

## Portland State University PDXScholar

---

Portland Institute for Computational Science  
Publications

Portland Institute for Computational Science

---

6-2018

# Space-Time Discretizations Using Constrained First-Order System Least Squares (CFOSLS)

Kirill Voronin

*Portland State University*, [kvoronin@pdx.edu](mailto:kvoronin@pdx.edu)

Chak Shing Lee

*Center for Applied Scientific Computing*

Martin Neumüller

*Institute of Computational Mathematics*, [neumueller@numa.uni-linz.ac.at](mailto:neumueller@numa.uni-linz.ac.at)

Paulina Sepulveda

*Portland State University*, [spaulina@pdx.edu](mailto:spaulina@pdx.edu)

Panayot S. Vassilevski

*Portland State University*, [panayot@pdx.edu](mailto:panayot@pdx.edu)

Let us know how access to this document benefits you.

Follow this and additional works at: [https://pdxscholar.library.pdx.edu/pics\\_pub](https://pdxscholar.library.pdx.edu/pics_pub)

 Part of the [Mathematics Commons](#)

---

### Citation Details

Voronin, Kirill; Lee, Chak Shing; Neumüller, Martin; Sepulveda, Paulina; and Vassilevski, Panayot S., "Space-Time Discretizations Using Constrained First-Order System Least Squares (CFOSLS)" (2018). *Portland Institute for Computational Science Publications*. 4. [https://pdxscholar.library.pdx.edu/pics\\_pub/4](https://pdxscholar.library.pdx.edu/pics_pub/4)

This Pre-Print is brought to you for free and open access. It has been accepted for inclusion in Portland Institute for Computational Science Publications by an authorized administrator of PDXScholar. For more information, please contact [pdxscholar@pdx.edu](mailto:pdxscholar@pdx.edu).

# SPACE-TIME DISCRETIZATIONS USING CONSTRAINED FIRST-ORDER SYSTEM LEAST SQUARES (CFOSLS)

KIRILL VORONIN, CHAK SHING LEE, MARTIN NEUMÜLLER, PAULINA SEPULVEDA,  
AND PANAYOT S. VASSILEVSKI

ABSTRACT. This paper studies finite element discretizations for three types of time-dependent PDEs, namely heat equation, scalar conservation law and wave equation, which we reformulate as first order systems in a least-squares setting, subject to a space-time conservation constraint (coming from the original PDE). Available piecewise polynomial finite element spaces in  $(n + 1)$ -dimensions for functional spaces from the  $(n + 1)$ -dimensional de Rham sequence for  $n = 3, 4$  are used for the implementation of the method. Computational results illustrating the error behavior, iteration counts and performance of block-diagonal and monolithic geometric multigrid preconditioners are presented for the discrete CFOSLS system. The results are obtained from a parallel implementation of the methods for which we report reasonable scalability.

## 1. INTRODUCTION

Space-time finite element methods have been widely studied recently, see e.g. [N13], [S18] and references therein. Among the reasons for considering these approaches we can mention their potential for performing space-time adaptive schemes (e.g. [HH88], [T09], [GS18], [B08]), retaining convergence even under limited space-time regularity [DGNS17] and exploitation of parallelism without causality constraints [FFKMSV17]. In [TS04], [T06], [T09] least squares space-time methods are used to solve flow problems for moving domains. Several space-time first-order systems least squares (FOSLS) formulations has been studied in the literature for convection-diffusion [BVZ17], convection-reaction system with transformed variables [NBP11], Navier-Stokes equations for moving domain problems [MH97].

In the present paper we use a constrained least squares method similar to the approach in [AV14] within the setting from [NVV16] for combined space-time discretizations. The resulting formulation is referred to as *constrained first-order system least-squares*, or CFOSLS. More specifically, we rewrite the original PDE as a first-order system and consider the minimization problem for the corresponding least

---

1991 *Mathematics Subject Classification.* 65F10, 65N30.

*Key words and phrases.* CFOSLS, space-time, parabolic problems, hyperbolic problems, scalar conservation laws, wave equation, multigrid, finite element method.

This work was supported by the ARO under US Army Federal Grant # W911NF-15-1-0590 and # W911NF-16-1-0307. Numerical studies were partially facilitated by equipment acquired under NSF's Major Research Instrumentation grant DMS-1624776. Part of this work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

squares functional. To maintain conservative properties we add a constraint by imposing the original PDE in a weak sense elementwise.

Rewriting the original PDE as a first-order system introduces additional unknowns (hence additional memory demand) which is compensated by the advantage coming from the least squares formulation which leads to symmetric positive definite systems. The additional conservation constraint makes the overall CFOSLS formulation of a saddle-point form. For error analysis of this type of formulations using conforming vector  $H^1$  elements on a mesh  $\mathcal{T}_h$  and piecewise-constants for the Lagrange multipliers on a coarser mesh  $\mathcal{T}_H$  we refer to [AV14].

One of the main goals of the paper is to present a comprehensive convergence study in terms of order of approximation with respect to the mesh size for the recently designed 4D elements (available in [MFEM] for the entire 4D de Rham complex of finite element spaces, see [GNV17]). More precisely, for the discretization in 4D we use Raviart-Thomas finite elements to approximate vector functions in  $H(\text{div})$ , Lagrange finite elements or piecewise constant finite elements to approximate the original scalar unknown in  $H^1$  or  $L^2$  respectively, and special finite element spaces to approximate a divergence-free correction in the space prior to  $H(\text{div})$  (in 3D it corresponds to  $H(\text{curl})$  in the 4D de Rham sequence, see [GNV17]). The Lagrange multiplier is sought in the space of piecewise constants which is compatible with the corresponding  $H(\text{div})$  finite element space approximation.

We also present a first attempt to construct efficient linear solvers for the resulting systems of equations. It is clear that a direct approach to design scalable solvers is quite a challenging task since the space-time FOSLS functional does not exhibit full ellipticity. For the saddle point systems which arise immediately after discretization we study performance of block-diagonal algebraic preconditioners. A more sophisticated approach involves a formulation in the divergence-free subspace. The resulting system for a divergence-free correction turns out to be symmetric positive semi-definite and for solving it we exploit a multigrid (MG) preconditioner with two different options for a smoother. As our results show, multigrid preconditioners perform better than block-diagonal preconditioners both in terms of iteration counts and time. However, even multigrid preconditioners showed non-flat (dependent on mesh step) iteration counts. Thus, a question of constructing an optimal preconditioner remains open.

Due to the presence of time direction (causality of the problems), a feasible approach would be to explore specialized directional smoothers (e.g., like in [N13]) to improve MG convergence. Another way might be to split the space-time domain into a sequence of time slabs, and utilize a parallel-in-time method (e.g., [LMT01] and [GA14]).

Another future direction of the project is to deal with the increased memory demand implied by the space-time discretizations in 4D. In [NVV16] the authors developed a so called AMGe upscaling approach where the main idea was to perform local adaptive coarsening of a sufficiently fine discretization. This technique allowed to avoid storing the memory-expensive global fine grid problem. A more natural approach handles the memory issue by employing appropriate adaptive mesh refinement, or AMR, (i.e. from coarse-to-fine direction) utilizing 4D elements. To decide where to refine, for CFOSLS formulation one can use the least-squares functional evaluated on individual elements.

Designing efficient scalable AMR strategies for the 4D elements is an ongoing project that we already have in place.

The structure of the remainder of this paper is as follows. In Section 2 we introduce a unified CFOSLS formulation for three types of time-dependent PDEs we consider. Next, in Section 3 we show how to reduce the saddle-point problem which comes from the variational formulation to a symmetric positive semi-definite system in the div-free subspace. To this end we utilize the space prior to  $H(\text{div})$  in the 4D de Rham sequence (see [GNV17]). The finite element discretization and the resulting linear systems are presented in Section 4. Within an additional Subsection 4.1 we sketch a practical algorithm for finding a particular solution to the divergence constraint at discrete level. In Section 5 we describe two types of iterative solvers which we used for solving the discrete problems. Subsection 5.1 is focused on the algebraic block-diagonal preconditioners while in Subsection 5.2 we present a geometric multigrid preconditioner used for the systems arising in the div-free formulation. A main section of the paper, Section 6, presents convergence study of the CFOSLS discretizations (Subsection 6.1) accompanied by some results for parallel solvers performance (Subsection 6.2) in both 3D and 4D cases. Finally, in Section 7 we draw the conclusions and outline future research directions.

## 2. CFOSLS FORMULATION OF TIME-DEPENDENT PDES

Let  $\Omega \subset \mathbb{R}^n$  be an open bounded spatial domain with Lipschitz boundary  $\partial\Omega$  and let  $\Omega_T = \Omega \times (0, T) \subset \mathbb{R}^{n+1}$  be the corresponding space-time domain, where  $T > 0$  represents the final time.

We consider three classical time-dependent PDEs: heat equation, scalar conservation law (or linear transport equation), and wave equation. All of them can be written in the unified partial differential equation:

$$(2.1) \quad \text{div } \mathcal{L}(u) := \frac{\partial}{\partial t}(\mathcal{L}_t(u)) + \text{div}_{\mathbf{x}}(\mathcal{L}_{\mathbf{x}}(u)) = f(\mathbf{x}, t), \quad \text{for all } (\mathbf{x}, t) \in \Omega_T,$$

where we define the space-time differential operator  $\mathcal{L}$  to be

$$\mathcal{L}(u) = \begin{bmatrix} \mathcal{L}_{\mathbf{x}}(u) \\ \mathcal{L}_t(u) \end{bmatrix},$$

while the definitions of  $\mathcal{L}_{\mathbf{x}}(u)$  and  $\mathcal{L}_t(u)$  for the considered PDEs can be found in Table 1. The operators  $\text{div}$  and  $\text{div}_{\mathbf{x}}$  stand for the  $(n+1)$ - and  $n$ -dimensional divergence operators, i.e., for a  $C^1$ -vector function  $\mathbf{g}(\mathbf{x}, t)$  in  $\mathbb{R}^{n+1}$  with components

$$\mathbf{g} = \begin{bmatrix} \mathbf{g}^{\mathbf{x}} \\ g^t \end{bmatrix}, \quad \mathbf{g}^{\mathbf{x}} \in \mathbb{R}^n, \quad g^t \in \mathbb{R} \quad \forall (\mathbf{x}, t) \in \Omega_T$$

we can write

$$\text{div } \mathbf{g} = \partial_t g^t + \sum_{i=1}^n \partial_{x_i} \mathbf{g}_i^{\mathbf{x}} = \partial_t g^t + \text{div}_{\mathbf{x}} \mathbf{g}^{\mathbf{x}}.$$

In order to have a well-posed problem for equation (2.1) we impose boundary conditions on  $\partial\Omega_T$  specified in Table 1. Notice that these “boundary conditions” on  $\partial\Omega_T$  will contain both initial and boundary conditions in the usual sense of time-dependent PDEs. We will consider additional notations for specific parts of the space-time boundary of  $\Omega_T$ , namely

$$\Gamma_0 := \Omega \times \{0\}, \quad \Gamma_s := \partial\Omega \times (0, T), \quad \Gamma_D := \Gamma_0 \cup \Gamma_s,$$

and, for scalar conservation law, we will also introduce

$$\Gamma_- := \{\mathbf{x} \in \partial\Omega \mid \mathbf{v} \cdot \mathbf{n} < 0\} \times (0, T)$$

where  $\mathbf{v}(\mathbf{x}, t) : \Omega_T \rightarrow \mathbb{R}^n$  is a given advection vector for which we use standard assumptions that guarantee the well-posedness [EV98] of problem (2.1). Boundary conditions for (2.1) can be formally written as

$$tr(u) = 0,$$

where the actual definitions of the trace operator  $tr$  for the different PDEs are listed in Table 1. For simplicity, we restrict ourselves to homogeneous boundary conditions since the treatment of inhomogeneous boundary conditions is straightforward.

TABLE 1. Definitions of  $\mathcal{L}_{\mathbf{x}}(u)$ ,  $\mathcal{L}_t(u)$ , and  $tr(u)$  for different PDEs.

PDEs	$\mathcal{L}_{\mathbf{x}}(u)$	$\mathcal{L}_t(u)$	$tr(u)$
Heat equation	$-\nabla_{\mathbf{x}}u$	$u$	$u _{\Gamma_D}$
Conservation law	$\mathbf{v}u$	$u$	$u _{\Gamma_0 \cup \Gamma_-}$
Wave equation	$-\nabla_{\mathbf{x}}u$	$\partial_t u$	$[u _{\Gamma_D}, \partial_t u _{\Gamma_0}]$

**2.1. Constrained space-time formulation.** Following [NVV16] we rewrite equation (2.1) as a first-order system by introducing a new variable  $\boldsymbol{\sigma} := \mathcal{L}(u)$ , obtaining

$$(2.2) \quad \begin{aligned} \boldsymbol{\sigma} - \mathcal{L}(u) &= 0, \\ \operatorname{div} \boldsymbol{\sigma} &= f. \end{aligned}$$

We will also make use of the following well-known functional spaces:  $L^2(\Omega_T)$ , the space of square-integrable functions in  $\Omega_T$ ;  $H^1(\Omega_T) = \{u \in L^2(\Omega_T) : \nabla u \in L^2(\Omega_T)^{n+1}\}$ ;  $H(\operatorname{div}; \Omega_T) = \{\boldsymbol{\sigma} \in L^2(\Omega_T)^{n+1} : \operatorname{div} \boldsymbol{\sigma} \in L^2(\Omega_T)\}$  and omit  $\Omega_T$  for the sake of brevity. Also, we denote by  $(\cdot, \cdot)$  the inner product with respect to scalar and vector  $L^2$ , and by  $\|\cdot\|$  the corresponding norm. From now on we will understand all differential operators in the weak sense.

To formulate the least squares problem we consider the following spaces with weakly imposed boundary conditions:

$$(2.3) \quad \begin{aligned} R &:= \{\boldsymbol{\tau} \in H(\operatorname{div}) \mid tr^\sigma(\boldsymbol{\tau}) = 0\}; \\ V &:= \{v \in H^1 \mid tr^u(v) = 0\}, \end{aligned}$$

where the definitions of the trace operators  $tr^\sigma$  and  $tr^u$  are given in Table 2. These trace operators are well defined, see [GT14].

TABLE 2. Definitions of  $tr^\sigma(\boldsymbol{\sigma})$ , and  $tr^u(u)$  for the different PDEs.

PDEs	$tr^\sigma(\boldsymbol{\sigma})$	$tr^u(u)$
Heat equation	N/A	$u _{\Gamma_D}$
Conservation law	N/A	$u _{\Gamma_0 \cup \Gamma_-}$
Wave equation	$(\boldsymbol{\sigma} \cdot \mathbf{n}) _{\Gamma_0}$	$u _{\Gamma_D}$

Then, for any given  $f \in L^2$  the FOSLS functional for problem (2.2)

$$(2.4) \quad J(\boldsymbol{\sigma}, u) = \|\boldsymbol{\sigma} - \mathcal{L}(u)\|^2 + \|f - \operatorname{div} \boldsymbol{\sigma}\|^2$$

is well defined in  $R \times V$ . Now, the constrained space-time first-order system least squares (CFOSLS) problem is to find the minimizer of the functional (2.4) under the constraint given by the conservation equation:

$$(2.5) \quad (\boldsymbol{\sigma}, u) = \operatorname{argmin}_{(\boldsymbol{\tau}, v) \in R \times V} J(\boldsymbol{\tau}, v) \quad \text{subject to} \quad \operatorname{div} \boldsymbol{\sigma} = f.$$

The constraint is imposed in a weak sense by introducing a Lagrange multiplier from  $L_2$ . Then we can omit the second term  $\|f - \operatorname{div} \boldsymbol{\sigma}\|^2$  in the functional(2.4).

**2.2. Variational CFOSLS formulation.** For the system (2.2) we define the operator

$$\mathcal{A}(\boldsymbol{\sigma}, u) = \boldsymbol{\sigma} - \mathcal{L}(u).$$

It is well known that the minimizer of the constrained minimization problem (2.5) is characterized by the first order (or Karush-Kuhn-Tucker) optimality condition, which leads in our case to the following saddle point system:

Find  $(\boldsymbol{\sigma}, u) \in R \times V$  and  $\lambda \in L^2$  such that

$$(2.6) \quad \begin{aligned} (\mathcal{A}(\boldsymbol{\sigma}, u), \mathcal{A}(\boldsymbol{\tau}, v)) + (\lambda, \operatorname{div} \boldsymbol{\tau}) &= 0 & \forall (\boldsymbol{\tau}, v) \in R \times V, \\ (\operatorname{div} \boldsymbol{\sigma}, \mu) &= (f, \mu) & \forall \mu \in L^2. \end{aligned}$$

The continuity of the bilinear form corresponding to the operator  $\mathcal{A}$  is obvious due to the choice of spaces  $R$  and  $V$ . Also, this form can be considered to be weakly-coercive in the sense of the definition given in [AV14].

**Remark 2.1.** *A traditional variational FOSLS formulation would be written in our notations as:*

Find  $(\boldsymbol{\sigma}, u) \in R \times V$  such that

$$(\mathcal{A}(\boldsymbol{\sigma}, u), \mathcal{A}(\boldsymbol{\tau}, v)) + (\operatorname{div} \boldsymbol{\sigma}, \operatorname{div} \boldsymbol{\tau}) = (f, \operatorname{div} \boldsymbol{\tau}) \quad \forall (\boldsymbol{\tau}, v) \in R \times V.$$

**Remark 2.2.** *Notice that, since  $u$  belongs to  $H^1$ , for the scalar conservation law we can consider a different second term  $\|\operatorname{div}(\mathbf{v}u) - f\|^2$  in the FOSLS functional (2.4). Then we will have an extra weighted diffusion term  $(\mathbf{v} \nabla_x u, \mathbf{v} \nabla_x u)$  for  $u$  (which already exists for heat and wave equations due to the definition of  $\boldsymbol{\sigma}$  for these problems).*

*Then, instead of (2.6), we will have the following system:*

$$(2.7) \quad \begin{aligned} (\mathcal{A}_+(\boldsymbol{\sigma}, u), \mathcal{A}_+(\boldsymbol{\tau}, v)) + (\lambda, \operatorname{div} \boldsymbol{\tau}) &= (f, \operatorname{div}(\mathbf{v}v)) & \forall (\boldsymbol{\tau}, v) \in R \times V, \\ (\operatorname{div} \boldsymbol{\sigma}, \mu) &= (f, \mu) & \forall \mu \in L^2. \end{aligned}$$

with

$$(2.8) \quad \mathcal{A}_+(\boldsymbol{\sigma}, u) = \begin{bmatrix} \mathcal{A}(\boldsymbol{\sigma}, u) \\ \operatorname{div}(\mathbf{v}u) \end{bmatrix}.$$

We use this modified formulation when we consider scalar conservation law with  $u \in H^1$ .

As one can notice, in equation (2.6) we seek for  $u$  in  $V \subset H^1$ . However, in case of the scalar conservation law we can also allow  $u$  to have less regularity. This is due to the fact that unlike the other example problems, the definition of  $\mathcal{L}(u)$  does not involve any derivatives of  $u$ . Hence, we can obtain a new formulation by seeking  $u \in L^2(\Omega)$  instead of  $V$  and imposing the boundary conditions on  $\boldsymbol{\sigma}$  instead of  $u$ , i.e. replacing  $u|_{\Gamma_0 \cup \Gamma_-}$  with  $(\boldsymbol{\sigma} \cdot \mathbf{n})|_{\Gamma_0 \cup \Gamma_-}$ .

Next, as it follows from the first equation of (2.2), the scalar unknown  $u$  can be expressed in terms of  $\boldsymbol{\sigma}$ :

$$(2.9) \quad u = \frac{1}{\mathbf{b}^T \mathbf{b}} \mathbf{b}^T \boldsymbol{\sigma},$$

where we introduced the space-time vector function

$$\mathbf{b} = \mathbf{b}(\mathbf{x}, t) := \begin{bmatrix} \mathbf{v}(\mathbf{x}, t) \\ 1 \end{bmatrix}.$$

Notice that  $\mathbf{b}^T \mathbf{b} \equiv \mathbf{v}^T \mathbf{v} + 1 \geq 1$ , thus  $\mathbf{b}^T \mathbf{b}$  is a strictly positive function.

Now we substitute (2.9) for  $u$  in (2.6) and redefine  $\mathcal{A}(\boldsymbol{\sigma}, u)$  in (2.6) to depend on  $\boldsymbol{\sigma}$  only. The final variational problem then reads as:

Find  $\boldsymbol{\sigma} \in R$  and  $\lambda \in L^2$  such that

$$(2.10) \quad \begin{aligned} (K\boldsymbol{\sigma}, \boldsymbol{\tau}) + (\lambda, \operatorname{div} \boldsymbol{\tau}) &= 0 & \forall \boldsymbol{\tau} \in R, \\ (\operatorname{div} \boldsymbol{\sigma}, \mu) &= (f, \mu) & \forall \mu \in L^2. \end{aligned}$$

where we have introduced a symmetric positive semi-definite matrix coefficient  $K(x, t)$  defined as

$$(2.11) \quad K := I - \frac{1}{\mathbf{b}^T \mathbf{b}} \mathbf{b} \mathbf{b}^T.$$

Notice that (2.10) can be obtained as a variational formulation of the minimization problem for the following modified functional:

$$\tilde{J}(\boldsymbol{\sigma}) = \|\boldsymbol{\sigma}\|_K^2 + \|f - \operatorname{div} \boldsymbol{\sigma}\|^2,$$

where  $\|\boldsymbol{\sigma}\|_K$  represents the  $K$ -weighted  $L^2$  seminorm for  $\boldsymbol{\sigma}$ .

**Remark 2.3.** *One should be careful when considering the alternative formulation (2.10). Although it looks similar to a standard mixed formulation for a Laplace equation, the presence of the matrix weight  $K$  makes it very different.  $K(x, t)$  is rank-deficient (since  $K \cdot \mathbf{b} \equiv 0$ ) at any point  $(\mathbf{x}, t)$  but the corresponding global operator is non-singular at the kernel of the divergence operator due to the underlying assumption that the original differential problem is well-posed, which implies, in particular, that the global matrix corresponding to  $(K, \cdot)$  should be positive definite in the subspace*

$\{\boldsymbol{\sigma} \in R : \operatorname{div} \boldsymbol{\sigma} = 0\}$ . Local rank deficiency certainly affects the conditioning number of the resulting linear system and makes the problem of finding optimal iterative solvers much more complicated.

**Remark 2.4.** Once  $\boldsymbol{\sigma}$  is found as a solution to (2.10),  $u$  can be recovered from the weak form of (2.9) if needed.

### 3. DIVERGENCE-FREE FORMULATION

Making the solution satisfy the divergence constraint explicitly is an attractive and highly desirable property. However, as we will see later in section 6, straightforward numerical approximations of (2.6), (2.7) or (2.10) lead to significant difficulties for the preconditioners. This motivates us to consider alternative formulations in the divergence-free subspace.

**3.1. Characterization of divergence-free functions.** First we introduce an additional functional space  $N$  which will be used for characterization of divergence-free functions as

$$N = \begin{cases} H(\operatorname{curl}), & n = 3 \\ H(\operatorname{Div}), & n = 4 \end{cases}$$

and exterior derivative operator  $d : N \mapsto R$ :

$$d = \begin{cases} \operatorname{curl}, & n = 3, \\ \operatorname{Div}, & n = 4. \end{cases}$$

Let also

$$(3.1) \quad N^0 := \{\boldsymbol{\psi} \in N \mid \operatorname{tr}^d(\boldsymbol{\psi}) = 0\}.$$

For our problem we assume that the boundary condition imposed on the elements of  $N^0$  is imposed on the same part of  $\partial\Omega_T$  on which boundary conditions for  $\boldsymbol{\sigma}$  are used. For example, in the three-dimensional case we use  $\operatorname{tr}^{\operatorname{curl}}(\boldsymbol{\psi}) = \boldsymbol{\psi} \times \mathbf{n}|_\Gamma$  where  $\Gamma = \Gamma_0$  for the wave equation and  $\Gamma = \Gamma_0 \cup \Gamma_-$  for the scalar conservation law with  $u \in L^2$ .

Since the four-dimensional case might be not well known, we briefly recall the definition of the space  $H(\operatorname{Div})$  and the trace in this space: [GNV17]:

$$H(\operatorname{Div}) = \left\{ \mathbf{F} \in [L^2(\Omega_T)]_{skew}^{4 \times 4} \mid \operatorname{Div} \mathbf{F} \in [L^2(\Omega_T)]^4 \right\}$$

The operator  $\operatorname{Div}$  is defined as vector divergence operator:

$$[\operatorname{Div} \mathbf{F}]_i = \sum_j \partial_j F_{ij}$$

for a skew-symmetric  $\mathbf{F} = [F_{ij}]$ . The trace operator in the definition of  $N^0$  for  $n = 4$  can be then defined as:

$$\operatorname{tr}^{\operatorname{Div}}(\mathbf{F}) = \mathbf{F}\mathbf{n}|_\Gamma.$$

where  $\Gamma$  is the part of the boundary where boundary condition on  $\boldsymbol{\sigma}$  is imposed. Further details on the four-dimensional case can be found, e.g. in [GNV17].



Spaces  $N^0$  and  $R^0 := \{\boldsymbol{\tau} \in R \text{ s.t. } \text{tr}^\sigma(\boldsymbol{\tau}) = 0\}$  represent a part of the exact de Rham sequence

$$\dots \longrightarrow N^0 \xrightarrow{\text{d}} R^0 \xrightarrow{\text{div}} \dots$$

In particular, it follows that any  $\boldsymbol{\sigma} \in R^0$  s.t.  $\text{div } \boldsymbol{\sigma} = 0$  can be represented as

$$(3.2) \quad \boldsymbol{\sigma} = \text{d } \boldsymbol{\psi}$$

for some  $\boldsymbol{\psi} \in N^0$ . Moreover, an important de Rham sequence property states that

$$\text{div } \text{d } \boldsymbol{\psi} = 0 \quad \forall \boldsymbol{\psi} \in N^0.$$

**3.2. Divergence-free setting.** Now we can proceed with reformulating (2.6) in the divergence-free subspace. The main idea is to represent  $\boldsymbol{\sigma}$  as a sum

$$(3.3) \quad \boldsymbol{\sigma} = \tilde{\boldsymbol{\sigma}} + \bar{\boldsymbol{\sigma}},$$

where  $\tilde{\boldsymbol{\sigma}} \in R$  is a particular solution which satisfies the divergence constraint

$$(3.4) \quad (\text{div } \tilde{\boldsymbol{\sigma}}, \boldsymbol{\mu}) = (f, \boldsymbol{\mu}) \quad \forall \boldsymbol{\mu} \in L^2,$$

and the boundary conditions imposed on  $\boldsymbol{\sigma}$ , and  $\bar{\boldsymbol{\sigma}} \in R^0$  is a divergence-free correction which satisfies homogeneous boundary conditions  $\text{tr}^\sigma(\bar{\boldsymbol{\sigma}}) = 0$ . Using the key property (3.2) we can then represent such  $\bar{\boldsymbol{\sigma}}$  as  $\text{d } \boldsymbol{\psi}$  and recast the problem into  $N^0$ .

In other words, we can solve (2.6) (or, similarly (2.10)) in two steps:

**Step 1.** Find any  $\tilde{\boldsymbol{\sigma}} \in R$  which satisfies

$$(3.5) \quad (\text{div } \tilde{\boldsymbol{\sigma}}, \boldsymbol{\mu}) = (f, \boldsymbol{\mu}) \quad \forall \boldsymbol{\mu} \in L^2.$$

Notice that solution to (3.5) is determined only up to an arbitrary divergence-free function.

**Step 2.** Find a divergence free-correction  $\bar{\boldsymbol{\sigma}}$  in the form of  $\text{d } \boldsymbol{\psi}$ . After **step 1** is completed we can substitute (3.3) into (2.6) and arrive at the following problem for  $\bar{\boldsymbol{\sigma}}$ :

Find  $(\bar{\boldsymbol{\sigma}}, u) \in R^0 \times V$  and  $\lambda \in L^2$  such that

$$(3.6) \quad \begin{aligned} (\mathcal{A}(\bar{\boldsymbol{\sigma}}, u), \mathcal{A}(\boldsymbol{\tau}, v)) + (\lambda, \text{div } \boldsymbol{\tau}) &= -(\mathcal{A}(\tilde{\boldsymbol{\sigma}}, u), \mathcal{A}(\boldsymbol{\tau}, v)) \quad \forall (\boldsymbol{\tau}, v) \in R^0 \times V, \\ (\text{div } \bar{\boldsymbol{\sigma}}, \boldsymbol{\mu}) &= (f - \text{div } \tilde{\boldsymbol{\sigma}}, \boldsymbol{\mu}) \equiv 0 \quad \forall \boldsymbol{\mu} \in L^2. \end{aligned}$$

Next, using (3.2) we can substitute  $\text{d } \boldsymbol{\psi}$  for  $\bar{\boldsymbol{\sigma}}$  with  $\boldsymbol{\psi} \in N^0$ . Then it is sufficient to consider only  $\boldsymbol{\tau} = \text{d } \boldsymbol{\zeta}$  for  $\boldsymbol{\zeta} \in N^0$  in the first equation of (3.6) while the second equation is satisfied automatically. Hence (3.6) simplifies to:

Find  $(\boldsymbol{\psi}, u) \in N^0 \times V$  which solve

$$(3.7) \quad (\mathcal{A}(\text{d } \boldsymbol{\psi}, u), \mathcal{A}(\text{d } \boldsymbol{\zeta}, v)) = -(\mathcal{A}(\tilde{\boldsymbol{\sigma}}, u), \mathcal{A}(\text{d } \boldsymbol{\zeta}, v)) \quad \forall (\boldsymbol{\zeta}, v) \in N^0 \times V,$$

which is the final form of the problem to be solved at **step 2**.

The resulting  $\boldsymbol{\sigma}$  is computed using (3.3) as a sum of  $\tilde{\boldsymbol{\sigma}}$  and  $\bar{\boldsymbol{\sigma}}$ .

**Remark 3.1.** Obviously the operator in the divergence-free subspace has the kernel coming from the previous space of the de Rham sequence (for  $n = 3$ , e.g. it's  $\nabla H^1(\Omega_T) \subset \ker \text{curl}$  due to the well-known identity  $\text{curl } \nabla v = 0 \quad \forall v \in H^1(\Omega_T)$ ). However, we only need  $\text{d } \boldsymbol{\psi}$  to reconstruct  $\boldsymbol{\sigma}$ .

## 4. FINITE ELEMENT DISCRETIZATION

Since the considered formulations make use of standard well-known functional spaces (defined in (2.3) and (3.1)), we want to exploit properties of discrete de Rham sequences. To this end for discretization we use canonical finite element spaces  $\mathcal{P}_r^- \Lambda^k$ , following theory of finite element exterior calculus [AFW10].

Assume that there is a conforming triangulation  $\mathcal{T}_h$  of the considered space-time domain  $\Omega_T$  into  $(n+1)$ -dimensional simplices:

$$\bar{\Omega}_T = \bigcup_{T \in \mathcal{T}_h} \bar{T},$$

where  $h = \max_{T \in \mathcal{T}_h} \text{diam}(T)$ . Then for approximation of (2.6), (2.7), (2.10) and (3.7) we use the finite element spaces listed in Table 3. As one can notice these are standard finite element spaces such as conforming Lagrange, Raviart-Thomas and Nedelec finite element families. In the last column of Table 3 local basis functions are given for the lowest order case in terms of Whitney forms [AFW10] and barycentric coordinates  $\{\lambda_i\}$ . In the general case,  $\mathcal{P}_r^- \Lambda^k$  can be defined through the spaces of finite element differential forms and the Koszyl differential:

$$\mathcal{P}_r^- \Lambda^k = \mathcal{P}_{r-1} \Lambda^k + \kappa \mathcal{P}_{r-1} \Lambda^{k+1}.$$

Here  $\mathcal{P}_{r-1} \Lambda^k$  is the full polynomial space for  $k$ -forms and  $\kappa$  is the Koszyl differential. Degrees of freedom are then given on subsimplices  $f$  of dimension  $n_f \leq k$  by moments of the trace:

$$(4.1) \quad u \rightarrow \int_f (tr_f u) \wedge q, \quad \forall q \in \mathcal{P}_{r+k-n_f-1} \Lambda^{n_f-k}(f).$$

For the further details about finite element spaces defined through differential forms we refer to [AFW10]. The case  $n = 4$  can be found in [GNV17].

TABLE 3. Table of finite element spaces.

Functional space	Finite element space	Local basis for $r = 0$
$V \subset H^1$	Lagrange, $\mathcal{P}_r^- \Lambda^0$	$\lambda_i$
$N \subset H(\text{curl})$	Nedelec, $\mathcal{P}_r^- \Lambda^1, n = 3$	$\lambda_i d\lambda_j - \lambda_j d\lambda_i$
$N \subset H(\text{Div})$	Divskew, $\mathcal{P}_r^- \Lambda^2, n = 4$	$\lambda_i d\lambda_j \wedge d\lambda_k - \lambda_j d\lambda_i \wedge d\lambda_k + \lambda_k d\lambda_i \wedge d\lambda_j$
$R \subset H(\text{div})$	Raviart-Thomas, $\mathcal{P}_r^- \Lambda^3$	$\sum_{i=0}^d (-1)^i \lambda_{\pi(i)} d\lambda_{\pi(0)} \wedge \dots \wedge \widehat{d\lambda_{\pi(i)}} \wedge d\lambda_{\pi(d)}$
$L^2$	Discontinuous $L^2$ , $\mathcal{P}_r^- \Lambda^4$	$d\lambda_i \wedge d\lambda_j \wedge d\lambda_k \wedge d\lambda_l$

The finite element discretization of (2.6), (2.7) and (2.10) leads to linear systems of the form

$$(4.2) \quad \begin{bmatrix} \mathcal{M} & \mathcal{G}^T & \mathcal{D}^T \\ \mathcal{G} & \mathcal{X} & 0 \\ \mathcal{D} & 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\sigma}_h \\ \mathbf{u}_h \\ \boldsymbol{\lambda}_h \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{g} \\ \mathbf{f} \end{bmatrix}$$

and

$$(4.3) \quad \begin{bmatrix} \mathcal{M} & \mathcal{D}^T \\ \mathcal{D} & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\sigma}_h \\ \boldsymbol{\lambda}_h \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{f} \end{bmatrix}$$

correspondingly. Here the matrices  $\mathcal{M}$ ,  $\mathcal{G}$  and  $\mathcal{X}$  come from the discretizations of the bilinear form  $\mathcal{A}(\boldsymbol{\sigma}, u)$ , and  $\mathcal{D}$  is the discrete divergence operator.

Now we derive a finite element discretization of the divergence-free formulation (3.7). We start with applying the approach described in Section 3 at discrete level directly.

To this end, we consider a discrete decomposition (cf. (3.3)):  $\boldsymbol{\sigma}_h = \tilde{\boldsymbol{\sigma}}_h + \bar{\boldsymbol{\sigma}}_h$ , where  $\tilde{\boldsymbol{\sigma}}_h$  is a particular solution that satisfies the discrete divergence constraint

$$(4.4) \quad \mathcal{D}\tilde{\boldsymbol{\sigma}}_h = \mathbf{f}$$

and imposed boundary conditions, while  $\bar{\boldsymbol{\sigma}}_h$  is a divergence-free correction. An efficient method for finding  $\tilde{\boldsymbol{\sigma}}_h$  will be described in Subsection 4.1. Now, assuming we have already found a particular solution  $\tilde{\boldsymbol{\sigma}}_h$  for (4.4) we can rewrite (4.2) as

$$(4.5) \quad \begin{bmatrix} \mathcal{M} & \mathcal{G}^T & \mathcal{D}^T \\ \mathcal{G} & \mathcal{X} & 0 \\ \mathcal{D} & 0 & 0 \end{bmatrix} \begin{bmatrix} \bar{\boldsymbol{\sigma}}_h \\ \mathbf{u}_h \\ \boldsymbol{\lambda}_h \end{bmatrix} = \begin{bmatrix} -\mathcal{M}\tilde{\boldsymbol{\sigma}}_h \\ \mathbf{g} - \mathcal{G}\tilde{\boldsymbol{\sigma}}_h \\ 0 \end{bmatrix}$$

Since  $\bar{\boldsymbol{\sigma}}_h$  is divergence free,  $\bar{\boldsymbol{\sigma}}_h$  belongs to the range of the discretized exterior derivative  $\mathcal{C} : N_h \rightarrow R_h$ . That is,  $\bar{\boldsymbol{\sigma}}_h = \mathcal{C}\boldsymbol{\psi}_h$  for some  $\boldsymbol{\psi}_h \in N_h$ , where  $\mathcal{C}$  satisfies  $\mathcal{D}\mathcal{C} = 0$  (which is in three-dimensional case a well-known identity  $\text{div curl} = 0$ ). Hence, (4.5) is equivalent to

$$(4.6) \quad \begin{bmatrix} \mathcal{C}^T \mathcal{M} \mathcal{C} & (\mathcal{G} \mathcal{C})^T \\ \mathcal{G} \mathcal{C} & \mathcal{X} \end{bmatrix} \begin{bmatrix} \boldsymbol{\psi}_h \\ \mathbf{u}_h \end{bmatrix} = \begin{bmatrix} -\mathcal{C}^T \mathcal{M} \tilde{\boldsymbol{\sigma}}_h \\ \mathbf{g} - \mathcal{G} \tilde{\boldsymbol{\sigma}}_h \end{bmatrix}.$$

which is the final form of the linear system related to (3.7). The same transformations lead to the following discretized form of (2.10) in the div-free formulation:

$$(4.7) \quad \mathcal{C}^T \mathcal{M} \mathcal{C} \boldsymbol{\psi}_h = -\mathcal{C}^T \mathcal{M} \tilde{\boldsymbol{\sigma}}_h.$$

In the end,  $\boldsymbol{\sigma}_h$  is computed as

$$\boldsymbol{\sigma}_h = \tilde{\boldsymbol{\sigma}}_h + \mathcal{C}\boldsymbol{\psi}_h.$$

Unlike (4.2) and (4.3), systems (4.6) and (4.7) are not of a saddle-point type but symmetric positive semi-definite. Their main advantage over (4.2) is that in this case we can exploit powerful multigrid preconditioners.

**Remark 4.1.** *To perform a finite element discretization of (2.6), (2.7) or (3.7) which corresponds to  $(n+1)$ -dimensional problems when  $n = 3$ , one needs four-dimensional meshes. Since it is not a commonly supported case for the standard mesh generators, we have implemented a parallel mesh generating algorithm. The algorithm creates a simplicial  $(n+1)$ -dimensional mesh in a space-time cylinder with a given unstructured  $n$ -dimensional mesh using uniform time stepping, so that the resulting mesh is a tensor-product extension of the given space mesh into the space-time cylinder.*

The algorithm uses a standard idea from [KB] of tessellating  $(n + 1)$ -dimensional prism into simplices. For each  $n$ -dimensional simplex given in the base we tessellate a corresponding space-time prism and ensure conformity of the mesh by imposing a suitable local vertex ordering for the lateral faces.

In the following subsection we will describe one feasible approach for finding a particular solution of discrete divergence constraint.

**4.1. Multilevel algorithm for finding a particular solution.** In this subsection we briefly sketch an efficient multilevel algorithm for finding a particular solution to the divergence constraint (4.4), see Appendix F.3 in [V08].

Assume that for  $\ell = \{0, \dots, L\}$  we have a sequence of nested meshes  $\{\mathcal{T}_{h_\ell}\}$  where  $\mathcal{T}_{h_0}$  is the finest mesh, and  $\mathcal{T}_{h_L} := \mathcal{T}_H$  is the coarsest mesh. For each  $\ell = \{0, \dots, L - 1\}$  we denote by  $W_{\ell,T}$  and  $R_{\ell,T}$  the local Raviart-Thomas and discontinuous constant spaces defined on each  $\mathcal{T}_{h_\ell}$ . Also, let  $R_{h_\ell}$  and  $W_{h_\ell}$  be the Raviart-Thomas space and the spaces of piecewise constants associated with  $\mathcal{T}_{h_\ell}$ .

Then, for each  $\ell = \{0, \dots, L - 1\}$  we solve following local problems on coarser elements  $T \in \mathcal{T}_{h_{\ell+1}}$  viewed as level  $\ell$  subdomains:

Find  $\tilde{\sigma}_\ell \in R_{\ell,T}$  and  $\lambda_\ell \in W_{\ell,T}$  s.t:

$$\begin{aligned} (\tilde{\sigma}_\ell, \boldsymbol{\tau}) + (\lambda_\ell, \operatorname{div} \boldsymbol{\tau}) &= 0 && \text{for all } \boldsymbol{\tau} \in R_{\ell,T}, \\ (\operatorname{div} \tilde{\sigma}_\ell, v) &= (Q_\ell f - Q_{\ell+1} f, v) && \text{for all } v \in W_{\ell,T}. \end{aligned}$$

Here  $Q_\ell$  are computable  $L^2$ -projections  $Q_\ell : L^2 \mapsto W_\ell = W_{h_\ell}$ . Notice that the local problems are decoupled via imposing homogeneous boundary conditions for  $\tilde{\sigma}_\ell$  on  $\partial T$  for each coarse element  $T$ . Because  $(Q_\ell - Q_{\ell+1})f$  is orthogonal to the constant functions on  $T$ , the above problem is solvable.

Finally, for  $\ell = L$  (i.e., at the coarsest level) we solve the global problem of small size:

$$\begin{aligned} (\tilde{\sigma}_L, \boldsymbol{\tau}) + (\lambda_L, \operatorname{div} \boldsymbol{\tau}) &= 0, && \text{for all } \boldsymbol{\tau} \in R_{h_L}, \\ (\operatorname{div} \tilde{\sigma}_L, v) &= (Q_L f, v), && \text{for all } v \in W_{h_L}. \end{aligned}$$

For the coarsest level problem we impose on  $\tilde{\sigma}_L$  homogeneous boundary conditions on the part of the domain boundary where, we specified boundary conditions for  $\boldsymbol{\sigma}$  in the original problem.

In the end we accumulate solutions at each level and the desired result is given as

$$\tilde{\boldsymbol{\sigma}} = \sum_{\ell=0}^L \tilde{\boldsymbol{\sigma}}_\ell.$$

Obviously, the constructed  $\tilde{\boldsymbol{\sigma}}$  satisfies (4.4).

## 5. ITERATIVE SOLVERS

To solve discretized systems (4.2) and (4.6) we use the following two iterative solvers. The first option is algebraic block-diagonal preconditioners for (4.2) or (4.3). The second option is a geometric multigrid applied to the divergence-free formulations (4.6) or (4.7).

**5.1. Algebraic block-diagonal preconditioners.** Since (4.2) is symmetric but indefinite, it seems natural to consider the minimal residual method [PS75] with a suitable preconditioner. The preconditioner we opt for is the following block-diagonal preconditioner. Let  $\mathcal{B}_{\mathcal{M}}$  be the Jacobi smoother for  $\mathcal{M}$  and let  $\mathcal{S} = \mathcal{D}\mathcal{B}_{\mathcal{M}}^{-1}\mathcal{D}^T$ , then the block-diagonal preconditioner for (4.2) is defined as

$$\mathcal{B} = \begin{bmatrix} \mathcal{B}_{\mathcal{M}} & 0 & 0 \\ 0 & \mathcal{B}_{\mathcal{X}} & 0 \\ 0 & 0 & \mathcal{B}_{\mathcal{S}} \end{bmatrix},$$

where  $\mathcal{B}_{\mathcal{X}}$  and  $\mathcal{B}_{\mathcal{S}}$  are some algebraic multigrid (AMG) preconditioners for  $\mathcal{X}$  and  $\mathcal{S}$  respectively. Similarly, for (4.3) we can write a similar preconditioner in the form

$$\mathcal{B} = \begin{bmatrix} \mathcal{B}_{\mathcal{M}} & 0 \\ 0 & \mathcal{B}_{\mathcal{S}} \end{bmatrix}.$$

In our experiments, we used the algebraic multigrid preconditioner (BoomerAMG) from *hypre*, a library of scalable linear solvers developed at Lawrence Livermore National Laboratory, cf. [HY02, BFKY12].

**5.2. Geometric multigrid for divergence-free formulation.** The solvers to be described here are proposed for solving discretized divergence-free formulations (4.6) and (4.7) developed in section 4. Notice that the corresponding systems are symmetric positive semi-definite, so we can solve them by the preconditioned conjugate gradients (PCG) method.

Let  $\{\mathcal{P}_{\ell}^N\}_{\ell}$  and  $\{\mathcal{P}_{\ell}^V\}_{\ell}$  be hierarchies of prolongation matrices for the discrete spaces  $N_h$  and  $V_h$  obtained through successive mesh refinement. Next, we define

$$\mathcal{P}_{\ell} = \begin{bmatrix} \mathcal{P}_{\ell}^N & 0 \\ 0 & \mathcal{P}_{\ell}^V \end{bmatrix}.$$

Then, our multigrid preconditioner is a monolithic geometric multigrid constructed using  $\{\mathcal{P}_{\ell}\}_{\ell}$  for the entire system (4.6) or just a standard geometric multigrid constructed with  $\mathcal{P}_{\ell}^N$  for (4.7).

We considered two options for the smoother. First was a standard  $\ell^1$  Gauss-Seidel preconditioner, second was a combination of the first smoother with an additional smoother based on a local non-overlapping Schwarz-type method (similar to the multilevel algorithm discussed in Subsection 4.1). We take coarser level elements as subdomains. An action of the smoother consists of solving a set of independent local saddle point problems which are decoupled via imposing zero boundary conditions on the interfaces between neighboring subdomains. Hence, one smoothing iteration provides an update in  $H(\text{div})$  which is well defined since the local updates match at the subdomain interfaces because they vanish there by construction. We note that this Schwarz smoother (at any given level) does not update the interface values of the previous iterate. This is compensated by additional smoothing in the div-free subspace exploiting the explicit basis for that subspace, namely, the bases of  $H(\text{curl})$  and  $H(\text{Div})$  when  $n = 3$  or 4 correspondingly.

At the coarsest level multigrid preconditioners use the conjugate gradients method with  $\ell^1$  Gauss-Seidel preconditioner.

## 6. NUMERICAL RESULTS

In this section, we present numerical examples showing the convergence of the space-time CFOSLS discretizations and solvers performance for the three types of PDEs considered in this paper.

The convergence of the discretizations for different PDEs,  $n = 3, 4$ , is analyzed in Subsection 6.1, in Tables 4-11. Performance of suggested preconditioners is reported for each problem in the Subsection 6.2, in Tables 12-21. In those tables we report observed iteration counts as well as weak scaling results.

Our implementation is based on MFEM, a scalable C++ library for finite element discretizations developed at Lawrence Livermore National Laboratory [MFEM]. Numerical experiments are conducted on the Coeus HPC cluster of the Portland Institute for Computational Science, which has 128 compute nodes, each with 20 cores and 128 GB RAM, and on the Quartz cluster of the Livermore Computing Center.

**6.1. Discretization.** For all considered PDEs, the space-time domain  $\Omega_T$  is  $(0, 1)^{n+1}$ , and the exact solution is given by

$$u(\mathbf{x}, t) = t^2 e^t \prod_{i=0}^{n-1} \sin\left((3-i)\pi x_i\right)$$

where  $n = 2, 3$ . The right-hand side  $f$  in (2.1) is computed using the exact  $u$ . For the conservation law, the velocity was given by

$$\mathbf{v}(\mathbf{x}, t) = \begin{bmatrix} \sin(\pi x_1) \cos(\pi x_2) \\ -\cos(\pi x_1) \sin(\pi x_2) \end{bmatrix}$$

for  $n = 2$ , and by

$$\mathbf{v}(\mathbf{x}, t) = \begin{bmatrix} \sin(\pi x_1) \cos(\pi x_2) \cos(\pi x_3) \\ -0.5 \cos(\pi x_1) \sin(\pi x_2) \cos(\pi x_3) \\ -0.5 \cos(\pi x_1) \cos(\pi x_2) \sin(\pi x_3) \end{bmatrix}$$

for  $n = 3$ .

We start with an unstructured initial mesh containing 384 tetrahedrons if  $\Omega_T \subseteq \mathbb{R}^3$  or 96 pentatopes if  $\Omega_T \subseteq \mathbb{R}^4$  and refine it uniformly several times.

To study convergence we define the following relative error measurements

$$\mathcal{E}_{L^2}(u) = \frac{\|u_h - u\|}{\|u\|}, \quad \mathcal{E}_{\mathcal{A}}(\boldsymbol{\sigma}, u) = \frac{\|\mathcal{A}(\boldsymbol{\sigma}_h - \boldsymbol{\sigma}, u_h - u)\|}{\|\mathcal{A}(\boldsymbol{\sigma}, u)\|}.$$

In the tables below #dofs are given for system (4.2). For (4.6) number of dofs would be different since different finite element spaces are used.

From these tables we can conclude that practical convergence rates are approaching the expected rates as the mesh is being refined. In particular, for all examples except conservation law, when  $u \in H^1$ , CFOSLS energy error  $\mathcal{E}_{\mathcal{A}}(\boldsymbol{\sigma}, u)$  converges with the first order. Also the error estimates for the scalar unknown  $u$  in  $L^2$  norm are very close

TABLE 4. Convergence rates for heat equation when  $\Omega_T \subseteq \mathbb{R}^3$ .

#dofs	$\mathcal{E}_{L^2}(u)$	rate	$\mathcal{E}_A(\boldsymbol{\sigma}, u)$	rate
631,905	0.0679631	-	0.0760962	-
5,017,793	0.0302238	1.17	0.0380784	1.00
39,993,729	0.0132747	1.19	0.0190443	1.00
319,357,697	0.005677	1.22	0.0095298	1.00

TABLE 5. Convergence rates for heat equation when  $\Omega_T \subseteq \mathbb{R}^4$ .

#dofs	$\mathcal{E}_{L^2}(u)$	rate	$\mathcal{E}_A(\boldsymbol{\sigma}, u)$	rate
1,421,857	0.367486	-	0.216416	-
22,513,729	0.180438	1.03	0.10917	0.99
358,359,169	0.0837263	1.11	0.0548004	0.99

TABLE 6. Convergence rates for wave equation when  $\Omega_T \subseteq \mathbb{R}^3$ .

#dofs	$\mathcal{E}_{L^2}(u)$	rate	$\mathcal{E}_A(\boldsymbol{\sigma}, u)$	rate
631,905	0.126338	-	0.0758716	-
5,017,793	0.0594012	1.09	0.0379642	1.00
39,993,729	0.0305297	0.96	0.0189864	1.00
319,357,697	0.0159308	0.93	0.0094938	1.00

TABLE 7. Convergence rates for wave equation when  $\Omega_T \subseteq \mathbb{R}^4$ .

#dofs	$\mathcal{E}_{L^2}(u)$	rate	$\mathcal{E}_A(\boldsymbol{\sigma}, u)$	rate
1,421,857	0.42149	-	0.215885	-
22,513,729	0.284831	0.57	0.108779	0.99
358,359,169	0.154014	0.89	0.0545992	0.99

to one. We would expect the convergence rates to reach 1.0 for finer discretizations also in the case of conservation law, although the current results clearly show some preasymptotic behavior.

**6.2. Performance.** In the Tables 12- 21 below we use the following notations. In the first column number of MPI processes is given which were used for the computations. The corresponding mesh refinement level is given in the second column. The rows show how the solution time varies with the number of processes for a fixed problem size per process. The rest of the columns show iteration counts and timings in seconds (only for the solution part) in format *#iter (time)* for different preconditioners: BLKDIAG stands for MINRES preconditioned by a block-diagonal preconditioner (first option listed in section 6), MG for PCG preconditioned by a monolithic geometric multigrid with  $l^1$  Gauss-Seidel smoother, and MG+S for PCG preconditioned by a monolithic geometric multigrid with additional Schwarz-type smoother. For all solvers we set the relative tolerance of the preconditioner residual to  $10^{-12}$ . In the

TABLE 8. Convergence rates for conservation law,  $u \in H^1$ , when  $\Omega_T \subseteq \mathbb{R}^3$ .

#dofs	$\mathcal{E}_{L^2}(u)$	rate	$\mathcal{E}_{\mathcal{A}}(\sigma, u)$	rate
631,905	0.0847963	-	0.108649	-
5,017,793	0.0332752	1.35	0.0559334	0.96
39,993,729	0.011964	1.48	0.0283935	0.98
319,357,697	0.0041479	1.53	0.0143018	0.99

TABLE 9. Convergence rates for conservation law,  $u \in H^1$ , when  $\Omega_T \subseteq \mathbb{R}^4$ .

#dofs	$\mathcal{E}_{L^2}(u)$	rate	$\mathcal{E}_{\mathcal{A}}(\sigma, u)$	rate
1,421,857	0.417028	-	0.440867	-
22,513,729	0.25144	0.73	0.260322	0.76
358,359,169	0.130435	0.95	0.147036	0.82

TABLE 10. Convergence rates for conservation law,  $u \in L^2$ , when  $\Omega_T \subseteq \mathbb{R}^3$ .

#dofs	$\mathcal{E}_{L^2}(u)$	rate	$\mathcal{E}_{\mathcal{A}}(\sigma, u)$	rate
595,968	0.137152	-	0.273584	-
4,743,168	0.0657198	1.06	0.136863	1.0
37,847,040	0.0312554	1.07	0.0684406	1.0
302,383,104	0.015107	1.05	0.0342215	1.0

TABLE 11. Convergence rates for conservation law,  $u \in L^2$ , when  $\Omega_T \subseteq \mathbb{R}^4$ .

#dofs	$\mathcal{E}_{L^2}(u)$	rate	$\mathcal{E}_{\mathcal{A}}(\sigma, u)$	rate
1,400,832	0.376594	-	0.36774	-
22,216,704	0.205292	0.88	0.184801	0.99
353,894,400	0.103456	0.99	0.09253	1.0

results within each table we keep the fixed size of the coarsest problem for the multi-grid preconditioners, thus increasing number of levels when the mesh is refined more times.

First, we want to show that the geometric multigrid shows excellent iteration counts for Laplace equation, for which multigrid is known to give optimal convergence rates. Notice, that we can write Laplace equation in the form (2.1) by considering one of the dimensions as “time” and the rest coordinates as “space”, and then defining  $\mathcal{L}(u)$  as

$$\mathcal{L}(u) = \begin{bmatrix} \nabla_x u \\ \frac{\partial u}{\partial t} \end{bmatrix}.$$

This will lead to a mixed formulation of Laplace equation in the least squares setting. The discrete problem will then take the form of (4.6).

As one can see from the results in Tables 12 and 13, for the Laplace equation geometric multigrid gives ideal flat iteration counts in  $\mathbb{R}^3$  and almost flat in  $\mathbb{R}^4$ . Of course, it is natural to expect that we cannot get those for the time-dependent PDEs.



TABLE 12. Performance of linear solvers for Laplace equation in CFOSLS formulation, when  $\Omega_T \subseteq \mathbb{R}^3$ .

#proc	#refs	MG	MG+S
5	1	10 (0.58)	9 (0.98)
40	2	10 (0.34)	9 (1.01)
320	3	10 (0.49)	9 (1.58)
2560	4	10 (0.77)	10 (2.95)

TABLE 13. Performance of linear solvers for Laplace equation (in CFOSLS formulation, just for comparison), when  $\Omega_T \subseteq \mathbb{R}^4$ .

#proc	#refs	MG	MG+S
2	1	17 (6.7)	16 (11.5)
32	2	20 (5.9)	19 (16.7)
512	3	22 (9.3)	21 (28.0)

One of the reasons for that is the time-related anisotropy which is built-in in the considered space-time formulations.

We also notice a relatively good weak scaling of the solver. Comparing the last two columns we see that the implementation of Schwarz-type smoother is quite expensive and worsens the scaling while giving only a tiny reduction of the iteration count.

TABLE 14. Performance of linear solvers for heat equation, when  $\Omega_T \subseteq \mathbb{R}^3$ .

#proc	#refs	BLKDIAG	MG	MG+S
5	1	266 (2.7)	36 (3.0)	36 (4.5)
40	2	376 (4.7)	55 (2.2)	55 (6.0)
320	3	518 (10.0)	85 (4.2)	84 (13.0)
2560	4	747 (25.3)	111 (9.1)	110 (29.5)

TABLE 15. Performance of linear solvers for heat equation, when  $\Omega_T \subseteq \mathbb{R}^4$ .

#proc	#refs	BLKDIAG	MG	MG+S
2	1	259 (15.9)	17 (7.4)	16 (11.9)
32	2	436 (32.7)	33 (9.1)	33 (29.1)
512	3	867 (116.6)	57 (23.2)	57 (70.3)

As the preliminary results given in Tables14-21 show, geometric multigrid is obviously superior to algebraic block-diagonal preconditioners, both in terms of iteration count and timings. For algebraic block-diagonal preconditioners we observe linear growth of iteration counts with respect to discretization step for all problems.

The multigrid iteration counts are also not independent from  $h$  and exhibit a similar (but smaller) growth. This shows that there still remains the question of constructing

TABLE 16. Performance of linear solvers for wave equation, when  $\Omega_T \subseteq \mathbb{R}^3$ .

#proc	#refs	BLKDIAG	MG	MG+S
5	1	1058 (10.1)	58 (4.7)	59 (7.4)
40	2	2053 (23.5)	104 (4.3)	105 (11.5)
320	3	3922 (67.2)	147 (6.9)	145 (22.0)
2560	4	7749 (216.2)	242 (18.8)	233 (59.7)

TABLE 17. Performance of linear solvers for wave equation, when  $\Omega_T \subseteq \mathbb{R}^4$ .

#proc	#refs	BLKDIAG	MG	MG+S
2	1	463 (27.9)	28 (14.1)	26 (20.8)
32	2	927 (75.3)	52 (15.3)	49 (41.7)
512	3	1815 (224.0)	90 (37.2)	87 (110.7)

TABLE 18. Performance of linear solvers for conservation law,  $u \in H^1$ , when  $\Omega_T \subseteq \mathbb{R}^3$ .

#proc	#refs	BLKDIAG	MG	MG+S
5	1	448 (4.5)	41 (3.2)	39 (5.1)
40	2	777 (9.7)	71 (3.0)	63 (7.1)
320	3	1397 (26.7)	115 (5.4)	109 (16.7)
2560	4	2445 (78.7)	199 (15.6)	186 (47.9)

TABLE 19. Performance of linear solvers for conservation law,  $u \in H^1$ , when  $\Omega_T \subseteq \mathbb{R}^4$ .

#proc	#refs	BLKDIAG	MG	MG+S
2	1	273 (17.4)	31 (13.1)	30 (24.0)
32	2	597 (45.9)	56 (16.1)	56 (50.1)
512	3	1233(160)	91 (38.1)	91 (119.1)

the optimal preconditioners, e.g by taking care of the implicit causality and time anisotropy via more sophisticated smoother constructions.

As in the case of Laplace equation, the improved smoother construction (MG+S column) doesn't improve the overall result, although a certain iteration count reduction might be noticed for  $n = 4$ . But in the end, the cost of the current smoother implementation increases the total solution time thus making the improved smoother impractical so far.

## 7. CONCLUSION AND OUTLOOK

In this paper we presented a comprehensive computational study for the approximation properties of the recently designed 4D finite elements used to discretize three classes of time-dependent PDEs (heat equation, transport equation (scalar conservation law), and wave equation) and compared the obtained results with 3D case. We

TABLE 20. Performance of linear solvers for conservation law,  $u \in L^2$ , when  $\Omega_T \subseteq \mathbb{R}^3$ .

#proc	#refs	BLKDIAG	MG	MG+S
5	1	2424 (19.4)	80 (4.1)	75 (5.5)
40	2	4442 (42.8)	144 (3.0)	132 (7.6)
320	3	8560 (116.2)	220 (5.7)	201 (15.8)
2560	4	16185 (359.4)	329 (13.5)	279 (40.5)

TABLE 21. Performance of linear solvers for conservation law,  $u \in L^2$ , when  $\Omega_T \subseteq \mathbb{R}^4$ .

#proc	#refs	BLKDIAG	MG	MG+S
2	1	2948 (162.4)	129 (56.6)	111 (59.8)
32	2	5817 (439.6)	220 (44.3)	202 (102.0)
512	3	11104 (1185.4)	341 (98.4)	321 (229.8)

used a unified constrained first-order system least square framework, CFOSLS, for discretizing the PDEs in the combined space-time domain. The performed numerical study confirmed the expected maximal rate of approximation of the underlined spaces although we only can theoretically guarantee that only for certain types of pairs of spaces (based on the result proven in [AV14]). We also investigated a div-free formulation, which is feasible due to the available finite element spaces for the whole 4D de Rham complex. For this formulation we exploited the space that equals the null-space of the 4D divergence operator and the respective finite element counterparts (known in the general  $n$ -dimensional case from the exterior calculus theory, see [AFW10]); its specific 4D finite element counterparts were considered in detail in [GNV17].

Additionally, we presented our first attempts for designing preconditioners, both block and monolithic ones, employing multigrid. As one might expect, due to lack of full ellipticity of the FOSLS functionals at hand, the results are generally not mesh independent. This topic requires further study, exploiting in some form the fact that the underlying PDEs have time direction (i.e., causality) which means that specialized smoothers and appropriate semi-coarsening can be developed. In addition, a parallel-in-time strategy based on a time-stepping scheme for space-time slabs can be exploited, e.g., in a form of a parareal iteration. This should improve the convergence of the overall iterative scheme (as demonstrated previously, e.g., by [XBraid]).

A major on-going direction for further research, which is a project on its own, is to design efficient and scalable AMR (adaptive mesh refinement) algorithms for the 4D case. That will make a more convincing argument for the usefulness of the combined space-time discretization since the solution will then follow more naturally the underlined physics while the FOSLS functional provides a natural tool to determine which elements are to be refined.

## REFERENCES

- [AFW10] D. N. ARNOLD, R. S. FALK, AND R. WINTHER, “*Finite element exterior calculus: from Hodge theory to numerical stability*”, Bull. Amer. Math. Soc. (N.S.) 2010, 47:281-353.
- [AV14] J. H. ADLER, P. S. VASSILEVSKI, “*Error Analysis for Constrained First-Order System Least-Squares Finite-Element Methods*,” SIAM J. Sci. Comput., 36(3), A1071-A1088, 2014.
- [B08] M. BEHR, “*Simplex Space-Time Meshes in Finite Element Simulations*,” Int. J. Numer. Meth. Fluids, 57:1421-1434, 2008.
- [BFKY12] A. BAKER, R. FALGOUT, T. KOLEV, AND U. YANG, “*Scaling hypre’s Multigrid Solvers to 100,000 Cores*”, High Performance Scientific Computing: Algorithms and Applications, Springer, 261-279, 2012.
- [BFKY11] A. H. BAKER, R. D. FALGOUT, T. V. KOLEV, AND U. M. YANG “*Multigrid Smoothers for Ultraparallel Computing*”, SIAM J. Sci. Comput., 33(5):2864-2887, 2011.
- [BVZ17] R. E. BANK, P. S. VASSILEVSKI, AND L. T. ZIKATANOV, “*Arbitrary Dimension Convection-Diffusion Schemes for Space-Time Discretizations*,” Journal of Computational and Applied Mathematics 310, pp. 19-31, 2017.
- [DGNS17] L. DEMKOWICZ, S. NAGARAJ, J. GOPALAKRISHNAN, AND P. SEPÚLVEDA. “*A spacetime DPG method for Schrödinger equation*. SIAM Journal on Numerical Analysis, Vol. 55, No. 4, pp. 1740-1759, 2017
- [EV98] L.C. EVANS, “*Partial Differential Equations*,” Providence: American Mathematical Society, ISBN 0-8218-0772-2.
- [FFKMSV17] R. D. FALGOUT, S. FRIEDHOFF, Tz. V. KOLEV, S. P. MACLACHLAN, J. B. SCHRODER, AND S. VANDERWALLE “*Multigrid methods with space-time concurrency*.” Computing and Visualization in Science, 18, Issue 4-5, pp 123-143, 2017.
- [GA14] M. J. GANDER, “*50 years of time parallel time integration , in Multiple Shooting and Time Domain Decomposition Methods*”, Springer, Berlin, 2014.
- [GT14] G. N. GATICA, “*A simple introduction to the mixed finite element method: Theory and applications*,” Springer Briefs in Mathematics. Cham: Springer. MR3157367, 2014.
- [GNV17] J. GOPALAKRISHNAN, M. NEUMUELLER, AND P. S. VASSILEVSKI, “*The Auxiliary Space Preconditioner for the de Rham Complex*, (submitted). Available as Lawrence Livermore National Laboratory Technical Report LLNL-JRNL-740551, October 25, 2017.
- [GS18] J. GOPALAKRISHNAN, AND P. SEPULVEDA “*A spacetime DPG method for acoustic waves*,” Manuscript submitted for publication, 2018.
- [HH88] T. HUGHES, AND G. HULBERT “*Space-time finite element methods for elastodynamics: formulations and error estimates* Comp. Methods in Applied Mechanics and Engineering, 66, 339-363, 1988.
- [HY02] V. E. HENSON AND U. M. YANG, “*BoomerAMG: A parallel algebraic multigrid solver and preconditioner*”, Applied Numerical Mathematics 2002; 41(1):155-177.
- [KB] E. KARABELAS, M. NEUMUELLER. “*Generating admissible space-time meshes for moving domains in  $d + 1$ -dimensions*,” <https://arxiv.org/abs/1505.03973>.
- [LMT01] J. L. LIONS , Y. MADAY , AND G. TURICINI, “*A parareal in time discretization of PDEs*,” C.R. Acad. Sci. Paris, Serie I, 332, pp. 661-668, 2001.
- [MFEM] “*MFEM: Modular finite element methods*”, [mfem.org](http://mfem.org) and <https://github.com/mfem/mfem> (developer’s branches).
- [MH97] A. MASUD AND T.J.R. HUGHES, “*A space-time Galerkin/least-squares finite element formulation of the Navier-Stokes equations for moving domain problems*.” Comput. Methods Appl. Mech. Engrg. , 146:91-126, 1997
- [N13] M. NEUMUELLER, “*Space-Time Methods: Fast Solvers and Applications*,” Dissertation, Graz University of Technology, June 2013.
- [NBP11] J. NAM, M. BEHR, AND M. PASQUALI, “*Space-time least-squares finite element method for convection-reaction system with transformed variables*.” Comput. Methods Appl. Mech. Engrg. , 200:2562-2576, 2011

- [NVV16] M. NEUMUELLER, P. S. VASSILEVSKI, AND U. VILLA, “*Space-time CFOSLS methods with AMGe upscaling*,” Available as Lawrence Livermore National Laboratory Technical Report LLNL-CONF-683318, February 18, 2016.
- [PS75] C. C. PAIGE, AND M. A. SAUNDERS, “*Solution of Sparse Indefinite Systems of Linear Equations*”, SIAM Journal on Numerical Analysis 1975; 12(4):617-629.
- [S18] P. SEPULVEDA, “*Spacetime Numerical Techniques for the Wave and Schrödinger Equations*,” (2018). Dissertations and Theses. Paper 4206.
- [T06] T.E. TEZDUYAR. “*Interface-tracking and interface-capturing techniques for finite element computation of moving boundaries and interfaces*,” Comput. Methods Appl. Mech. Engrg. , 195:2983-3000, 2006.
- [T09] S. THITE, “*Adaptive spacetime meshing for discontinuous Galerkin methods*.” Comput. Geom. , 42:20-44, 2009.
- [TS04] T.E. TEZDUYAR, AND S. SATHE, “*Enhanced-discretization space-time technique (ED-STT)*.” Comput. Methods Appl. Mech. Engrg. , 193:1385-1401, 2004.
- [V08] P. S. VASSILEVSKI, “*Multilevel Block-Factorization Preconditioners. Matrix-based Analysis and Algorithms for Solving Finite Element Equations*,” Springer, New York, 2008.
- [XBraid] “*XBraid, Parallel multigrid in time*”, <http://llnl.gov/casc/xbraid>.

PORTLAND STATE UNIVERSITY, PO Box 751 (MTH), PORTLAND, OR 97201, (KVORONIN@PDX.EDU)

CENTER FOR APPLIED SCIENTIFIC COMPUTING, LLNL, MAIL STOP L-561, LIVERMORE, CA 94550, LEE1029@LLNL.GOV

INSTITUTE OF COMPUTATIONAL MATHEMATICS, JOHANNES KEPLER UNIVERSITY, ALTENBERGER STR., A-4040 LINZ, AUSTRIA, (NEUMUELLER@NUMA.UNI-LINZ.AC.AT)

PORTLAND STATE UNIVERSITY, PO Box 751 (MTH), PORTLAND, OR 97201, (SPAULINA@PDX.EDU)

PORTLAND STATE UNIVERSITY, PO Box 751 (MTH), PORTLAND, OR 97201, (PANAYOT@PDX.EDU)

CENTER FOR APPLIED SCIENTIFIC COMPUTING, LLNL, MAIL STOP L-561, LIVERMORE, CA 94550, (PANAYOT@LLNL.GOV)