Higher Order Operator-Splitting Methods via Zassenhaus product formula: Theory and Applications

Jürgen Geiser *and Gamze Tanoğlu †
Humboldt Univeristät zu Berlin, D- Berlin, Germany,
Izmir Institute of Technology, Gulbahce Campus, Urla, Izmir,35430, Turkey

July 26, 2009

Abstract

In this paper, we contribute higher order operator-splitting method improved by Zassenhaus product. We apply the contribution to classical and iterative splitting methods. The underlying analysis to obtain higher order operator-splitting methods is presented. While applying the methods to partial differential equations, the benefits of balancing time and spatial scales are discussed to accelerated the methods.

The verification of the improved splitting methods are done with numerical examples. An individual handling of each operators with adapted standard higher order time-integrators is discussed. Finally we conclude the higher order operator-splitting methods.

Keywords. Operator splitting method, Iterative solver method, Weighting methods, Zassenhaus product, Parabolic differential equations.

AMS subject classifications. 80A20, 80M25, 74S10, 76R50, 35J60, 35J65, 65M99, 65Z05, 65N12.

1 Introduction

The motivation to study the splitting methods are coming from model equations which simulate bio-remediation [2] or radioactive contaminants [9], [8].

The mathematical equations are given by

$$\partial_t \, R \, c + \nabla \cdot (\mathbf{v}c - D(c)\nabla c) = f(c) \;, \tag{1}$$

$$f(c) = c^p \;, \text{ chemical-reaction and } p > 0 \tag{2}$$

$$f(c) = \frac{c}{1-c} \;, \text{ bio-remediation} \tag{3}$$

The unknown $c = c(x,t)$ is considered in $\Omega \times (0,T) \subset \mathbb{R}^d \times \mathbb{R}$, the space-dimension is given by d . The Parameter $R \in \mathbb{R}^+$ is constant and is named as retardation factor. The other parameters $f(c)$ are nonlinear functions, for example bio-remediation or chemical reaction. $D(c)$ is the nonlinear diffusion-dispersion tensor and $\mathbf{v}$ is the velocity.

The aim of this paper is to study a novel splitting method which improve operator splitting methods. By weighting methods which embed the so called is Zassenhaus product, see [23], we

*email: geiser@mathematik.hu-berlin.de
†email: gamzetanoglu@iyte.edu.tr

improve the initial and starting conditions of the splitting process. To apply the methods, the discretization for the time-scales is done by combining explicit and implicit methods. The main advantage is using standard implicit and explicit Runge-Kutta or BDF-method and embed this methods in an iterative solver.

For the iterative operator-splitting methods, the delicate problem of low convergence (see [26]) can be improved by starting with sufficient accurate initial conditions. This is satisfied by weighting the method with the help of the Zassenhaus products.

2 Operator splitting methods

We focus our attention on the case of two linear operators (i.e we consider the Cauchy problem):

$$\frac{\partial c(t)}{\partial t} = Ac(t) \; + \; Bc(t), \text{ with } t \in [0, T], \; c(0) = c_0, \tag{4}$$

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 corresponds to physical operators such as convection and diffusion operators. We consider the following operators splitting schemes:

1. Sequential operator-splitting: A-B splitting

$$\frac{\partial c^*(t)}{\partial t} = Ac^*(t) \text{ with } t \in [t^n, t^{n+1}] \quad \text{and} \quad c^*(t^n) = c_{sp}^n \tag{5}$$

$$\frac{\partial c^{**}(t)}{\partial t} = Bc^{**}(t) \quad \text{with } t \in [t^n, t^{n+1}] \quad \text{and} \quad c^{**}(t^n) = c^*(t^{n+1}), \tag{6}$$

for $n = 0, 1, ..., N-1$ whereby $c_{sp}^n = c_0$ is given from (4). The approximated split solution at the point $t = t^{n+1}$ is defined as $c_{sp}^{n+1} = c^{**}(t^{n+1})$.

2. Strang-Marchuk operator-splitting : A-B-A splitting

$$\frac{\partial c^*(t)}{\partial t} = Ac^*(t) \text{ with } t \in [t^n, t^{n+1/2}] \quad \text{and} \quad c^*(t^n) = c_{sp}^n \tag{7}$$

$$\frac{\partial c^{**}(t)}{\partial t} = Bc^{**}(t) \quad \text{with } t \in [t^n, t^{n+1/2}] \quad \text{and} \quad c^{**}(t^n) = c^*(t^{n+1/2}) \;, \tag{8}$$

$$\frac{\partial c^{***}(t)}{\partial t} = Ac^*(t) \text{ with } t \in [t^{n+1/2}, t^{n+1}] \quad \text{and} \quad c^{***}(t^n + 1/2) = c^{**} t^{n+1}, \tag{9}$$

where $t^{n+1/2} = t^n + 0.5\tau_n$, and the approximated split solution at the point $t = t^{n+1}$ is defined as $c_{sp}^{n+1} = c^{***}(t^{n+1})$.

3. Iterative splitting with respect to one operator

$$\frac{\partial c_i(t)}{\partial t} = Ac_i(t) \; + \; Bc_{i-1}(t), \text{ with } \; c_i(t^n) = c^n, i = 1, 2, \ldots, m \tag{10}$$

4. Iterative splitting with respect to alternating operators

$$\frac{\partial c_i(t)}{\partial t} = Ac_i(t) \ + \ Bc_{i-1}(t), \text{ with } \ c_i(t^n) = c^n \tag{11}$$
$$i = 1, 2, \ldots, j \ ,$$
$$\frac{\partial c_i(t)}{\partial t} = Ac_{i-1}(t) \ + \ Bc_i(t), \quad \text{ with } \ c_{i+1}(t^n) = c^n \ , \tag{12}$$
$$i = j+1, j+2, \ldots, m \ ,$$

In addition, $c_0(t^n) = c^n$, $c_{-1} = 0$ and c^n is 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 $c^{n+1} = c_{2m+1}(t^{n+1})$. (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.)

3 Higher Order Operator Splitting Methods

Often standard splitting methods have the problem to be less effective in the rate of the convergence and CPU times.

Here we propose the followings to overcome these difficulties:

- Initialization: Improve the starting conditions via Zassenhaus product formula,
- Accelerated the subproblems via *Weighted Polynomials*,
- Extended operator Splitting methods via Zassenhaus product formula

3.1 Classical Operator Splitting Errors

The main problem is the initialization.

Often the $c_0(t) = c(t^n)$ or $c_0(t) = 0$ are to fare from the result, see

$$||c - c_0|| \leq err \tag{13}$$

where err is a given starting error.

By the way the standard initialization errors are

$$||c(t) - c_n|| \leq ||(\exp((A+B)t) - I)c_n|| \tag{14}$$
$$||c(t) - 0|| = ||\exp((A+B)t)c_n|| \tag{15}$$

and are of zero order and to large at all.

Here the ideas of prestepping methods,e.g. A-B splitting or Strang splitting as first or second order exponential splitting schemes can reduce the initial error.

See for the A-B splitting we have global a first order scheme

$$c_0(t) = \exp(At)\exp(Bt)c_n, \tag{16}$$
$$||c(t) - c_0(t)|| \leq O(t^2) \tag{17}$$

where for the Strang or A-B-A splitting we have global a second order scheme

$$c_0(t) = \exp(At/)\exp(Bt)\exp(A/2t)c_n, \tag{18}$$
$$||c(t) - c_0(t)|| \leq O(t^3) \tag{19}$$

3.2 Higher order A-B splitting by Initialization

In this subsection, we improve the order of the A-B splitting via Zassenhaus product formula as follows:

Theorem 3.2.1 *We solve the initial value problem (5) and (6). We assume bounded and constant operators A and B.*

The consistency error of the A-B splitting is $\mathcal{O}(t)$, then we can improve the error of the A-B splitting scheme to $\mathcal{O}(t^p), p > 1$ by improving the starting conditions c_0 as

$$c_0 = (\pi_{j=2}^{p} \exp(C_j t^j))c_0$$

where C_j is called as Zassenhaus exponents given in [24], thus local splitting error of A-B splitting method can be read as follows

$$\begin{aligned} \rho_n &= (\exp(\tau_n(A+B)) - \exp(\tau_n B)\exp(\tau_n A))c_{sp}^n \\ &= C_T\ \tau_n^{p+1} + \mathcal{O}(\tau_n^{p+2}) \end{aligned} \tag{20}$$

where C_T is a function of Lie brackets of A and B.

Proof 3.2.1 *Let us consider the subinterval [0, t], where $\tau = t$, the solution of the subproblem (5) is:*

$$c^*(t) = \exp(At)c_0 \tag{21}$$

after improving the initialization we have

$$c^*(t) = \exp(At)(\pi_{j=2}^{p} \exp(C_j t^j))c_0 \tag{22}$$

the solution of the subproblem (6) becomes

$$\begin{aligned} c^{**}(t) &= \exp(Bt)\ \exp(At)(\pi_{j=2}^{p} \exp(C_j t^j))c_0 \\ &= \exp((B+A)t)c_0 + \mathcal{O}(t^{p+1}) \end{aligned} \tag{23}$$

with the help of the Zassenhaus product formula.

Remark 3.2.2 *For example, the second order A-B splitting after improving the initialization is*

$$\begin{aligned} c^{**}(t) &= \exp(Bt)\ \exp(At)\ \exp(-\frac{1}{2}[B,A]t^2)c_0 \\ &= \exp((B+A)t)c_0 + \mathcal{O}(t^3) \end{aligned} \tag{24}$$

and the third order A-B splitting after improving the initialization is

$$\begin{aligned} c^{**}(t) &= \exp(Bt)\ \exp(At)\ \exp(-\frac{1}{2}[B,A]t^2)\ \exp((\frac{1}{6}[B,[B,A]] - \frac{1}{3}[A,[A,B]])t^3)c_0 \\ &= \exp((B+A)t)c_0 + \mathcal{O}(t^4) \end{aligned} \tag{25}$$

3.3 Higher order A-B-A splitting by accelerating the subproblems via *Weighted Polynomials*

In literature, Strang-Marchuk or A-B-A splitting is given as

$$\exp(At/2)\,\exp(Bt)\,\exp(At/2) = \exp((A+B)t) + \mathcal{O}(t^3)$$

since we would like to use the Zassenhaus product formula given as

$$\exp((A+B)t) \;=\; \exp(At)\,\exp(Bt)(\pi_{j=2}^{p}\exp(C_j t^j)) + \mathcal{O}(t^{p+1}) \tag{26}$$

in order to obtain higher order A-B-A splitting, we present the idea of the *Weighted Polynomials* in the following theorem:

Theorem 3.3.1 *We solve the initial value problem (7), (8) and (9) on the subinterval [0,t]. We assume bounded and constant operators A and B.*

The consistency error of the A-B-A splitting is $O(t^2)$, then we can improve the error of the A-B-A splitting scheme to $O(t^p), p > 2$ by applying the following steps:

- *Step 1: Improve the starting conditions $c^*(0) = c_0$ as*

  $$c^*(0) = (\pi_{j=2}^{p}\exp(C_j t^j))c_0$$

  where C_j is called as Zassenhaus exponents given in [24],

- *Step 2 : Accelerate $c^{**}(0)$ as*

  $$c^{**}(0) = (\exp(-A\,t))c^*(t/2),$$

- *Step 3: Accelerate $c^{***}(t/2)$ as*

  $$c^{***}(t/2) = (\exp(At/2))c^{**}(t),$$

thus the order of the A-B-A splitting method can be read as follows

$$\exp(At/2)\,\exp(Bt)\,\exp(At/2) = \exp((A+B)t) + \mathcal{O}(t^{p+1}).$$

Proof 3.3.1 *Let us consider the subinterval [0, t], the solution of the subproblem (7) is:*

$$c^*(t) \;=\; \exp(At)c_0 \tag{27}$$

after improving the initialization we have

$$c^*(t) \;=\; \exp(At)(\pi_{j=2}^{p}\exp(C_j t^j))c_0. \tag{28}$$

Next accelerate $c^*(t)$ as

$$c^*(t) \;=\; \exp(-At)c^*(t) \tag{29}$$

the solution of the subproblem (8) becomes

$$\begin{aligned} c^{**}(t) &= \exp(Bt)c^*(t/2) && (30)\\ &= \exp(Bt)\,\exp(-At/2)\exp(At/2)(\pi_{j=2}^{p}\exp(C_j (t/2)^j))c_0 && (31) \end{aligned}$$

or

$$c^{**}(t) = \exp(Bt)\,(\pi_{j=2}^{p}\exp(C_j(t/2)^j))c_0 \tag{32}$$

since [-A/2, A/2]=0. Finally, the acceleration of $c^{**}(t)$ is given by the equation

$$c^{**}(t) = \exp(At/2)\exp(Bt)\,(\pi_{j=2}^{p}\exp(C_j(t/2)^j))c_0, \tag{33}$$

then the solution of the subproblem (8) becomes

$$c^{***}(t) = \exp(At/2)\exp(At/2)\exp(Bt)\,(\pi_{j=2}^{p}\exp(C_j(t/2)^j))c_0 \tag{34}$$

or

$$c^{***}(t) = \exp(At)\exp(Bt)\,(\pi_{j=2}^{p}\exp(C_j(t/2)^j))c_0 \tag{35}$$

since [A/2, A/2]=0. This can be rewritten as

$$\begin{aligned} c^{***}(t) &= \exp(At)\exp(Bt)\,(\pi_{j=2}^{p}\exp(\tilde{C}_j(t)^j))c_0 & (36)\\ &= \exp((A+B)t) + \mathcal{O}(t^{p+1}). & (37)\end{aligned}$$

where $\tilde{C}_p = \frac{1}{2^p}C_j$ with the help of the Zassenhaus product formula.

3.4 Higher order A-B-A splitting based on Zassenhaus product formula

In this subsection, we first derive the Zassenhaus exponents by using the same approach given [24] for the A-B-A splitting. Again, using the formal power series expansion of exponential function, the Zassenhaus product for two non-commutative variables A t and B t for A-B-A splitting may be written as

$$\begin{aligned} e^{(A+B)t} &= \sum_{k=0}^{\infty}\frac{1}{k!}(A+B)^k t^k = I + (A+B)t + (\frac{1}{2}A^2 + \frac{1}{2}AB + \frac{1}{2}BA + \frac{1}{2}B^2)t^2 + \ldots. & (38)\\ &= (I + \frac{At}{2} + \frac{A^2t^2}{8} + \ldots)(I + Bt + \frac{B^2t^2}{2} + \ldots)(I + \frac{At}{2} + \frac{A^2t^2}{8} + \ldots)\prod_{n=3}^{\infty}(e^{D_nt^n}) & (39)\\ &= e^{\frac{At}{2}}e^{B}e^{\frac{At}{2}}e^{D_3t^3}e^{D_4t^4}\ldots. & (40)\end{aligned}$$

Our aim is to compute the polynomials D_n which are function of commutators $[., [[.,]]]$. One can find these polynomials by comparison method or Witschel's method. But, we use the ideas and notations which were first presented in Ref. [24] as follows:

Let $\tau_1, \ldots\tau_n$ be arbitary commutative variables and let $J = (J_{ij})$, $K = (K_{ij})$, and $L = (L_{ij})$ be three $(n+1)\times(n+1)$ matrices defined by $J_{ij} = 0$, $K_{ij} = 0$ and $L_{ij} = 0$ for $i > j$ and

$$J_{ij} = \frac{1}{(j-i)!}.\prod_{k=i}^{j-1}(1+\tau_k), \quad K_{ij} = \frac{(-1)^{(i+j)}}{(j-i)!} \quad \text{and} \quad L_{ij} = \frac{(-1)^{(i+j)}}{(j-i)!}\prod_{k=i}^{j-1}\tau_k \quad \text{for} \quad i \le j$$

Furthermore, they define the $(n+1)\times(n+1)$ matrices P and Q by

$$P_{ij} = \delta_{i+1,j} \quad \text{and} \quad Q_{ij} = \delta_{i+1,j}\tau_i$$

where $\delta_{i,j}$ is Kroneckerdelta. The operator U is defined in Ref. [24] but $a = At, b = Bt, c_n = D_nt^n$ We state the following corollary:

Corollary 3.4.1 *The Zassenhaus exponent c_3 defined in Eq. (40) is obtained in terms of the 4×4 matrices L, K, H where $H = \exp(1/2P + Q + 1/2P)$, $K = \exp(-1/2P)$ and $L = exp(-Q)$ as $c_3 = U(K.L.K.H)_{1,4}$. For $n > 2$, the Zassenhaus exponents c_k is given in terms of the corresponding $(n+1) \times (n+1)$ matrices as*

$$c_k = U((e^{-C_{k-1}}........e^{-C_3}.K.L.K.H)_{1,n+1}). \tag{41}$$

Here, $C_m(1 < m < n)$ are the the Zassenhaus exponents written in terms of the $(n+1) \times (n+1)$ matrices P and Q, and the index (1,n+1) indicates the upper right element of a matrix.

Proof 3.4.1 *Each element $n \in N$ can ce written as*

$$e^{P+Q} = e^{P/2}.e^{Q}.e^{P/2} \prod_{i=3}^{n} (e^{C_i}), \tag{42}$$

Therefore one obtains

$$e^{C_n} = e^{-C_{n-1}}....e^{-C_3}.e^{P/2}.e^{Q}.e^{P/2}.e^{P+Q}. \tag{43}$$

The rest of the proof is the same as in Ref. [24].

Corollary 3.4.2 *The Zassenhaus exponent D_3 given in Eq. (39) can be found as*

$$D_3 = \frac{1}{24}[B,[B,A]] - \frac{1}{12}[A,[A,B]] \tag{44}$$

by comparing the exact solution given in (38) with the expansion up to the order $\mathcal{O}(t^4)$ given Eq. (40). Thus if the weight $w_3 = I + D_3 t^3$ is chosen and multiplied by the initial condition, the order of the A-B-A splitting becomes $\mathcal{O}(t^3)$.

Proof 3.4.2 *The splitting error of Strang splitting or A-B-A splitting is*

$$\rho = \exp((A+B)t) - \exp(At/2)\exp(Bt)\exp(At/2) \tag{45}$$

$$= (\frac{1}{24}[B,[B,A]] - \frac{1}{12}[A,[A,B]])t^3 \tag{46}$$

The coefficient of t^3 given in the expansion

$$e^{(A+B)t} = e^{\frac{At}{2}} e^{B} e^{\frac{At}{2}} e^{D_3 t^3} + \mathcal{O}(t^4) \tag{47}$$

is

$$D_3 + \frac{(A+B)^3}{3!} - \rho,$$

thus if we choose $D_3 = \rho$, the splitting error becomes $\mathcal{O}(t^3)$.

3.5 Higher order iterative splitting based on Zassenhaus product formula

Waveform relaxation (one operator):

Theorem 3.5.1 *We solve the initial value problem (10). We assume bounded and constant operators A_1, A_2. The initial step is given as $c_1(t) = \exp(At)c_0$.*

Then we can improve the error of the iterative scheme to $\mathcal{O}(t^{i+j})$ by multiplying a weighted function with the kernel $\omega_j(t) = \exp(Bt)\Pi_{k=2}^{i}\exp(\hat{c}_k t^k) + O(t^j)$ to $c_{i-1}(t)$, where $\hat{c}_k$ are the so called Zassenhaus exponents, see [24].

Proof 3.5.1 *The iterative scheme is for the step c_2 as:*

$$c_i = \exp(At)c_0 + \int_0^t \exp(A(t-s))B\exp(As)c_0 ds \tag{48}$$

where $c_1(t) = \exp(As)c_0$.

We improve the method as:

$$\tilde{c}_1(t) = \exp(At)\exp(Bt)\Pi_{k=2}^{i}\exp(\hat{c_k}t^k)c_0 \tag{49}$$

where we obtain the exact solution:

$$c_i = \exp(At)c_0 + \int_0^t \exp(A(t-s))B\exp((A+B)s)c_0 ds, \tag{50}$$

based on the formulation of the Zassenhaus product formula (see [23])

$$\exp((A+B)t) = \exp(At)\exp(Bt)\Pi_{k=2}^{\infty}\exp(\hat{c_k}t^k), \tag{51}$$

we can derive the weights are given as:

$$w_i(t) = \exp(Bt)\exp(\hat{c_2})\exp(\hat{c_3}).....\exp(\hat{c_i}) + \mathcal{O}(t^{i+1}) \tag{52}$$

where $\hat{c_i}, i = 2, ...\infty$ are Zassenhaus exponents as follows:

$$\begin{aligned} &\hat{c_2} = -1/2[A,B] \\ &\hat{c_3} = (-1/3[B,[B,A]] + 1/6[A,[A,B]]) \\ &\hat{c_4} = (-1/24[[[A,B],A],A] - 1/8[[[A,B],A],B] - 1/8[[[A,B],B],B]) \end{aligned} \tag{53}$$

Thus some examples for weights are given as:

$$w_1(t) = I + Bt \tag{54}$$

$$w_2(t) = I + Bt + B^2t^2/2 - 1/2[A,B]t^2 \tag{55}$$

$$\begin{aligned} &w_3(t) = I + Bt + B^2t^2/2 - 1/2[A,B]t^2 \\ &+B^3/3t^3 + (-1/3[B,[B,A]] + 1/6[A,[A,B]])t^3 - 1/2B[A,B]t^3 \end{aligned} \tag{56}$$

$$\begin{aligned} &w_4(t) = I + Bt + B^2t^2/2 - 1/2[A,B]t^2 \\ &+B^3/3!t^3 + (-1/3[B,[B,A]] + 1/6[A,[A,B]])t^3 - 1/2B[A,B]t^4 \\ &+B^4/4!t^4 + (-1/24[[[A,B],A],A] \\ &-1/8[[[A,B],A],B] - 1/8[[[A,B],B],B])t^4 \end{aligned} \tag{57}$$

$$(-1/3B[B,[B,A]] + 1/6B[A,[A,B]])t^4 - (\frac{B^2}{4}[A,B])t^4 + 1/4[A,B]^2t^4 \tag{58}$$

Same can be done for the iterative splitting method.

3.5.1 Higher order Iterative Splitting with respect alternating operators based Comparison or Witschel's Method

Consider the Equation (11) and (12), the exact solution of the iteration can be found by using variation of constant formula as follows:

$$c_i(t) = \exp(At)c_0 + \exp(At)\int_0^t \exp(-As)Bc_{i-1}\, ds \tag{59}$$

$$c_{i+1}(t) = \exp(Bt)c_0 + \exp(Bt)\int_0^t \exp(-Bs)Ac_i\,ds \tag{60}$$

Assume that $c_{i-1} = 0$ then
for i=1,

$$c_1(t) = \exp(At)c_0 \tag{61}$$

for i=2,

$$\begin{aligned}
c_2(t) &= \exp(Bt)(I + \int_0^t \exp(-Bs)A\exp(As)\,ds)c_0 \qquad (62)\\
&= \exp(Bt)(I + \int_0^t (I - Bs)A(I + As)\ ds)c_0\\
&= \exp(Bt)(I + \int_0^t (A + A^2 s - ABs + \mathcal{O}(s^2)\ ds)c_0\\
&= \exp(Bt)(I + At + \frac{A^2t^2}{2} - \frac{ABt^2}{2})c_0 + \mathcal{O}(t^3)\\
&= (I + Bt + \frac{B^2t^2}{2})(I + At + \frac{A^2t^2}{2} - \frac{ABt^2}{2})c_0 + \mathcal{O}(t^3)\\
&= (I + (A+B)t + \frac{B^2t^2}{2} + BAt^2 + \frac{A^2t^2}{2} - \frac{ABt^2}{2})c_0 + \mathcal{O}(t^3)\\
&= \exp(At + Bt) + \mathcal{O}(t^2).
\end{aligned}$$

In next theorem, we show how to increase the order of the accuracy with respect to the *Weighted Polynomials.*

Theorem 3.5.2 *There exists a Weighted Polynomial so that the order of the accuracy of iterative splitting with alternating operators can be increased up to $\mathcal{O}(t^3)$.*

Proof 3.5.2 *We give the proof by construction in the following steps:*

- *Step 1 : Start the initiation as $c_{i-1} = 0$*
- *Step 2 : Accelerate the c_1 as $c_1 = (I + Wt)c_1$*
- *Step 3 : Compute c_2 by using the Equation (60) as*

$$\begin{aligned}
c_2(t) &= \exp(Bt)(I + \int_0^t \exp(-Bs)\exp(As)(I + Ws)\,ds)c_0 \qquad (63)\\
&= \exp(Bt)(I + \int_0^t (I - Bs)A(I + As)(I + Ws) + \mathcal{O}(s^2)\,ds)c_0\\
&= \exp(Bt)(I + \int_0^t (A + AWs - BAs + A^2 s + \mathcal{O}(s^2)\,ds)c_0\\
&= \exp(Bt)(I + At + \frac{AWt^2}{2} - \frac{BAt^2}{2} + \frac{A^2t^2}{2})c_0 + \mathcal{O}(t^3)
\end{aligned}$$

- *Step 4 : Next expand* $\exp(Bt)$ *up to* $\mathcal{O}(t^3)$,

$$\begin{aligned} c_2(t) &= (I + Bt + \frac{B^2t^2}{2})(I + At + \frac{AWt^2}{2} - \frac{BAt^2}{2} + \frac{A^2t^2}{2})c_0 + \mathcal{O}(t^3) \\ &= (I + (A+B)t + \frac{B^2t^2}{2} + BAt^2 + \frac{AWt^2}{2} + \frac{A^2t^2}{2})c_0 + \mathcal{O}(t^3) \end{aligned} \tag{64}$$

- *Step 5 : Finally compare this with exact solution up to* $\mathcal{O}(t^3)$ *to find the commutator and* w *as follows. The exact solution of the problem is given by,*

$$\begin{aligned} c_{exact} &= e^{(A+B)t} \\ &= (I + (A+B)t + (\frac{A^2}{2} + \frac{AB}{2} + \frac{BA}{2} + \frac{B^2}{2})t^2)c_0 + \mathcal{O}(t^3) \end{aligned} \tag{65}$$

and the error can be found by subtracting the Equation (65) from (64) as follows

$$|c_{exact} - c_2| \leq (\frac{AB}{2} - \frac{AW}{2})O(t^2) + \mathcal{O}(t^3), \tag{66}$$

From this expression if $W = B$*, the order of the accuracy of iterative splitting with respect to alternating operators can be increased up to* $O(t^3)$*, thus we can find the Weighted Polynomial as follows:*

$$w_1 = I + Bt. \tag{67}$$

Note that this is the same as the weight found in the Equation (54). We proved that the order of the accuracy of iterative splitting with respect to alternating operators can be increased up to $\mathcal{O}(t^3)$ *via Weighted Polynomial defined in Equation (67), Therefore,*

$$|c_{exact} - c_{(i=2)}| \leq C.\,\mathcal{O}(t^3). \tag{68}$$

where C *is the function of Commutators.*

Theorem 3.5.3 *There exists a Weighted Polynomial so that the order of the accuracy of iterative splitting with alternating operators can be increased up to* $O(t^4)$ *after the second iteration.*

Proof 3.5.3 *We give the proof by construction in the following steps:*
Step 1 : Start the initiation as $c_{i-1} = 0$
Step 2 : Accelerate the c_1 *as* $c_1 = (I + W_1t + W_2t^2)c_1$
Step 3 : Compute c_2 *by using the Equation* (62) *as*

$$\begin{aligned} c_2(t) &= \exp(Bt)(I + \int_0^t \exp(-Bs)A\exp(As)(I + W_1s + W_2s^2)\,ds)c_0 \\ &= \exp(Bt)(I + \int_0^t (I - Bs + \frac{B^2s^2}{2})A(I + As + \frac{A^2s^2}{2})(I + W_1s + W_2s^2) + \mathcal{O}(s^2)\,ds)c_0 \\ &= \exp(Bt)(I + (\int_0^t (A + (A^2 + AW_1 - BA)s)\,ds \\ &\quad + \int_0^t (\frac{A^3}{2} + AW_1A - BA^2 + AW_2 - BAW_1 + \frac{B^2A}{2})s^2)\,ds)c_0 + \mathcal{O}(s^4) \end{aligned} \tag{69}$$

After integrating the the expression in Equation (69) on the right, $c_2(t)$ becomes

$$\begin{aligned} c_2(t) &= \exp(Bt)(I + At + (A^2 + AW_1 - BA)\frac{t^2}{2} \\ &+ (\frac{A^3}{2} + (A^2W_1 - BA^2 + AW_2 - BAW_1 + \frac{B^2A}{2})\frac{t^3}{3})c_0 + \mathcal{O}(t^4) \end{aligned} \tag{70}$$

Step 4 : Next expand $\exp(Bt)$ *up to* $O(t^3)$ *and insert this into equation (69)*

$$\begin{aligned} c_2(t) &= (I + Bt + \frac{B^2t^2}{2} + \frac{B^3t^3}{3})(I + At + (A^2 + AW_1 - BA)\frac{t^2}{2} + \\ & (\frac{A^3}{2} + A^2W_1 - BA^2 + AW_2 - BAW_1 + \frac{B^2A}{2})\frac{t^3}{3})c_0 + \mathcal{O}(t^4) \end{aligned} \tag{71}$$

Step 5 : Finally by comparing this with exact solution up to $\mathcal{O}(t^3)$, which is given by

$$\begin{aligned} c_{exact} &= e^{(A+B)t} \\ &= (I + (A+B)t + (\frac{A^2}{2} + \frac{AB}{2} + \frac{BA}{2} + \frac{B^2}{2})t^2 \\ &+ (\frac{A^3}{6} + \frac{A^2B}{6} + \frac{ABA}{6} + \frac{AB^2}{6} + \frac{BA^2}{6} + \frac{BAB}{6} + \frac{B^2A}{6} + \frac{B^3}{6}))c_0 + \mathcal{O}(t^4), \end{aligned} \tag{72}$$

then we find a weight function $w_2 = I + W_1t + W_2t^2$ where

$$W_1 = B, \quad W_2 = \frac{B^2 - [A, B]}{2}.$$

Note that this is the same as the weight found in the Equation (55). Finally, by putting these values into the error which can be found by subtracting the Equation (72) from (71), we have

$$|c_{exact} - c_2| \leq D.\mathcal{O}(t^4), \tag{73}$$

where D is the function of Commutators.

4 Extended splitting method based on Zassenhaus-formula

The standard exponential splitting methods are based on the following decomposition idea:

$$\exp((A + B)t) = \pi_{i=1}^{j} \exp(a_iAt)\exp(b_iBt) + O(t^{j+1}). \tag{74}$$

The extension to the exponential splitting schemes are given as:

$$\exp((A + B)t) = \pi_{i=1}^{j} \exp(a_iAt)\exp(b_iBt)\pi_{k=j}^{m} \exp(C_jt^j) + O(t^{m+1}). \tag{75}$$

where C_j is a function of Lie brackets of A and B.

Theorem 4.0.4 *The initial value problem (10) is solved by classical exponential splitting schemes. We assume bounded and constant operators A_1, A_2.*

Then we can find extensions based on the Zassenhaus formula given as

$$\exp((A + B)t) = \pi_{i=1}^{j} \exp(a_iAt)\exp(b_iBt)\pi_{k=j}^{m} \exp(C_jt^j) + O(t^{m+1}). \tag{76}$$

where C_j is a function of Lie brackets of A and B.

Proof 4.0.4 *1.) Lie-Trotter splitting:*

For the Lie-Trotter splitting there exists coefficients with respect to the extension:

$$\exp((A+B)t) = \exp(At)\exp(Bt)\Pi_{k=2}^{\infty}\exp(C_k t^k), \tag{77}$$

where the coefficients C_k are given in [24].

Based on an existing BCH formula of the Lie-Trotter splitting one can apply the Zassenhaus formula.

2.) Strang Splitting:

A existing BCH formula is given as:

$$\exp(At/2)\exp(Bt)\exp(At/2) = \exp(tS_1 + t^3 S_3 + t^5 S_5 + \ldots)), \tag{78}$$

where the coefficients S_i are given as in [17].

There exists an Zassenhaus formula based on the BCH formula.

See:

$$\exp((A/2+B/2)t) = \Pi_{k=2}^{\infty}\exp(\tilde{C}_k t^k)\exp(A/2t)\exp(B/2t), \tag{79}$$

and

$$\exp((B/2+A/2)t) = \exp(B/2t)\exp(A/2t)\Pi_{k=2}^{\infty}\exp(C_k t^k), \tag{80}$$

then there exists a new product:

$$\Pi_{k=3}^{\infty}\exp(D_k t^k) = \Pi_{k=2}^{\infty}\exp(\tilde{C}_k t^k)\Pi_{k=2}^{\infty}\exp(C_k t^k), \tag{81}$$

with one order higher, see also [28].

3.) General exponential splitting:

Same can be done with the general exponential splitting schemes.

5 Balancing of Time- and spatial discretization

Splitting methods are important for partial differential equations, because of reducing computational time to solve the equations and accelerating the solver process, see [13].

Here additional balancing is taken into account, because of the spatial step.

The following theorem, addresses the delicate situation of time- and spatial steps and the fact of reducing the theoretical promised order of the scheme:

Theorem 5.0.5 *We solve the initial value problem by applying iterative operator splitting schemes (11) and (12). We assume bounded and constant operators A, B. While iterating i-time with A and j-time with A_2 the theoretical order is given as $O(t^{i+j})$ The initial step is given as $c_1(t) = \exp(At)\exp(Bt)c_0$.*

Then we reduce order of the iterative scheme to $O(t^i)$, while norm of B is larger or equal than $O(\frac{1}{t})$ same is also with the operator A.

So the balancing below the so called CFL condition is important to preserve the order of the splitting method.

Proof 5.0.5 *The theoretical order of the iterative splitting scheme is given as:*
$||c_{i+j} - c|| \leq ||A|| \, ||B||$
$; t^{i+j} + O(t^{i+j+1})$
where $||A|| = \rho(A)$ *is the spectral or the maximum eigenvalue of operator* A *and* $||B|| = \rho(B)$ *is the spectral or the maximum eigenvalue of operator* B.
Based on the spatial discretization we have the following eigenvalues:
$\rho(A) = \frac{a_1}{\Delta x^p}$
$\rho(B) = \frac{a_2}{\Delta x^q}$

where we have a p-th order spatial discretization of A *and a q-th order spatial discretization of* B, a_1, a_2 *are the diagonal entries of the finite difference stencil, see [14].*
If we assume to have a CFL-condition ≥ 1 *for the operator* B *we obtain:*

$$\frac{a_1}{\Delta x^p} t \geq 1 \tag{82}$$

and therefore:

$$||A_2||^j t^j = O(1). \tag{83}$$

We lost the order for operator B *and reduce to the order of the operator* A.
Same can be done for operator A.
Therefor we have a necessary restriction to preserve the order of the splitting method given as:
$O(1) \geq \rho(A) \geq O(\frac{1}{t})$.
We preserve the order:

$$||B||^j t^j = O(t^j). \tag{84}$$

Remark 5.0.6 *By using implicit method for the discretization scheme, we did not couple the time-scale and the spatial scale by a CFL condition and are so fare independent of the reduction but taken into account less accurate results.*

6 Numerical Examples

We consider the following test problems in order to verify our theoretical findings in the previous sections.

6.1 First Test- Example : Eigenvalue Problem

We first deal with the following ordinary differential Equation

$$\frac{\partial c(t)}{\partial t} = \lambda c(t) \text{with } t \in [0, T], \; c(0) = 1. \tag{85}$$

We divide our ODE's in subequations after applying the one operator splitting method as following

$$\frac{\partial c_i(t)}{\partial t} = -\lambda_1 c_i(t) + \lambda_2 c_{i-1}(t), \text{ with } c_i(t^n) = c^n, i = 1, 2, \ldots, m \tag{86}$$

where $\lambda_1 = -1$ and $\lambda_2 = 2$, initial condition is $c(0) = 1$. The exact solution of the problem is $c_{exact} = e^x$. We applied the midpoint rule to find the approximate solution. Since there is no splitting error we have following proposition:

Proposition 6.1.1 *The order of the accuracy of the iterative splitting (10) is two after applying the midpoint to each sub equations.*

Proof 6.1.1 *We obtain following finite difference approximation after discretization the Equation (10) by midpoint method on* $[0, \tau]$,

$$c_i(\tau) = \chi_1 c^0 + \chi_2 \frac{\tau}{2} \lambda_2 (c_{i-1}(0) + c_{i-1}(\tau))(\tau) \quad i = 1, 2, \ldots, m \tag{87}$$

where χ_1 *is defined as follows if* $|\frac{\lambda_1}{2}\tau| < 1$,

$$\begin{aligned} \chi_1 &= \frac{1 - \frac{\lambda_1}{2}\tau}{1 + \frac{\lambda_1}{2}\tau} && (88) \\ &= 1 - (\lambda_1 \tau) + \frac{\lambda_1^2 \tau^2}{2} + O(\tau^3), && (89) \\ &= e^{-\lambda_1 \tau} + O(\tau^3) && (90) \end{aligned}$$

and Pade Approximation of the $e^{-\lambda_1 \tau}$ *up to the order* $O(\tau^3)$ *and* χ_2 *is defined as follows if* $|\frac{\lambda_1}{2}\tau| < 1$

$$\begin{aligned} \chi_2 &= \frac{\lambda_2 \frac{\tau}{2}}{1 + \frac{\lambda_1 \tau}{2}} && (91) \\ &= \frac{\lambda_2 \tau}{2} - \frac{\lambda_1 \lambda_2 \tau^2}{4} + O(\tau^3), && (92) \end{aligned}$$

assume that $c_{i-1} = 0$, by inserting this into Equation (87), we have for i= 1,

$$c_1(\tau) = \chi_1 c^0 \tag{93}$$

$$\tag{94}$$

for i=2,

$$c_2(\tau) = (\chi_1 + \chi_2(1 + \chi_1))c^0 \tag{95}$$

We can easily see that this approximation does not give the exact solution up to the second order, then we need to compute next iteration as follows,

$$\begin{aligned} c_3(\tau) &= (\chi_1 + \chi_2(1 + (\chi_1 + \chi_2(1 + \chi_1))))c^0 && (96) \\ &= \chi_1(1 + {\chi_1}^{-1}\chi_2(1 + (\chi_1 + \chi_2(1 + \chi_1))))c^0 && (97) \end{aligned}$$

after expanding the terms up to third order we may see that following result

$$c_3(\tau) = e^{(-\lambda_1 + \lambda_2)\tau} + O(\tau^3). \tag{98}$$

In the first experiment, we exhibit the solution of the eigenvalue problem by using the weight in Equation (54) as $w = 1 + \tau\lambda_2$, since $W = B = \lambda_2$. The Figure (1) shows that the same order of accuracy can be reached by using the less iteration via *Weighted Polynomial.*

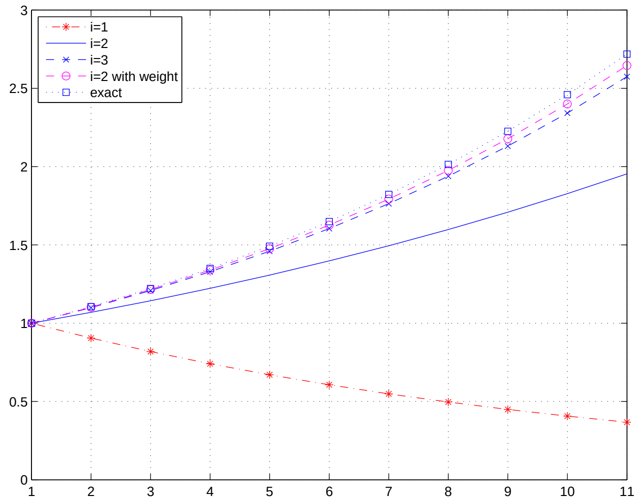

Figure 1: Comparison of the solutions obtained by midpoint method for different number of iterations and iteration with weight

6.2 Second Test-Example: Matrix Problem

We deal with the following problem:

$$\frac{\partial u(t)}{\partial t} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} u \tag{99}$$

with the initial conditions $u_0 = (1 - 1)$ on the interval [0, T].

The analytical solution is given by :

$$u(t) = \begin{pmatrix} e^{-t} \\ e^{t} \end{pmatrix} \tag{100}$$

We split our linear operators into two operators by setting:

$$\frac{\partial u(t)}{\partial t} = \begin{pmatrix} 2 & -1 \\ -1 & 0 \end{pmatrix} u + \begin{pmatrix} -2 & 2 \\ 2 & 0 \end{pmatrix} u \tag{101}$$

Not that the matrices are not commute. For integration constants we use a step size of $h = 10^{-2}$. We apply the third order Runge-Kutta method to our iterative scheme with respect to the one operator. We compare the first component of the solution obtained from weighted and without weighted iterative scheme with exact solution in Figure 2.

In the Figure (3), we show the rate of convergence on $[0, \triangle t]$ obtained from weighted and without weighted iterative scheme.

In Figures (4) and (5), we compare the different weight polynomials, one term weight we mean $w_1 = I + Bt$, two term weight we mean

$$w_2 = I + Bt + (B^2 - [A, B])\frac{t^2}{2}$$

for $dt = 0.04$ and $dt = 0.02$, respectively for alternating operator splitting.

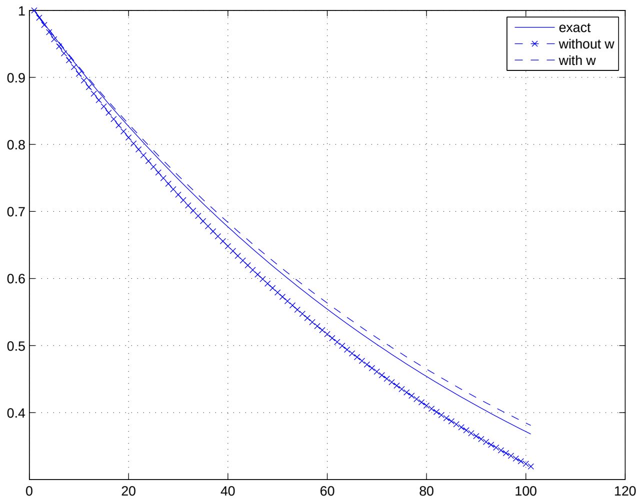

Figure 2: Comparison of the solutions of matrix problem by one-operator splitting solved by the third order Runge-Kutta method with weight or without weight for dt=0.01.

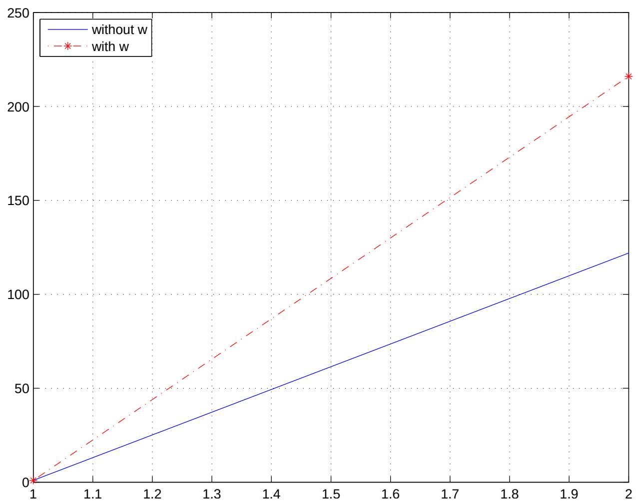

Figure 3: Rate of the convergency of the Matrix Problem solved one-operator splitting solved by the third order Runge-Kutta Method.

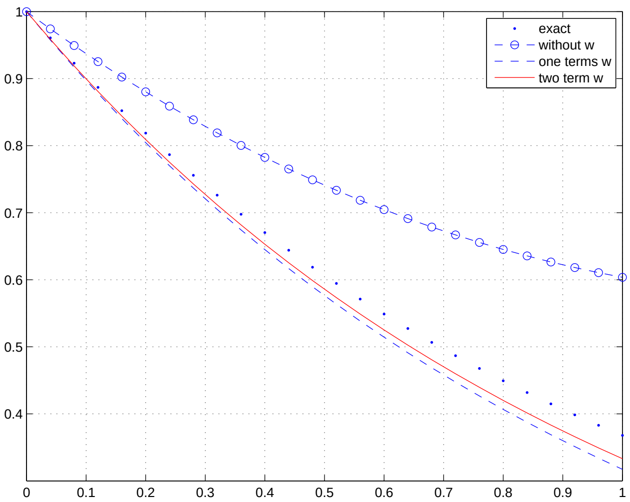

Figure 4: Comparison of the solutions of Matrix Problem obtained by iterative-splitting method solved the third order Runge-Kutta method with weight, without weight, one term weight, two terms weight for dt=0.04.

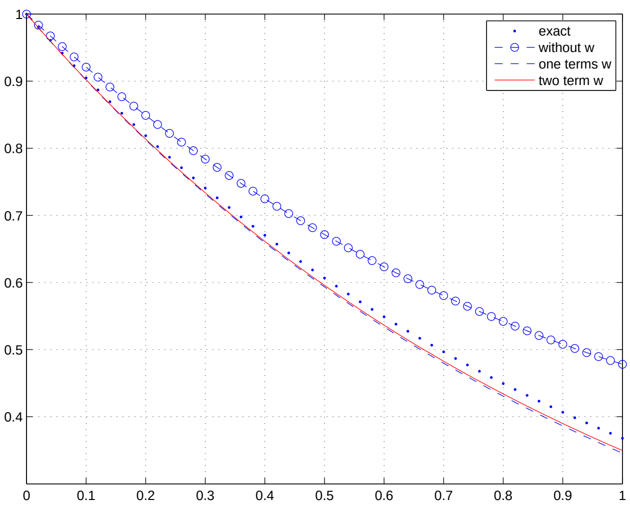

Figure 5: Comparison of the solutions of Matrix Problem obtained by iterative-splitting method and the third order Runge-Kutta method with weight, without weight, one term weight, two terms weight for dt=0.02

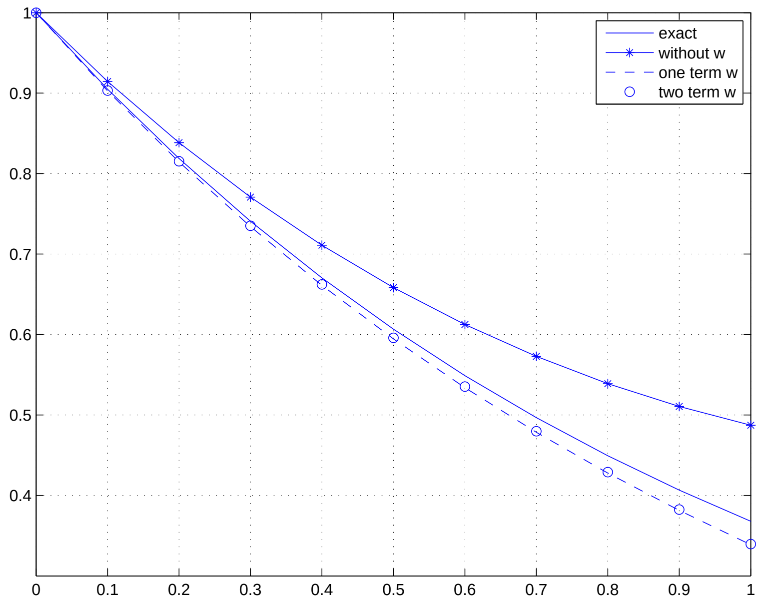

Figure 6: Comparison of the solutions of Matrix Problem obtained by Lie-Trotter splitting solved by fourth order Runge-Kutta Method without weight , one term weight,two term weight for dt=0.01.

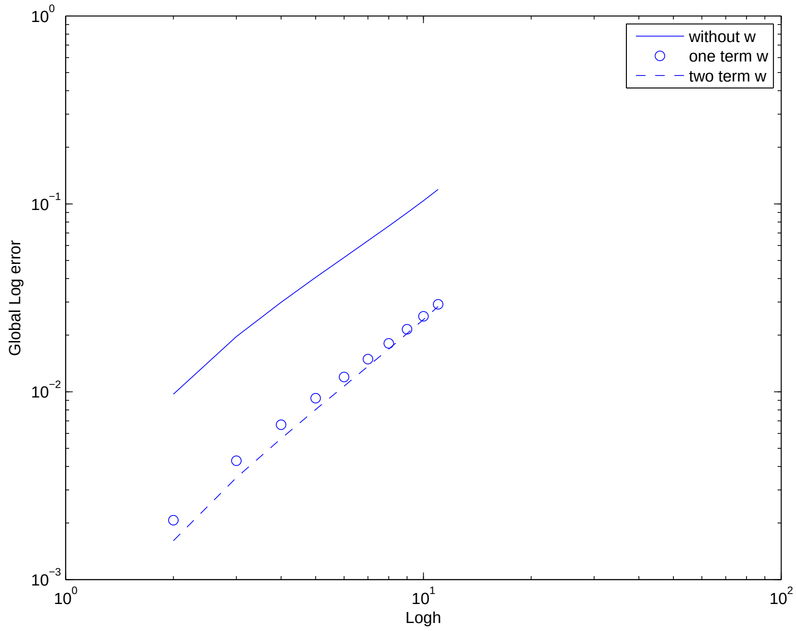

Figure 7: Comparison of the errors for Matrix Problem obtained by Lie-Trotter splitting solved by fourth order Runge-Kutta Method without weight , one term weight,two term weight for dt=0.01.

		err_{L_∞}	err_{L_1}
Lie Trotter Splitting	Without w	0.1194	0.0060
	With one w	0.0292	0.0014
	With two w	0.0284	0.0013

Table 1: Comparison of errors for matrix problem solved by Lie-Trotter Splitting and Runge -Kutta 4th-order method

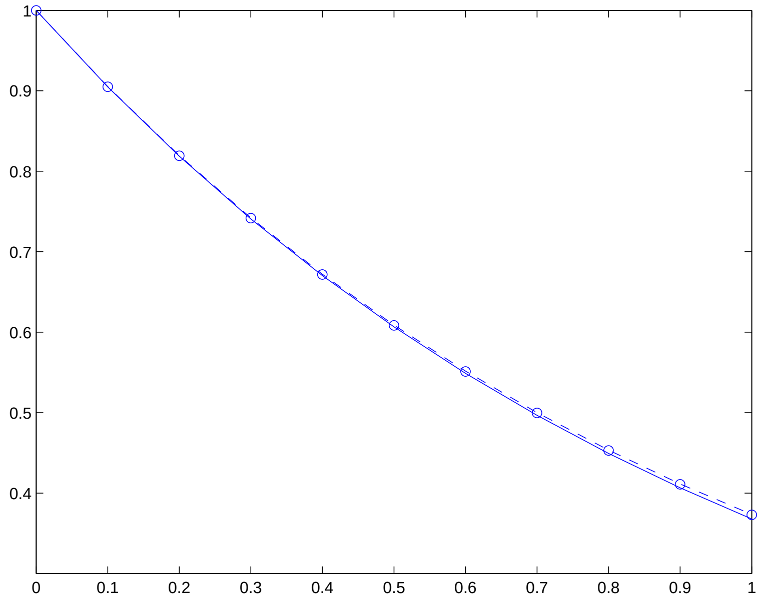

Figure 8: Comparison of the solutions obtained by Strang splitting solved fourth order Runge-Kutta Method without weight , one term weight for dt=0.01

Next, We apply fourth order Runge-Kutta method with Lie-Trotter Splitting to the same problem and compare the solutions without weight, one term weight and two term weight. Results are given in the following figures.

In the following Table, we used the weight obtained in Corollary 3.4.2 for Strang Splitting solved by fourth order Runge-Kutta Method :

		err_{L_∞}	err_{L_1}
Strang Splitting	Without w	0.0055	2.7104e-004
	With one w	0.0051	2.3562e-004

Table 2: Comparison of errors for Matrix Problem with Strang Splitting and Runge -Kutta 4th order method for $dt = 0.01$.

6.3 Third Test-Example: Hyperbolic Equation

We consider the following test problem:

$$u_t + au_x - bu = 0 \tag{102}$$

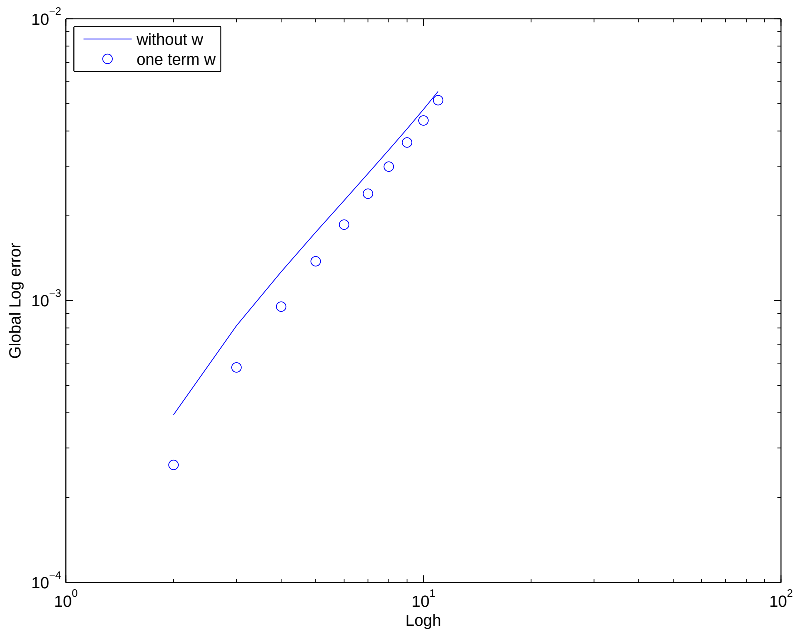

Figure 9: Comparison of the errors for Matrix problem obtained by Strang splitting solved by fourth order Runge-Kutta Method without weight , one term weight for dt=0.01

where $(x,t) \in [0,20] \times [1900,2000]$, $R = 1$, $v = 0.001$, $D = 0.0001$ and $\lambda = 10^{-5}$ with exact solution $u(x,t) = \frac{1}{2\sqrt{D\Pi t}} e^{\frac{-(x-vt)^2}{4D\Pi t}} e^{-\lambda t}$ and initial conditions ,boundary conditions are taken from exact solution.

Comparison of errors with Iterative splitting solved by Trapezoidal rule for space discretization, $h = 0.02$ and time discretization $dt = 0.01$ is exhibited in Table (3) for hyperbolic test problem.

		err_{L_∞}	err_{L_1}
Iterative Method	Without w	0.0284	0.0216
	w_1	0.0072	0.0025
	w_2	0.0048	0.0023

Table 3: Comparison of errors for Hyperbolic Problem with Iterative splitting solved by Trapezoidal rule for $h = 0.02$ and $dt = 0.01$.

The L_∞ norm is given by

$$err_{L_\infty} := max(max(|u(x_i, y_j, t^n) - u_{analy}(x_i, y_j, t^n)|))$$

.

The numerical convergence rate is given by

$$err_p := \frac{\ln(err_{L_\infty}(\triangle_{t_1})/err_{L_\infty}(\triangle_{t_2}))}{\ln(\triangle_{t_1}/\triangle_{t_2})} \tag{103}$$

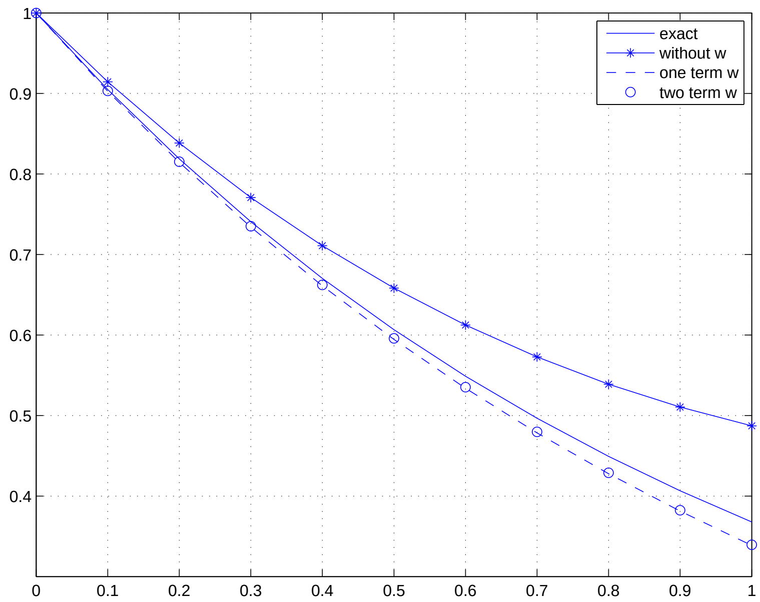

Figure 10: Comparison of the solutions of Matrix Problem with Strang Splitting solved by fourth order Runge -Kutta with weight, without weight, one term weight for dt=0.1

In Table (6.3), the rate of the convergence of iterative splitting solved by trapezoidal rule for hyperbolic Equation is exhibited.

Convergence Rate for iterative Splitting	Without w	0.5883
	w_1	1.7843
	w_2	2.6630

Table 4: Rate of convergence with L_∞ norm when $\triangle_{t_1} = 0.01$ and $\triangle_{t_2} = 0.02$

Convergence Rate	Without w	-0.3370
	w_1	1.2630
	w_2	1.4537

Table 5: Rate of convergence with L_1 norm when $\triangle_{t_1} = 0.01$ and $\triangle_{t_2} = 0.02$

6.4 Fourth Test-Example: Parabolic Equation

We consider the following test problem as final example:

$$Ru_t + vu_x - Du_{xx} = -\lambda u \tag{104}$$

where $(x, t) \in [0, 10] \times [1900, 2000]$, $R = 1$, $v = 0.001$, $D = 0.0001$ and $\lambda = 10^{-5}$ with exact solution $u(x, t) = \frac{1}{2\sqrt{D\Pi t}} e^{\frac{-(x-vt)^2}{4D\Pi t}} e^{-\lambda t}$ and initial conditions ,boundary conditions are taken from exact solution. Figure (11) exhibits the solution of the parabolic problem by iterative

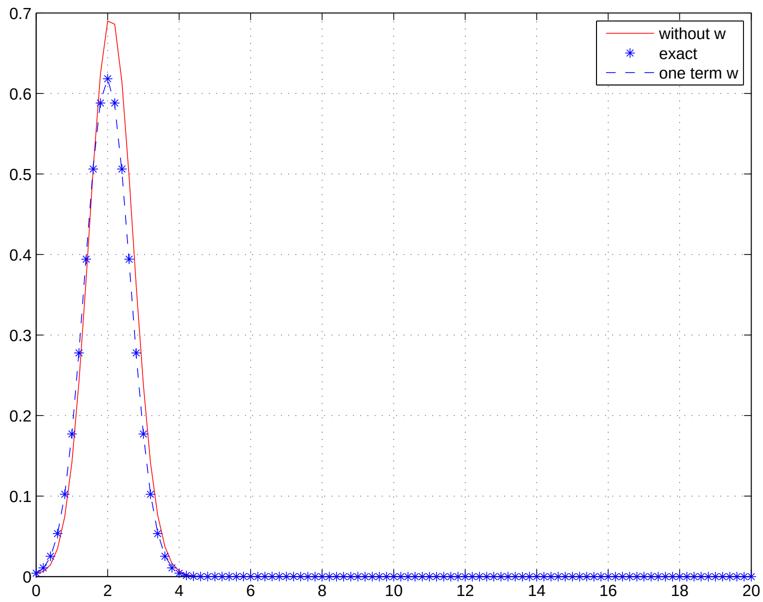

Figure 11: Comparison of the solutions of Parabolic Problem obtained by Iterative splitting and fourth order Runge-Kutta Method without weight , one term weight for h=0.2 and dt=1.

splitting method solved by 4-th order Runge-Kutta method for without weight, w_1, w_2 given in the Equations (54) and (55), respectively.

Errors computed by L_∞ norm and L_1 norm are presented in the next table.

		err_{L_∞}	err_{L_1}
Iterative Method	Without w	0.0574	1.2108
	w_1	0.0037	0.0569
	w_2	0.0037	0.0569

Table 6: Comparison of errors in Parabolic problem measured by L_∞ norm and andL_1 norm after applying Iterative splitting solved by fourth order Runge-Kutta Method for $h = 0.2$ and $dt = 1$.

The errors of solutions of parabolic problem using Iterative splitting solved by fourth order Runge-Kutta for without weight, w_1 are presented in Figure (12).

The errors of solutions of parabolic problem using Iterative splitting solved by fourth order Runge-Kutta for without weight, w_1, w_2 are presented in Figure (13).

7 Conclusion

In the paper we presented the benefits of improving standard splitting methods with weighting schemes. Here the benefit of accelerating the wel-known Lie-Trotter and Strang splitting methods by the Zassenhaus product help to understand a general framework. By adding additional terms the starting conditions of the splitting methods improved and all weighted methods achieved more accurate results. The applications in parabolic equations shows the verification of the theoretical results. In future, we present a frame work for iterative and non-iterative operator-splitting

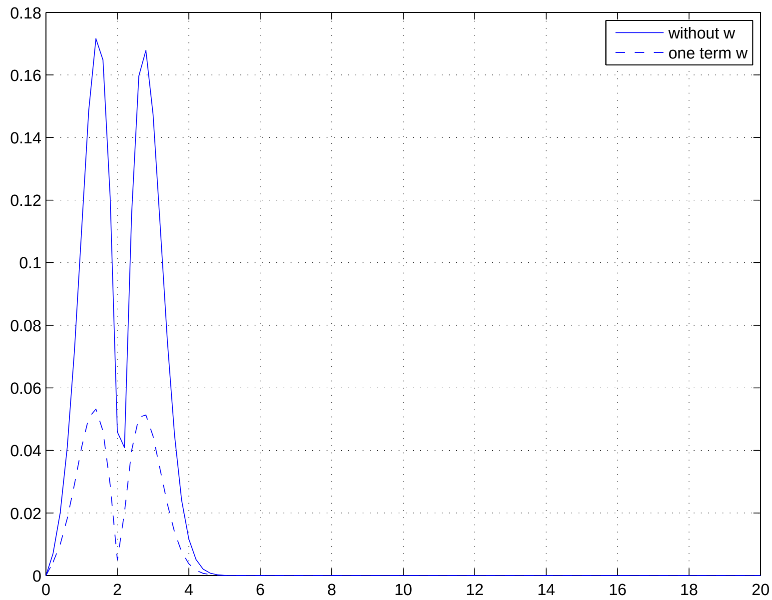

Figure 12: Comparison of errors for Parabolic Problem obtained by Iterative splitting and fourth order Runge-Kutta Method without weight , w_1 when h=0.2 and dt=1.

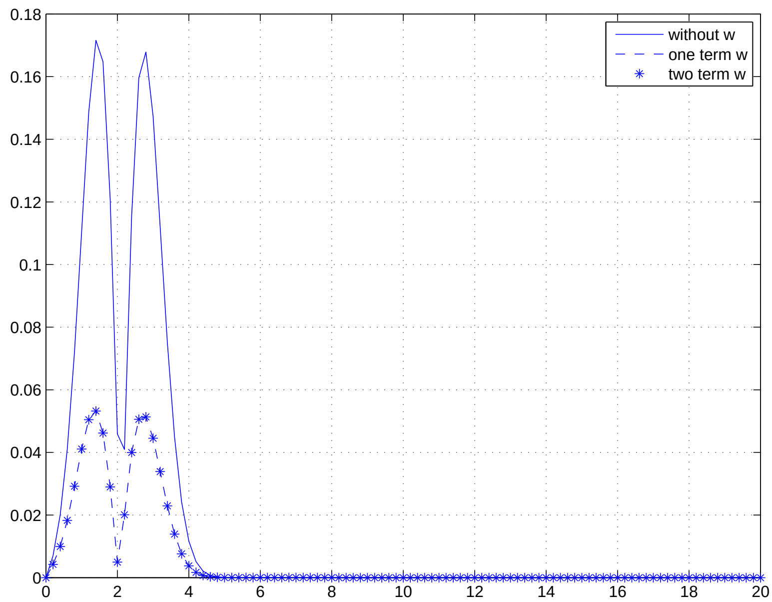

Figure 13: Comparison of errors for Parabolic Problem obtained by Iterative splitting and fourth order Runge-Kutta Method without weight , w_1, w_2 for h=0.2 and dt=1.

methods.

References

[1] R.J. Braun B.T Murray and J Soto Jr. *Adaptive Finite-difference computations of dendritic growth using a phase-field model.* Modelling Simul. Mater., vol. 5, 19957 365-380.

[2] 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.

[3] I. Farago, and Agnes Havasi. *On the convergence and local splitting error of different splitting schemes.* Eötvös Lorand University, Budapest, 2004.

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

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

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

[7] I. Farago, J. Geiser. *Iterative Operator-Splitting methods for Linear Problems.* Preprint No. 1043 of the Weierstrass Institute for Applied Analysis and Stochastics, Berlin, Germany, June 2005.

[8] P. Frolkovič and J. Geiser. *Numerical Simulation of Radionuclides Transport in Double Porosity Media with Sorption.* Proceedings of Algorithmy 2000, Conference of Scientific Computing, 28-36, 2000.

[9] 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.

[10] J. Geiser. *Gekoppelte Diskretisierungsverfahren für Systeme von Konvektions-Dispersions-Diffusions-Reaktionsgleichungen.* Doktor-Arbeit, Universität Heidelberg, 2003.

[11] J. Geiser. R^3T *: Radioactive-Retardation-Reaction-Transport-Program for the Simulation of radioactive waste disposals.* Proceedings: Computing, Communications and Control Technologies: CCCT 2004, The University of Texas at Austin and The International Institute of Informatics and Systemics (IIIS), to appear, 2004.

[12] J. Geiser. *Iterative Operator-Splitting methods for Parabolic Differential Equations : Convergence theory.* Humboldt-Preprint, to be submitted, February 2006.

[13] J. Geiser. *Decomposition Methods for Differential Equations: Theory and Applications.* Chapman Hall/CRC Press, Taylor and Francis Group, edited by Magoules and Lai, 2009.

[14] B. Gustafsson. *High Order Difference Methods for Time dependent PDE.* Springer Series in Computational Mathematics, Springer Verlag, Berlin, New York, Heidelberg, 38, 2007.

[15] W. Hackbusch. *Multi-Gird Methods and Applications.* Springer-Verlag, Berlin, Heidelberg, 1985.

[16] W. Hackbusch. *Iterative L”osung gro”ser schwachbesetzter Gleichungssysteme.* Teubner Studienb”ucher: Mathematik, B.G. Teubner Stuttgart, 1993.

[17] E. Hairer, C. Lubich and G. Wanner. *Geometric Numerical Integration.* Springer Series in Computational Mathematics, Springer Verlag, Berlin, New York, Heidelberg, 31, 2002.

[18] W. Hundsdorfer, L. Portero. A Note on Iterated Splitting Schemes. CWI Report MAS-E0404, Amsterdam, Netherlands, 2005.

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

[20] 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.

[21] R.J. LeVeque. *Finite Volume Methods for Hyperbolic Problems.* Cambridge Texts in Applied Mathematics ,

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

[23] J.A. Oteo. *The Baker-Campbell-Hausdorff formula and nested commutator identities.* Journal of mathematical physics, 32, 419, 1991.

[24] D. Scholz and M. Weyrauch. *A note on the Zassenhaus product formula.* Journal of mathematical physics, 47, 033505, 2006.

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

[26] S. Vandewalle. *Parallel Multigrid Waveform Relaxation for Parabolic Problems.* Teubner Skripten zur Numerik, B.G. Teubner Stuttgart, 1993.

[27] J.,G. Verwer and B. Sportisse. *A note on operator splitting in a stiff linear case.* MAS-R9830, ISSN 1386-3703, 1998.

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