Iterative operator-splitting method: Error analysis and examples

Jürgen Geiser

geiser@mathematik.hu-berlin.de

Abstract. In this paper we describe an iterative operator-splitting method for bounded operators. The contribution is a novel iterative method, that can be applied as a splitting method for ordinary and partial differential equations. A simple relation between the number of iterative steps and the order of the splitting scheme, can made it as an alternative method to consider it as a time decomposition method. The iterative splitting scheme is interested on physical problem, while the original problem is not divided as in standard splitting schemes. We present error bounds for the iterative splitting methods in the presence of bounded operators. We discuss efficient algorithms to compute the integral formulation of the splitting scheme. In experiments, we consider the benefits of the novel splitting method with respect to their number of iterations and time steps. Ordinary differential equations and convection-diffusion-reaction equations are presented in the numerical results.

Keywords Iterative operator-splitting method, Error analysis, convection-diffusion equation.

AMS subject classifications. 65M15, 65L05, 65M71.

1 Introduction

Iterative operator splitting methods have been considered since the last years, see [4], [7] and [13], as efficient solver for differential equations. Historically they can be seen as alternating Waveform relaxation methods, while exchanging their operators in the Waveform relaxation scheme. Such schemes are wel-known and are considered as efficient solver in many application fields since the 80ties of the last century, see [18] and [20]. Here the iterative scheme is a novel method that generalize such schemes.

Thus, for the iterative splitting methods we state the following theses.

- For non-commuting operators, we may reduce the local splitting error, by using more iteration steps to obtain a higher-order accuracy.
- We must solve the original problem within a full splitting step, while keeping all operators in the equations.

– Splitting the original problem into the different subproblems including all operators of the problem, is physically the best. We obtain consistent approximations after each inner step, because of the exact or approximative starting conditions for the previous iterative solution.

In our paper, we taken into account the iterative operator splitting schemes for a PDE solver and deal with bounded operators. Here the balance between the operator in an integral formulation is important to bound the scheme. Such a balance, which is not necessary in the bounded case, reduce the order of the scheme. Such deficits can be reduced by using additional iteration steps and balance the spatial and time step in the discretised form, see [4] and [5].

To dicuss the analysis and the application to the differential equations, we concentrate in this paper on an approximate solution of the linear evolution equation

$$\partial_t \, c = Lc = (A + B)c, \; c(0) = c_0, \tag{1}$$

where L, A and B are bounded operators.

As numerical method we will apply a two-stage iterative splitting scheme:

$$c_i(t) = \exp(At)c_0 + \int_0^t \exp(A(t-s))Bc_{i-1}(s) \, ds, \tag{2}$$

$$c_{i+1}(t) = \exp(Bt)c_0 + \int_0^t \exp(B(t-s))Ac_i(s) \, ds, \tag{3}$$

where $i = 1, 3, 5, \ldots$ and the initialization $c_0(t) = 0$.

The outline of the paper is as follows. The operator-splitting methods are introduced in Section 2 and the error analysis of the operator-splitting methods is presented. In Section 3, we discuss the error analysis of the iterative methods. In Section 4, we discuss an efficient computation of the iterative splitting method with ϕ-functions. In Section 5 we introduce the application of our methods to existing software tools. Finally we discuss future works in the area of iterative splitting methods.

2 Iterative splitting method

The following algorithm is based on an iteration with fixed-splitting discretization step-size τ, that is, on the time-interval $[t_n, t_{n+1}]$ we solve the following sub-problems consecutively for $i = 0, 2, \ldots 2m$. (cf. [8, 12]):

$$\frac{\partial c_i(t)}{\partial t} = Ac_i(t) \; + \; Bc_{i-1}(t), \tag{4}$$

$$\text{with} \;\; c_i(t_n) = c^n$$

$$\frac{\partial c_{i+1}(t)}{\partial t} = Ac_i(t) \; + \; Bc_{i+1}(t), \tag{5}$$

$$\text{with} \;\; c_{i+1}(t_n) = c^n \; ,$$

where c^n is the known split approximation at the time-level $t = t_n$. The initialization is given as $c_0(t_n) = c^n$ and starting solution is given as $c_{-1} = 0.0$ (here also improvement are discussed, see [3] and [6]). The split approximation at the time-level $t = t_{n+1}$ is defined as $c^{n+1} = c_{2m+1}(t_{n+1})$. (Clearly, the function $c_{i+1}(t)$ depends on the interval $[t_n, t_{n+1}]$, too, but, for the sake of simplicity, in our notation we omit the dependence on n.)

In the following we will analyze the convergence and the rate of convergence of the method (4)–(5) for m tends to infinity for the linear operators $A, B : \mathbf{X} \to \mathbf{X}$, where we assume that these operators and their sum are generators of the C_0 semigroups. We emphasize that these operators are bounded, so the convergence is examined in a Banach space setting. The Banach norm is given as $||\cdot||_{\mathbf{X}} = ||\cdot||$ and also its corresponding matrix norm, see [3].

3 Error analysis

In the following we discuss the error analysis of the iterative splitting method.

We deal with the following assumptions:

Assumption 31 *The linear operators $A + B, A, B$ generate C_0 semigroups on $\mathbf{X}$, and the operators A, B satisfy in addition the bounds:*

$$|| \exp(A\tau)|| \leq \exp(\omega |t|) \textit{ and } || \exp(B\tau)|| \leq \exp(\omega |t|) \tag{6}$$

for some $\omega ge 0$ and all $t \in \mathbb{R}$.

The following theorem is given the convergence of an iterative operator splitting schemes for one-sided iterations and a next assumption $B = A^{1-\alpha}$:

Assumption 32 *For the consistency proofs we have to assume the following:*

The linear operators $A + B, A, B$ generate analytical semigroups on $\mathbf{X}$, and the operators A, B satisfy in addition the bounds:

$$||B^\alpha \exp(B\tau_n)|| \leq \kappa \tau_n^{-\alpha}. \tag{7}$$

$$||B \exp((A + B)\tau_n)|| \leq \kappa \tau_n^{-1+\alpha}, \tag{8}$$

$$|| \exp(A\tau_n)B|| \leq \tilde{\kappa} \tau_n^{-1+\alpha)}, \tag{9}$$

$$||A^\beta \exp(A\tau_n)|| \leq \kappa \tau_n^{-\beta}. \tag{10}$$

$$||A^\beta \exp((A + B)\tau_n)|| \leq \kappa \tau_n^{-\beta}, \tag{11}$$

where $\alpha, \beta \in (0, 1)$ and $\tau_n = (t^{n+1} - t^n)$.

Theorem 1. *For the numerical solution of (4), consider an iterative operator splitting scheme on operator A with i-th iterative steps.*

If the assumptions 31 and 32 are valid, then

$$||S_i^n - \exp((A + B)n\tau)|| \leq C\tau^{i-1}, \; n\tau \leq T, \tag{12}$$

where the constant C can be chosen uniformly on bounded time intervals and in particular, independent of n and τ.

Proof. By applying the telescopic identity we obtain

$$
\begin{aligned}
&(S_i^n - \exp((A+B)n\tau)c_0 \\
&= \sum_{\nu=0}^{n-1} S_i^{n-\nu-1}(S - \exp((A+B)\tau))\exp(\nu\tau(A+B))c_0. \quad (13)
\end{aligned}
$$

If we assume the stability bound:

$$||S_i|| \leq \exp(c\omega\tau), \quad (14)$$

with a constant c only depends on the estimation of the method.

Furthermore, if we assume the consistency bound:

$$
\begin{aligned}
&||S_i^n - \exp((A+B)n\tau)|| \\
&\leq \exp(c\omega T)\sum_{\nu=0}^{n-1} ||(S - \exp(\tau(A+B)))\exp(\nu\tau(A+B))|| \quad (15) \\
&\leq C\tau^i, \; n\tau \leq T, \quad (16)
\end{aligned}
$$

The desired consistency and stability bound is given in the next subsections.

3.1 Consistency analysis

We present the results of the consistency of our iterative method. We assume for the system of operator the generator of an analytical semigroup based on their underlying norms (see the previous Section 2).

In the following we discuss the consistency of the 2 stage iterative method, taken into account to iterate over both operators.

Theorem 2. *Let us consider the abstract Cauchy problem in a Banach space* $\mathbf{X}$.

$$
\begin{aligned}
\partial_t c(x,t) &= Ac(x,t) + Bc(x,t), \quad 0 < t \leq T \text{ and } x \in \Omega, \\
c(x,0) &= c_0(x), \; x \in \Omega, \quad (17) \\
c(x,t) &= c_1(x,t), \; x \in \partial\Omega \times [0,T],
\end{aligned}
$$

where $A, B : \mathbf{X} \to \mathbf{X}$ *are given linear operators which are generators of the analytical semigroups and* $c_0 \in \mathbf{X}$ *is a given element. We assume* $dom(B) \subset dom(A)$*, so we are restricted to balance the operators. Further, we assume the estimations of an unbounded operator, see [11]:*

$$B = A^{1-\alpha}, \quad (18)$$

where $\alpha \in (0,1)$ *and we assume*

$B = A^{1-\alpha}$ *is the infinitesimal generator of an analytical semigroup for all* $\alpha \in (0,1)$*, see [17].*

The error of the first time-step is of accuracy $\mathcal{O}(\tau_n^{i_A\alpha})$*, where* $\tau_n = t^{n+1} - t^n$ *and we have equidistant time-steps, with* $n = 1, \ldots, N$*. Further* i_A *are the iterative steps with operator* A.

Then the iteration process (4)–(5) for $i = 1, 3, \ldots, 2m+1$ *is consistent with the order of the consistency* $\mathcal{O}(\tau_n^{\alpha i_A})$*, where* $0 \leq \alpha < 1$.

Proof. Let us consider the iteration (4)–(5) on the sub-interval $[t^n, t^{n+1}]$.

For the first iterations we have:

$$\partial_t c_1(t) = A c_1(t), \quad t \in (t^n, t^{n+1}], \tag{19}$$

and for the second iteration we have:

$$\partial_t c_2(t) = A c_1(t) + B c_2(t), \quad t \in (t^n, t^{n+1}], \tag{20}$$

In general we have:
for the odd iterations: $i = 2m + 1$ for $m = 0, 1, 2, \ldots$

$$\partial_t c_i(t) = A c_i(t) + B c_{i-1}(t), \quad t \in (t^n, t^{n+1}], \tag{21}$$

where for $c_0(t) \equiv 0$.
for the even iterations: $i = 2m$ for $m = 1, 2, \ldots$

$$\partial_t c_i(t) = A c_{i-1}(t) + B c_i(t), \quad t \in (t^n, t^{n+1}], \tag{22}$$

We have the following solutions for the iterative scheme:
the solutions for the first two equations are given by the variation of constants:

$$c_1(t) = \exp(A(t - t^n)) c(t^n), \quad t \in (t^n, t^{n+1}], \tag{23}$$

$$\begin{aligned} c_2(t) \ &= \exp(B(t - t^n)) c(t^n) \\ &+ \int_{t^n}^{t^{n+1}} \exp(B(t^{n+1} - s)) A c_1(s) ds, \quad t \in (t^n, t^{n+1}]. \end{aligned} \tag{24}$$

For the recursive even and odd iterations we have the solutions.
We start with the odd iterations: $i = 2m + 1$ for $m = 0, 1, 2, \ldots$

$$\begin{aligned} &c_i(t) = \exp(A(t - t^n)) c(t^n) \\ &+ \int_{t^n}^{t} \exp((t - s)A) B c_{i-1}(s) \, ds, \quad t \in (t^n, t^{n+1}], \end{aligned} \tag{25}$$

For the even iterations: $i = 2m$ for $m = 1, 2, \ldots$

$$\begin{aligned} &c_i(t) = \exp(B(t - t^n)) c(t^n) \\ &+ \int_{t^n}^{t} \exp((t - s)B) A c_{i-1}(s) \, ds, \quad t \in (t^n, t^{n+1}], \end{aligned} \tag{26}$$

The consistency is given as:
For e_1 we have:

$$c_1(t^{n+1}) = \exp(A\tau_n) c(t^n), \tag{27}$$

$$\begin{aligned} &c(t^{n+1}) = \exp((A + B)\tau_n) c(t^n) = \exp(A\tau_n) c(t^n) \\ &+ \int_{t^n}^{t^{n+1}} \exp(A(t^{n+1} - s)) B \exp((s - t^n)(A + B)) c(t^n) \, ds. \end{aligned} \tag{28}$$

We obtain:

$$
\begin{aligned}
&||e_1|| = ||c - c_1|| \leq || \exp((A+B)\tau_n)c(t^n) - \exp(A\tau_n)c(t^n)|| \qquad (29)\\
&\leq || \int_{t^n}^{t^{n+1}} \exp(A(t^{n+1}-s))B \exp((s-t^n)(A+B))c(t^n)\; ds||\\
&\leq || \int_{t^n}^{t^{n+1}} \exp(A(t^{n+1}-s))A^{1-\alpha} \exp((s-t^n)(A+B))c(t^n)\; ds||\\
&\leq \int_{t^n}^{t^{n+1}} || \exp(A(t^{n+1}-s))A^{(1-\alpha})/2A^{(1-\alpha)/2} \exp((s-t^n)(A+B))||\; ds\; ||c(t^n)||\\
&\leq \int_{t^n}^{t^{n+1}} \frac{1}{(t^{n+1}-s)^{-(1-\alpha)/2}}\; \frac{\kappa}{(s-t^n)^{(1-\alpha)/2}}\; ds\; ||c(t^n)||\\
&+ \left. \int_{t^n}^{t^{n+1}} Cs^(\alpha - 1)ds \right) ||c(t^n)||\\
&\leq C\tau^\alpha\; ||c(t^n)|| \qquad (30)
\end{aligned}
$$

where $\alpha \in (0,1)$ and $\tau = (t^{n+1} - t^n)$.

For e_2 we have:

$$
\begin{aligned}
&c_2(t^{n+1}) = \exp(B\tau_n)c(t^n)\\
&+ \int_{t^n}^{t^{n+1}} \exp(B(t^{n+1}-s))A \exp((s-t^n)A)c(t^n)\; ds, \qquad (31)
\end{aligned}
$$

$$
\begin{aligned}
&c(t^{n+1}) = \exp(B\tau_n)c(t^n)\\
&+ \int_{t^n}^{t^{n+1}} \exp(B(t^{n+1}-s))A \exp((s-t^n)A)c(t^n)\; ds\\
&+ \int_{t^n}^{t^{n+1}} \exp(B(t^{n+1}-s))A \qquad (32)\\
&\int_{t^n}^{s} \exp(A(s-\rho))B \exp((\rho-t^n)(A+B))c(t^n)\; d\rho\; ds.
\end{aligned}
$$

We obtain:

$$
\begin{aligned}
&||e_2|| \leq || \exp((A+B)\tau_n) c(t^n) - c_2|| && (33)\\
&= || \int_{t^n}^{t^{n+1}} \exp(B(t^{n+1}-s)) A && (34)\\
&\int_{t^n}^{s} \exp(A(s-\rho)) A^{1-\alpha} \exp((\rho - t^n)(A+B)) c(t^n) \ d\rho \ ds|| \\
&= \int_{t^n}^{t^{n+1}} || \exp(B(t^{n+1}-s))|| && (35)\\
&\int_{t^n}^{s} || \exp(A(s-\rho)) A^{2-\alpha} \exp((\rho - t^n)(A+B)) c(t^n) \ d\rho|| ds \\
&= \int_{t^n}^{t^{n+1}} \kappa \int_{t^n}^{s} (s-\rho)^{\alpha-2} d\rho ds ||c(t^n)|| && (36)\\
&\leq C\tau^{\alpha} \ ||c(t^n)||
\end{aligned}
$$

For odd and even iterations, the recursive proof is given in the following. In the next steps, we shift $t^n \rightarrow 0$ and $t^{n+1} \rightarrow \tau_n$ for simpler calculations, see [11]. The initial conditions are given with $c(0) = c(t^n)$.

For the odd iterations means the iteration over operator A: $i = 2m+1$, with $m = 0, 1, 2, \ldots$, we obtain for c_i and c:

$$
\begin{aligned}
&c_i(\tau_n) = \exp(A\tau_n) c(0) && (37)\\
&+ \int_0^{\tau_n} \exp(As) B \exp((\tau_n - s)B) c(0) \ ds \\
&+ \int_0^{\tau_n} \exp(As_1) B \int_0^{\tau_n - s_1} \exp(s_2 B) A \exp((\tau_n - s_1 - s_2)A) c(0) \ ds_2 \ ds_1 \\
&+ \ldots + \\
&+ \int_0^{\tau_n} \exp(As_1) B \int_0^{\tau_n - s_1} \exp(s_2 A) B \int_0^{\tau_n - s_1 - s_2} \exp(s_3 A) B \ldots \\
&\int_0^{\tau_n - \sum_{j=1}^{i-1} s_j} \exp(As_i) B \exp((\tau_n - \sum_{j=1}^{i-1} s_j) A) c(0) \ ds_i \ldots ds_1,
\end{aligned}
$$

$$
\begin{aligned}
c(\tau_n) &= \exp(A\tau_n)c(0) \\
&+ \int_0^{\tau_n} \exp(As)B\exp((\tau_n - s)B)c(0)\ ds \\
&+ \int_0^{\tau_n} \exp(As_1)B \int_0^{\tau_n - s_1} \exp(s_2 B)A\exp((\tau_n - s_1 - s_2)A)c(0)\ ds_2\ ds_1 \\
&+ \ldots + \\
&+ \int_0^{\tau_n} \exp(As_1)B \int_0^{\tau_n - s_1} \exp(s_2 A)B \int_0^{\tau_n - s_1 - s_2} \exp(s_3 A)B \ldots \\
&\int_0^{\tau_n - \sum_{j=1}^{i-1} s_j} \exp(As_i)B\exp((\tau_n - \sum_{j=1}^{i-1} s_j)A)c(0)\ ds_i \ldots ds_1 \\
&+ \int_0^{\tau_n} \exp(As_1)B \int_0^{\tau_n - s_1} \exp(s_2 A)B \int_0^{\tau_n - s_1 - s_2} \exp(s_3 A)B \ldots \\
&\int_0^{\tau_n - \sum_{j=1}^{i} s_j} \exp(As_{i+1})B\exp((\tau_n - \sum_{j=1}^{i} s_j)(A+B))c(0)\ ds_{i+1} \ldots ds_1.
\end{aligned} \tag{38}
$$

By shifting $0 \rightarrow t^n$ and $\tau_n \rightarrow t^{n+1}$, we obtain our result:

$$
\begin{aligned}
&||e_i|| \leq ||\exp((A+B)\tau_n)c(t^n) - c_i|| \\
&\leq \tilde{C}\tau_n^{i_A \alpha}||c(t^n)||,
\end{aligned} \tag{39}
$$

where $\alpha = \min_{j=1}^{i}\{\alpha_i\}$ and $0 \leq \alpha_i < 1$ and i_A is the number of odd iteration steps means over the operator A.

The same proof idea can be applied to the even iterative scheme.

Where for the even scheme we could not obtain a higher order, see:

$$
\begin{aligned}
c(\tau_n) &= \exp(A\tau_n)c(0) \\
&+ \int_0^{\tau_n} \exp(Bs)A\exp((\tau_n - s)B)c(0)\ ds \\
&+ \int_0^{\tau_n} \exp(Bs_1)A \int_0^{\tau_n - s_1} \exp(s_2 B)A\exp((\tau_n - s_1 - s_2)B)c(0)\ ds_2\ ds_1 \\
&+ \ldots + \\
&+ \int_0^{\tau_n} \exp(Bs_1)A \int_0^{\tau_n - s_1} \exp(s_2 B)A \int_0^{\tau_n - s_1 - s_2} \exp(s_3 B)A \ldots \\
&\int_0^{\tau_n - \sum_{j=1}^{i-1} s_j} \exp(Bs_i)A\exp((\tau_n - \sum_{j=1}^{i-1} s_j)B)c(0)\ ds_i \ldots ds_1 \\
&+ \int_0^{\tau_n} \exp(Bs_1)A \int_0^{\tau_n - s_1} \exp(s_2 B)A \int_0^{\tau_n - s_1 - s_2} \exp(s_3 B)A \ldots \\
&\int_0^{\tau_n - \sum_{j=1}^{i} s_j} \exp(Bs_{i+1})A\exp((\tau_n - \sum_{j=1}^{i} s_j)(A+B))c(0)\ ds_{i+1} \ldots ds_1.
\end{aligned} \tag{40}
$$

and we have

$$||e_{i_B}|| \leq || \exp((A+B)\tau_n)c(t^n) - c_{i_B}|| \tag{41}$$

$$= || \int_{t^n}^{t^{n+1}} \exp(B(t^{n+1} - s_1))A \ldots \tag{42}$$

$$\int_{t^n}^{s_{i_B-1}} \exp(B(s_{i_B-1} - s_{i_B}))A \exp((s_{i_B} - t^n)(A+B))c(t^n) \, ds_{i_B} \, ds_1||$$

$$= \int_{t^n}^{t^{n+1}} \kappa \tag{43}$$

$$\int_{t^n}^{s} \kappa ||A^2 \exp((\rho - t^n)(A+B))c(t^n) \, d\rho||ds$$

$$= \int_{t^n}^{t^{n+1}} \kappa \ldots \int_{t^n}^{s_{i_B}} (s-\rho)^{-i_B} ds_{i_B} \ldots ds_1 ||c(t^n)|| \tag{44}$$

$$\leq C\tau^0 \; ||c(t^n)||$$

So we could not improve the order in the weaker iterations, so we need at least some strong iterative steps.

Remark 1. The applications are given for A-bounded problems, e.g.

- Convection-Diffusion equation: $A = \frac{\partial^2}{\partial x^2}$ and $B = -\frac{\partial}{\partial x}$,
- Diffusion-Reaction equation: $A = \frac{\partial^2}{\partial x^2}$ and $B = -\lambda$

In the next section we describe the stability analysis.

3.2 Stability Analysis

For stability bound we have the following theorem:

Theorem 3. *Let us consider the abstract Cauchy problem (4) and (5) in a Banach space* $\mathbf{X}$*. Then the stability of the method given in equation (12) is given as*

$$||S_i|| \leq \exp(c\omega\tau) \tag{45}$$

where c depends only on the coefficients of the method and ω *is a bound for the operators, see (31), and* τ *is the time-step size.*

Proof. We apply the assumption:
$B = A^{1-\alpha}$

Based on the definition of S_i we have:

$$
\begin{aligned}
& S_i = \exp(A\tau_n) && (46)\\
& + \int_0^{\tau_n} \exp(As)B\exp((\tau_n - s)A)\; ds \\
& + \int_0^{\tau_n} \exp(As_1)B \int_0^{\tau_n - s_1} \exp(s_2 A)B\exp((\tau_n - s_1 - s_2)A)\; ds_2\; ds_1 \\
& + \ldots + \\
& + \int_0^{\tau_n} \exp(As_1)B \int_0^{\tau_n - s_1} \exp(s_2 A)B \int_0^{\tau_n - s_1 - s_2} \exp(s_3 A)B \ldots \\
& \int_0^{\tau_n - \sum_{j=1}^{i-1} s_j} \exp(As_i)B\exp((\tau_n - \sum_{j=1}^{i-1} s_j)A)\; ds_i \ldots ds_1,
\end{aligned}
$$

After application of B we have:

$$
\begin{aligned}
& ||S_i|| \leq ||\exp(A\tau_n)|| && (47)\\
& +|| \int_0^{\tau_n} \exp(As)B\exp((\tau_n - s)A)\; ds|| \\
& +|| \int_0^{\tau_n} \exp(As_1)B \int_0^{\tau_n - s_1} \exp(s_2 B)A\exp((\tau_n - s_1 - s_2)A)\; ds_2\; ds_1|| \\
& + \ldots + \\
& +|| \int_0^{\tau_n} \exp(As_1)B \int_0^{\tau_n - s_1} \exp(s_2 A)B \int_0^{\tau_n - s_1 - s_2} \exp(s_3 A)B \ldots \\
& \int_0^{\tau_n - \sum_{j=1}^{i-1} s_j} \exp(As_i)B\exp((\tau_n - \sum_{j=1}^{i-1} s_j)A)\; ds_i \ldots ds_1||, \\
& \leq \exp(\omega\tau) + \sum_{j=1}^{i} C_j t^{\alpha j} \leq \exp(tilde\omega\tau) && (48)
\end{aligned}
$$

where for all $\omega, C_j \leq 0$ for all $j = 1, \ldots, i$ and $\alpha \in (0, 1)$, we find $\tilde{\omega} \leq 0$.

Therefore $||S_i|| \leq \exp(\tilde{\omega}\tau)$ is bounded.

4 Computation of the iterative splitting method

Based on the integral formulation of the iterative splitting schemes, it is important to formulate efficient algorithms to solve the schemes. One efficient idea is to compute integral-formulation of exp-functions with so called ϕ-functions, which reduces the integration to a product of exp-functions, see [9]. Such algorithms can be discrete formulated with exponential Runge-Kutta methods, see [10].

As regards computations of the matrix exponential, an overview is given in [15].

For linear operators $A, B : \mathcal{D}(X) \subset X \rightarrow X$ generating a C_0 semigroup and a scalar $t \in \mathbb{R}^+$, we define the operator $a = tA$ and $b = tB$, and the bounded operators $\phi_{0,A} = \exp(a)$, $\phi_{0,B} = \exp(b)$ and:

$$\phi_{k,A} = \int_0^1 \exp((1-s)\tau A) \frac{s^{k-1}}{(k-1)!} ds, \tag{49}$$

$$\phi_{k,B} = \int_0^1 \exp((1-s)\tau B) \frac{s^{k-1}}{(k-1)!} ds, \tag{50}$$

for $k \geq 1$.

From this definition it is a straightforward matter to prove the recurrence relation:

$$\phi_{k,A} = \frac{1}{k!} I + \tau A \phi_{k+1}, \tag{51}$$

$$\phi_{k,B} = \frac{1}{k!} I + \tau B \phi_{k+1}. \tag{52}$$

We apply equations (51) and (52) to our iterative schemes (**??**) and (**??**) and obtain:

$$c_1(\tau) = \exp(A\tau) c(t_n) = \phi_{0,A} c(t_n), \tag{53}$$

$$c_2(\tau) = \phi_{0,A} c(t_n) + \sum_{k=1}^{\infty} B^k A \phi_{k,A}, \tag{54}$$

where we assume that B is bounded and $\exp B = \sum_{k=0}^{\infty} \frac{1}{k} B^k$.

For an bounded operator B we can apply the convolution of integrals, exactly with the Laplacian transformation or numerically with integration rules.

4.1 Exact Computation of the Integrals

The algorithm of the iterative splitting method (4)-(5) can be written in the form:

$$\partial_t c_1 = A c_1 \tag{55}$$

$$\partial_t c_2 = A c_1 + B c_2 \tag{56}$$

$$\vdots$$

$$\partial_t c_{i+1} = A c_{i+1} + B c_{i+1}, \tag{57}$$

where $c(t_n)$ is the initial condition and A, B are bounded operators.

We use the Laplace transform to solve the ordinary differential equations (55)-(57), see also [1].

The transformations for this cases are given in [1]. We need to define the transformed function $\hat{u} = \hat{u}(s, t)$:

$$\hat{u}_i(s, t) := \int_0^{\infty} u_i(x, t) \, e^{-sx} \, dx \, . \tag{58}$$

We obtain the following analytical solution of the first iterative steps with the re-transformation:

$$c_1 = \exp(At)c(t_n), \tag{59}$$
$$c_2 = A(B-A)^{-1}\exp(At)c(t_n) + A(A-B)^{-1}\exp(Bt)c(t_n). \tag{60}$$

The solutions of the next steps can be done recursively.
The Laplacian transform is given as :

$$\tilde{c}_1 = (Is + A)^{-1} c_{01} \tag{61}$$
$$\tilde{c}_2 = (Is + B)^{-1} c_{02} + (Is + B)^{-1} A\tilde{c}_1 \tag{62}$$
$$\tilde{c}_3 = (Is + A)^{-1} c_{03} + (Is + A)^{-1} A\tilde{c}_2 \tag{63}$$
$$\ldots .$$

Here we assume that we fulfill the following commutator:

$$(A-B)^{-1}A = A(A-B)^{-1}, \tag{64}$$

and we can apply the decomposition of partial fraction:

$$(Is + A)^{-1}A(Is + B)^{-1} = (Is + A)^{-1}(B-A)^{-1}A$$
$$+A(B-A)^{-1}(Is + B)^{-1}. \tag{65}$$

Here we can derive our solutions:

$$c_2 = \exp(Bt)c(t_n) \tag{66}$$
$$+A(B-A)^{-1}\exp(At)c(t_n) + A(A-B)^{-1}\exp(Bt)c(t_n).$$

We have the following recurrent argument for the Laplace transform:
for the odd iterations: $i = 2m + 1$
for $m = 0, 1, 2, \ldots$

$$\tilde{c}_i = (Is - A)^{-1}\; c_n + (Is - A)^{-1}B\; \tilde{c}_{i-1}, \tag{67}$$

for the even iterations: $i = 2m$ for $m = 1, 2, \ldots$

$$\tilde{c}_i(t) = (Is - B)^{-1}A\; \tilde{c}_{i-1} + (Is - B)^{-1}\; c_n, \tag{68}$$

We develop the next iterative solution c_3 as follows:

$$c_3 = \exp(At)c(t_n) \tag{69}$$
$$+BA(B-A)^{-1}t\exp(At)c(t_n)$$
$$+BA(A-B)^{-1}(B-A)^{-1}\exp(At)c(t_n)$$
$$+BA(A-B)^{-1}(A-B)^{-1}\exp(Bt)c(t_n).$$

We apply the iterative steps recursively and obtain for the odd iterative scheme the following recurrent argument:

$$\begin{aligned}
&c_i = \exp(At)c(t_n) \\
&+BA(B-A)^{-1}t\exp(At)c(t_n) \\
&+\ldots \\
&+BA\ldots BA(B-A)^{-1}\ldots(B-A)^{-1}t^{i-2}\exp(At)c(t_n) \\
&+BA\ldots BA(B-A)^{-1}\ldots(B-A)^{-1}(A-B)^{-1}(B-A)^{-1}\exp(At)c(t_n) \\
&+\ldots + BA\ldots BA(B-A)^{-1}\ldots(B-A)^{-1}(A-B)^{-1}\exp(Bt)c(t_n).
\end{aligned} \tag{70}$$

Remark 2. The same recurrent argument can be applied to the even iterative scheme. Here we have only to apply matrix multiplications and can skip the time-consuming integral computations. Only two evaluations for the exponential function for A and B are necessary. The main disadvantage of computing the iterative scheme exactly are the time-consuming inverse matrices. These can be skipped with numerical methods.

4.2 Numerical Computation of the Integrals

Here our main contributions are to skip the integral formulation of the exponential functions and to apply only matrix multiplication of given exponential functions. Such operators can be computed at the beginning of the evaluation.

Evaluation with Trapezoidal rule (two iterative steps).

We have to evaluate:

$$c_2(t) = \exp(Bt)c(t_n) + \int_{t_n}^{t_{n+1}} \exp(B(t_{n+1}-s))Ac_1(s)ds, \quad t \in (t_n, t_{n+1}], \tag{71}$$

where $c_1(t) = \exp(At)\exp(Bt)c(t_n)$.

We apply the Trapezoidal rule and obtain:

$$c_2(t) = \exp(Bt)c(t_n) + \frac{1}{2}\Delta t\left(B\exp(At)\exp(Bt) + exp(At)B\right), \tag{72}$$

where $c_1(t) = \exp(At)\exp(Bt)c(t_n)$ and $\Delta t = t - t_n$.

Evaluation with Simpson rule (three iterative steps).

We have to evaluate:

$$c_3(t) = \exp(At)c(t_n) + \int_{t_n}^{t_{n+1}} \exp(A(t_{n+1}-s))Bc_2(s)ds, \quad t \in (t_n, t_{n+1}], \tag{73}$$

where $c_1(t) = \exp(A\frac{t}{2})\exp(Bt)\exp(A\frac{t}{2})c(t_n)$.

We apply the Simpson rule and obtain:

$$\begin{aligned}
c_3(t) = \exp(At)c(t_n) + \frac{1}{6}\Delta t \Bigg(& B\exp(A\frac{t}{2})\exp(Bt)\exp(A\frac{t}{2}) \\
& + 4\exp(A\frac{t}{2})Bexp(A\frac{t}{4})\exp(B\frac{t}{2})\exp(A\frac{t}{4}) + \exp(At)B\Bigg),
\end{aligned} \tag{74}$$

where $c_1(t) = \exp(At)\exp(Bt)c(t_n)$ and $\Delta t = t - t_n$.

Remark 3. The same result can also be derived by applying BDF3 (Backward Differentiation Formula of Third Order).

Evaluation with Bode rule (four iterative steps).
We have to evaluate:

$$c_4(t) = \exp(Bt)c(t_n) + \int_{t_n}^{t_{n+1}} \exp(B(t_{n+1} - s))Ac_3(s)ds, \quad t \in (t_n, t_{n+1}], \quad (75)$$

where $c_3(t)$ has to be evaluated with a third-order method.
We apply the Bode rule and obtain:

$$\begin{aligned} c_4(t) = \exp(At)c(t_n) + \frac{1}{90}\Delta t \Bigg(& 7Ac_3(0) + 32\exp(B\frac{t}{4})Ac_3(\frac{t}{4}) \\ & + 12\exp(B\frac{t}{2})Ac_3(\frac{t}{2}) + 32\exp(B\frac{3t}{4})Ac_3(\frac{3t}{4}) + 7\exp(Bt)Ac_3(t)\Bigg), \end{aligned} \quad (76)$$

where $c_3(t)$ is evaluated with the Simpson rule or a further third order method. We have $\Delta t = t - t_n$.

Remark 4. The same result can also be derived by applying the fourth order Gauss Runge Kutta method.

In the next section we describe the numerical results of our methods.

5 Numerical Examples

In the first example, we applied our iterative scheme with its underlying numerical approximations to differential equations.

5.1 First Experiment: Linear Ordinary Differential Equation

We deal with the linear ordinary differential equation:

$$\frac{\partial u(t)}{\partial t} = \begin{pmatrix} -\lambda_1 & \lambda_2 \\ \lambda_1 & \lambda_2 \end{pmatrix} c, \quad (77)$$

with initial condition $u_0 = (1, 1)$ on the interval $[0, T]$.
The analytical solution is given by:

$$u(t) = \begin{pmatrix} c_1 - c_2 \exp(-(\lambda_1 + \lambda_2)t) \\ \frac{\lambda_1}{\lambda_2} c_1 + c_2 \exp(-(\lambda_1 + \lambda_2)t) \end{pmatrix}, \quad (78)$$

where

$$c_1 = \frac{2}{1 + \frac{\lambda_1}{\lambda_2}} \quad , \quad c_2 = \frac{1 - \frac{\lambda_1}{\lambda_2}}{1 + \frac{\lambda_1}{\lambda_2}}. \quad (79)$$

We split our linear operator into two operators by setting:

$$\frac{\partial u(t)}{\partial t} = \begin{pmatrix} -\lambda_1 & 0 \\ \lambda_1 & 0 \end{pmatrix} u + \begin{pmatrix} 0 & \lambda_2 \\ 0 & -\lambda_2 \end{pmatrix} u. \tag{80}$$

We choose $\lambda_1 = 0.25$ and $\lambda_2 = 0.5$ on the interval [0,1].

We therefore have the operators:

$$A = \begin{pmatrix} -0.25 & 0 \\ 0.25 & 0 \end{pmatrix} \quad , \quad B = \begin{pmatrix} 0 & 0.5 \\ 0 & -0.5 \end{pmatrix}. \tag{81}$$

For the integration method we use a time-step size of $h = 10^{-3}$.

As initialization of our iterative method we use $c_{-1} \equiv 0$

From the examples one can see that the order increases by one per iteration step.

In Tables 1-3 we apply the different integration rules to our iterative scheme.

Iterative Steps	Number of splitting-partitions	err_1	err_2
1	1	4.5321e-002	4.5321e-002
1	10	3.9664e-003	3.9664e-003
1	100	3.9204e-004	3.9204e-004
2	1	7.6766e-003	7.6766e-003
2	10	6.6383e-005	6.6383e-005
2	100	6.5139e-007	6.5139e-007
3	1	4.6126e-004	4.6126e-004
3	10	4.1883e-007	4.1883e-007
3	100	5.9520e-009	5.9521e-009
4	1	4.6828e-005	4.6828e-005
4	10	1.3954e-009	1.3953e-009
4	100	5.5352e-009	5.5351e-009
5	1	1.9096e-006	1.9096e-006
5	10	5.5527e-009	5.5528e-009
5	100	5.5355e-009	5.5356e-009

Table 1. Numerical results for the first example with the iterative splitting method and the second-order Trapezoidal rule.

Remark 5. Here we see the benefit of higher quadrature rules in combination with the iterative operator splitting scheme, (see Figure 1). We obtain the best result with a fourth-order Gauss Runge-Kutta method. Such improved quadrature rules and the expansion of the integral formulation show that our method has considerable computational benefits.

Iterative Steps	Number of splitting-partitions	err_1	err_2
1	1	4.5321e-002	4.5321e-002
1	10	3.9664e-003	3.9664e-003
1	100	3.9204e-004	3.9204e-004
2	1	7.6766e-003	7.6766e-003
2	10	6.6385e-005	6.6385e-005
2	100	6.5312e-007	6.5312e-007
3	1	4.6126e-004	4.6126e-004
3	10	4.1334e-007	4.1334e-007
3	100	1.7864e-009	1.7863e-009
4	1	4.6833e-005	4.6833e-005
4	10	4.0122e-009	4.0122e-009
4	100	1.3737e-009	1.3737e-009
5	1	1.9040e-006	1.9040e-006
5	10	1.4350e-010	1.4336e-010
5	100	1.3742e-009	1.3741e-009

Table 2. Numerical results for the first example with the iterative splitting method and third-order BDF3.

5.2 Second Experiment: Large ODE systems

In the second experiment, we deal a large ODE system to verify the benefit to more complicate differential equations. Here the computational cost is increasing and the balance between iterative steps and time-step size is considered.

We deal with the 10×10 ODE system:

$$\partial_t u_1 = -\lambda_{1,1} u_1 + \lambda_{2,1} u_2 + \cdots + \lambda_{10,1} u_{10}, \tag{82}$$

$$\partial_t u_2 = \lambda_{1,2} u_1 - \lambda_{2,2}(t) u_2 + \cdots + \lambda_{10,2} u_{10} \ , \tag{83}$$

$$\vdots \tag{84}$$

$$\partial_t u_{10} = \lambda_{1,10} u_1 + \lambda_{2,10}(t) u_2 + \cdots - \lambda_{10,10} u_{10} \ , \tag{85}$$

$$u_1(0) = u_{1,0}, \ldots, \ u_{10}(0) = u_{10,0} \text{ (initial conditions)} \ , \tag{86}$$

where $\lambda_1(t) \in \mathbb{R}^+$ and $\lambda_2(t) \in \mathbb{R}^+$ are the decay factors and $u_{1,0}, \ldots, u_{10,0} \in \mathbb{R}^+$. We have the time-interval $t \in [0, T]$.

We rewrite the equation (82) in operator notation, we concentrate us to the following equations :

$$\partial_t u = A(t) u + B(t) u \ , \tag{87}$$

where $u_1(0) = u_{10} = 1.0$, $u_2(0) = u_{20} = 1.0$ are the initial conditions, where the operators are

$$A = \begin{pmatrix} -\lambda_{1,1}(t) \ldots \lambda_{10,1}(t) \\ \lambda_{1,5}(t) \ \ldots \lambda_{10,5}(t) \\ 0 \quad \ldots \quad 0 \end{pmatrix} \ , \ B = \begin{pmatrix} 0 \quad 0 \\ \lambda_{1,6}(t) \ \ldots \ \ \lambda_{10,6}(t) \\ \lambda_{1,10}(t) \ldots -\lambda_{10,10}(t) \end{pmatrix} \ . \tag{88}$$

Iterative Steps	Number of splitting-partitions	err_1	err_2
1	1	4.5321e-002	4.5321e-002
1	10	3.9664e-003	3.9664e-003
1	100	3.9204e-004	3.9204e-004
2	1	7.6766e-003	7.6766e-003
2	10	6.6385e-005	6.6385e-005
2	100	6.5369e-007	6.5369e-007
3	1	4.6126e-004	4.6126e-004
3	10	4.1321e-007	4.1321e-007
3	100	4.0839e-010	4.0839e-010
4	1	4.6833e-005	4.6833e-005
4	10	4.1382e-009	4.1382e-009
4	100	4.0878e-013	4.0856e-013
5	1	1.9040e-006	1.9040e-006
5	10	1.7200e-011	1.7200e-011
5	100	2.4425e-015	1.1102e-016

Table 3. Numerical results for the first example with the iterative splitting method and fourth-order Gauss RK.

$$\lambda_{1,1} = 0.09, \lambda_{2,1} = 0.01, \ldots, \lambda_{10,1} = 0.01 \quad (89)$$

$$\vdots$$

$$\lambda_{1,10} = 0.01, \ldots \lambda_{9,10} = 0.01, \ldots, \lambda_{10,10} = 0.09. \quad (90)$$

Further we also considered $\tilde{A} = A^t$ and $\tilde{B} = B^t$ as a combination of operators A and B. Such combinations have not influence the splitting scheme, see also [5], and we have considered a balance of time and iteration.

In various tests, we considered an optimal balancing time and iteration, which is presented in Figure 2.

Remark 6. For larger systems of differential equations, we obtain the same higher order results as for lower systems. For more accuracy also the computational time for at least a 3rd order iterative scheme is less expensive. At least a balance between the order and the computational time is important, while lower order schemes save computational time with moderate accuracy, e.g. 10^{-12}, higher order schemes have their computational benefits above an accuracy of 10^{-15}.

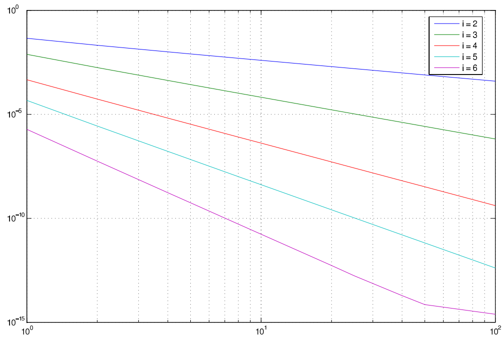

Fig. 1. Convergence rates from two to six iterations.

5.3 Third Experiment: Diffusion Equation

In a next example, we deal with partial differential equation and verify also the theoretical results. The equation is given as:

$$\frac{\partial u}{\partial t} = D \frac{\partial^2 u}{\partial x^2}, \text{ in } \Omega \times [0,T], \tag{91}$$

$$u(x,0) = \sin(\pi x), \text{ on } \Omega, \tag{92}$$

$$u = 0, \text{ on } \partial\Omega \times [0,T], \tag{93}$$

with exact solution

$$u_{\text{exact}}(x,t) = \sin(\pi x) \exp(-D\pi^2 t) .$$

We choose $D = 0.0025$, $t \in [0,T] = [0,1]$ and $x \in \Omega = [0,1]$.
For the spatial discretization we use an upwind finite difference discretization:

$$\partial^- \partial^+ u_i = \frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2} . \tag{94}$$

and we set the space step size to $\Delta x = \frac{1}{100}$.

Our operator is then given as

$$A = \frac{D}{\Delta x^2} \cdot \begin{pmatrix} 0 & & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & & 0 \end{pmatrix} \tag{95}$$

We split the space-interval into two intervals by splitting the Matrix A into two Matrixes:

$$\begin{pmatrix} A_1 \\ A_2 \end{pmatrix} := A .$$

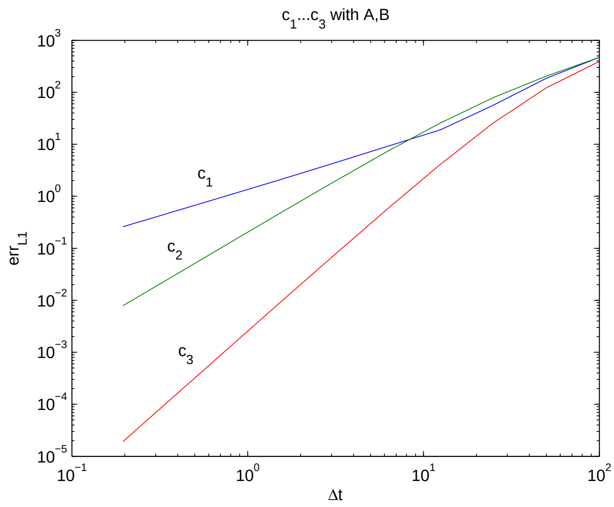

Fig. 2. Numerical errors of the splitting schemes, x-axis: time, y-axis: L_1-error, results of the iterative splitting schemes with (1- 3 iterative steps).

We now solve the problem

$$\partial_t u = A_1 u + A_2 u.$$

We use the Iterative Operator Splitting with Pade approximant and different discretization steps of Δx.

Based on the CFL condition of discretized scheme with the iterative splitting scheme we have:

$$\frac{2D}{\Delta x^2} \, \tau^i \leq err \leq 1, \tag{96}$$

$$2D \, N^2 \, \tau^i \leq err \leq 1, \tag{97}$$

where $\Delta x = \frac{1}{N}$ is the spatial step and N are the number of spatial points, τ is the time step and i the order of the iterative scheme. Further D is the diffusion parameter.

We obtain for the restriction of the time-step :

$$\frac{2D}{\Delta x^2} \, \tau^i \leq err \leq 1 \tag{98}$$

$$\tau \leq \left(\frac{err}{2D \, N^2})^{\frac{1}{i}} \right). \tag{99}$$

We have to consider a balance between the spatial steps size, time step size and also iterative steps.

Here we could not obtain more accurate results, while using additionally more iterative steps concluded improved results. The CFL condition is important, while balancing time and space. Additionally we see the improvement with more iterative steps, if we consider sufficient small time steps.

The numerical results of the iterative schemes are given in Figure 3.

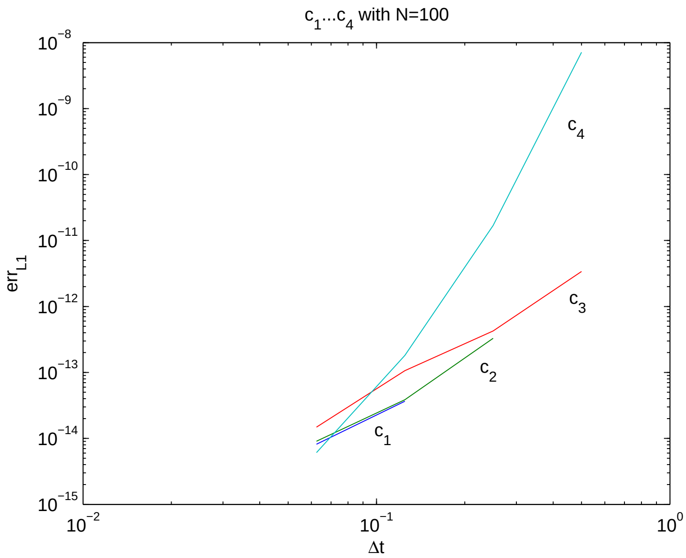

Fig. 3. Numerical result for the exponential splitting schemes of the diffusion equation with $N = 100$ spatial points.

Remark 7. For partial differential equations, additional balancing problems between time and spatial scales are involved and we have to deal with the CFL condition. We obtain the same higher order results as for ODE systems, by considering on optimal CFL conditions and control the spatial scale with the time scale. For more accuracy we have taken into account at least four iterative steps, which benefits for small time steps and we obtain the higher accuracy. Therefore also fine spatial grids are necessary to see the benefits of iterative splitting schemes.

5.4 Fourth Example: Convection-Diffusion-Reaction Equation

In a next example, we consider a convection-diffusion-reaction equation, which is given as

$$R\partial_t u + v\partial_x u - D\partial_{xx} u = -\lambda u \ , \ \text{in } \Omega \times [t_0, t_{\text{end}}), \tag{100}$$

$$u(x, t_0) = u_{\text{exact}}(x, t_0) \ , \ \text{in } \Omega, \tag{101}$$

$$u(0, t) = u_{\text{exact}}(0, t) \ , \ u(L, t) = u_{\text{exact}}(L, t). \tag{102}$$

We choose $x \in [0, L] = \Omega$ with $L = 30$ and $t \in [t_0, t_{\text{end}}] = [10^4, 2 \cdot 10^4]$.
Furthermore, we have $\lambda = 10^{-5}$, $v = 0.001$, $D = 0.0001$ and $R = 1.0$. The analytical solution is given by

$$u_{\text{exact}}(x, t) = \frac{1}{2\sqrt{D\pi t}} \exp\left(-\frac{(x - vt)^2}{4Dt}\right) \exp(-\lambda t). \tag{103}$$

To begin away from the singular point of the exact solution, we start from the time point $t_0 = 10^4$.

Our split operators are

$$A = \frac{D}{R}\partial_{xx}, \ B = -\frac{1}{R}(\lambda + v\partial_x). \tag{104}$$

For the spatial discretization we use the finite differences with a spatial grid width of $\Delta x = \frac{1}{10}$ and $\Delta x = \frac{1}{100}$.
To solve the linear ordinary differential equation we solve the underlying integrals of the iterative scheme, is with a Simpson rule, which is a third order time integration method, see Subsection 4.2. Our numerical results are presented in Table 4. We choose different iteration steps and time partitions, and show the error between the analytical and numerical solution in the supremum norm.

Iteration steps i	Number of splitting partitions n	err at $x = 18$	err at $x = 20$	err at $x = 22$
1	10	9.8993e-002	1.6331e-001	9.9054e-002
2	10	9.5011e-003	1.6800e-002	8.0857e-003
3	10	9.6209e-004	1.9782e-002	2.2922e-004
4	10	8.7208e-004	1.7100e-002	1.5168e-005

Table 4. Numerical results for the third example with iterative operator splitting method and Simpson rule with $\Delta x = 10^{-2}$ and $\Delta t = 10^3$.

Figure 4 shows the initial solution at $t_0 = 10^4$, and the analytical as well as the numerical solutions at $t_{\text{end}} = 2 \cdot 10^4$ of the convection-diffusion-reaction equation.

Here we have also to balance the spatial and time steps. We can reduce the error between the analytical and the numerical solution by using more iteration

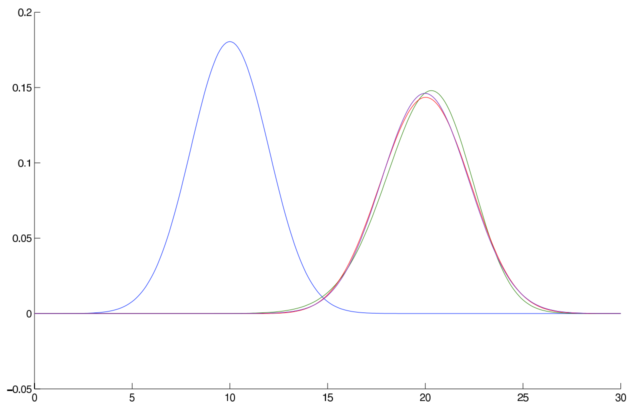

Fig. 4. Initial and computed results with iterative splitting method and Simpson rule; the initial solution is plotted as blue line, the iterative solution is plotted as green line and the exact solution is plotted as red line.

steps, but taken into account additionally sufficient time-steps. If we restrict ourselves to an error of 10^{-4}, we obtain an effective computation with 3 iteration steps and 10 time partitions.

Remark 8. For the convection-diffusion-reaction equation, we have also obtained the theoretical results. More iterative steps reduce the splitting error and simplify the computation. We also need to take into account the spatial discretization and the influence to the time step size. We applied a fine grid step on the spatial discretization, such that the error of the time discretization method is dominant. We obtain an optimal efficiency of the iteration steps and the time partitions, if we use 3 iteration steps and 10 time partitions. Here we see also an order reduction of the method.

6 Conclusions and Discussions

We have presented an iterative operator-splitting method as competitive method to compute split-able differential equations. On the basis of integral formulation of the iterative scheme, we presented the error analysis and its local error for bounded operators. Numerical examples confirm the method's application to ordinary differential and partial differential equations. Here an optimal balance of time, space and iteration steps are necessary and an order reduction is obtained. In the future we will focus on the development of improved operator-splitting methods with respect to their application in nonlinear differential equations.

References

1. B. Davis. *Integral Transform and Their Applications.* Applied Mathematical Sciences, 25, Springer Verlag, New York, Heidelberg, Berlin, 1978 .
2. K.-J. Engel, R. Nagel, *One-Parameter Semigroups for Linear Evolution Equations.* Springer-Verlag, Heidelberg, New York, 2000.
3. I. Farago and J. Geiser. *Iterative Operator-Splitting Methods for Linear Problems.* Preprint No. 1043 of the Weierstrass Institute for Applied Analysis and Stochastics, (2005) 1-18. International Journal of Computational Science and Engineering, accepted September 2007.
4. J. Geiser. *Iterative Operator-Splitting Methods with higher order Time-Integration Methods and Applications for Parabolic Partial Differential Equations.* Journal of Computational and Applied Mathematics, Elsevier, Amsterdam, The Netherlands, 217, 227-242, 2008.
5. J. Geiser. *Higher order splitting methods for differential equations: Theory and applications of a fourth order method. Numerical Mathematics: Theory, Methods and Applications.* Global Science Press, Hong Kong, China, accepted, April 2008.
6. J. Geiser and L. Noack. *Iterative operator-splitting methods for nonlinear differential equations and applications of deposition processes* Preprint 2008-4, Humboldt University of Berlin, Department of Mathematics, Germany, 2008.
7. J. Geiser. *Decomposition Methods for Differential Equations: Theory and Applications.* Chapman & Hall/CRC, Series: Numerical Analysis and Scientific Computing Series, edited by Magoules and Lai, 2009.
8. R. Glowinski. *Numerical methods for fluids.* Handbook of Numerical Analysis, Gen. eds. P.G. Ciarlet, J. Lions, Vol. IX, North-Holland Elsevier, Amsterdam, The Netherlands, 2003.
9. E. Hansen and A. Ostermann. *Exponential splitting for unbounded operators.* Mathematics of Computation, 78, 1485-1496, 2009.
10. M. Hochbruck and A. Ostermann. *Explicit Exponential Runge-Kutta Methods for Semilinear Parabolic Problems.* SIAM Journal on Numerical Analysis, 43:3, 1069-1090, 2005.
11. T. Jahnke and C. Lubich. *Error bounds for exponential operator splittings.* BIT Numerical Mathematics, 40:4, 735-745, 2000.
12. J. Kanney, C. Miller and C. Kelley. *Convergence of iterative split-operator approaches for approximating nonlinear reactive transport problems.* Advances in Water Resources, 26:247–261, 2003.
13. C.T. Kelly. *Iterative Methods for Linear and Nonlinear Equations.* Frontiers in Applied Mathematics, SIAM, Philadelphia, USA, 1995.
14. G.I Marchuk. *Some applications of splitting-up methods to the solution of problems in mathematical physics.* Aplikace Matematiky, 1, 103-132, 1968.
15. I. Najfeld and T.F. Havel. *Derivatives of the matrix exponential and their computation.* Adv. Appl. Math, ftp://ftp.das.harvard.edu/pub/cheatham/tr-33-94.ps.gz, 1995.
16. R.S. Palais. *Seminar on the Atiyah-Singer Index Theorem.* Annals of Mathematics Studies, no. 57, Princeton University Press, 1965.
17. A. Pazy. *Semigroups of Linear Operators and Applications to Partial Differential Equations.* Applied Mathematical Sciences 44, Springer, Berlin, 1983.
18. A. Quarteroni and A. Valli. *Domain Decomposition Methods for Partial Differential Equations* Series: Numerical Mathematics and Scientific Computation, Clarendon Press, Oxford, 1999.

19. G. Strang. *On the construction and comparison of difference schemes*. SIAM J. Numer. Anal., 5, 506-517, 1968.
20. S. Vandewalle. *Parallel Multigrid Waveform Relaxation for Parabolic Problems*. Teubner Skripten zur Numerik, B.G. Teubner Stuttgart, 1993.