Weierstraß-Institut
für Angewandte Analysis und Stochastik

im Forschungsverbund Berlin e.V.

Preprint ISSN 0946 – 8633

Iterative Operator-Splitting Methods for Linear Problems

István Faragó[1] and Jürgen Geiser[2]

submitted: 11th July 2005

[1] Eötvös Loránd University
Pázmány Péter s. 1/c
H-1117 Budapest
Hungary
E-Mail: faragois@cs.elte.hu

[2] Weierstrass Institute
for Applied Analysis
and Stochastics
Mohrenstrasse 39
D-10117 Berlin
Germany
E-Mail: geiser@wias-berlin.de

No. 1043
Berlin 2005

2000 *Mathematics Subject Classification.* 80A20 80M25 74S10 76R50 35J60 35J65 65M99 65Z05.

Key words and phrases. Numerical simulation. Operator-Splitting Methods. Iterative Operator-Splitting Method. Ordinary Differential Equations. Consistency Analysis .

2003 *Physics Abstract Classifications.* 02.60.Cb 44.05.+e.

This work has been supported by the DFG Research Center MATHEON – "Mathematics for key technologies" (FZT 86) in Berlin.

Edited by
Weierstraß-Institut für Angewandte Analysis und Stochastik (WIAS)
Mohrenstraße 39
10117 Berlin
Germany

Fax: + 49 30 2044975
E-Mail: preprint@wias-berlin.de
World Wide Web: http://www.wias-berlin.de/

Abstract

The operator-splitting methods base on splitting of the complex problem into the sequence of the simpler tasks. A useful method is the iterative splitting method which ensures a consistent approximation in each step. In our paper, we suggest a new method which is based on the combination of splitting the time interval and the traditional iterative operator splitting. We analyze the local splitting error of the method. Numerical examples are given in order to demonstrate the method.

1 Introduction

Traditionally there are two fundamentally different algorithmic approaches to the solution to mathematical model of complex physical processes

- For the fully coupling approach, considered the discrete form, the governing equations are solved as a single, usually very complicated system.
- For the decoupling approach, considered the discrete form, the governing equations are decoupled in more simpler uncoupled sub-problems and are solved as more simpler tasks.

The operator-splitting methods belong to the second type and they are used to solve complex physical models of different nature especially in the geophysical and environmental physics. They are developed and applied in different works, see, e.g. [8] and [14] and references therein. The basic idea of the operator-splitting methods based on splitting of complex problem into a sequence of simpler tasks, called split sub-problems. In the traditional operator splitting methods the solutions to the several sub-split problems are not approximations of the solution to the original un-split problem, only when executing the full cycle for one splitting step, we have consistency, see, e.g. [5] and [8]. There is an other class of splitting methods, the iterative operator method, where a system of one-step iterative methods is constructed on the whole interval and each sub-iteration requires to solve a problem only with one sub-operator from the complex original problem, see, e.g. [9] and the detailed references therein. In this case, the different splitting solution are consistent to the original solution on each steps. However, the algorithmic realization of this method leads to some difficulties.

In our paper, we propose a new iterative operator splitting method which is a combination of the traditional operator splitting (de-coupling of the time interval into

the smaller parts with the splitting time-step) and the iterative splitting method (on each split time-interval we use the one-step iterative methods). In some sense, our approach is similar to the ADI-iteration process.

Our paper is organized as follows.

In the Section 2, we investigate the traditional operator splitting methods and analyse their accuracy. In the next section, we introduce the iterative splitting method on split time intervals. In the Section 4, we analyze the accuracy (local splitting error) of this method, which, in case of stability, ensures also the convergence. The convergence-order of the method is derived. For the practical realization of this method we propose the interpolation of initial values of the intermediate split solution. We also investigate the consistency (and convergence) of this modified (interpolated) problem and give a useful recommendation to the choice of the iteration step in the algorithm. In the Section 5, we investigate different numerical examples which shows the validity of our theoretical results. We close the paper with some conclusions and plans for the further work.

2 Traditional Operator-Splitting Methods

In the following, we describe those traditional operator-splitting methods which are widely used for the solution to the real-life problems. We focus our attention to the case of two linear operators, i.e. we consider the Cauchy problem

$$\partial_t c(t) = Ac(t) + Bc(t) \quad t \in (0,T), \quad c(0) = c_0, \tag{2.1}$$

whereby the initial function c_0 is given, and A and B are assumed to be bounded linear operators in the Banach-space $\mathbf{X}$ with $A, B : \mathbf{X} \to \mathbf{X}$. In realistic applications the operators correspond to physical operators, e.g. space convection and diffusion operators.

2.1 Sequential operator-splitting method

First, we describe the simplest operator-splitting, which is called *sequential operator splitting*. The sequential operator-splitting method is introduced in [5], as a method, which solves two sub-problems sequentially on sub-intervals $[t^n, t^{n+1}]$, where $n = 0, 1, \ldots, N-1$, $t^0 = 0$ and $t^N = T$. The different sub-problems are connected via the initial conditions. This means that we replace the original problem (2.1) with the sub-problems on the sub-intervals

$$\begin{aligned} &\frac{\partial c^*(t)}{\partial t} = Ac^*(t), \quad t \in (t^n, t^{n+1}) \quad \text{with } c^*(t^n) = c^n_{\text{sp}}, \\ &\frac{\partial c^{**}(t)}{\partial t} = Bc^{**}(t), \quad t \in (t^n, t^{n+1}) \quad \text{with } c^{**}(t^n) = c^*(t^{n+1}), \end{aligned} \tag{2.2}$$

for $n = 0, 1, \ldots, N-1$, whereby $c_{\rm sp}^0 = c_0$ is given from (2.1). The approximated split solution at the point $t = t^{n+1}$ is defined as $c_{\rm sp}^{n+1} = c^{**}(t^{n+1})$.

Clearly, the change of the original problems with the sub-problems usually results some error, called *local splitting error*. Obviously, the local splitting error of the sequential operator splitting method can be derived as follows, cf. [8],

$$\begin{aligned} \rho_n &= \frac{1}{\tau_n}(\exp(\tau_n(A+B)) - \exp(\tau_n B)\exp(\tau_n A))\, c_{\rm sp}^n \\ &= \frac{1}{2}\tau_n [A,B]\, c(t^n) + \mathcal{O}(\tau_n^2), \end{aligned} \tag{2.3}$$

whereby the splitting time-step is defined as $\tau_n = t^{n+1} - t^n$. We define $[A,B] := AB - BA$ as commutator of A and B. Consequently, the splitting error is $O(\tau_n)$ when the operators A and B do not commute. When the operators commute then the method is exact, see [8]. Hence, by definition, the sequential operator splitting is called *first order splitting method.*

2.2 Symmetrically weighted sequential operator splitting

For non commuting operators the sequential operator splitting is not symmetric w.r.t. the operators A and B and it has of first order accuracy. However, in many practical cases we require splittings of higher order accuracy. We can achieve this by the following modified splitting method, called symmetrically weighted sequential operator splitting which is already symmetrical w.r.t. the operators.

The algorithms reads as follows. We consider again the Cauchy problem (2.1) and we define the operator-splitting on the time interval $[t^n, t^{n+1}]$ (where $t^{n+1} = t^n + \tau_n$) as

$$\begin{aligned} \frac{\partial c^*(t)}{\partial t} &= Ac^*(t), \quad \text{with } c^*(t^n) = c_{\rm sp}^n, \\ \frac{\partial c^{**}(t)}{\partial t} &= Bc^{**}(t), \quad \text{with } c^{**}(t^n) = c^*(t^{n+1}), \end{aligned} \tag{2.4}$$

and

$$\begin{aligned} \frac{\partial v^*(t)}{\partial t} &= Bv^*(t), \quad \text{with } v^*(t^n) = c_{\rm sp}^n, \\ \frac{\partial v^{**}(t)}{\partial t} &= Av^{**}(t), \quad \text{with } v^{**}(t^n) = v^*(t^{n+1}), \end{aligned} \tag{2.5}$$

where $c_{\rm sp}^n$ is known.

Then the approximation at the next time-level t^{n+1} is defined as

$$c_{\rm sp}^{n+1} = \frac{c^{**}(t^{n+1}) + v^{**}(t^{n+1})}{2}. \tag{2.6}$$

The splitting error of this operator splitting method is derived as follows, cf. [2]

$$\begin{aligned}\rho_n = \frac{1}{\tau_n}\{&\exp(\tau_n(A+B))- \\ -\frac{1}{2}&[\exp(\tau_n B)\exp(\tau_n A)+\exp(\tau_n A)\exp(\tau_n B)]\}c(t^n).\end{aligned} \tag{2.7}$$

An easy computation shows that in the general case

$$\rho_n = \mathcal{O}(\tau_n^2), \tag{2.8}$$

i.e. the method is of second order accurate. We note that in the case of commuting operators A and B the method is exact, i.e. the splitting error vanishes.

2.3 Strang-Marchuk operator-splitting method

One of the most popular and widely used operator-splittings is the so-called *Strang operator-splitting (or Strang-Marchuk operator-splitting)*, which reads as follows [11, 12]

$$\begin{aligned}&\frac{\partial c^*(t)}{\partial t} = Ac^*(t), \text{ with } t^n \le t \le t^{n+1/2} \text{ and } c^*(t^n) = c^n_{\text{sp}}, \\ &\frac{\partial c^{**}(t)}{\partial t} = Bc^{**}(t), \text{ with } t^n \le t \le t^{n+1} \text{ and } c^{**}(t^n) = c^*(t^{n+1/2}), \\ &\frac{\partial c^{***}(t)}{\partial t} = Ac^{***}(t), \text{ with } t^{n+1/2} \le t \le t^{n+1} \text{ and } c^{***}(t^{n+1/2}) = c^{**}(t^{n+1}),\end{aligned} \tag{2.9}$$

where $t^{n+1/2} = t^n + 0.5\tau_n$ and the approximation on the next time level t^{n+1} is defined as $c^{n+1}_{\text{sp}} = c^{***}(t^{n+1})$.

The splitting error of the Strang splitting is

$$\rho_n = \frac{1}{24}\tau_n^2([B,[B,A]] - 2[A,[A,B]])\, c(t^n) + O(\tau_n^3), \tag{2.10}$$

see, e.g. [8]. This means that this operator-splitting is of second order, too. We note that under some special conditions for the operators A and B, the Strang splitting has third order accuracy and even can be exact, see [5].

In the next section, we present some other type of operator-splitting methods which are based on the combination of the operator-splitting and the iterative methods.

3 Iterative operator-splitting method

The traditional operator-splittings have several drawbacks –besides their benefits–

- For non-commuting operators we may have a very large constant in the local splitting error which requires the use of unrealistically small splitting time step.

- Within a full splitting step in one sub-interval the inner values are not approximation to the solution to the original problem.
- Splitting the original problem into the different sub-problems with one operator, i.e. neglecting the other components, is physically questionable.

In order to avoid the above problems, one can use the iterative operator splitting on the interval $[0, T]$, cf. [9]. In the following, we suggest the modification of this method by introducing the splitting time discretization. We suggest an algorithm which is based on the iteration for the fixed sequential operator splitting discretization with the step-size τ_n. On the time interval $[t^n, t^{n+1}]$ we solve the following sub-problems consecutively, for $i = 1, 3, 5, \ldots, 2m+1$.

$$\frac{\partial c_i(t)}{\partial t} = Ac_i(t) \ + \ Bc_{i-1}(t), \text{ with } c_i(t^n) = c_{\text{sp}}^n, \tag{3.1}$$

$$\frac{\partial c_{i+1}(t)}{\partial t} = Ac_i(t) \ + \ Bc_{i+1}(t), \text{ with } c_{i+1}(t^n) = c_{\text{sp}}^n, \tag{3.2}$$

where $c_0(t)$ is any fixed function for each iteration. (As before, c_{sp}^n denotes the known split approximation at the time level $t = t^n$.) The split approximation at the time-level $t = t^{n+1}$ is defined as $c_{\text{sp}}^{n+1} = c_{2m+1}(t^{n+1})$. (Clearly, the functions $c_k(t)$ $(k = i-1, i, i+1)$ depend on the interval $[t^n, t^{n+1}]$, too, but, for the sake of simplicity, in our notation, we omit the dependence on n.)

The algorithm (3.1) and (3.2) is an iterative method which on each steps consists of both operators A and B. Hence, in these equations, there is no real separation of the different physical processes. However we note that, due to the sub-division of the time interval into the sub-intervals, this process differs from the simple fix-point iteration and turns it into a more efficient numerical method.

We want remark that the algorithm (3.1) and (3.2) is a real operator splitting dealing with the equation (3.1) requires to solve a problem with the operator A, and (3.2) requires to solve a problem with the operator B. Hence, like in the sequential operator splitting we separate the two operators.

4 Analysis of the iterative operator-splitting method

In this section, we analyze the consistency and the order of the iterative operator-splitting method. First, in the Section 4.1, we consider the original (3.1) and (3.2) algorithm and prove its consistency and define the order of the local splitting error.

Dealing with (3.1) and (3.2) requires the knowledge of the functions $c_{i-1}(t)$ and $c_i(t)$ on the whole interval $[t^n, t^{n+1}]$, which is typically not the case, since we know their values only at several points of the split interval. Hence, typically we can define only some interpolation to these functions. In the Section 4.2, we prove the consistency of such a modified algorithm.

4.1 Local error analysis of the iterative operator-splitting method

In the following we will analyze the consistency and the order of the local splitting error of the method (3.1)–(3.2) for the linear bounded operators $A, B : \mathbf{X} \to \mathbf{X}$, where $\mathbf{X}$ is a Banach-space, cf. [13].

Theorem 4.1. *Let $A, B \in \mathcal{L}(X)$ are given linear bounded operators. We consider the abstract Cauchy problem*

$$\begin{aligned} \partial_t c(t) &= Ac(t) + Bc(t), \quad 0 < t \leq T, \\ c(0) &= c_0. \end{aligned} \tag{4.1}$$

Then the problem (4.1) has a unique solution; the iteration (3.1)–(3.2) by $i = 1, 3, \ldots, 2m+1$ is consistent with the order of the consistency $\mathcal{O}(\tau_n^{2m})$.

Proof. Since $A + B \in \mathcal{L}(X)$ therefore it is a generator of a uniformly continuous semi-group, hence the problem (4.1) has a unique solution $c(t) = \exp((A+B)t)c_0$.

Let us consider the iteration (3.1)–(3.2) on the sub-interval $[t^n, t^{n+1}]$. For the local error function $e_i(t) = c(t) - c_i(t)$ we have the relations

$$\begin{aligned} \partial_t e_i(t) &= Ae_i(t) + Be_{i-1}(t), \quad t \in (t^n, t^{n+1}], \\ e_i(t^n) &= 0, \end{aligned} \tag{4.2}$$

and

$$\begin{aligned} \partial_t e_{i+1}(t) &= Ae_i(t) + Be_{i+1}(t), \quad t \in (t^n, t^{n+1}], \\ e_{i+1}(t^n) &= 0, \end{aligned} \tag{4.3}$$

for $m = 0, 2, 4, \ldots$, with $e_0(0) = 0$ and $e_{-1}(t) = c(t)$. In the following, we use the notations $\mathbf{X}^2$ 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 $\mathcal{E}_i(t)$, $\mathcal{F}_i(t) \in \mathbf{X}^2$ and the linear operator $\mathcal{A} : \mathbf{X}^2 \to \mathbf{X}^2$ are defined as follows

$$\mathcal{E}_i(t) = \begin{bmatrix} e_i(t) \\ e_{i+1}(t) \end{bmatrix}, \quad \mathcal{F}_i(t) = \begin{bmatrix} Be_{i-1}(t) \\ 0 \end{bmatrix}, \quad \mathcal{A} = \begin{bmatrix} A & 0 \\ A & B \end{bmatrix}. \tag{4.4}$$

Then, using the notations (4.4), the relations (4.2) and (4.3) can be written in the form

$$\begin{aligned} \partial_t \mathcal{E}_i(t) &= \mathcal{A}\mathcal{E}_i(t) + \mathcal{F}_i(t), \quad t \in (t^n, t^{n+1}], \\ \mathcal{E}_i(t^n) &= 0. \end{aligned} \tag{4.5}$$

Due to our assumptions, $\mathcal{A}$ is a generator of the one-parameter C_0-semi-group $(\exp \mathcal{A}t)_{t \geq 0}$, hence using the variations of constants formula, the solution to the abstract Cauchy problem (4.5) with homogeneous initial condition can be written as

$$\mathcal{E}_i(t) = \int_{t^n}^{t} \exp(\mathcal{A}(t-s))\mathcal{F}_i(s)ds, \quad t \in [t^n, t^{n+1}]. \tag{4.6}$$

(See, e.g. [3].) Hence, using the denotation

$$\|\mathcal{E}_i\|_\infty = \sup_{t\in[t^n,t^{n+1}]} \|\mathcal{E}_i(t)\| \ , \tag{4.7}$$

we have

$$\begin{aligned} \|\mathcal{E}_i\|(t) &\le \|\mathcal{F}_i\|_\infty \int_{t^n}^t \|\exp(\mathcal{A}(t-s))\| ds \\ &= \|B\|\|e_{i-1}\| \int_{t^n}^t \|\exp(\mathcal{A}(t-s))\| ds, \quad t \in [t^n, t^{n+1}]. \end{aligned} \tag{4.8}$$

Since $(\mathcal{A}(t))_{t\ge 0}$ is a semi-group, therefore the so called *growth estimation*

$$\|\exp(\mathcal{A}t)\| \le K \exp(\omega t), \quad t \ge 0, \tag{4.9}$$

holds with some numbers $K \ge 0$ and $\omega \in \mathbb{R}$, cf. [3].

- Assume that $(\mathcal{A}(t))_{t\ge 0}$ is a bounded or exponentially stable semi-group, i.e. (4.9), holds with some $\omega \le 0$. Then obviously the estimate

  $$\|\exp(\mathcal{A}t)\| \le K, \quad t \ge 0, \tag{4.10}$$

  holds, and hence, on base of (4.8), we have the relation

  $$\|\mathcal{E}_i\|(t) \le K\|B\|\tau_n\|e_{i-1}\|, \quad t \in [t^n, t^{n+1}]. \tag{4.11}$$

- Assume that $(\exp \mathcal{A}t)_{t\ge 0}$ has an exponential growth with some $\omega > 0$. Using (4.9), we have

  $$\int_{t^n}^t \|\exp(\mathcal{A}(t-s))\| ds \le K_\omega(t), \quad t \in [t^n, t^{n+1}], \tag{4.12}$$

  where

  $$K_\omega(t) = \frac{K}{\omega}\left(\exp(\omega(t-t^n)) - 1\right), \quad t \in [t^n, t^{n+1}]. \tag{4.13}$$

  Hence

  $$K_\omega(t) \le \frac{K}{\omega}\left(\exp(\omega\tau_n) - 1\right) = K\tau_n + \mathcal{O}(\tau_n^2). \tag{4.14}$$

The estimations (4.11) and (4.14) result in that

$$\|\mathcal{E}_i\|_\infty = K\|B\|\tau_n\|e_{i-1}\| + \mathcal{O}(\tau_n^2). \tag{4.15}$$

Taking into the account the definition of $\mathcal{E}_i$ and the norm $\|\cdot\|_\infty$, we obtain

$$\|e_i\| = K\|B\|\tau_n\|e_{i-1}\| + \mathcal{O}(\tau_n^2), \tag{4.16}$$

and hence

$$\|e_{i+1}\| = K_1\tau_n^2\|e_{i-1}\| + \mathcal{O}(\tau_n^3), \tag{4.17}$$

which proves our statement.

□

Remark 4.2. When A and B are matrices, i.e. (3.1) and (3.2) is a system of ordinary differential equations, for the growth estimation (4.9) we can use the concept of the logarithmic norm. see, e.g. [8]. Hence, for many important class of matrices we can prove the validity of (4.9) with $\omega \leq 0$.

Remark 4.3. We note that a huge class of important differential operators generate contractive semi-group. This means that for such problems –assuming the exact solvability of the split sub-problems– the iterative splitting method is convergent in second order to the exact solution.

Remark 4.4. We note that the assumption $A \in \mathcal{L}(X)$ can be weakened: It is enough to assume that the operator A is the generator of a C_0-semi-group.

Remark 4.5. When T is a sufficiently small number then we don't need the partition of the interval $[0, T]$ into the subintervals. For this case the convergence of the iteration (3.1) and (3.2) to the solution to the problem (4.1) follows immediately from Theorem 4.1 and the rate of the convergence is equal to the order of the local splitting error.

Remark 4.6. The estimate (4.25) shows that after the final iteration step ($i = 2m + 1$) we have the estimation

$$\|e_{2m+1}\| = K_m \|e_0\| \tau_n^{2m} + \mathcal{O}(\tau_n^{2m+1}). \tag{4.18}$$

This relation shows that the constant in the leading term strongly depends on the choice of the initial guess $c_0(t)$. When the choice is $c_0(t) = 0$ (see [9]) then $\|e_0\| = c$ (where c is the exact solution to the original problem) *and hence the error maybe very significant.*

4.2 Consistency analysis of the iterative operator-splitting method with interpolated split solutions

The algorithm (3.1) and (3.2) requires the knowledge of the functions $c_{i-1}(t)$ and $c_i(t)$ on the whole interval $[t^n, t^{n+1}]$. However, when we solve the split sub-problems, usually we apply some numerical methods which allow us to know the values of the above functions only at some points of the interval. Hence, typically we can define only some interpolation to the exact functions.

In the following, we consider and analyze the modified iterative process

$$\frac{\partial c_i(t)}{\partial t} = Ac_i(t) \; + \; Bc_{i-1}^{\text{int}}(t), \text{ with } \; c_i(t^n) = c_{\text{sp}}^n, \tag{4.19}$$

$$\frac{\partial c_{i+1}(t)}{\partial t} = Ac_i^{\text{int}}(t) \; + \; Bc_{i+1}(t), \text{ with } \; c_{i+1}(t^n) = c_{\text{sp}}^n, \tag{4.20}$$

where $c_k^{\text{int}}(t)$ (for $k = i - 1, i$) denotes an approximation of the function $c_k(t)$ on the interval $[t^n, t^{n+1}]$ with the accuracy $\mathcal{O}(\tau_n^p)$. (For simplicity, we assume the same order of accuracy p on each sub-intervals.)

Then the iteration (4.19) and (4.20) for the error function $\mathcal{E}_i(t)$ implies again the relation (4.5) with the modified right side, namely

$$\mathcal{F}_i(t) = \begin{bmatrix} Be_{i-1}(t) + Bh_{i-1}(t) \\ Ah_i(t) \end{bmatrix}, \tag{4.21}$$

where $h_k(t) = c_k(t) - c_k^{\text{int}}(t) = \mathcal{O}(\tau_n^p)$ for $k = i-1, i$. Hence

$$\|\mathcal{F}_i\|_\infty \leq \max\{\|B\|\,\|e_{i-1}\| + \|h_{i-1}\|; \|A\|\,\|h_i\|\}, \tag{4.22}$$

which results in the estimation

$$\|\mathcal{F}_i\|_\infty \leq \|B\|\,\|e_{i-1}\| + C\,\tau_n^p. \tag{4.23}$$

Consequently, for these assumptions the estimation (4.16) becomes the following

$$\|e_i\| \leq K(\|B\|\tau_n\|e_{i-1}\| + C\,\tau_n^{p+1}) + \mathcal{O}(\tau_n^2). \tag{4.24}$$

Therefore, for these assumptions the estimation (4.25) takes the modified form

$$\|e_{i+1}\| \leq K_1\tau_n^2\|e_{i-1}\| + KC\tau_n^{p+2} + KC\tau_n^{p+1} + \mathcal{O}(\tau_n^3). \tag{4.25}$$

Hence, we have

Theorem 4.7. *Let $A, B \in \mathcal{L}(X)$ are given linear bounded operators and consider the abstract Cauchy problem (4.1). Then for any interpolation of order $p \geq 1$ the iteration (4.19) and (4.20) by $i = 1, 3, \ldots 2m+1$ is consistent with the order of the consistency α where $\alpha = \min\{2m-1, p\}$.*

Remark 4.8. The above Theorem 4.7 shows that the number of the iteration should be chosen according to the order of the interpolation formula. For additional iteration we cannot expect more accurate solution.

Remark 4.9. We can use the piecewise constant approximation of the function $c_k(t)$, namely, $c_k^{\text{int}}(t) = c_k(t^n) = \text{const}$ which is known from the split solution. In this case, it is enough to take only two iterations in the case of sufficiently small discretization step-size.

Remark 4.10. The above analysis was done for the local error. The global error analysis –as usual– is the same and leads to the α-order convergence.

5 Numerical Results

In this section, we deal with test examples to verify our theoretical results, presented in the previous sections. We have chosen these model problems where the exact solutions are known such that we can compute the exact values of the errors.

In our examples, first we considered a simple scalar equation for the ordinary differential equation (ODE) and then the system of ODE's. We present the flexibility and improvement of the iterative operator splitting method. In the scheme of the various operator-splitting methods, we use also the analytical method of such reduced ODE's. We can verify the number of iteration-steps with respect to the order of the approximation of the functions.

5.1 First test-example: Scalar equation

We consider the following Cauchy problem for the scalar equation

$$u^{'}(t) = (-\lambda_1 - \lambda_2)u(t), \quad u(0) = u_0, \tag{5.1}$$

which has the exact solution

$$u(t) = \exp(-(\lambda_1 + \lambda_2))t)u_0. \tag{5.2}$$

For the problem (5.1), we split the right hand side into the sum of two scalar operators $A + B$, where $Au = -\lambda_1 u$ and $Bu = -\lambda_2 u$. According to the iterative splitting method (3.1) and (3.2), we apply the following algorithm

$$u_i^{'}(t) = -\lambda_1 u_i - \lambda_2 u_{i-1}, \tag{5.3}$$

$$u_{i+1}^{'}(t) = -\lambda_1 u_i - \lambda_2 u_{i+1}, \tag{5.4}$$

on the interval $t \in [0, \Delta t]$, with $u_i(0) = u_{i+1}(0) = u_0$ and $u_0(t) = 0$ and $i = 1, 3, 5, \ldots, 2m + 1$, where m is a positive integer.

For two equations (5.3) and (5.4), we can derive the analytical solutions as

$$u_i(\Delta t) = \exp(\lambda_1 \Delta t)\, u_i(0) + \frac{\lambda_2}{\lambda_1} u_{\mathrm{approx,i-1}}(t)\, (\exp(-\lambda_1 \Delta t) - 1), \tag{5.5}$$

$$u_{i+1}(\Delta t) = \frac{\lambda_1}{\lambda_2} u_{\mathrm{approx,i}}(t)\, (\exp(-\lambda_2 \Delta t) - 1) + \exp(\lambda_2 \Delta t)\, u_{i+1}(0), \tag{5.6}$$

where we have $u_{i+1}(0) = u_0$ and $u_i(0) = u_0$ with the index $i = 0, 2, 4, \ldots$. The initial conditions are $u_0(0) = u_0$ and $u_{-1}(0) = 0$.

Based on this solutions, we compare the results of iterative splitting method with the analytical solution to the complex equation.

The combination of handling the iterative steps and the time-partitions is therefore important. We consider a time-interval $[0, \Delta t]$ and divide this interval in n intervals with length $\tau = \frac{\Delta t}{n}$. We can improve the results by using smaller time-steps and more iterative steps. We can optimize the cost of computation and use more larger time-steps with less than $2 - 4$ iteration-steps, cf. Theorem 4.7.

For our example, we choose $\lambda_1 = 0.25$, $\lambda_2 = 0.5$ and $\Delta t = 1.0$, such that we get our exact solution with $u_{\mathrm{exact}} = \exp(-0.75) \approx 0.4723665$.

Number of time-partitions	Number of iterations i	u_{num}	$err = \lvert u_{\text{exact}} - u_{\text{num}} \rvert$
1	2	0.540346	6.798×10^{-2}
1	4	0.50034	2.797×10^{-2}
1	10	0.49653	2.416×10^{-2}
1	100	0.49653	2.416×10^{-2}
5	2	0.48207	9.710×10^{-3}
5	4	0.48032	7.955×10^{-3}
5	10	0.48031	7.946×10^{-3}
5	100	0.48031	7.946×10^{-3}
10	2	0.477004	4.637×10^{-3}
10	4	0.47656	4.196×10^{-3}
10	10	0.476562	4.196×10^{-3}
10	100	0.476562	4.196×10^{-3}
100	2	0.47281	4.449×10^{-4}
100	4	0.472807	4.4047×10^{-4}
100	10	0.472807	4.4047×10^{-4}
100	100	0.472807	4.4047×10^{-4}

Table 1: Numerical results for the first example of a ODE.

In the Table 1, we show the errors between the analytical and numerical results.

For small time-partitions and more iteration-steps, we get the best results and can improve them be more refinement. For an error-interval about 10^{-3} we suggest a coarser time-partition and see efficient and accurate results with 2 and 4 iteration-steps. From the theory we derive a interpolation order of 1 and convergent results after 3 time-steps, these fits with our results.

In the next example we present improved results done by the iterative method for a more complex example. We compare the iterative with the traditional results.

5.2 Second test-example of a systems of an ODE

Let us consider in this computations a more complicate example, where the motivation behind is a chemical reaction process for educts and products. The educts transform to the products with the velocity-rate λ_1 and reverse the products transform to the educts by the velocity-rate λ_2. Chemical reaction models and bio-remediation have such processes cf. [4] and [6].

We deal with the following equation :

$$\partial_t u_1 = -\lambda_1 u_1 + \lambda_2 u_2, \tag{5.7}$$

$$\partial_t u_2 = \lambda_1 u_1 - \lambda_2 u_2, \tag{5.8}$$

$$u_{10}(0) = u_{10},\ u_{20}(0) = u_{20}, \tag{5.9}$$

where $\lambda_1 \in \mathbb{R}^+$ and $\lambda_2 \in \mathbb{R}^+$ are the velocity-rate. Further, u_1 is the concentration of the educts and u_2 is the concentration of products.

We rewrite the equation-system (5.7)–(5.9) in operator notation, and end up with the following equations

$$\begin{aligned} \partial_t u &= Au + Bu\ , & (5.10)\\ u_0(0) &= u_0\ , u_1(0) = 0\ , & (5.11) \end{aligned}$$

where $u(t) = (u_1(t), u_2(t))^T$ for $t \in [0, \Delta t]$, and our spit operators are

$$A = \begin{pmatrix} -\lambda_1 & \lambda_2 \\ 0 & 0 \end{pmatrix},\ B = \begin{pmatrix} 0 & 0 \\ \lambda_1 & -\lambda_2 \end{pmatrix}. \quad (5.12)$$

We chose such an example to have $AB \neq BA$, therefore, we have a splitting error of first order for the usual sequential splitting methods, called A-B splitting.

For the complex equation-system (5.7)-(5.9) we can derive the analytical solution by integrating the system of ODE's

$$u_1(t) = u_{10} + u_{20} \exp(-(\lambda_1 + \lambda_2)t), \quad (5.13)$$

$$u_2(t) = \frac{\lambda_1}{\lambda_2}\, u_{10} - u_{20} \exp(-(\lambda_1 + \lambda_2)t). \quad (5.14)$$

To validate the methods and obtain the improved results, we compare the results of a first order method with the iterative method.

We have the parameters $\lambda_1 = 0.25$, $\lambda_2 = 0.5$ and the end-time $\Delta t = 1.0$ with these values we get the analytical solutions to our equation : $u_{1,\text{exact}} = 1.0$ and $u_{2,\text{exact}} = 0.73618$.

5.2.1 The A-B splitting method (first order method)

The traditional sequential splitting (A-B splitting method) is used as a first method and various time-partions are computed.

For this A-B splitting, we define the following numerical algorithm in an A- and B-step

A-step

$$\begin{aligned} \partial_t u_1^* &= -\lambda_1 u_1^* + \lambda_2 u_2^*, & (5.15)\\ \partial_t u_2^* &= 0, & (5.16)\\ u_1^*(0) &= u_{10},\ u_2^*(0) = u_{20}, & (5.17) \end{aligned}$$

B-step

$$\begin{aligned} \partial_t u_1^{**} &= 0, & (5.18)\\ \partial_t u_2^{**} &= \lambda_1 u_1^{**} - \lambda_2 u_2^{**}, & (5.19)\\ u_1^{**}(0) &= u_1^*(\Delta t),\ u_2^{**}(0) = u_2^*(\Delta t), & (5.20) \end{aligned}$$

where $t \in [0, \Delta t]$ and the result of the computation is $u(\Delta t) = (u_1^{**}(\Delta t), u_2^{**}(\Delta t))^t$.

For the equation-systems (5.15)–(5.17) and (5.18)–(5.20) we derive the analytical solutions and apply them in our numerical scheme, leading to

$$u_1^*(t) = u_{10} \exp(-\lambda_1 t) + u_{20} \frac{\lambda_2}{\lambda_1} , \tag{5.21}$$

$$u_2^*(t) = u_{20} , \tag{5.22}$$

and

$$u_1^{**}(t) = u_1^{**}(0) , \tag{5.23}$$

$$u_2^{**}(t) = u_1^{**}(0) \frac{\lambda_1}{\lambda_2} + u_2^{**}(0) \exp(-\lambda_2 t) , \tag{5.24}$$

and $u_1^{**}(0) = u_1^*(t)$, $u_2^{**}(0) = u_2^*(t)$.

We compute the A-B splitting with our given parameters $\lambda_1 = 0.25$, $\lambda_2 = 0.5$ and the initial conditions $u_0 = 1$, $u_1 = 1$ and the end-time $\Delta t = 1.0$. The results are compared with the analytical solution, cf. (5.13) and (5.14).

We present the numerical errors for the A-B splitting method and variation in the time-partitions in the Table 2.

Number of time-partitions	$u_{1,\text{num}}$	$u_{2,\text{num}}$	err_1	err_2
1	1.2211	0.8476	2.211×10^{-1}	1.105×10^{-1}
10	1.1802	0.8263	1.802×10^{-1}	9.01×10^{-2}
100	1.1763	0.8243	1.763×10^{-1}	8.815×10^{-2}

Table 2: Numerical results for the second example with the first order A-B splitting method.

In the Table 2, we see the decreasing of the error by smaller time-steps but we obtain a slow convergence rate. The classical splitting-method could not halfen the previous error and we propose an acceleration with the new iterative method.

The improved method is presented in the next subsection.

5.2.2 The iterative splitting method (improved method of higher order)

For the iterative splitting method, we have the following splitting equations of our system of ODE's. We divide in step i and $i+1$ as following

Step i

$$\partial_t u_1^i = -\lambda_1 u_1^i + \lambda_2 u_2^i, \tag{5.25}$$

$$\partial_t u_2^i = \lambda_1 u_1^{i-1} - \lambda_2 u_2^{i-1}, \tag{5.26}$$

$$u_1^i(0) = u_{10}, u_2^i(0) = u_{10},$$

where we have the initial conditions as $u_1^{-1}(0) = 0$ and $u_2^{-1}(0) = 0$.

Step $i+1$

$$\begin{aligned} \partial_t u_1^{i+1} &= -\lambda_1 u_1^i + \lambda_2 u_2^i, && (5.27)\\ \partial_t u_2^{i+1} &= \lambda_1 u_1^{i+1} - \lambda_2 u_2^{i+1}, && (5.28)\\ u_1^{i+1}(0) &= u_{10}, u_2^{i+1}(0) = u_{10}, \end{aligned}$$

where $t \in [0, \Delta t]$ and $i = 0, 2, 4, \ldots, 2m$ and $m > 0$.

For the step i and $i+1$, we can derive the analytical solutions and apply them in our numerical scheme. The analytical solutions are given as

$$\begin{aligned} u_1^i(t) &= u_{10}\exp(-\lambda_1 t) + u_{20}\frac{\lambda_2}{\lambda_1} \\ &+ u_1^{i-1}(t)(\lambda_2 t - \frac{\lambda_2}{\lambda_1}) + u_2^{i-1}(t)(-\frac{\lambda_2^2}{\lambda_1}t - \frac{\lambda_2^2}{\lambda_1^2}), && (5.29)\\ u_2^i(t) &= u_1^{i-1}(t)\lambda_1 t - u_2^{i-1}(t)\lambda_2 t + u_{20}, && (5.30) \end{aligned}$$

and

$$\begin{aligned} u_1^{i+1}(t) &= -u_1^i(t)\lambda_1 t + u_2^i(t)\lambda_2 t + u_{10}, && (5.31)\\ u_2^{i+1}(t) &= u_{10}\frac{\lambda_1}{\lambda_2} + u_{20}\exp(-\lambda_2 t) \\ &+ u_1^i(t)(-\frac{\lambda_1^2}{\lambda_2}t - \frac{\lambda_1^2}{\lambda_2^2}) + u_2^i(t)(\lambda_1 t - \frac{\lambda_1}{\lambda_2}), && (5.32) \end{aligned}$$

where $u_1^{-1}(0) = 0$ and $u_2^{-1}(0) = 0$ and $i = 0, 2, 4, \ldots, 2m$ and $m > 0$.

We compute with our given scheme, cf. equations (5.25)–(5.28) and our numerical results are presented in Table 3.

The numerical results show an improvement of effectivity in larger time-steps and lesser iteration steps by the iterative splitting method. Because of the higher order and of the accelerate algorithm for th iterative method. For non commutative operators, we get a first order result with the A-B splitting method and with the iterative splitting method we obtain improved convergence-rates by the higher order method. The iterative method is at least of second order and 3 iteration steps are enough to reach the optimal results, cf. Theorem 4.7.

For such complex situations, we will propose the new iterative splitting methods. In further works we will design new methods with both characteristics, robust A-B splitting method and higher order iterative splitting method.

6 Conclusions and Discussions

We present the mathematical background for the coupling of simple physical and one-dimensional software tools to multi-physical and multi-dimensional software-tools. Based on the operator splitting methods, we present the possible splitting

Number of time-partitions	Number of iterations	$u_{1,num}$	$u_{2,num}$	err_1	err_2
1	1	1.1743	0.7799	1.743×10^{-1}	4.373×10^{-2}
1	2	1.1316	0.7753	1.316×10^{-1}	3.919×10^{-2}
1	4	1.1279	0.7749	1.279×10^{-1}	3.879×10^{-2}
1	10	1.1276	0.7749	1.276×10^{-1}	3.875×10^{-2}
10	1	1.025	0.8053	2.52×10^{-2}	6.916×10^{-2}
10	2	1.024	0.8050	2.4×10^{-2}	6.88×10^{-2}
10	4	1.024	0.8050	2.4×10^{-2}	6.88×10^{-2}
10	10	1.024	0.8050	2.4×10^{-2}	6.88×10^{-2}
100	1	1.0025	0.8035	2.502×10^{-3}	6.732×10^{-2}
100	2	1.00248	0.8035	2.48×10^{-3}	6.732×10^{-2}
100	4	1.00248	0.8035	2.48×10^{-3}	6.732×10^{-2}
100	10	1.00248	0.8035	2.48×10^{-3}	6.732×10^{-2}

Table 3: Numerical results for the second example with the iterative splitting method.

methods and the errors. The discussion about the application of the splitting methods is done. We have compared different splitting methods and obtain improved convergence results for the iterative method. In the future, we will focus on the development of improved operator-splitting methods with respect to the application in nonlinear convection-diffusion-reaction-equations, arising from chemical and biological models, cf. [6] and [4] and heat-transfer problems, arising from crystal-growth, cf. [7],[10] and [1].

References

[1] N. Bubner, O. Klein, P. Philip, J. Sprekels, and K. Wilmanski. *A transient model for the sublimation growth of silicon carbide single crystals.* Journal of Crystal Growth, 205: 294-304, 1999.

[2] P. Csomós, I. Faragó, and A. Havasi. *Weighted sequential splittings and their analysis.* Comput. Math. Appl., (to appear)

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

[4] R.E. Ewing. Up-scaling of biological processes and multiphase flow in porous media. *IIMA Volumes in Mathematics and its Applications*, Springer-Verlag, 295 (2002), 195-215.

[5] I. Farago. *Splitting methods for abstract Cauchy problems.* Lect. Notes Comp.Sci. 3401, Springer Verlag, Berlin, pp. 35-45, 2005.

[6] J. Geiser. *Numerical Simulation of a Model for Transport and Reaction of Radionuclides.* Proceedings of the Large Scale Scientific Computations of Engineering and Environmental Problems, Sozopol, Bulgaria, 2001.

[7] J. Geiser, O. Klein, and P. Philip. *Numerical simulation of heat transfer in materials with anisotropic thermal conductivity: A finite volume scheme to handle complex geometries.* Preprint No.1033, Weierstraß-Institut für Angewandte Analysis und Stochastik, Berlin, 2005.

[8] W.H. Hundsdorfer and J. Verwer *Numerical solution of time-dependent advection-diffusion-reaction equations*, Springer, Berlin, (2003).

[9] 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.

[10] O. Klein, P. Philip, and J. Sprekels. *Modeling and simulation of sublimation growth of SiC bulk single crystals.* Interfaces and Free Boundaries, 6: 295–314, 2004.

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

[12] G. Strang. *On the construction and comparision of difference schemes.* SIAM J. Numer. Anal., 5:506–517, 1968.

[13] R.S. Varga. *Matrix Iterative Analysis.* Prentice-Hall, Inc., Englewood Cliffs, New Jersy, 1962.

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