



**HAL**  
open science

# An Iterative Splitting Method via Waveform Relaxation

Juergen Geiser

► **To cite this version:**

Juergen Geiser. An Iterative Splitting Method via Waveform Relaxation. International Journal of Computer Mathematics, 2011, pp.1. 10.1080/00207160.2011.610892 . hal-00733565

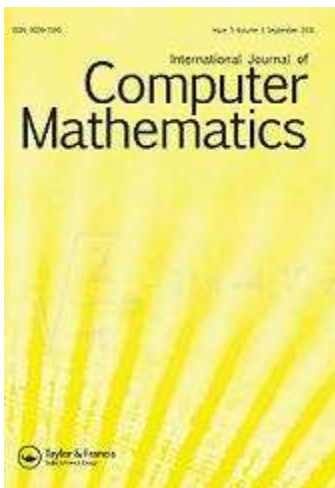
**HAL Id: hal-00733565**

**<https://hal.science/hal-00733565>**

Submitted on 19 Sep 2012

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



**An Iterative Splitting Method via Waveform Relaxation**

Journal:	<i>International Journal of Computer Mathematics</i>
Manuscript ID:	GCOM-2009-0813-B.R2
Manuscript Type:	Original Article
Date Submitted by the Author:	19-Oct-2010
Complete List of Authors:	Geiser, Juergen; Humboldt-Universität zu Berlin, Mathematics
Keywords:	Iterative operator-splitting method, waveform relaxation method, convergence analysis, Pade approximations, fixed-point schemes
<p>Note: The following files were submitted by the author for peer review, but cannot be converted to PDF. You must view these files (e.g. movies) online.</p> <p>geiser_ijcm_2011.tar.gz</p>	

SCHOLARONE™  
Manuscripts

# An Iterative Splitting Method via Waveform Relaxation

Jürgen Geiser

geiser@mathematik.hu-berlin.de

**Abstract.** This paper explores a new numerical strategy for a closed formulation of iterative splitting methods and their embedding in classical Waveform-Relaxation methods. Since iterative splitting has been developed in several papers, an abstract framework that relates these methods to other classical splitting methods would be useful and is needed. Here, we present an embedding of the iterative splitting method in Waveform-Relaxation and exponential splitting methods. While we can use the theoretical background of the classical schemes, a simpler iterative splitting analysis is obtained. This is achieved by basing the analysis on semigroup and fix-point schemes.

Our approach is illustrated with numerical results obtained on differential equations with constant and time-dependent coefficients.

**Keyword** Iterative splitting method, Waveform-relaxation method, Pade approximations.

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

## 1 Introduction

Historically, iterative splitting methods are related to iterative solver methods. The underlying ideas are based on waveform relaxation methods, which are fix-point iterations or successive approximation schemes, see [24]. Their convergence analysis and application to second-order differential equations are studied in [14–18]. They are developed to simplify the solver process, while saving time when computing, e.g. simple diagonal matrices, see [26]. On the other hand, they can be used to accelerate the iterative process of solving partial differential equations, see [21]. In the next step, the generalization of iterative splitting schemes to unbounded operators allows them to be applied to partial differential equations, see [8], [9]. In this paper, we deal with a general scheme, a so called multi-stage scheme, which gives a significant improvement in terms of accuracy, numerical stability and reduction of local and global errors.

We concentrate on an approximation to the solution of the linear evolution equation

$$\frac{\partial u}{\partial t} = Lu = (A + B)u, \quad u(0) = u_0, \quad (1)$$

2

where  $L, A$  and  $B$  are linear operators.

1.) For the numerical method, we employ an 1-stage iterative splitting scheme, see [26], also called the *Waveform-Relaxation method*:

$$u_i(t) = \exp(At)u_0 + \int_0^t \exp(A(t-s))Bu_{i-1}(s) ds, \quad (2)$$

where  $i = 1, 2, 3, \dots$  and  $u_0(t) = u(0)$  is the initial condition.

2.) As a second numerical method, we employ a 2-stage *iterative splitting scheme*, see [3], [5]:

$$u_i(t) = \exp(At)u_0 + \int_0^t \exp(A(t-s))Bu_{i-1}(s) ds, \quad (3)$$

$$u_{i+1}(t) = \exp(Bt)u_0 + \int_0^t \exp(B(t-s))Au_i(s) ds, \quad (4)$$

where  $i = 1, 3, 5, \dots$  and  $u_0(t) = u(0)$  is the initial condition.

3.) As a third numerical method, the 2-stage scheme is improved by combining it with equations (3) and (4) in a *multiple-stage scheme* and can be written generally as an embedded iterative scheme with inner and outer layers.

$$u_{i_k}(t) = \exp(At)u_0 + \int_0^t \exp(A(t-s))Bu_{i_k+J_{k-1}-1}(s) ds, \quad (5)$$

$$u_{j_k+I_k}(t) = \exp(Bt)u_0 + \int_0^t \exp(B(t-s))Au_{j_k+I_k-1}(s) ds, \quad (6)$$

where  $i_k = 1, 2, 3, \dots, I_k$ ,  $j_k = 1, 2, 3, \dots, J_k$ ,  $k = 1, \dots, K$ ,  $I_1, \dots, I_K$  are the number of iterations done with the  $A$ -operator, where  $J_1, \dots, J_K$  are the number of iterations done with the  $B$ -operator. The initialization is given as  $u_0(t) = u(0)$  and  $J_0 = 0$ . Here, we can control the iterative steps on each operator  $A$  and  $B$ .

*Remark 1.* The motivation for expanding the 1-stage scheme to a 2-stage scheme comes from the reduction of local and global errors, as illustrated in the following example:

*Example 1.* Let  $A, B$  be constant matrices, not dependent of  $t$ , not commuting ( $[A, B] \neq \mathbb{O}$ ) and no zero matrices ( $A, B \neq \mathbb{O}$ ). We obtain the following schemes: For an 1-stage scheme (2), for  $i = 1$ :

$$u_{1,one}(t) = \exp(At)u_0 + \int_0^t \exp(A(t-s))Bu_0 ds, \quad (7)$$

further for an 1-stage scheme (2), for  $i = 2$ :

$$u_{2,one}(t) = \exp(At)u_0 + \int_0^t \exp(A(t-s))Bu_1(s) ds, \quad (8)$$

and for a 2-stage scheme for  $i = 2$ :

$$u_{2,two}(t) = \exp(Bt)u_0 + \int_0^t \exp(B(t-s))Au_{1,one}(s) ds. \quad (9)$$

So we find that equation (7) is locally a second order approximation of  $\exp((A+B)t)$  for  $t > 0$ , meaning that:

$$\|\exp((A+B)t)u_0 - u_{1,one}(t)\| \leq Ct^2\|u_0\| + O(t^3), \quad (10)$$

where  $C = \frac{\|BA\|}{2} + \frac{\|B^2\|}{2}$  and  $\|\cdot\|$  is the  $L_1$ -norm.

Further equation (8) is locally a third order approximation of  $\exp((A+B)t)$  for  $t > 0$ , meaning that:

$$\|\exp((A+B)t)u_0 - u_{2,one}(t)\| \leq \hat{C}t^3\|u_0\| + O(t^4), \quad (11)$$

where  $\hat{C} = \frac{\|BAB\|}{6} + \frac{\|B^3\|}{6}$  and  $\|\cdot\|$  is the  $L_1$ -norm.

and equation (9) is locally a third order approximation of  $\exp((A+B)t)$  for  $t > 0$ , meaning that:

$$\|\exp((A+B)t)u_0 - u_{2,two}(t)\| \leq \tilde{C}t^3\|u_0\| + O(t^4), \quad (12)$$

where  $\tilde{C} = \frac{\|ABA\|}{6} + \frac{\|AB^2\|}{6}$  and  $\|\cdot\|$  is the  $L_1$ -norm.

*Remark 2.* The motivation for expanding the 2-stage scheme to a multi-stage scheme comes from the improvement in numerical stability of the scheme. In numerical examples, the operators are not equal and often one operator can be bounded by the other operator. So, in a multi-stage scheme, we can control the iterative stage over each operator.

*Example 2.* Let  $A, B$  be bounded operators, but  $\|A\| \gg \|B\|$ , while  $\|\cdot\|$  is a matrix norm. We assume  $0 < \|B\| \leq 1$  otherwise we have trivially to solve only with operator  $A$ .

We apply a 2-stage method until  $i = 2$  and obtain the error bound of equation (9) for bounded operators given as:

$$\tilde{C} = \frac{\|ABA\|}{6} + \frac{\|AB^2\|}{6} \leq \frac{\|A^2\|}{6}. \quad (13)$$

We can reduce the error bound and therefore the stability of the scheme, while applying an 1-stage method to operator  $A$ , see equation (2), and obtain the error bound of equation (8) for bounded operators as:

$$\hat{C} = \frac{\|BAB\|}{6} + \frac{\|B^3\|}{6} \leq \frac{\|A\|}{6} \quad (14)$$

where we have three cases:

4

1. If  $\|A\| \gg 1$ , then  $\hat{C} \ll \tilde{C}$ .
2. If  $\|A\| \ll 1$ , then  $\hat{C} \gg \tilde{C}$ .
3. If  $\|A\| \approx 1$ , then  $\hat{C} < \tilde{C}$ .

Here we have some benefits in applying different stages with respect to the underlying operators. Such benefits are discussed later in the error analysis.

The outline of the paper is as follows. The iterative splitting methods and their error analysis are presented in Section 2. In Section 3, we discuss an efficient computation of the iterative splitting method with a closed formulation. In Section 4, we discuss the numerical experiments and the benefits of the iterative splitting schemes. Finally, we discuss future works in the area of iterative methods.

## 2 Splitting Method and Error Analysis

In the following, the iterative splitting scheme is presented as a multi-stage scheme.

### 2.1 General Iterative Splitting Method (multiple stage scheme)

A general setting of an iterative splitting method is given with inner and outer iterative schemes, so that each part of a 2-stage scheme can be chosen independently to control the stages over each operator. The method is given in the following:

$$\frac{\partial u_{i_k+J_{k-1}}(t)}{\partial t} = Au_{i_k+J_{k-1}}(t) + Bu_{i_k+J_{k-1}-1}(t), \quad (15)$$

$$\text{with } u_{i_k+J_{k-1}}(t^n) = u^n,$$

$$\frac{\partial u_{j_k+I_k}(t)}{\partial t} = Au_{j_k+I_k-1}(t) + Bu_{j_k+I_k}(t), \quad (16)$$

$$\text{with } u_{j_k+I_k}(t^n) = u^n,$$

where  $i_k = J_{k-1} + 1, \dots, I_k$ ,  $j_k = I_k + 1, \dots, J_k$ ,  $k = 1, \dots, K$ ,  $I_k - J_{k-1}$  are the number of iterations done with the  $A$ -operator, where  $J_k - I_k$  is the number of iterations done with the  $B$ -operator. The initialization is given as  $u_0(t) = u(0)$  and  $I_0 = J_0 = 0$ .

### 2.2 Error Analysis for the General Scheme

In this section, we analyze the convergence of the general scheme (15) and (16) in which Waveform Relaxation and the iterative splitting method are embedded.

We consider the error of the iterative splitting method on a Banach space  $\mathbf{X}$  with norm and induced operator norm denoted by  $\|\cdot\|$ .

Further we have the following assumptions:

**Assumption 21** The linear operators  $A + B, A, B$  generate  $C_0$  semigroups on  $X$ , and the operators  $A, B$  satisfy in addition the bounds:

$$\|\exp(A\tau)\| \leq \exp(\omega|t|) \text{ and } \|\exp(B\tau)\| \leq \exp(\omega|t|) \quad (17)$$

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

In the next we present the convergence of the iterative splitting scheme.

**Theorem 1.** For the numerical solution of (3), consider an iterative operator splitting scheme on operator  $A$  with  $i$ -th iterative steps.

If the assumption 21 is valid, then

$$\|S_i^n - \exp((A + B)n\tau)\| \leq C\|B^i\|\tau^{i-1}, \quad n\tau \leq T, \quad (18)$$

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

*Proof.* By applying the telescopic identity we obtain

$$\begin{aligned} & (S_i^n - \exp((A + B)n\tau))u_0 \\ &= \sum_{\nu=0}^{n-1} S_i^{n-\nu-1} (S - \exp((A + B)\tau)) \exp(\nu\tau(A + B))u_0, \end{aligned} \quad (19)$$

if we assume the stability bound:

$$\|S_i\| \leq \exp(c\omega\tau), \quad (20)$$

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

Furthermore, if we assume the consistency bound:

$$\begin{aligned} & \|S_i^n - \exp((A + B)n\tau)\| \\ & \leq \exp(c\omega T) \sum_{\nu=0}^{n-1} \|(S - \exp(\tau(A + B))) \exp(\nu\tau(A + B))\| \end{aligned} \quad (21)$$

$$\leq C\tau^i, \quad n\tau \leq T, \quad (22)$$

where  $C$  is a constant and independent of  $\tau$ .

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

#### Consistency and Stability:

*Proof.* Let us consider the iteration (15)–(16) on the sub-interval  $[t^n, t^{n+1}]$ .

For the first iteration of (15), we have:

$$\frac{\partial u_1(t)}{\partial t} = Au_1(t), \quad t \in (t^n, t^{n+1}], \quad (23)$$

and for the second iteration, we have:

$$\frac{\partial u_2(t)}{\partial t} = Au_2(t) + Bu_1(t), \quad t \in (t^n, t^{n+1}], \quad (24)$$

6

In general we have:

for  $m = 1, 2, \dots$ ,

$$\frac{\partial u_i(t)}{\partial t} = Au_i(t) + Bu_{i-1}(t), \quad t \in (t^n, t^{n+1}], \quad (25)$$

where  $u_0(t) \equiv 0$ .

We have the following solutions for the iterative scheme:

The solutions for the first two equations are given by the variation of constants:

$$u_1(t) = \exp(A(t^{n+1} - t))u(t^n), \quad t \in (t^n, t^{n+1}], \quad (26)$$

$$u_2(t) = \exp(At)u(t^n) + \int_{t^n}^{t^{n+1}} \exp(A(t^{n+1} - s))Bu_1(s)ds, \quad t \in (t^n, t^{n+1}], \quad (27)$$

For  $m = 0, 1, 2, \dots$

$$u_i(t) = \exp(A(t - t^n))u(t^n) + \int_{t^n}^t \exp(sA)Bu_{i-1}(t^{n+1} - s) ds, \quad t \in (t^n, t^{n+1}]. \quad (28)$$

The stability is given as:

$$\|S_i\| \leq \exp(c\omega\tau), \quad (29)$$

where  $c$  only depends on the coefficients of the method,  $\tau = t - t^n$  and  $\omega \geq 0$ . and we assume the bound of the operators as:

$$\|\exp(A\tau)\| \leq \exp(\omega_1\tau), \quad (30)$$

$$\|\exp(B\tau)\| \leq \exp(\omega_2\tau), \quad (31)$$

where  $\omega_1, \omega_2 \geq 0$ .

The consistency is given as:

For  $e_1$  we have:

$$u_1(\tau) = \exp(A\tau)u(t^n), \quad (32)$$

$$u(\tau) = \exp((A + B)\tau)u(t^n) = \exp(A\tau)u(t^n) + \int_{t^n}^{t^{n+1}} \exp(As)B \exp((t^{n+1} - s)(A + B))u(t^n) ds. \quad (33)$$

We obtain:

$$\|e_1\| = \|c - u_1\| \leq \|\exp((A + B)\tau)u(t^n) - \exp(A\tau)u(t^n)\| \leq C_1\|B\|\tau\|u(t^n)\|. \quad (34)$$

where  $C_1$  is a constant and independent of  $\tau$ .

For  $e_2$  we have:

$$u_2(\tau) = \exp(A\tau)u(t^n) + \int_{t^n}^{t^{n+1}} \exp(As)B \exp((t^{n+1} - s)A)u(t^n) ds, \quad (35)$$



$$\begin{aligned}
u(\tau) &= \exp(A\tau)u(t^n) + \int_{t^n}^{t^{n+1}} \exp(As)B \exp((t^{n+1} - s)A)u(t^n) ds \\
&+ \int_{t^n}^{t^{n+1}} \exp(As)B \\
&\int_{t^n}^{t^{n+1}-s} \exp(A\rho)B \exp((t^{n+1} - s - \rho)(A + B))u(t^n) d\rho ds.
\end{aligned} \tag{36}$$

We obtain:

$$\begin{aligned}
\|e_2\| &\leq \|\exp((A + B)\tau)u(t^n) - u_2\| \\
&\leq C_2\|B^2\|\tau^2\|u(t^n)\|.
\end{aligned} \tag{37}$$

For the iterations, a recursive proof is given as follows:

For  $m = 0, 1, 2, \dots$ , for  $e_i$ , we have :

$$\begin{aligned}
u_i(\tau) &= \exp(A\tau)u(t^n) \\
&+ \int_{t^n}^{t^{n+1}} \exp(As)B \exp((t^{n+1} - s)A)u(t^n) ds \\
&+ \int_{t^n}^{t^{n+1}} \exp(As_1)B \int_{t^n}^{t^{n+1}-s_1} \exp(s_2A)B \exp((\tau - s_1 - s_2)A)u(t^n) ds_2 ds_1 \\
&+ \dots + \\
&+ \int_{t^n}^{t^{n+1}} \exp(As_1)B \int_{t^n}^{t^{n+1}-s_1} \exp(s_2A)B \exp((\tau - s_1 - s_2)A)u(t^n) ds_2 ds_1 + \dots + \\
&+ \int_{t^n}^{t^{n+1}} \exp(As_1)B \int_{t^n}^{t^{n+1}-\sum_{j=1}^{i-1} s_1} \exp(s_2A)A \exp((\tau - s_1 - s_2)A)u(t^n) ds_2 ds_1 \dots ds_i,
\end{aligned} \tag{38}$$

$$\begin{aligned}
u(\tau) &= \exp(A\tau) + \int_{t^n}^{t^{n+1}} \exp(As)B \exp((t^{n+1} - s)A)u(t^n) ds \\
&+ \dots + \\
&+ \int_{t^n}^{t^{n+1}} \exp(As_1)B \int_{t^n}^{t^{n+1}-s_1} \exp(s_2A)B \exp((\tau - s_1 - s_2)A)u(t^n) ds_2 ds_1 + \dots + \\
&+ \int_{t^n}^{t^{n+1}} \exp(As_1)B \int_{t^n}^{t^{n+1}-\sum_{j=1}^{i-1} s_1} \exp(s_2A)B \exp((\tau - s_1 - s_2)A)u(t^n) ds_2 ds_1 \dots \\
&\int_{t^n}^{t^{n+1}-\sum_{j=1}^i s_2} \exp(s_2A)B \exp((\tau - s_1 - s_2)(A + B))u(t^n) ds_i,
\end{aligned} \tag{39}$$

We obtain:

$$\begin{aligned}
\|e_i\| &\leq \|\exp((A + B)\tau)u(t^n) - u_i\| \\
&\leq C\|B^i\|\tau^i\|u(t^n)\|,
\end{aligned} \tag{40}$$

where  $i$  is the iterative step,  $C$  is a constant independent of  $t$ .

The same idea can be applied to the operator  $B$ .

8

*Remark 3.* At least we obtain for the general iterative splitting method with  $I$  iterative steps to operator  $A$  and  $J$  iterative steps to operator  $B$  the following consistency error:

$$\begin{aligned} \|e_{I+J}\| &\leq \|\exp((A+B)\tau)u(t^n) - u_{I+J}\| \\ &\leq C\|A^I\| \|B^J\| \tau^{I+J} \|u(t^n)\|, \end{aligned} \quad (41)$$

where  $C$  is a constant and independent of  $\tau$ .

Embedding the result to the convergence results we obtain at least an error of  $\mathcal{O}(\tau^{I+J-1})$ .

In the next section, we describe the computation of the integral formulation with exponential functions.

### 3 Computation of the Iterative Splitting Method: Closed Formulation

In the last few years, the computational effort to compute integrals with exp-functions has increased. We present a closed form and re-substitute the integral with closed functions. Such benefits accelerate the computation and allow the ideas to be parallelized.

Here, we present a closed form for the iterative splitting method for the first 4 splitting iterations.

For  $i = 1$ , we have:

$$u_1(t) = \exp(At) \exp(Bt) u(t^n). \quad (42)$$

where we have a first order method, also known as an  $AB$  splitting method, see [3].

For  $i = 2$ , we have:

$$u_2(t) = \frac{1}{2}(\exp(At) \exp(Bt) + \exp(Bt) \exp(At))u(t^n), \quad (43)$$

where we have a second order method, also known as a parallel  $AB$  splitting method, see [3].

For  $i = 3$ , we have:

$$\begin{aligned} u_3(t) &= \frac{1}{6}(\exp(At) \exp(Bt) \exp(At) + \exp(Bt) \exp(At) \exp(At) \\ &\quad + \exp(Bt) \exp(Bt) \exp(At) + \exp(At) \exp(At) \exp(Bt) \\ &\quad + \exp(At) \exp(Bt) \exp(Bt) + \exp(Bt) \exp(At) \exp(Bt))u(t^n), \end{aligned} \quad (44)$$

where we can reduce the operators by making assumptions about the commutators, e.g.  $[A, [A, B]] = [B, [A, A]]$ .

*Remark 4.* Higher order iterative splitting methods involve at the very least the derivation of the remaining forms for all commutations between operators  $A$  and  $B$ . Here an optimization is possible by assuming that commutators are equal or at least zero, see [2] and [8].

#### **Exp-Approximations with Pade Approximations**

In applications, we have to extend differential equations to systems of differential equations. Therefore, we have to apply matrix functions to our analytical tools.

To approximate matrix functions in the following section, we apply Pade approximations.

For the matrix exponential we apply:

$$\frac{I + \frac{1}{2}At}{1 - \frac{1}{2}At} = \exp(At) + O((At)^3), \quad (45)$$

$$\frac{I + \frac{2}{3}(At) + \frac{1}{6}(At)^2}{I - \frac{1}{3}At} = \exp(At) + O((At)^4), \quad (46)$$

where  $A \in \mathbb{R}^{n \times n}$  is the matrix.

*Remark 5.* The general formulation for different Pade approximations applied to exponential functions  $\exp(At)$  is given in [1].

In the next experiments, we apply the Pade approximations for  $m = n = 1$ ,  $m = n = 2$  and  $m = n = 3$ .

## **4 Numerical Experiments**

In the following experiment, we start by presenting the enhancement of the multi-stage procedure to a standard Waveform relaxation method. Further, we present the improvement of the iterative splitting scheme to a classical splitting method.

### **4.1 First Example: Benchmark with $2 \times 2$ Matrix**

In the first experiment, we deal with an ODE and separate the complex operator into two simpler operators.

We deal with the following equation :

$$\frac{\partial u_1}{\partial t} = -\lambda_1 u_1 + \lambda_2 u_2, \quad (47)$$

$$\frac{\partial u_2}{\partial t} = \lambda_1 u_1 - \lambda_2 u_2, \quad (48)$$

$$u_1(0) = u_{10}, u_2(0) = u_{20} \text{ (initial conditions)}, \quad (49)$$

where  $\lambda_1, \lambda_2 \in \mathbb{R}^+$  are the decay factors and  $u_{10}, u_{20} \in \mathbb{R}^+$ . We have the time interval  $t \in [0, T]$ .

10

We rewrite equation (47) in operator notation, we concentrate on the following equations :

$$\frac{\partial u}{\partial t} = A(t)u + B(t)u, \quad (50)$$

$$(51)$$

where  $u_1(0) = u_{10} = 1.0$ ,  $u_2(0) = u_{20} = 1.0$  are the initial conditions, where we have  $\lambda_1(t) = t$  and  $\lambda_2(t) = t^2$ .

Our split operators are:

$$A = \begin{pmatrix} -\lambda_1 & \lambda_2 \\ 0 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 0 & 0 \\ \lambda_1 & -\lambda_2 \end{pmatrix}. \quad (52)$$

The actual parameters for the experiments are given as:

$$\lambda_1 = 0.05 \quad \lambda_2 = 0.01 \quad T = 1.0 \quad u_0 = (1, 1)^t$$

We apply  $AB$ , Strang and 3rd order splitting and compare with the unsplit solutions:

1.) Unsplit :

$$u_{exact}(t) = \exp((A + B)t)u(t^n). \quad (53)$$

2.) A-B splitting

$$u_1(t) = \exp(A\tau) \exp(B\tau)u(t^n), \quad (54)$$

where we have a first order method, also known as an  $AB$  splitting method, see [3].

3.) Strang splitting

$$u_2(t) = \frac{1}{2}(\exp(At) \exp(Bt) + \exp(Bt) \exp(At))u(t^n), \quad (55)$$

where we have a second order method, also known as a parallel  $AB$  splitting method, see [3].

4.) 3rd order splitting

$$u_3(t) = \frac{1}{6}(\exp(At) \exp(Bt) \exp(At) + \exp(Bt) \exp(At) \exp(At) + \exp(Bt) \exp(Bt) \exp(At) + \exp(At) \exp(At) \exp(Bt) + \exp(At) \exp(Bt) \exp(Bt) + \exp(Bt) \exp(At) \exp(Bt))u(t^n), \quad (56)$$

where the solution is derived from the iterative splitting methods.

The  $L_1$ -error is computed as:

$$err_{num} = \sum_{k=1}^N |u_{exact}(t_k) - u_{num}(t_k)|, \quad (57)$$

where  $t_k = k\Delta t$ , where  $t_0, t_1, \dots$  and  $\Delta t = 0.1$ .

number of time partitions	$err_1$ (2nd order)	$err_2$ (2nd order)	$err_1$ (3rd order)	$err_2$ (3rd order)
2	4.5321e-002	3.6077e-003	4.5321e-002	3.6077e-003
3	4.6126e-004	3.6077e-003	4.6126e-004	3.6077e-003
4	4.6126e-004	2.2459e-005	4.6126e-004	2.2464e-005
5	1.9096e-006	2.2459e-005	1.9040e-006	2.2464e-005
6	1.9096e-006	6.1224e-008	1.9040e-006	6.6759e-008

**Table 1.** Numerical results for the first example with the iterative splitting method and 2nd- and 3rd-order method.

*Remark 6.* Our numerical results are based on higher order iterative schemes in closed formulations. Table 1 presents the results in which the 3rd order methods can achieve more accurate results. The numerical results show that the splitting error decreases as long as the Pade approximations employed allow it. Therefore, we can say that more iterations are only sufficient when a higher order method is used. One can also see that the iterative operator-splitting method is of order  $i$  as long as the Pade approximation is also of order  $i$ .

#### 4.2 Second Experiment

In the second experiment, we deal with a time-dependent ODE and separate the complex operator into two simpler operators.

We deal with the following equation :

$$\frac{\partial u_1}{\partial t} = -\lambda_1(t)u_1 + \lambda_2(t)u_2, \quad (58)$$

$$\frac{\partial u_2}{\partial t} = \lambda_1(t)u_1 - \lambda_2(t)u_2, \quad (59)$$

$$u_1(0) = u_{10}, \quad u_2(0) = u_{20} \text{ (initial conditions)}, \quad (60)$$

where  $\lambda_1(t) \in \mathbb{R}^+$  and  $\lambda_2(t) \in \mathbb{R}^+$  are the decay factors and  $u_{10}, u_{20} \in \mathbb{R}^+$ . We have the time-interval  $t \in [0, T]$ .

We rewrite the equation (58) in operator notation and concentrate on the following equations :

$$\frac{\partial u}{\partial t} = A(t)u + B(t)u, \quad (61)$$

$$(62)$$

where  $u_1(0) = u_{10} = 1.0$ ,  $u_2(0) = u_{20} = 1.0$  are the initial conditions, where we have  $\lambda_1(t) = t$  and  $\lambda_2(t) = t^2$ .

and our split operators are

$$A(t) = \begin{pmatrix} -\lambda_1(t) & \lambda_2(t) \\ 0 & 0 \end{pmatrix}, \quad B(t) = \begin{pmatrix} 0 & 0 \\ \lambda_1(t) & -\lambda_2(t) \end{pmatrix}. \quad (63)$$

12

For equation (58), we could apply a higher order Pade approximation, e.g. 3rd order.

To start, we apply sequential splitting and the iterative operator-splitting, further we combine them by using pre-step methods to see the improvement in results.

For time-steps  $\Delta t$ , we have  $\Delta t = 1$  for 1 time-partition and  $\Delta t = 0.1$  for 10 time-partitions.

number of time partitions	$err_1$ (2nd order)	$err_2$ (2nd order)	$err_1$ (3rd order)	$err_2$ (3rd order)
1	4.5321e-002	3.6077e-003	4.5321e-002	3.6077e-003
10	4.6126e-004	3.6077e-003	4.6126e-004	3.6077e-003

**Table 2.** Numerical results for the second example with the iterative splitting method and 2nd and 3rd order method.

### 4.3 Third Experiment

We tackle the 2-dimensional advection-diffusion equation with periodic boundary conditions

$$\partial_t u = -\mathbf{v}\nabla u + D\Delta u, \quad (64)$$

$$= -v_x \frac{\partial u}{\partial x} - v_y \frac{\partial u}{\partial y} + D \frac{\partial^2 u}{\partial x^2} + D \frac{\partial^2 u}{\partial y^2}, \quad (65)$$

$$u(\mathbf{x}, t_0) = u_0(\mathbf{x}), \quad (66)$$

with parameters

$$v_x = v_y = 1, \quad (67)$$

$$D = 0.01, \quad (68)$$

$$t_0 = 0.25. \quad (69)$$

The advection-diffusion problem has an analytical solution

$$u_a(\mathbf{x}, t) = \frac{1}{t} \exp\left(\frac{-(\mathbf{x} - \mathbf{v}t)^2}{4Dt}\right), \quad (70)$$

which we will use as a convenient initial function:

$$u(\mathbf{x}, t_0) = u_a(\mathbf{x}, t_0). \quad (71)$$

We apply dimensional splitting to our problem:

$$\frac{\partial u}{\partial t} = A_x u + A_y u, \quad (72)$$

where:

$$A_x = -v_x \frac{\partial u}{\partial x} + D \frac{\partial^2 u}{\partial x^2} \quad (73)$$

$$A_y = -v_y \frac{\partial u}{\partial y} + D \frac{\partial^2 u}{\partial y^2}. \quad (74)$$

We use a 1st order upwind scheme for  $\frac{\partial}{\partial x}$  and a 2nd order central difference scheme for  $\frac{\partial^2}{\partial x^2}$ . By introducing the artificial diffusion constant  $D_x = D - \frac{v_x \Delta x}{2}$  we achieve a 2nd order finite difference scheme

$$L_x u(x) = -v_x \frac{u(x) - u(x - \Delta x)}{\Delta x} + D_x \frac{u(x + \Delta x) + u(x) + u(x - \Delta x)}{\Delta x^2}, \quad (75)$$

because the new diffusion constant eliminates the first order error (i.e. numerical viscosity) of the Taylor expansion of the upwind scheme.  $L_y u$  is derived in the same way.

We apply a BDF5 method to gain 5th order accuracy in time:<sup>1</sup>

$$L_t u(t) = \frac{1}{\Delta t} \left( \frac{137}{60} u(t + \Delta t) - 5u(t) + 5u(t - \Delta t) - \frac{10}{3} u(t - 2\Delta t) + \frac{5}{4} u(t - 3\Delta t) - \frac{1}{5} u(t - 4\Delta t) \right). \quad (76)$$

Our aim is to compare the iterative splitting method with AB splitting (Lie-Trotter splitting, see [3]). Since  $[A_x, A_y] = 0$  there is no splitting error for AB splitting and therefore we cannot expect to achieve better results with the iterative splitting in terms of general numerical accuracy. Instead, we will show that iterative splitting out-competes AB splitting in terms of computational effort and round-off errors. But first some remarks need to be made about the special behaviour of both methods when combined with high-order Runge-Kutta and BDF methods.

### Splitting and Schemes of High Order in Time, Concerning AB-Splitting:

The principle of AB-splitting is well known and simple. The equation  $\frac{du}{dt} = Au + Bu$  is broken up into:

$$\begin{aligned} \frac{du^{n+1/2}}{dt} &= Au^{n+1/2}, \\ \frac{du^{n+1}}{dt} &= Bu^{n+1}, \end{aligned}$$

which are connected via  $u^{n+1}(t) = u^{n+1/2}(t + \Delta t)$ . This is pointed out in figure (1). AB splitting works very well for any given one-step method like the Crank-

<sup>1</sup> Please note that the dependencies of  $u(\mathbf{x}, t)$  are suppressed for the sake of simplicity.

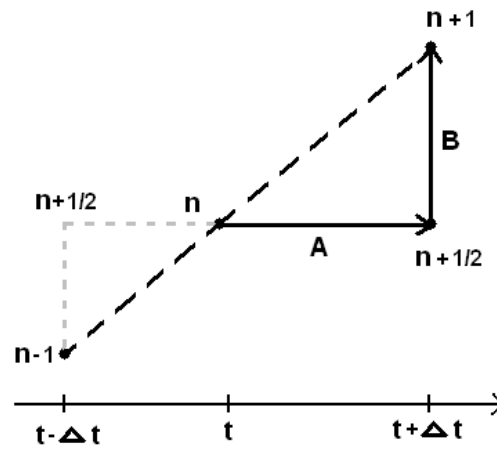


Fig. 1. Principle of the AB-Splitting.

Nicholson-Scheme. Not taking into account the splitting error (which is an error in time) it is also compatible with high order schemes such as explicit/implicit Runge-Kutta-schemes.

Things look different if one tries to use a multi-step method like the implicit BDF or the explicit Adams method with AB splitting, as these cannot be properly applied as shown by the following example:

Choose for instance a BDF2 method which, in the case of  $du/dt = f(u)$ , has the scheme

$$\frac{3}{2}u(t + \Delta t) - 2u(t) + \frac{1}{2}u(t - \Delta t) = \Delta t f(u(t + \Delta t)).$$

So, the first step of AB splitting looks like:

$$\frac{3}{2}u^{n+1/2}(t + \Delta t) - 2u^{n+1/2}(t) + \frac{1}{2}u^{n+1/2}(t - \Delta t) = \Delta t Au(t + \Delta t)$$

Clearly,  $u^{n+1/2}(t) = u^n(t)$  but what is  $u^{n+1/2}(t - \Delta t)$ ? This is also shown in figure (1) and it is obvious that we won't have knowledge about  $u^{n+1/2}(t - \Delta t)$  unless we compute it separately, which means additional computational effort. This overhead increases dramatically when we move to a multi-step method of higher order.

The mentioned problems with AB splitting will not occur with a higher order Runge-Kutta method since only knowledge of  $u^n(t)$  is needed.



**Remarks about iterative splitting:** The BDF methods apply very well to iterative splitting. Let us recall at this point that this method, although being a real splitting scheme, always remains a combination of the operators  $A$  and  $B$ , so no steps have to be done in one direction only <sup>2</sup>.

In particular, we make a subdivision of our existing time-discretization  $t_j = t_0 + j\Delta t$  into  $I$  parts. So we have sub-intervals  $t_{j,i} = t_j + i\Delta t/I$ ,  $0 \leq i \leq I$  on which we solve the following equations iteratively:

$$\frac{du^{i/I}}{dt} = Au^{i/I} + Bu^{(i-1)/I} \quad (77)$$

$$\frac{du^{(i+1)/I}}{dt} = Au^{i/I} + Bu^{(i+1)/I} \quad (78)$$

$$(79)$$

$u^{-1/I}$  is either 0 or a reasonable approximation<sup>3</sup> while  $u^0 = u(t_j)$  and  $u^1 = u(t_j + \Delta t)$ . The crucial point here is that we only know our approximations at given times which happen not to be the times at which a Runge-Kutta (RK) method needs to know them. Therefore, in the case of a RK method, the values of the approximations have to be interpolated with at least the accuracy one wishes to attain with the splitting and this means a lot of additional computational effort. We can now summarize our results in table 4.3 which shows which methods are practicable for each kind of splitting scheme.<sup>4</sup>

	low order s.s.m.	high order s.s.m.	m.s.m.
AB-splitting	X	X	-
Iterative splitting	X	-	X

**Table 3.** Practicability of single- and multi-step methods (s.s.m: single-step methods, m.s.m. multi-step methods).

**Numerical results** After resolving the technical aspects of this issue, we can now proceed to the actual computations. A question which arises is which of the

<sup>2</sup> As we will see there is an exception to this.

<sup>3</sup> In fact the order of the approximation is not of much importance if we fulfill a sufficient number of iterations. In the case of  $u^{-1/I} = 0$ , we have the exception that a step in the A-direction is done while B is left out. The error of this step certainly vanishes after a few iterations, but mostly after only one iteration

<sup>4</sup> Something in favour of the iterative splitting scheme is that it also takes into the account the fact that AB splitting may be used alongside the high order methods alluded to but cannot maintain the order if  $[A, B] \neq 0$ , while the iterative splitting scheme re-establishes the maximum order of the scheme after a sufficient number of iterations have been done.

splitting methods requires the least computational effort since we can expect them to solve the problem with more or less the same accuracy if we use practicable methods with equal order, as  $[A_x, B_x] = 0$ . We tested the dimensional splitting of the 2-dimensional advection-diffusion equation with AB splitting combined with a 5th order RK method after Dormand and Prince, and with iterative splitting in conjunction with a BDF5 scheme. We used  $40 \times 40$  and  $80 \times 80$  grids and completed  $n_t$  time steps each subdivided into 10 smaller steps until we reached time  $t_{end} = 0.6$  which is sufficient to see the main effects. Iterative splitting was performed with 2 iterations which was already enough to attain the desired order. In tables 4 and 5, the errors at time  $t_{end}$  and the computation times are shown.

Number of steps	Error AB	Error It.spl.	AB computation time	It. spl. computation time
5	0.1133	0.1154	0.203 s	0.141 s
10	0.1114	0.1081	0.500 s	0.312 s
30	0.1074	0.1072	1.391 s	0.907 s
50	0.1075	0.1074	2.719 s	1.594 s

**Table 4.** Errors and computation times of AB splitting and iterative splitting for a  $40 \times 40$  grid.

Number of steps	Error AB	Error It.spl.	AB computation time	It. spl. computation time
5	0.0288	0.0621	0.812 s	0.500 s
10	0.0276	0.0285	2.031 s	1.266 s
30	0.0268	0.0267	6.109 s	4.000 s
50	0.0265	0.0265	12.703 s	7.688 s

**Table 5.** Errors and computation times of AB splitting and iterative splitting for a  $80 \times 80$  grid.

*Remark 7.* As we can see, the error in the iterative splitting scheme reaches the same value as the AB splitting error after a certain number of time steps and stays below it for all additional steps we accomplish. Of course, the error cannot drop below a certain value which is governed by the spatial discretization increments. It is to be noticed that, while the computation time used for iterative splitting is always about 20%-40% less than that of AB splitting<sup>5</sup> the accuracy is, with a sufficient number of time steps, slightly better than that of AB splitting. This is due to the roundoff error which is higher for the Runge-Kutta method

<sup>5</sup> The code for both methods is kept in the simplest possible form.

because of the greater amount of basic operations needed to compute the RK steps.

A future task will be to introduce non-commuting operators in order to show the superiority of iterative splitting over AB splitting when the order in time is reduced due to the splitting error.

## 5 Conclusions and Discussions

We have presented an iterative operator-splitting method and analyzed the error bounds for linear operators. Under weak assumptions, we could prove the higher order error bounds. The benefit of higher accuracy and more computational efficiency with respect to the multi-stage scheme shows the importance of iterative splitting schemes. Numerical examples confirm the potential of applying our new scheme to differential equations. In the future, we will focus on the development of improved operator-splitting methods which are better adapted to applications involving nonlinear differential equations.

## References

1. G.A. Baker Jr., P.R. Graves-Morris. *Pade approximants*. In G.-C. Rota (Ed.), Encyclopedia of Mathematics and its Applications, Vol. 59, 2nd Edition, Cambridge University Press, New York, 1996.
2. S. Blanes and F. Casas. *Splitting Methods for Non-autonomous separable dynamical systems*, Journal of Physics A: Math. Gen., 39, 5405-5423, 2006.
3. I. Farago and J. Geiser. *Iterative Operator-Splitting Methods for Linear Problems*. International Journal of Computational Science and Engineering, 3(4): 255-263, 2007.
4. J. Geiser. *Higher order splitting methods for differential equations: Theory and applications of a fourth order method*. Numer.Math.Theor.Meth.Appl., 1(3): 321-339, 2008.
5. 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.
6. J. Geiser and C. Kravvaritis. *A Domain Decomposition method based on iterative Operator Splitting method*. Applied Numerical Mathematics, 59, 608-623, 2009.
7. J. Geiser and C. Kravvaritis. *Overlapping operator splitting methods and applications in stiff differential equations*. In: Qin Sheng and Johnny Henderson (Eds.). Special issue: Novel Difference and Hyprod Methods for Differential and Integro-Differential Equations and Applications. Neural, Parallel, and Scientific Computations (NPSC), Vol. 16, 189-200, 2008.
8. J. Geiser. *Consistency of Iterative Operator-Splitting Method: Theory and Applications*. NMPDE, published online, 22 Jan. 2009, <http://www3.interscience.wiley.com/journal/107630294/issue>. 2008.
9. J. Geiser. *Stability of Iterative Operator-Splitting Methods*. IJCM, Taylor, USA, accepted, 1 Oct. 2008, published online in 2009.

10. R. Glowinski. *Numerical methods for fluids*. In: P.G. Ciarlet, J. Lions (Eds.) Handbook of Numerical Analysis, Vol. IX, North-Holland Elsevier, Amsterdam, 2003.
11. M. Hochbruck and A. Ostermann. *Explicit Exponential Runge-Kutta Methods for Semilinear Parabolic Problems*. SIAM Journal of Numerical Analysis, Vol. 43, Iss. 3, 1069–1090, 2005.
12. E. Hansen and A. Ostermann. *Exponential splitting for unbounded operators*. Mathematics of Computation, accepted, 2008.
13. T. Jahnke and C. Lubich. *Error bounds for exponential operator splittings*. BIT Numerical Mathematics, Vol. 40, Iss. 4, 735–745, 2000.
14. Y.L. Jiang, R.M.M. Chen, and O. Wing. *Convergence analysis of waveform relaxation for nonlinear differential-algebraic equations of index one*. IEEE Trans. Circuits and Systems Part I, Vol. 47, No. 11, 16391645, 2000.
15. Y.L. Jiang, R.M.M. Chen, O. Wang. *Waveform relaxation of nonlinear second-order differential equations*. IEEE Transactions on Circuits and Systems I: fundamental theory and applications, 48(11): 1344-1347, 2001.
16. Y.L. Jiang. *Periodic waveform relaxation solutions of nonlinear dynamic equations*. Applied Mathematics and Computation, 135(2-3): 219-226, 2003.
17. Y.L. Jiang, R.M.M. Chen, and O. Wing. *Periodic waveform relaxation of nonlinear dynamic systems by quasi-linearization*. IEEE Trans. Circuits and Systems - Part I, Vol. 50, No.4, 589593, 2003.
18. Y.L. Jiang. *A general approach to waveform relaxation solutions of differential-algebraic equations: the continuous-time and discrete-time cases*. IEEE Trans. Circuits and Systems - Part I, Vol. 51, No. 9, 17701780, 2004.
19. J. Kanney, C. Miller and C. Kelley. *Convergence of iterative split-operator approaches for approximating nonlinear reactive transport problems*. Advances in Water Resources, Vol. 26, 247–261, 2003.
20. C.T. Kelly. *Iterative Methods for Linear and Nonlinear Equations*. Frontiers in Applied Mathematics, SIAM, Philadelphia, 1995.
21. C.T. Kelley. *Solving Nonlinear Equations with Newton's Method*. Computational Mathematics, SIAM, XIV, 2003.
22. G.I. Marchuk. *Some applications of splitting-up methods to the solution of problems in mathematical physics*. Aplikace Matematiky, Vol. 1, 103–132, 1968.
23. U. Miekala and O. Nevanlinna. *Convergence of dynamic iteration methods for initial value problems*. SIAM Journal of Scientific and Statistical Computing, Vol. 8, Iss. 4, 459–482, 1987.
24. R.E. Showalter. *Hilbert Space Methods for Partial Differential Equations*. Pitman Publisher, 1994, Available at: <http://ejde.math.txstate.edu/Monographs/01/abstr.html>.
25. G. Strang. *On the construction and comparison of difference schemes*. SIAM Journal of Numerical Analysis, Vol. 5, 506–517, 1968.
26. S. Vandewalle. *Parallel Multigrid Waveform Relaxation for Parabolic Problems*. Teubner Skripten zur Numerik, B.G. Teubner Stuttgart, 1993.