$	1.7525	1.6275

Table 10: Numerical results for the 2D momentum equation for the second component with $\mu = 50$, $v = (100, 0.01)^t$, initial condition $c_0(x, y, t) = c^n$, and two iterations per time step.

obtain a second-order method. Such iterative splitting methods can balance the different behaviors of the underlying operators. 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 new method that reveals the mixed behavior in an unsplitted method.

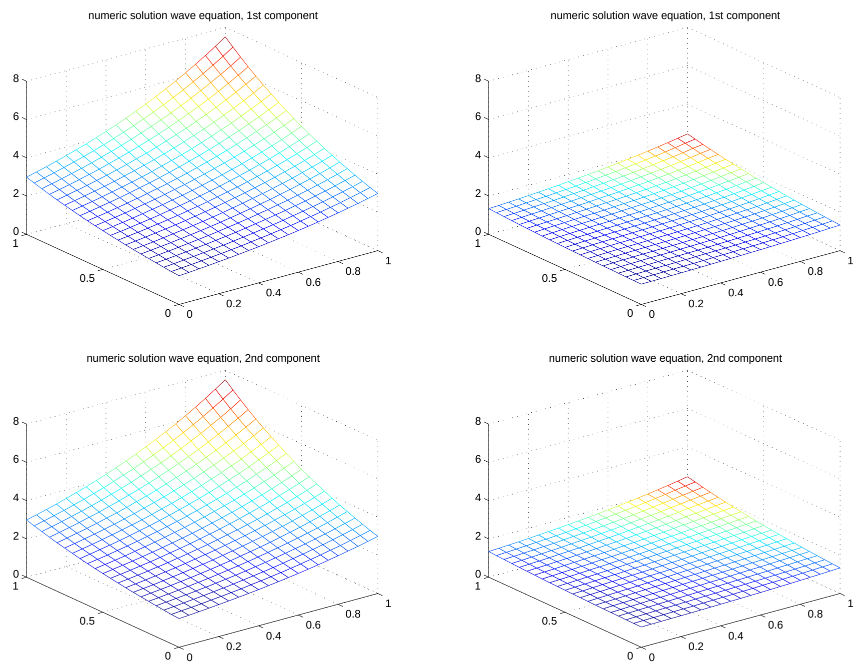

Figure 4: 2D momentum equation at initial time $t = 0.0$ (left figure) and end time $t = 1.25$ (right figure) for $\mu = 0.5$ and $v = (1, 1)^t$ for the first and second component of the numerical solution.

References

[1] W. Bangerth and R. Rannacher. *Adaptive finite Element Methods for differential equations.* Lectures in Mathematics, ETH Zürich, Birkhäuser Verlag, Basel-Boston-Berlin, 2003.

[2] X.C. Cai, Additive Schwarz algorithms for parabolic convection-diffusion equations, Numer. Math. 60 (1991) 41-61.

[3] X.C. Cai, Multiplicative Schwarz methods for parabolic problems, SIAM J. Sci Comput. 15 (1994) 587-603.

[4] W. Cheney, Analysis for Applied Mathematics, Graduate Texts in Mathematics., 208, Springer, New York, Berlin, Heidelberg, 2001.

[5] 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.

[6] P. Deuflhard, Newton Methods for Nonlinear Problems, Springer-Verlag, New-York, Berlin, 2004.

[7] K.-J. Engel and R. Nagel, One-Parameter Semigroups for Linear Evolution Equations, Springer, New York, 2000.

[8] 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.

[9] M.J. Gander and H. Zhao, Overlapping Schwarz waveform relaxation for parabolic problems in higher dimension, In A. Handlovičová, Magda Komorníkova, and Karol Mikula, editors, in: Proc. Algoritmy 14, Slovak Technical University, 1997, pp. 42-51.

[10] 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.

[11] 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.

[12] 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.

[13] 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.

[14] 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.

[15] 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: Stucture, Function, and Genetics, vol. 18, 231–245, 1994.

[16] 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.

[17] 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.

[18] K.H. Karlsen and N. Risebro. *An Operator Splitting method for nonlinear convection-diffusion equation.* Numer. Math., 77, 3 , 365–382, 1997.

[19] K.H. Karlsen and N.H. Risebro, Corrected operator splitting for nonlinear parabolic equations, SIAM J. Numer. Anal. 37 (2000) 980-1003.

[20] 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.

[21] C.T. Kelly. *Iterative Methods for Linear and Nonlinear Equations.* Frontiers in Applied Mathematics, SIAM, Philadelphia, USA, 1995.

[22] P. Knabner and L. Angermann, Numerical Methods for Elliptic and Parabolic Partial Differential Equations, Text in Applied Mathematics, Springer Verlag, Newe York, Berlin, vol. 44, 2003.

[23] 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.

[24] G.I. Marchuk, Some applicatons of splitting-up methods to the solution of problems in mathematical physics, Aplikace Matematiky 1 (1968) 103-132.

[25] 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.

[26] M. Ohring. *Materials Science of Thin Films.* Academic Press, San Diego, New York, Boston, London, Second edition, 2002.

[27] H. Roos, M. Stynes and L. Tobiska. Numerical Methods for Singular Perturbed Differential Equations, Springer-Verlag, Berlin, Heidelberg, New York, 1996.

[28] H.A. Schwarz, Über einige Abbildungsaufgaben, Journal für Reine und Angewandte Mathematik 70 (1869) 105-120.

[29] 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.

[30] G. Strang, On the construction and comparision of difference schemes, SIAM J. Numer. Anal. 5 (1968) 506-517.

[31] 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.

[32] H. Yoshida, Construction of higher order symplectic integrators, Physics Letters A, Vol. 150, no. 5,6,7, 1990.

[33] E. Zeidler. Nonlinear Functional Analysis and its Applications. II/B Nonlinear montone operators Springer-Verlag, Berlin-Heidelberg-New York, 1990.

[34] Z. Zlatev. *Computer Treatment of Large Air Pollution Models.* Kluwer Academic Publishers, 1995.