ITERATIVE OPERATOR-SPLITTING METHODS FOR WAVE EQUATIONS WITH STABILITY RESULTS AND NUMERICAL EXAMPLES.

JÜRGEN GEISER * AND LENA NOACK †

Abstract. We are motivated by simulating a three-dimensional wave equation for an anisotropic material with stress-free boundary conditions. The applications are suited in the earthquake simulation that is based on seismic model problems. In this paper we discuss the efficiency of a higher-order time-discretization method, that is based on an iterative operator-splitting method. The main contribution is deriving the initial starting conditions for the iterative method; we propose different ideas and results for a pre-stepping method. The operator-splitting methods are well-know to solve such complex multi-dimensional and multi-physical problems. By decoupling the complex systems of differential equations into simpler equations, we save memory and computational resources. The iterative splitting method is discussed with its stability and consistency analysis. We verify our numerical methods with computational results based on our software tool $OPERA-SPLITT$. We present 2D and 3D wave equations with different higher-order splitting ideas. Finally we discuss the next works.

Keywords: Partial differential equations, operator-splitting methods, iterative methods, seismic sources and waves, stability and consistency analysis.
AMS subject classifications. 80A20, 80M25, 74S10, 76R50, 35J60, 35J65, 65M99, 65Z05, 65N12

1. Introduction. Based on the motivation of simulating wave equations, we contribute iterative splitting methods as accurate solver methods with respect to decouple complicated differential equations. The splitting methods deal with simpler differential equations, with respecting time and space scales of the physical operators, and can be designed to save memory and computational resources. While using multi-steps or more iterations, the additional amount of work for the operator-splitting methods can be reduced by optimizing the balance between time partitions and number of iteration steps. In our case, the classical splitting methods for hyperbolic equations are the alternating direction implicit (ADI) methods [6], [14], as well as the locally one-dimensional (LOD) methods [4], [13]. The methods are based on locally reduced equations, e.g. explicitly parts, and sweep implicitly over all equation parts. These methods are often not stable and accurate enough, while neglecting the physical coupling of each operator, and delicate for designing higher-order methods, see [6]. We contribute new iterative methods based on fixed-point iterations, which gain higher-order results and can be easily implemented, see [8]. The first iterative splitting methods were designed for parabolic differential equations, see [8] and [12]. We generalize the methods to be used also for hyperbolic differential equations, see also [10]. The delicate initialization process, which is important to obtain higher-order methods, is presented with pre-stepping and boot-strapping methods. We prove stability and consistency of the methods with the transformation on an abstract first-order Cauchy problem. The numerical experiments show the application to wave equations and the benefit of the splitting methods.

The paper is organized as follows. A mathematical model based on the wave equations

*Department of Mathematics, Humboldt Universität zu Berlin, Unter den Linden 6, D-10099 Berlin, Germany, E-mail: geiser@mathematik.hu-berlin.de
†Department of Mathematics, Humboldt Universität zu Berlin, Unter den Linden 6, D-10099 Berlin, Germany, E-mail: noackl@math.hu-berlin.de

is introduced in Section 2. The iterative operator-splitting methods are discussed in Section 3. The improvements of the starting solutions, i.e. initialization process, is done in Section 4. The underlying discretization is presented in Section 5. The consistency and stability analysis is considered in Section 6. We introduce the numerical results in Section 7. Finally, in Section 8, we discuss our future works in the area of splitting and decomposition methods.

2. Mathematical model. The motivation for the study presented below is coming from a computational simulation of earthquakes [3] and the examination of seismic waves [1] and [2].

We concentrate on the scalar wave equation, see [13], for which the mathematical equations are given by

$$\begin{aligned} \partial_{tt}\, c \;=\; & D_1(x_1,\ldots,x_d)\,\partial_{x_1x_1}\, c \;+\; D_2(x_1,\ldots,x_d)\,\partial_{x_2x_2}\, c \\ & +\ldots+\; D_d(x_1,\ldots,x_d)\,\partial_{x_dx_d}\, c,\ \text{in } \Omega\times[0,T], \qquad (2.1)\\ c(x_1,\ldots,x_d) = c_0(x_1,\ldots,x_d),\ & c_t(x_1,\ldots,x_d,0) = c_1(x_1,\ldots,x_d),\ \text{in } \Omega. \qquad (2.2) \end{aligned}$$

The unknown function $c = c(x_1,\ldots,x_d,t)$ is considered to be in $\Omega\times(0,T)\subset I\!R^d\times I\!R$ where the spatial dimension is given by d .
The function $\mathbf{D}(x_1,\ldots,x_d) = (D_1(x_1,\ldots,x_d), D_2(x_1,\ldots,x_d),\ldots,D_d(x_1,\ldots,x_d))^t \in I\!R^{d,+}$ describes the wave propagation in $x_1,\ldots,x_d$. The functions $x_0(x_1,\ldots,x_d)$ and $c_1(x_1,\ldots,x_d)$ are the initial conditions for the wave equation.
We deal with the following boundary conditions:

$$c(x_1,\ldots,x_d,t) = c_3,\ \text{on } \partial\Omega\times T:\ \text{Dirichlet boundary condition}, \qquad (2.3)$$

$$\frac{\partial c(x_1,\ldots,x_d,t)}{\partial n} = 0,\ \text{on } \partial\Omega\times T:\ \text{Neumann boundary condition}, \qquad (2.4)$$

$$\mathbf{D}\nabla c(x_1,\ldots,x_d,t) = c_{\text{out}},\ \text{on } \partial\Omega\times T:\ \text{outflow boundary condition}, \qquad (2.5)$$

where $c_3, c_{\text{out}} : \partial\Omega\times[0,T]\to I\!R$ are know and sufficient smooth functions.
For simpler model equations we can derive the following analytical solution:

$$u(x_1,\ldots,x_d,t) = \sin(\frac{1}{\sqrt{D_1}}\pi x_1)\cdot\ldots\cdot\sin(\frac{1}{\sqrt{D_d}}\pi x_d)\cdot\cos(\sqrt{d}\pi t), \qquad (2.6)$$

where $D_1, D_2,\ldots,D_d\in I\!R^+$ are given, and the initial functions $c_0(x_1,\ldots,x_d)$ and $c_1(x_1,\ldots,x_d)$ as well as the function for the Dirichlet boundary condition $c_3(x_1,\ldots,x_d,t)$ are given analytically.

3. Operator-splitting methods for wave equations. In the next subsections we describe the classical and iterative operator-splitting methods for wave equations.

3.1. Classical operator-splitting methods for wave equations. We describe the ADI and the LOD methods, which are both locally one-dimensional methods with explicit operators and implicit coupling steps.

ADI methods
Our classical method is based on the splitting method of [6] and [14].
The classical splitting methods ADI (alternating direction methods) are based on the idea of computing the different directions of the given operators. Each direction is computed independently by solving more basic equations. The result combines all the solutions of the elementary equations. Thus we obtain more effectivity by decoupling

the operators.
The classical splitting method for the wave equation starts from

$$\partial_{tt}c(t) = (A+B+C)c(t) + f(t),\ t \in (t^n, t^{n+1}),\ c(t^n) = c_0,\ c'(t^n) = c_1, \quad (3.1)$$

where the initial functions c_0 and c_1 are given. We could also apply for c_1 that $c'(t^n) = \frac{c(t^n)-c(t^{n-1})}{\Delta t} + \mathcal{O}(\Delta t) = c_1$. Consequently, we have $c(t^{n-1}) \approx c_0 - \Delta t c_1$. The right-hand side $f(t)$ is given as a force term.
We could decouple the equation into three simpler equations obtaining a method of second order.

$$\begin{aligned}
\tilde{\tilde{c}} - 2c(t^n) + c(t^{n-1}) &= \Delta t^2 A(\eta\tilde{\tilde{c}} + (1-2\eta)c(t^n) + \eta c(t^{n-1})) \quad (3.2)\\
&\quad + \Delta t^2 Bc(t^n) + \Delta t^2 Cc(t^n)\\
&\quad + \Delta t^2(\eta f(t^{n+1}) + (1-2\eta)f(t^n) + \eta f(t^{n-1})),\\
\tilde{c} - 2c(t^n) + c(t^{n-1}) &= \Delta t^2 A(\eta\tilde{\tilde{c}} + (1-2\eta)c(t^n) + \eta c(t^{n-1})) \quad (3.3)\\
&\quad + \Delta t^2 B(\eta\tilde{c} + (1-2\eta)c(t^n) + \eta c(t^{n-1})) + \Delta t^2 Cc(t^n)\\
&\quad + \Delta t^2(\eta f(t^{n+1}) + (1-2\eta)f(t^n) + \eta f(t^{n-1})),\\
c(t^{n+1}) - 2c(t^n) + c(t^{n-1}) &= \Delta t^2 A(\eta\tilde{\tilde{c}} + (1-2\eta)c(t^n) + \eta c(t^{n-1})) \quad (3.4)\\
&\quad + \Delta t^2 B(\eta\tilde{c} + (1-2\eta)c(t^n) + \eta c(t^{n-1}))\\
&\quad + \Delta t^2 C(\eta c(t^{n+1}) + (1-2\eta)c(t^n) + \eta c(t^{n-1}))\\
&\quad + \Delta t^2(\eta f(t^{n+1}) + (1-2\eta)f(t^n) + \eta f(t^{n-1})),
\end{aligned}$$

where the result is given as $c(t^{n+1})$ with the initial conditions $c(t^n) = c_0$ and $c(t^{n-1}) = c_0 - \Delta t c_1$ and $\eta \in (0, 0.5)$. A fully coupled method is given for $\eta = 0$ and for $0 < \eta \leq 0.5$ the decoupled method consists of a composition of explicit and implicit Euler methods.
The spatial discretization is given by $A = \frac{\partial^2}{\partial^2 x},\ B = \frac{\partial^2}{\partial^2 y}\ , C = \frac{\partial^2}{\partial^2 z}$,
where the approximated discretization is derived with the finite difference method as follows.

$$\begin{aligned}
Ac(x,y,z) &\approx \tfrac{c(x+\Delta x,y,z)-2c(x,y,z)+c(x-\Delta x,y,z)}{\Delta x^2},\\
Bc(x,y,z) &\approx \tfrac{c(x,y+\Delta y,z)-2c(x,y,z)+c(x,y-\Delta y,z)}{\Delta y^2},\\
Cc(x,y,z) &\approx \tfrac{c(x,y,z+\Delta z)-2c(x,y,z)+c(x,y,z-\Delta z)}{\Delta z^2}.
\end{aligned}$$

We have to compute the first equation (3.2) to get the result $\tilde{\tilde{c}}$, which is a further initial condition for the second equation (3.3), and after whose computation we obtain $\tilde{c}$. In the third equation (3.4) we have to put $\tilde{c}$ as a further initial condition and get the result $c(t^{n+1})$.

The underlying idea consists of the approximation of the pairwise operators:

$$\begin{aligned}
\Delta t^2 A\eta(\tilde{\tilde{c}} - 2c(t^n) + c(t^{n-1})) &\approx 0,\\
\Delta t^2 B\eta(\tilde{c} - 2c(t^n) + c(t^{n-1})) &\approx 0,
\end{aligned}$$

which we can raise to second order.

LOD method
In the following we introduce the LOD (locally one-dimensional) method as an improved splitting method while using pre-stepping techniques.

The method was discussed in [13] and is given by:

$$c^{n+1,0} - 2c^n + c^{n-1} = dt^2(A+B)c^n, \tag{3.5}$$
$$c^{n+1,1} - c^{n+1,0} = dt^2\eta A(c^{n+1} - 2c^n + c^{n-1}), \tag{3.6}$$
$$c^{n+1} - c^{n+1,1} = dt^2\eta B(c^{n+1} - 2c^n + c^{n-1}), \tag{3.7}$$

where $\eta \in (0.0, 0.5)$ and A, B are the spatial discretized operators.
If we eliminate the intermediate values in (3.5)- (3.7) we obtain

$$\begin{aligned} c^{n+1} - 2c^n + c^{n-1} = \Delta t^2(A+B)(\eta c^{n+1} - (1-2\eta)c^n + \eta c^{n-1} \\ +B_\eta(c^{n+1} - 2c^n + c^{n-1}), \end{aligned} \tag{3.8}$$

where $B_\eta = \eta^2\Delta t^2(AB)$ and thus $B_\eta(c^{n+1} - 2c^n + c^{n-1}) = \mathcal{O}(\Delta t^4)$.
Therefore we obtain a higher-order method.

REMARK 3.1. *For $\eta \in (0.25, 0.5)$ we have unconditionally stable methods and for getting a higher-order method we use $\eta = \frac{1}{12}$. Then for sufficiently small time steps we get a conditionally stable splitting method.*

3.2. Iterative operator-splitting methods for wave equations. In the following we present the iterative operator-splitting method as an extension to the traditional splitting methods for wave equations.

The idea is to repeat the splitting steps with the improved computed solutions. At least one fixed-point iteration has to be solved and we obtain higher-order results. We iterate for $i = 1, 4, 7, \ldots, 3m+1$:

$$\begin{aligned} c_i - 2c(t^n) + c(t^{n-1}) = \Delta t^2 A(\eta c_i + (1-2\eta)c(t^n) + \eta c(t^{n-1})) \\ +\Delta t^2 B(\eta c_{i-1} + (1-2\eta)c(t^n) + \eta c(t^{n-1})) \\ +\Delta t^2 C(\eta c_{i-2} + (1-2\eta)c(t^n) + \eta c(t^{n-1})) \\ +\Delta t^2(\eta f(t^{n+1}) + (1-2\eta)f(t^n) + \eta f(t^{n-1}))\,, \end{aligned} \tag{3.9}$$

$$\begin{aligned} c_{i+1} - 2c(t^n) + c(t^{n-1}) = \Delta t^2 A(\eta c_i + (1-2\eta)c(t^n) + \eta c(t^{n-1})) \\ +\Delta t^2 B(\eta c_{i+1} + (1-2\eta)c(t^n) + \eta c(t^{n-1})) \\ +\Delta t^2 C(\eta c_{i-2} + (1-2\eta)c(t^n) + \eta c(t^{n-1})) \\ +\Delta t^2(\eta f(t^{n+1}) + (1-2\eta)f(t^n) + \eta f(t^{n-1}))\,, \end{aligned} \tag{3.10}$$

$$\begin{aligned} c_{i+2} - 2c(t^n) + c(t^{n-1}) = \Delta t^2 A(\eta c_i + (1-2\eta)c(t^n) + \eta c(t^{n-1})) \\ +\Delta t^2 B(\eta c_{i+1} + (1-2\eta)c(t^n) + \eta c(t^{n-1})) \\ +\Delta t^2 C(\eta c_{i+2} + (1-2\eta)c(t^n) + \eta c(t^{n-1})) \\ +\Delta t^2(\eta f(t^{n+1}) + (1-2\eta)f(t^n) + \eta f(t^{n-1}))\,, \end{aligned} \tag{3.11}$$

where the result is given as $c(t^{n+1})$ with the initial conditions $c(t^n) = c_0$ and $c(t^{n-1}) = c_0 - \Delta c_1$ with $\eta \in (0, 0.5)$. We have the fully coupled method for $\eta = 0$ and the decoupled method for $0 < \eta \leq 1$, which is a mixing of explicit and implicit Euler methods. For the initial conditions of the starting vector we use $c_1(t^{n+1}) = 0$ and $c_0(t^{n+1}) = 0$. The stop criterion is given as

$$|c_{\tilde{i}+3} - c_{\tilde{i}}| \leq \epsilon,$$

where $\tilde{i} \in 1, 4, 7, \ldots, 3m+1$ $\epsilon \in I\!R^+$.
Therefore the solution is given as $c(t^{n+1}) = c_{\tilde{i}+3}$.

REMARK 3.2. *The initializing process can be done with* $c_1(t^{n+1}) = 0$ *and* $c_0(t^{n+1}) = 0$*, but one can also improve the accuracy by dealing with approximated solutions to the starting vector. Further the amount of iteration steps has to be discussed in relation to the time step, such that a smaller time step can save iterations steps and vice versa.*

4. Strategies for improved approximations of the starting vector $c_0(t^{n+1})$**.** For the stability of the method, it is important to start the iterative method with a good initial value $c^{0,n+1} = c^0(t^{n+1})$, e.g. $c^{0,n+1} - c_{\text{sp}}^{0,n+1} = \mathcal{O}(\tau^{3m+1})$. For the strategies we contribute four cases, using

I.1) zero vector, $c^{0,n+1} \equiv 0$,
I.2) last time step, $c^{0,n+1} = c^n$,
I.3) average growth, for example $c^{0,n+1} = c^n + \frac{t-t^n}{\Delta t} \cdot (c^n - c^{n-1})$,
and $t \in (t^n, t^{n+1}]$, or
I.4) prestepping method, for example $c^{0,n+1} - 2c^n + c^{n-1} = \tau^2(D_1 \frac{\partial^2 c^n}{\partial x^2} + D^2 \frac{\partial^2 c^n}{\partial y^2})$.

For the last case, we have an improved initialization process.

LEMMA 4.1. *We have the wave equation given as in (2.1). For sufficient small time and space steps, we can assume the following,*

$$c^{n+1} = c^n + \frac{c^n - c^{n-1}}{\tau}, \tag{4.1}$$

where τ *is the equidistant time step and we assume* $\partial_t c^{n+1} = \partial_t c^{n-1}$*, (reversibility of the hyperbolic equations).*

Proof. Using the Taylor expansion for c^{n+1} we have

$$c(t) = c^n + (t - t^n)\partial_t c(t) + \mathcal{O}(\tau^2), \tag{4.2}$$

$$c(t) \approx c^n + (t - t^n)\frac{c(t) - c(t^n)}{\tau} + \mathcal{O}(\tau^2), \tag{4.3}$$

where $t \in [t^n, t^{n+1}]$.
We follow $t = t^{n+1}$ and use the reversibility of the hyperbolic equations.

$$c(t^{n+1}) \approx c^n + (t - t^n)\frac{c(t^n) - c(t^{n-1})}{\tau} + \mathcal{O}(\tau^2), \tag{4.4}$$

for all $t \in (t^n, t^{n+1}]$.
□

REMARK 4.1. *In the 2D and 3D test examples there were no significant differences between the results using one of the last three initial values. For the first possibility, one more iteration step is needed until the same results as with the third and fourth initial value are achieved, for the second one this depends on the PDE. Thus for 2D and 3D problems the third initial value (by average growth) is the best choose. This might be different for higher-dimensional PDEs, where more iterations are needed for the fixed-point iteration. There the last initial value should yield the best results and save some iteration steps.*

5. Discretization methods. For the time- and space-discretization method we first consider finite difference schemes for the discretization.

For a classical wave equation this is the well-known discretization in time and space. Based on this discretization the time is discretized as:

$$c_{tt,i} = \frac{c_i^{n+1} - 2c_i^n + c_i^{n-1}}{\Delta t^2}, \tag{5.1}$$

$$c(0) = c_0, c_t(0) = c_1, \tag{5.2}$$

where i denotes the space point x_i and $\Delta t = t^{n+1} - t^n$ is the time step. The space is discretized with

$$c_{xx,n} = \frac{c_{i+1}^n - 2c_i^n + c_{i-1}^n}{\Delta x^2}, \tag{5.3}$$

$$c(0) = c_0, c_t(0) = c_1, \tag{5.4}$$

where n denotes the time point t_n and $\Delta x = x_{i+1} - x_i$ is the grid width.
The underlying equation,

$$c_{tt} = D_1 c_{xx} + D_2 c_{yy} \text{ in } \Omega, \tag{5.5}$$

$$u(x, y, 0) = c_0(x, y), c_t(x, y, 0) = c_1(x, y), \tag{5.6}$$

$$u(x, y, t) = c_2 \text{ on } \partial\Omega, \tag{5.7}$$

is discretized with the unconditional stable implicit η-method, see [6].

$$\begin{aligned} &\frac{c_{i,j}^{n+1} - 2c_{i,j}^n + c_{i,j}^{n-1}}{\Delta t^2} \\ &= \frac{D_1}{\Delta x^2}\left(\eta\left(c_{i+1,j}^{n+1} - 2c_{i,j}^{n+1} + c_{i-1,j}^{n+1}\right)\right. \\ &+(1-2\eta)\left(c_{i+1,j}^n - 2c_{i,j}^n + c_{i-1,j}^n\right) + \eta\left(c_{i+1,j}^{n-1} - 2c_{i,j}^{n-1} + c_{i-1,j}^{n-1}\right)\Big) \\ &+\frac{D_2}{\Delta y^2}\left(\eta\left(c_{i,j+1}^n - 2c_{i,j}^n + c_{i,j-1}^n\right)\right. \\ &\left.+(1-2\eta)\left(c_{i,j+1}^n - 2c_{i,j}^n + c_{i,j-1}^n\right) + \eta\left(c_{i,j+1}^{n-1} - 2c_{i,j}^{n-1} + c_{i,j-1}^{n-1}\right)\right), \end{aligned} \tag{5.8}$$

where Δx and Δy are the grid width in x and y and $0 \leq \eta \leq 0.5$. The initial conditions are given as $c(x, y, t^n) = c_0(x, y)$ and $c(x, y, t^{n-1}) = c_0(x, y) - \Delta t c_1(x, y)$.

These discretization schemes are applied to our operator-splitting schemes.

We discretize on a finite difference grid, where k denotes the time step, and h_x, h_y and h_z are the grid sizes in the different spatial directions of the grid. The time nk is denoted by t^n, and i, j, l is the grid point with spatial coordinates ih_x, jh_y, kh_z. Let $\mathbf{u}^n$ be a the grid function on time level n, and $u_{i,j,l}^n$ the specific value of $\mathbf{u}^n$ at the point i, j, l. The value of the grid function during the iteration is denoted by an extra super script as $u_{i,j,l}^{n,m}$.

In the next section we apply our theoretical results to first test examples and in the next step to real-life models.

6. Consistency and stability analysis. For the stability and consistency we can rewrite the equations (3.9)-(3.10) in the continuous form in the operator equation as

$$\partial_{tt} C_i = \mathcal{A} C_i + \mathcal{F}_i, \tag{6.1}$$

where $C_i = (c_i, c_{i+1})^t$ and the operators are given as

$$\mathcal{A} = \begin{bmatrix} A & 0 \\ A & B \end{bmatrix}, \ \mathcal{F}_i = \begin{bmatrix} Bc_{i-1} \\ 0 \end{bmatrix}. \tag{6.2}$$

REMARK 6.1. *The stability and consistency results can be done as for the parabolic case. The operator equation with second-order time derivatives can be reformulated as a system of first-order time derivatives.*

6.1. Consistency for the iterative operator-splitting method for wave equations. In the following we analyze the consistency and the order of the local splitting error for the linear bounded operators $A, B : \mathbf{X} \to \mathbf{X}$, where $\mathbf{X}$ is a Banach space, see [17].

We assume our Cauchy problem for two linear operators with the second-order time derivative,

$$c_{tt} - Ac - Bc = 0,\ t \in (0,T), \tag{6.3}$$
$$c(0) = c_0\ ,\ c_t = c_1, \tag{6.4}$$

where c_0 and c_1 are given.
We rewrite the probem to a system of first-order time derivatives as

$$\partial_t c_1 - c_2 = 0 \text{ in } (0,T), \tag{6.5}$$
$$\partial_t c_2 - Ac_1 - Bc_1 = 0 \text{ in } (0,T), \tag{6.6}$$
$$\text{with } c_1(0) = c_1, c_2(0) = c_2,$$

where $c_1 = c$ and $c_2 = c_t$.

The iterative operator-splitting method (3.9)–(3.11), with respect to two operators and in the semi-discretized form, can be rewritten to a system of splitting methods.

The method is given as

$$\partial_t c_{1,i} = c_{2,i}, \tag{6.7}$$
$$\partial_t c_{2,i} = Ac_{1,i} + Bc_{1,i-1}, \tag{6.8}$$
$$\text{with } c_{1,i}(t^n) = c_1(t^n),$$
$$\partial_t c_{1,i+1} = c_{2,i+1}, \tag{6.9}$$
$$\partial_t c_{2,i+1} = Ac_{1,i} + Bc_{1,i+1}, \tag{6.10}$$
$$\text{with } c_{2,i+1}(t^n) = c_2(t^n).$$

We start with $i = 1, 3, 5, \ldots, 2m+1$.
We can obtain the consistency with the underlying fundamental solution of the equation system.

THEOREM 6.1. *Let $A, B \in \mathcal{L}(\mathcal{X})$ be given linear bounded operators. Then the abstract Cauchy problem (6.3)–(6.4) has a unique solution and the iterative splitting method (6.7)–(6.10) for $i = 1, 3, \ldots, 2m+1$ is consistent with the order of the consistency $\mathcal{O}(\tau_n^{2m})$.*
The error estimates are given as:

$$\|e_i\| = K\|B\|\tau_n\|e_{i-1}\| + \mathcal{O}(\tau_n^2), \tag{6.11}$$

where $e_i = \max\{|e_{1,i}|, |e_{i,2}|\}$.

Proof. In the following we derive the underlying consistency of the operator-splitting method.

Let us consider the iteration (3.9)–(3.10) on the subinterval $[t^n, t^{n+1}]$. For the local error function $e_i(t) = c(t) - c_i(t)$ we have the relations

$$\begin{aligned} &\partial_t e_{1,i}(t) = e_{2,i}(t), \quad t \in (t^n, t^{n+1}], \\ &\partial_t e_{2,i}(t) = Ae_{1,i}(t) + Be_{1,i-1}(t), \quad t \in (t^n, t^{n+1}], \\ &\partial_t e_{1,i+1}(t) = e_{2,i+1}(t), \quad t \in (t^n, t^{n+1}], \\ &\partial_t e_{2,i+1}(t) = Ae_{1,i}(t) + Be_{1,i+1}(t), \quad t \in (t^n, t^{n+1}], \end{aligned} \tag{6.12}$$

for $m = 1, 3, 5, \ldots$, with $e_1(0) = 0$ and $e_0(t) = c(t)$. We use the notations $\mathbf{X}^4$ for the product space $\mathbf{X} \times \mathbf{X} \times \mathbf{X} \times \mathbf{X}$ enabled with the norm $||(u_1, u_2, u_3, u_4)|| = \max\{||u_1||, ||u_2||, ||u_3||, ||u_4||\}(u_1, u_2, u_3, u_4 \in \mathbf{X})$.

The elements $\mathcal{E}_i(t)$, $\mathcal{F}_i(t) \in \mathbf{X}^4$ and the linear operator $\mathcal{A} : \mathbf{X}^4 \to \mathbf{X}^4$ are defined as follows:

$$\mathcal{E}_i(t) = \begin{bmatrix} e_{1,i}(t) \\ e_{2,i}(t) \\ e_{1,i+1}(t) \\ e_{2,i+1}(t) \end{bmatrix}, \mathcal{F}_i(t) = \begin{bmatrix} 0 \\ Be_{1,i-1}(t) \\ 0 \\ 0 \end{bmatrix}, \mathcal{A} = \begin{bmatrix} 0 & I & 0 & 0 \\ A & 0 & 0 & 0 \\ 0 & I & 0 & I \\ A & 0 & B & 0 \end{bmatrix}. \tag{6.13}$$

Then, using the notations (6.13), the relations (6.12) can be written in the form:

$$\begin{aligned} &\partial_{tt}\mathcal{E}_i(t) = \mathcal{A}\mathcal{E}_i(t) + \mathcal{F}_i(t), \quad t \in (t^n, t^{n+1}], \\ &\mathcal{E}_i(t^n) = 0. \end{aligned} \tag{6.14}$$

Due to our assumptions, $\mathcal{A}$ is a generator of the one-parameter C_0-semigroup $(\exp \mathcal{A}t)_{t\geq 0}$, hence using the variations of constants formula, the solution for the abstract Cauchy problem (6.14) with homogeneous initial conditions can be written as:

$$\mathcal{E}_i(t) = c_0 \int_{t^n}^{t} \exp(\mathcal{A}(t-s))\mathcal{F}_i(s)ds, \tag{6.15}$$

with $t \in [t^n, t^{n+1}]$.

Hence, using the denotation

$$\|\mathcal{E}_i\|_\infty = \sup_{t\in[t^n,t^{n+1}]} \|\mathcal{E}_i(t)\| \ , \tag{6.16}$$

we have

$$\begin{aligned} \|\mathcal{E}_i\|(t) &\leq \|\mathcal{F}_i\|_\infty \int_{t^n}^{t} \|\exp(\mathcal{A}(t-s))\|ds \\ &= \|B\|\|e_{1,i-1}\| \int_{t^n}^{t} \|\exp(\mathcal{A}(t-s))\|ds, \quad t \in [t^n, t^{n+1}]. \end{aligned} \tag{6.17}$$

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

$$\|\exp(\mathcal{A}t)\| \leq K \exp(\omega t), \quad t \geq 0, \tag{6.18}$$

holds with some numbers $K \geq 0$ and $\omega \in I\!R$, cf. [5].

The estimations (6.17) and (6.18) result in

$$\|\mathcal{E}_i\|_\infty = K\|B\|\tau_n\|e_{i-1}\| + \mathcal{O}(\tau_n^2). \tag{6.19}$$

where $||e_{i-1}|| = \max\{||e_{1,i-1}||, ||e_{2,i-1}||\}$.

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

$$\|e_i\| = K\|B\|\tau_n\|e_{i-1}\| + \mathcal{O}(\tau_n^2), \tag{6.20}$$

and hence

$$\|e_{i+1}\| = K_1\tau_n^2\|e_{i-1}\| + \mathcal{O}(\tau_n^3), \tag{6.21}$$

which proves our statement.

□

REMARK 6.2. The proof is aligned to scalar temporal first-order derivatives, see [8]. The generalization can also be done for higher-order hyperbolic equations, that are reformulated in first-order systems.

6.2. Stability for the iterative operator-splitting method for wave equations. The following stability theorem is given for the wave equation performed with the iterative splitting method, see (6.7) - (6.10).

The convergence is examined in a general Banach space setting and we can prove the following stability theorem.

THEOREM 6.2. *Let us consider the system of linear differential equations used for the spatial discretized wave equation*

$$\frac{dc_1}{dt} = c_2, \tag{6.22}$$

$$\frac{dc_2}{dt} = Ac_1 + Bc_1, \tag{6.23}$$

$$\text{with } c_1(t^n) = c(t^n),\ c_2(t^n) = \frac{dc(t^n)}{dt},$$

where the operators $A, B : \mathbf{X} \to \mathbf{X}$ are linear and densely defined in the real Banach space $\mathbf{X}$, see [17]. *We can define a norm on the product space $\mathbf{X}\times\mathbf{X}$ with $||(u,v)^T|| = \max\{||u||, ||v||\}$.*

We rewrite the equations (6.22)–(6.23) *and obtain*

$$\begin{aligned} \frac{d\tilde{c}(t)}{dt} &= \tilde{A}\tilde{c}(t) + \tilde{B}\tilde{c}(t), \\ \tilde{c}(t^n) &= \tilde{c}^n, \end{aligned} \tag{6.24}$$

where $\tilde{c}^n = (c(t^n), \frac{dc(t^n)}{dt})^T$, $\tilde{A} = \begin{pmatrix} 0 & 1/2I \\ A & 0 \end{pmatrix}$ and $\tilde{B} = \begin{pmatrix} 0 & 1/2I \\ B & 0 \end{pmatrix}$.

We assume that $\tilde{A},\ \tilde{B} : \mathbf{X}^2 \to \mathbf{X}^2$ are given linear bounded operators that generate the C_0-semigroup, and $\tilde{c}, \tilde{c}^n \in \mathbf{X}^2$ are given elements.

We also assume $\lambda_{\tilde{A}}$ is an eigenvalue of $\tilde{A}$ and $\lambda_{\tilde{B}}$ is an eigenvalue of $\tilde{B}$.

Then the linear iterative operator-splitting method for wave equations (6.7) - (6.10) *is stable with the following result:*

$$\|\tilde{c}_{i+1}(t^{n+1})\| \le \tilde{K}\sum_{j=0}^{i+1}\|\tilde{c}^n\|\tau^j\lambda_{\max}^j, \tag{6.25}$$

where $\tilde{K} > 0$ *is a constant and* $\tilde{c}^n = (c(t^n), \frac{dc(t^n)}{dt})^T$ *is the initial condition,* $\tau = (t^{n+1} - t^n)$ *is the time step and* $\lambda_{\max}$ *is the maximal eigenvalue of the linear and bounded operators* $\tilde{A}$ *and* $\tilde{B}$.

Proof. The proof can be found in [10].

□

7. Numerical examples. The numerical examples can support the theoretical results of the iterative operator-splitting methods.

7.1. Test example 1: 2D wave equation. We apply our methods on the following test example:

$$\begin{aligned}
&\partial_{tt} c = D_1 \partial_{x_1 x_1} c + \ldots + D_d \partial_{x_d x_d} c \text{ on } \Omega, \\
&u(x,0) = \sin(\frac{1}{\sqrt{D_1}} \pi x_1) \cdots \sin(\frac{1}{\sqrt{D_d}} \pi x_d), \\
&\partial_t u(x,0) = 0, \\
&u(x,t) = u_{\text{analy}}(x,t), \text{ or} \\
&\frac{\partial u(x,t)}{\partial n} = \frac{\partial u_{\text{analy}}(x,t)}{\partial n} \text{ on } \partial\Omega \times]0,T[.
\end{aligned} \tag{7.1}$$

The analytical solution is given as

$$u_{\text{analy}}(x,t) = \sin(\frac{1}{\sqrt{D_1}} \pi x_1) \cdots \sin(\frac{1}{\sqrt{D_d}} \pi x_d) \cos(\sqrt{d} \pi t). \tag{7.2}$$

We implement the methods by using the program *MATLAB*. The error will be calculated by

$$err_{L_1} := \sum_{i_1 \ldots i_d = 1}^{m} \Delta x_1 \cdot \ldots \cdot \Delta x_d |c(x_1, \ldots, x_d, t^n) - c_{\text{analy}}(x_1, \ldots, x_d, t^n)|.$$

We now see the simulation of a two-dimensional wave in the time interval $[0, 1/\sqrt{(2)}]$ with using the iterative operator-splitting method with two iterations per step, $\eta = 0.1$, $\Delta x = \frac{1}{16}, \Delta y = \frac{1}{32}, \Delta t = \frac{1}{16}$ and the initial value for the iterations $c^{0,n+1} = c^n + \frac{1}{\Delta t} \cdot (c^n - c^{n-1})$.

Figure 7.1 shows some typical error functions for the wave equation for the Dirichlet and Neumann boundary condition.

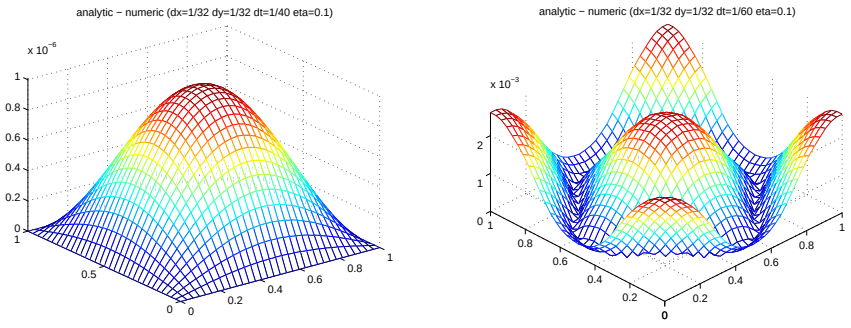

FIG. 7.1. *Left: Dirichlet boundary, right: Neumann boundary.*

η	err_{L_1}	err_{L_1}	err_{L_1}	err_{L_1}	err_{L_1}
tsteps	50	65	80	100	1000
0.0	1.3378e+024	2.9930e+013	3.1712e-006	7.0626e-006	1.7881e-005
0.01	4.3434e+019	8.6263e+010	4.2870e-006	8.0847e-006	1.7897e-005
0.1	0.0037	2.4075e-005	2.1859e-005	2.0379e-005	1.8040e-005
0.5	0.0011	4.9551e-004	2.6220e-004	1.4201e-004	1.8682e-005

TABLE 7.1

Iterative OSM; Dirichlet boundary, $D_1 = 1$, $D_2 = 1$, $\Delta x = \frac{1}{16}$, $\Delta y = \frac{1}{32}$, *2 iterations per step, initial condition* ***I.3)****, and* $\Omega = [0,1]^2 \times [0, 3 \cdot (1/\sqrt{2})]$.

Table 7.1 shows the result of the 2D non-stiff problem computed with the iterative operator-splitting method.

Table 7.2 shows the result of the 2D non-stiff problem computed with the classical operator-splitting method (ADI method).

η	err_{L_1}	err_{L_1}	err_{L_1}	err_{L_1}	err_{L_1}
tsteps	50	65	80	100	1000
0.0	1.6495e+024	6.2708e+013	3.2525e-006	7.2068e-006	1.7917e-005
0.01	5.0807e+021	2.4727e+011	4.3970e-006	8.2497e-006	1.7933e-005
0.1	0.0038	2.4863e-005	2.2429e-005	2.0799e-005	1.8076e-005
0.5	0.0012	5.1386e-004	2.6976e-004	1.4515e-004	1.8719e-005

TABLE 7.2

Classical operator-splitting method.

Figure 7.2 shows the dependence of the error, i.e. of the convergence, on the η-value.

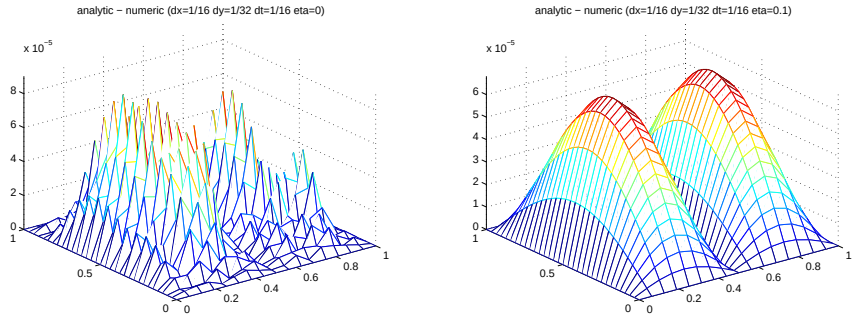

FIG. 7.2. *Left: η=0.0, right: η=0.1 .*

Table 7.3 shows the result of the 2D stiff problem computed with the iterative operator-splitting method.

In the following we have the 2D stiff problem done with the classical operator-splitting method (ADI), see Table 7.4.

Figure 7.3 shows the numerical and analytical results for the 2D wave equation.

REMARK 7.1. *The results show the accuracy of the iterative methods, which are as good as the classical methods. With respect to more iteration steps we could achieve*

η	err_{L_1}	err_{L_1}	err_{L_1}	err_{L_1}
tsteps	3	4	5	6
0	2.2799	0.3089	0.4845	0.5424
0.2000	0.5299	0.6056	0.6284	0.6334
0.4000	0.1610	0.3716	0.5047	0.5743
0.5000	0.0730	0.2671	0.4259	0.5237

TABLE 7.3

Iterative OSM; Dirichlet boundary, $D_1 = 1$, $D_2 = \frac{1}{1000}$, $\Delta x = \frac{1}{16}$, $\Delta y = \frac{1}{32}$, *2 iterations per step, initial condition* ***I.4)****, and* $\Omega = [0,1]^2 \times [0, 3 \cdot (1/\sqrt{2})]$.

η	err_{L_1}	err_{L_1}	err_{L_1}	err_{L_1}
tsteps	3	4	5	6
0	16.9876	0.4420	0.5144	0.5955
0.2000	0.5801	0.6469	0.6903	0.6474
0.4000	0.1697	0.3826	0.5886	0.6486
0.5000	0.0839	0.2950	0.4806	0.6142

TABLE 7.4

Classical operator-splitting method.

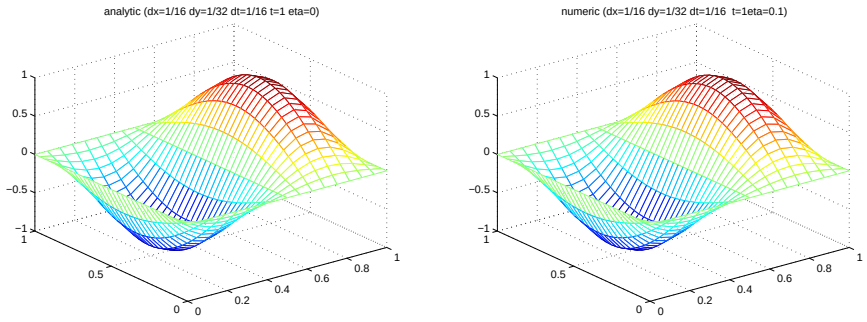

FIG. 7.3. *Left: analytical solution, right: numerical solution.*

more accurate solutions. The best initialization is given with the average method.

7.2. Test Example 2: 3D wave equation. In the following, we apply our iterative algorithm on the 3d wave equation.

$$\frac{\partial^2 c}{\partial t^2} = D_1 \frac{\partial^2 c}{\partial x^2} + D_2 \frac{\partial^2 c}{\partial y^2} + D_3 \frac{\partial^2 c}{\partial z^2}. \tag{7.3}$$

The analytical solution is given by (2.6) and for the error function we apply the L_1-norm.

Choosing the optimal starting condition is important for the 3D model. For the initial conditions (c_{i-1}, c_{i-2}), we apply the conditions (case 1 - case 4), see Section 4.

A first result for optimal starting conditions with non-stiff operators is presented in Table 7.5.

In comparison to the stiff case, the optimal starting condition is presented in Table 7.6. Here, we saw the higher amount of iteration steps and a finer time partition to

tsteps	err_{L_1} $\eta = 0.01$	η	err_{L_1} tsteps= 45
30	7.1499e+006	0	3.4227e-005
40	2.0628	0.009	4.9271e-007
44	4.4076e-006	0.01	8.1189e-009
45	8.1189e-009	0.011	1.3792e-007
46	2.8080e-006	0.05	5.0199e-004
50	6.2946e-005	0.1	0.0025
60	3.3983e-004	0.5	0.0541

TABLE 7.5

Iterative operator-splitting method; $D_1 = D_2 = D_3 = 1$, $\Delta x = \Delta y = \Delta z = \frac{1}{8}$, *2 iterations per step, initial condition* ***I.3)****, and* $\Omega = [0,1]^3 \times [0\ ,\ 6 \cdot (1/\sqrt{3})]$.

obtain acceptable results. If we choose $\eta = 1/12$, we obtain the best result, because of the fourth-order time-discretization.

η	err_{L_1}	err_{L_1}	err_{L_1}	err_{L_1}
tsteps	5	10	30	100
0.0	1394.2	67.7059	0.2921	0.3493
0.01	766.0236	30.0415	0.2980	0.3498
0.1	11.5627	0.3325	0.3486	0.3541
0.2	0.1143	0.4863	0.3963	0.3588
0.3	0.4160	0.2573	0.4340	0.3633
0.4	0.2633	0.1430	0.4622	0.3677
0.5	0.2225	0.1738	0.4824	0.3719

TABLE 7.6

Iterative operator-splitting method; $D_1 = \frac{1}{10}$, $D_2 = \frac{1}{10}$ *and* $D_3 = \frac{1}{10}$, $\Delta x = \Delta y = \Delta z = \frac{1}{8}$, *2 iterations per step, initial condition* ***I.3)****, and* $\Omega = [0,1]^3 \times [0\ ,\ 6 \cdot (1/\sqrt{3})]$.

If we choose a better initialization process, for example the pre-stepping method, we obtain also for the stiff case much more better results. In Table 7.7 the balancing between time and number of iterations is given, good results were already achieved for 9 time steps while using 3 iterations per step. If we choose $\eta = 1/12$, we also obtain higher-order results.

η	err_{L_1}	err_{L_1}	err_{L_1}	err_{L_1}	err_{L_1}
tsteps	8	9	10	11	12
0	169.4361	8.4256	0.5055	0.2807	0.3530
0.2000	0.0875	0.1315	0.1750	0.2232	0.2631
0.3000	0.1151	0.0431	0.0473	0.0745	0.1084
0.4000	0.3501	0.2055	0.0988	0.0438	0.0454
0.4500	0.4308	0.3002	0.1719	0.0844	0.0402
0.5000	0.4758	0.3792	0.2510	0.1402	0.0704

TABLE 7.7

Iterative operator-splitting method; $D_1 = 1$, $D_2 = \frac{1}{10}$ *and* $D_3 = \frac{1}{100}$, $\Delta x = \Delta y = \Delta z = \frac{1}{8}$, *3 iterations per step, initial condition* ***I.4)****, and* $\Omega = [0,1]^3 \times [0\ ,\ 6 \cdot (1/\sqrt{3})]$.

In Table 7.8, we compare all 4 different initialization cases. The most effective

results are given in the cases 3 and 4. Therefore higher accuracy results are related with improved approximated starting solutions. The pre-stepping method can be proposed for more iteration steps, where the averaging method also fits with less iterations and more time partitions.

iterations per step	**I.1)**	**I.2)**	**I.3)**	**I.4)**
1	5.5271e-006	5.6045e-006	7.5923e-005	8.5346e-008
2	2.0669e-008	2.0650e-008	7.3888e-009	2.2775e-008
3	2.2794e-008	2.2793e-008	2.2810e-008	2.2792e-008
4	2.2792e-008	2.2792e-008	2.2792e-008	2.2792e-008

TABLE 7.8

Comparison of the initial values for the iterative operator-splitting method; $D_1 = 1$, $D_2 = 1$ *and* $D_3 = 1$, $\Delta x = \Delta y = \Delta z = \frac{1}{8}$, *tsteps=45, η=0.01, and* $\Omega = [0,1]^3 \times [0\ ,\ 6 \cdot (1/\sqrt{3})]$.

REMARK 7.2. *The results show the delicate choice of the initialization methods, and also the balancing of the iterations and time partitions. Higher accuracy is achieved with the pre-stepping method, that can calculate the solution approximatively and explicitly. With respect to more iteration steps we can achieve more accurate solutions, but we have also taken into account the additional computational time. So a balancing might be usefull, where very less iteration steps, e.g. 2 or 3, and fine time partitions are used to obtain the same accurate results.*

8. Conclusions and discussions. We present an iterative operator-splitting method to solve hyperbolic partial differential equations. The smoothing process for more accurate solutions is done with more iterations. The theoretical results stated the higher-order method to be dependent of the iterations. Consistency and stability is fulfilled for the method. For applications to numerical examples the delicate control of the initialization process can be done with a pre-stepping method. Further the balance of the iteration steps and steps is discussed. The simple implementation and the stable results are advantages of the iterative methods. The additional splitting error, that can be controlled by the iteration steps, has also to be taken into account. All in all the iterative splitting method is an attractive method and has as accurate results as the classical splitting methods. In future an automatization between the iterations and time partitions has to be done as well as the error control of the splitting method.

REFERENCES

[1] Z.S. Alterman and A. Rotenberg. *Seismic waves in a quarter plane.* Bulletin of the Seismological Society of America, 59, 1969.
[2] A. Ben-Menahem and S. Jit Singh. *Seismic Waves and Sources.* Dover Publications, 2000.
[3] St.M. Day et al. *Test of 3D elastodynamic codes: Final report for lifelines project 1A01.* Technical report, Pacific Earthquake Engineering Center, 2001.
[4] J. Douglas, Jr. and S. Kim. *Improved accuracy for locally one-dimensional methods for parabolic equations.* Mathematical Models and Methods in Applied Sciences, 11, 1563–1579, 2001.
[5] K.-J. Engel, R. Nagel, *One-Parameter Semigroups for Linear Evolution Equations.* Springer, New York, 2000.
[6] G. Fairweather and A.R. Mitchell. *A high accuracy alternating direction method for the wave equations.* J. Industr. Math. Appl., Vol.1, 309–316, 1965.
[7] I. Farago. *Splitting methods for abstract Cauchy problems.* Lect. Notes Comp.Sci. 3401, Springer Verlag, Berlin, 2005, pp. 35-45

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

[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. *Decomposition Methods for Partial Differential Equations Theory and Applications in Multiphysics Problems.* Habilitation Thesis, Humboldt University of Berlin, Germany, submitted, 2007.

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

[12] J. Kanney, C. Miller and C. Kelley. *Convergence of iterative split-operator approaches for approximating nonlinear reactive transport problems.* Advances in Water Resources, 26:247–261, 2003.

[13] S. Kim and H. Lim *High-order schemes for acoustic waveform simulation.* Appl. Numer. Math. (accepted, 2006)

[14] M. Lees. *Alternating direction methods for hyperbolic differential equations.* J. Soc. Industr. Appl. Math., Vol. 10, No. 4, 610–616, 1962.

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

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

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