Università degli Studi di Pavia
Università della Svizzera Italiana

Joint PhD program in Computational Mathematics
and Decision Sciences – XXXVI cycle

Isogeometric discretizations of evolutionary equations and fast solvers

Supervisor:
Prof. Giancarlo SANGALLI
Co-supervisor:
Ph.D. Andrea BRESSAN

Ph.D. Candidate:
Alen KUSHOVA

Academic years 2020-2023

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Partial differential equations (PDE) arise throughout physics, engineering, and natural sciences, typically modelling problems where the exact solutions are impossible to calculate. Numerical analysis is the branch of mathematics that addresses such continuous problems through numeric approximation. It involves developing methods that provide approximate yet accurate numeric solutions. Among the numerical methods designed for solving PDEs, the finite element method (FEM) stands as the state of the art. However, a primary challenge with FEM is the need to approximate the computational domain, usually described by a computer-aided design (CAD) file, for instance, through triangulation. This approximation of the geometry systematically introduces an error source, and refining the triangulation becomes a computationally expensive process.

To address this disparity between FEM and CAD, isogeometric analysis (IgA) has been introduced by Hughes et al. in the seminal paper [63]; see also the book [29]. IgA serves as an extension of FEM, significantly improving interoperability between CAD and solvers for PDEs. Indeed, IgA is based on the adoption of the same functions that describe the CAD geometry (usually B-splines and Non-Uniform Rational B-splines, that is NURBS) to construct and represent the approximated solution of the PDE. Thanks to an exact representation of the computational domain, the error due to the approximation of the geometry is thus eliminated.

From a more theoretical perspective, the mathematical analysis of isogeometric methods, on one hand, borrows from classical spline theory, for instance see [34, 95], but on the other hand it has stimulated new developments and interesting new open questions. The theory of h-refinement of isogeometric spaces and methods, that is, the study of convergence that is achieved by refining the mesh, was developed in [4] and [14], the latter covering anisotropic refinements. The refinement of the spline spaces can be achieved not only by the classical p-refinement (order elevation) and h-refinement (knot insertion) procedures, already present in FEM, but also by the new k-refinement, that is order elevation followed by knot insertion, see [63]. The k-refinement leads to possibilities previously unavailable in FEM, for instance the direct discretization of high order PDEs, the use of continuous stresses and the development of collocation methods. In addition, the higher regularity of the basis functions brings several advantages: higher accuracy per degree-of-freedom [41], robust approximation of non-smooth functions [25], better approximation of the spectrum [30].

Isogeometric methods have been used and tested on a variety of problems of engineering interest. There is indeed a large engineering literature showing the beneficial effects of higher regularity in several practical problems. We report here a few references. Isogeometric methods have been proposed for flow simulations in the presence of turbulence [5, 9, 8], with conservative schemes [23, 43, 44], and for higher-order models such as Cahn-Hilliard [54, 55]; for plate and shell analysis [66, 65, 15, 11], where, in particular, the use of Kirchhoff models is made possible by the higher regularity of splines; for nonlinear solid mechanics [40, 74, 39, 3], shape optimization [110], fluid-structure interaction [6, 7]; for electromagnetic problems [26, 31, 109, 108, 27], and more recently for plasma physics and magnetohydrodynamics [68, 62]. Indeed, in the latter ones, the authors introduce energy conservative isogeometric discretizations specifically for the long-term study and simulation of particle dynamics in plasma, as well as for the electrodynamics of phenomena resulting from particle motion. In particular, in [62], high order B-spline spaces are used together with implicit time splitting schemes, which could also be of high order. However, a numerical analysis of the method is not easily attainable and remains an open problem.

Moreover, in the recent theory of space-time methods, isogeometric analysis allows high order discretizations simultaneously in space and time. It has been applied to evolutionary equations of the parabolic type [71, 72], flow simulations [103, 102], linear and non-linear elastodynamics [92] and wave propagation [46]. The modern theory of linear solvers, and in particular the setup of preconditioners is under development. This is an important aim from the viewpoint of real-world application of isogeometric methods, and also a mathematical challenge, particularly with regard to the high-degree high-regularity case. Some of the modern approaches for finite elements have been extended to the isogeometric context. As regards space-time discretizations, in particular for parabolic problems, multigrid solvers have been proposed in [50, 60], while low rank approximations has been investigated in [82]. Sylvester type preconditioners based on Fast Diagonalization (FD) techniques have been introduced in [85, 75].

Finally, well posed space-time variational formulations has been recently introduced for the linear Schrödinger equation. In [36], the authors propose two variational formulations that are proved to be well posed in one dimensional space domains: a strong formulation, with no relaxation of the original equation, and an ultraweak formulation, that transfers all derivatives onto test functions. The proposed discretization for the ultraweak form is based on a discontinuous Petrov-Galerkin (DPG) method, and B-spline basis functions. In [56] a space–time ultraweak Trefftz discontinuous Galerkin (DG) method for the Schrödinger equation has been proposed, proving well-posedness and stability of the method, and optimal high-order h-convergence error estimates in a skeleton norm, for one and two dimensional cases. Recently, in [57], Hain and Urban proposed a well posed space–time ultraweak variational formulation that uses high order B-splines with maximum regularity and can be extended to the isogeometric analysis framework.

Main contributions and structure of the thesis

The aim of this thesis is to contribute to the development of recent interesting numerical methods in the context of isogeometric analysis: firstly, energy conservative discretizations for initial-boundary value problems in mixed form, here proposed for the wave equation, and secondly, fast preconditioners for solving linear systems arising from space-time discretizations, here proposed for the heat – and Schrödinger – equation.

The initial segment of this thesis is intricately linked to the aforementioned open topic of the numerical analysis of the methods introduced in [68, 62]. Instead of Vlasov-Maxwell's equations, we consider a simplified model problem of hyperbolic PDEs, specifically the wave equation in mixed form. We introduce, for our model problem, an isogeometric semi-discretization in space, building upon the methodologies outlined in [68] for Maxwell's equations. In particular, we employed B-splines discrete spaces forming a De Rham complex along with suitable commutative projectors. The semi-discretization is coupled with a Crank-Nicolson time stepping scheme, and the fully discrete system is proved to be energy conservative. Our main contribution lies in the development of a numerical analysis for the convergence of the method, firstly for rigorous assumptions on the projections, and then for more relaxed and practical conditions. The theoretical convergence analysis is complemented by numerical results, and the numerical examples confirm the theoretically shown energy conservation property of the proposed approach.

As mentioned above, the second part of this thesis is devoted to fast solvers for space-time discretizations of evolutionary equations in the framework of isogeometric analysis. We start with a review of recently proposed preconditioners for the heat equation, see [85, 75]. These preconditioners are represented by a suitable sum of Kronecker products of matrices. The core idea for a fast application, is the factorization of univariate pencils in the Kronecker structure, that makes the computational cost of setup and application of the preconditioners very appealing. Indeed, the cost of the setup is $O(N_{dof})$ FLoating Point Operations (FLOPs), while their application cost is $O(N_{dof}^{(d+2)/(d+1)})$ FLOPs in d-dimensions, with $d > 1$. Unfortunately, the fast diagonalization technique directly applied in time direction results unstable. However, a simultaneous diagonalization can be achieved up to a low rank term, typically related to the final degree of freedom, leading to *ad-hoc* stable factorizations like the arrow-head structure. Alongside the preconditioners mentioned above, we propose a third approach relying on FD method and on Sherman-Morrison formula, in order to deal with the low rank term. The computational cost of setup and application of this new preconditioner is equivalent to the previous ones, resulting in an competitive approach.

Lastly, we extend to Schrödinger type equations the ideas developed for the heat equation. First of all, we extend the well posedness of the strong variational formulation of [36] to smoothly parametrized isogeometric domains. We derive a well posed space-time isogeometric Petrov-Galerkin discretization, that is essentially a Galerkin approximation of the space-time least squares variational formulation of the Schrödinger equation. We compare our discretization to the ultraweak space-time discretization of [57]. The two discrete operators have essentially the same Kronecker structure between space and time, thus we proposed a preconditioner for the least squares problem that can easily fit in the ultraweak framework. Our preconditioner is stable and leads to a fast solver for the problem modeled in the parametric domain. Indeed, analogously to the preconditioning techniques mentioned for the heat equation, the computational cost of the setup is $O(N_{dof})$ FLOPs, while its application cost is $O(N_{dof}^{(d+2)/(d+1)})$ FLOPs, again for $d > 1$. An extension of the preconditioner to parametrized geometries is yet an open problem, and will require future investigations. More in general, both for heat and Schrödinger equations, the extension of the preconditioners to multi-patch geometries is yet to be understood, and hopefully this thesis may contribute to future developments in such direction.

In the remaining we give an overview of the structure of this thesis.

Chapter 2 We analyze the wave equation in mixed form, with periodic and/or Dirichlet homogeneous boundary conditions, and nonconstant coefficients that depend on the spatial variable. For the discretization, the weak form of the second equation is replaced by a strong form, written in terms of a projection operator. The system of equations is discretized with B-splines forming a De Rham complex along with suitable commutative projectors for the approximation of the second equation. The discrete scheme is energy conservative when discretized in time with a conservative method such as Crank-Nicolson. We propose a convergence analysis of the method to study the dependence with respect to the mesh size h, with focus on the consistency error. Numerical results show optimal convergence of the error in energy norm, and a relative error in energy conservation for long-time simulations of the order of machine precision.

Chapter 3 We review preconditioning techniques based on fast diagonalization methods for space-time isogeometric discretization of the heat equation. Three formulation are considered: the Galerkin approach, a Galerkin L^2 least squares form and a continuous least squares approach. For each formulation, the heat differential operator is written as a sum of terms that are Kronecker products of univariate operators. These are used to speed-up the application of the operator in iterative solvers and to construct a suitable preconditioner. Contrary to the fast diagonalization technique for the Laplace equation, where all univariate operators acting on the same direction can be simultaneously diagonalized, in the case of the heat equation this is not possible. Luckily, this can be done up to an additional term that has low rank, allowing for the utilization of arrow-head like factorization or inversion by Sherman-Morrison formula. The proposed preconditioners work extremely well on the parametric domain and, when the domain is parametrized or when the equation coefficients are not constant, they can be adapted and retain good performance characteristics.

Chapter 4 We present a space-time least squares isogeometric discretization of the Schrödinger equation and propose a preconditioner for the arising linear system in the parametric domain. Exploiting the tensor product structure of the basis functions, the preconditioner is written as the sum of Kronecker products of matrices. Thanks to an extension to the Fast Diagonalization method, the application of the preconditioner is efficient and robust w.r.t. the polynomial degree of the spline space. The time required for the application is almost proportional to the number of degrees-of-freedoms, for a serial execution.

Chapter 2

Wave equation

The purpose of this chapter is to provide a numerical analysis of discretization schemes applied to the wave equation in mixed form, with nonconstant coefficients, in which one of the variational equations is replaced by a suitably projected equation within the discrete space, introducing a consistency error. The motivation for our work stems from analogous discretization techniques recently proposed for Maxwell's equations, specifically in the long-term study and simulation of particle dynamics in plasma, as well as for the electrodynamics of phenomena resulting from particle motion [68, 62]. The simplified model problem of the wave equation allows us to analyze the proposed method, paying particular attention to the introduced consistency error.

The wave equation has been studied extensively in theory and numerical approximations. Existence and uniqueness results are well known in literature [32, 45]. The present work considers the wave equation as a first order hyperbolic system, introducing velocity and pressure fields following approaches found in prior works [10, 53, 18]. For the resulting variational formulations in mixed form it is acknowledged that the successful approximation of the problem requires suitable compatibility conditions within the involved discrete spaces [17]. In general, mixed methods for the wave equation consider the discretization of vector fields in some $\mathbf{H}(\mathrm{div})$-conforming spaces while scalar fields in some L^2-conforming spaces. To meet these conditions, we construct discrete spaces that adhere to the De Rham complex of exterior calculus [1], within the framework of isogeometric analysis [63].

Briefly, isogeometric analysis (IgA) utilizes spline functions, or their generalizations, for both representing the computational domain and approximating solutions to the partial differential equation that models the relevant problem. This approach aims to streamline the interoperability between computer-aided design and numerical simulations. Moreover, IgA derives advantages from the approximation properties of splines, where their high continuity contributes to enhanced accuracy compared to C^0 piecewise polynomials. This characteristic is well-documented in literature, [41, 20, 94]. As regards the De Rham complex for tensor-product B-splines, initially introduced and analyzed by [24], it has found wide applications in Galerkin approximation of Maxwell's equations [91], and divergence-free methods for incompressible fluid flow [42, 43, 44].

In this chapter, together with the discrete B-spline spaces of the De Rham complex, we build quasi-interpolant projections that commute with the divergence operator. Various families of quasi-interpolant operators have been defined and presented in the context of spline approximation, [35, 78]. Here, analogously to what has been done in [62, 69], we implement a local quasi-interpolant operator, as presented in [73], wherein explicit formulas are provided for computing the coefficients of the projection. These explicit formulas entail pointwise evaluation of the function to be projected. Employing this operator, we project the second equation of the variational formulation, reducing the problem to a single equation for the velocity. Subsequently, the pressure field is determined after the computation of velocity. The resulting semi-discrete problem preserves the total energy of the system, as for a standard Galerkin method.

Given our interest in preserving energy for long time simulations, the choice of an energy-preserving method in time is mandatory. We adopt Crank-Nicolson method which is a second order energy conservative time discretization. Other time discretization methods that conserve energy can be applied with appropriate modifications to the fully discrete system. The numerical tests show a relative error in energy conservation for long time simulations of the order of machine precision.

Finally, error estimates in energy norm for Galerkin discretizations of mixed formulations are well known in literature [17], and in particular for the wave equation in mixed form [18]. Here, as the main contribution of this work, we present an error convergence analysis for our method with a generic family of projections, focusing on the consistency error. By assuming good approximation properties of the projections, together with its formal adjoint operator, we prove high-order convergence with respect to the mesh size h. Since the specific quasi-interpolant that we tested numerically does not fulfill the requirements on the adjoint operator, we relax the assumption and prove that the numerical scheme converges linearly with respect to the mesh size. Numerical simulations confirm optimal high-order convergence for the implemented quasi-interpolant, similar to a standard Galerkin scheme, which suggests that an improvement of our theoretical results can be achieved. The global convergence of the scheme is of second order, as expected from the use of Crank-Nicolson method.

The chapter is organized as follows. In Section 2.1 we present the model problem. In Section 2.2 we provide a brief overview of the isogeometric framework. Section 2.3 introduces the discretization spaces along with commutative projections and presents the discrete problem. In Section 2.4 we present the a priori error estimates in energy norm. Implementation details are covered in Section 2.5, with numerical results presented in Section 2.6, and finally, in Section 2.7 we draw our conclusions.

2.1 Model problem

We start presenting the wave equation and its formulation as a first order hyperbolic system.

2.1.1 Strong formulation and boundary conditions

Let $\Omega \subset \mathbb{R}^d$ be our domain, with $d = 2, 3$, and let the time domain interval $I = [0, T]$ with $T > 0$. We consider our model problem, the second-order scalar wave equation

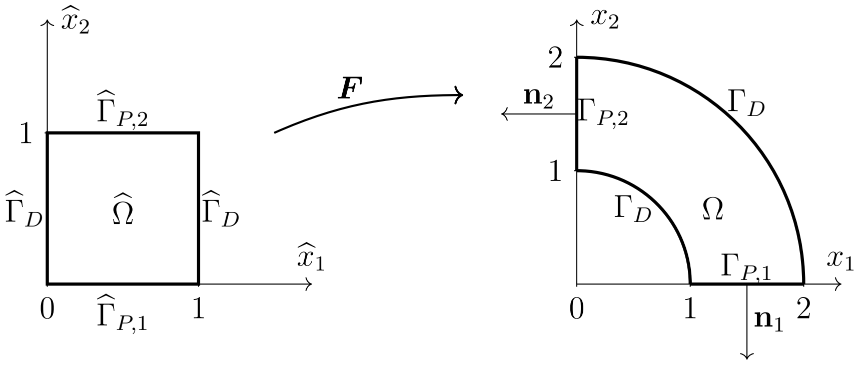


Figure 2.1. Isogeometric parameterization for the quarter of a bidimensional ring Ω with mixed Dirichlet and periodic boundaries.

with space dependent coefficients, that reads like

$$\begin{cases} u_{tt} - \operatorname{div}(c^2 \boldsymbol{\nabla} u) = 0 & \text{in } \Omega \times I, \\ u(\cdot, 0) = u_0, \quad u_t(\cdot, 0) = u_1 & \text{in } \Omega, \end{cases} \tag{2.1.1}$$

with the subscript $\cdot_t$ indicating the partial derivative with respect to time, u_0 and u_1 are the initial conditions, and the coefficient c is a positive, uniformly bounded and smooth scalar field in Ω. The problem has to be completed with boundary conditions, that we will detail below. Following the idea in [53], by introducing the new variables, $\mathbf{v} = c\boldsymbol{\nabla} u$ and $\phi = u_t$, we rewrite (2.1.1) as a first order hyperbolic system, in the form

$$\begin{cases} \mathbf{v}_t = c\boldsymbol{\nabla}\phi & \text{in } \Omega \times I, \\ \phi_t = \operatorname{div}(c\mathbf{v}) & \text{in } \Omega \times I, \\ \mathbf{v}(\cdot, 0) = c\boldsymbol{\nabla} u_0, \quad \phi(\cdot, 0) = u_1 & \text{in } \Omega. \end{cases} \tag{2.1.2}$$

We will discretize the problem with an isogeometric method, for which we will assume that the domain is given by a parameterization of the form $\Omega = \boldsymbol{F}(\widehat{\Omega})$, where $\widehat{\Omega} = [0, 1]^d$ is the parametric domain, and $\boldsymbol{F}$ is the isogeometric map to be detailed in Section 2.2. We split the boundary of Ω in two parts, with mutually disjoint interiors, denoted $\Gamma_D = \boldsymbol{F}(\widehat{\Gamma}_D)$ and $\Gamma_P = \boldsymbol{F}(\widehat{\Gamma}_P)$, respectively corresponding to Dirichlet and periodic boundary sides. Moreover, for simplicity we assume that periodicity occurs only in the last parametric direction, and split the periodic boundary into two parts, $\Gamma_{P,1}$ and $\Gamma_{P,2}$, such that $\Gamma_P = \Gamma_{P,1} \cup \Gamma_{P,2}$, where $\Gamma_{P,i} = \boldsymbol{F}(\widehat{\Gamma}_{P,i})$ for $i = 1, 2$ with $\widehat{\Gamma}_{P,1} = [0, 1]^{d-1} \times \{0\}$ and $\widehat{\Gamma}_{P,2} = [0, 1]^{d-1} \times \{1\}$. An illustration of the isogeometric map and the split of the boundary is given in Figure 2.1.

To impose the boundary conditions, for scalar fields we introduce the trace operator $\gamma : H^1(\Omega) \to H^{\frac{1}{2}}(\partial\Omega)$, $\gamma : u \mapsto u|_{\partial\Omega}$, and for vector fields the normal trace operator $\gamma_n : H(\operatorname{div}; \Omega) \to H^{-\frac{1}{2}}(\partial\Omega)$, $\gamma_n : \mathbf{v} \mapsto \mathbf{v} \cdot \mathbf{n}$, with $\mathbf{n}$ the outgoing unit normal at $\partial\Omega$. In what follows, we make the assumptions that g does not depend on the time t, and c is periodic, in the sense that its pull-back $\widehat{c} = c \circ \boldsymbol{F}$, is periodic in the last parametric direction of $\widehat{\Omega}$, that is, it satisfies periodicity conditions on $\widehat{\Gamma}_P$. Thus, Dirichlet boundary condition reads as

$$\gamma(u) = g \quad \text{on } \Gamma_D \times I,$$

while periodic boundary conditions are such that

$$\gamma(u), \gamma(u_t), \text{ and } \gamma_n(c\nabla u) \text{ are periodic on } \Gamma_P, \text{ for all } t \in I.$$

Therefore, the Dirichlet and periodic boundary conditions for the first order system (2.1.2) read as

$$\begin{aligned}
&\phi = 0 \quad \text{on } \Gamma_D \times I, \\
&\gamma_n(\mathbf{v}),\ \gamma(\phi) \text{ are periodic on } \Gamma_P \text{ for all } t \in I.
\end{aligned}$$

We remind that existence and uniqueness results for the solution u are well known, see [45, Section 7.2.2], while the regularity of the solution will depend on the regularity of the initial conditions [45, Section 7.2.3].

2.1.2 Weak formulation and conservation of energy

Let us define the following Hilbert spaces over $\Omega \subset \mathbb{R}^d$. $L^2(\Omega)$ is the usual Hilbert space of square integrable functions, endowed with the classical L^2-norm $\|\cdot\|_{L^2(\Omega)}$. By $\mathbf{L}^2(\Omega)$ we denote its vectorial counterpart. The Hilbert spaces $H^k(\Omega)$ denote the functions in $L^2(\Omega)$ such that their kth-order derivatives also belong to $L^2(\Omega)$, and their vectorial counterparts will be denoted by $\mathbf{H}^k(\Omega)$. We define

$$\begin{aligned}
\mathbf{H}(c, \mathrm{div}; \Omega) &:= \{\mathbf{v} \in \mathbf{L}^2(\Omega) : \mathrm{div}\ (c\mathbf{v}) \in L^2(\Omega)\}, \\
\mathbf{H}_P(c, \mathrm{div}; \Omega) &:= \{\mathbf{v} \in \mathbf{H}(c, \mathrm{div}; \Omega) : \gamma_n(c\mathbf{v}) \text{ is periodic on } \Gamma_P\}.
\end{aligned}$$

We will also make use of the space $\mathbf{H}^k(\mathrm{div}; \Omega)$, the space of functions in $\mathbf{H}^k(\Omega)$ such that their divergence belongs to $H^k(\Omega)$. We can then write the weak formulation of (2.1.2), that is: find $\mathbf{v} \in \mathbf{H}_P(c, \mathrm{div}; \Omega)$ and $\phi \in L^2(\Omega)$ such that the following equations hold

$$\int_\Omega \mathbf{v}_t \cdot \mathbf{w}\ \mathrm{d}\boldsymbol{x} = -\int_\Omega \mathrm{div}(c\mathbf{w})\ \phi\ \mathrm{d}\boldsymbol{x} \qquad \forall \mathbf{w} \in \mathbf{H}_P(c, \mathrm{div}; \Omega), \tag{2.1.3a}$$

$$\int_\Omega \phi_t \psi\ \mathrm{d}\boldsymbol{x} = \int_\Omega \mathrm{div}(c\mathbf{v})\ \psi\ \mathrm{d}\boldsymbol{x}, \qquad \forall \psi \in L^2(\Omega). \tag{2.1.3b}$$

Notice that adding equation (2.1.3a) to equation (2.1.3b) and choosing the test functions as $\mathbf{w} = \mathbf{v}$ and $\psi = \phi$, we obtain

$$\int_\Omega \mathbf{v}_t \cdot \mathbf{v}\, \mathrm{d}\boldsymbol{x} + \int_\Omega \phi_t \phi\, \mathrm{d}\boldsymbol{x} = 0,$$

and defining the quantity of total energy as

$$E(t) := \frac{1}{2} \int_\Omega |\mathbf{v}|^2 + |\phi|^2 \mathrm{d}\boldsymbol{x}, \tag{2.1.4}$$

it is obviously seen that $\frac{\mathrm{d}E(t)}{\mathrm{d}t} = 0$, that is, the energy is preserved.

Building upon the work conducted by Kraus *et al.* in the field of plasma physics [68], we aim to introduce and analyze, for this simplified model problem, a numerical method that preserves the total energy of the system for long-term simulations, and for which (2.1.3a) is solved in a weak sense, while (2.1.3b) is solved in strong form. To achieve this purpose, we introduce a generic linear operator

$$\mathcal{T} : L^2(\Omega) \to L^2(\Omega),$$

and replace the second equation (2.1.3b) by

$$\phi_t = \mathcal{T}(\mathrm{div}(c\mathbf{v})). \tag{2.1.5}$$

Note that we recover (2.1.3b) if $\mathcal{T}$ is the identity operator. This change in the equation affects the conservation of the total energy of the system. In order to retrieve this conservation we replace (2.1.3a) with

$$\int_\Omega \mathbf{v}_t \cdot \mathbf{w} \, \mathrm{d}\boldsymbol{x} = -\int_\Omega \mathcal{T}(\mathrm{div}(c\mathbf{w}))\phi \, \mathrm{d}\boldsymbol{x}.$$

The modified problem that we obtain, with mixed homogeneous Dirichlet and periodic boundary conditions, reads: find $\mathbf{v} \in \mathbf{H}_P(c, \mathrm{div}; \Omega)$ and $\phi \in L^2(\Omega)$ such that the following equations hold

$$\int_\Omega \mathbf{v}_t \cdot \mathbf{w} \, \mathrm{d}\boldsymbol{x} = -\int_\Omega \mathcal{T}(\mathrm{div}(c\mathbf{w}))\phi \, \mathrm{d}\boldsymbol{x}, \quad \forall \mathbf{w} \in \mathbf{H}_P(c, \mathrm{div}; \Omega), \tag{2.1.6a}$$

$$\phi_t = \mathcal{T}(\mathrm{div}(c\mathbf{v})). \tag{2.1.6b}$$

2.2 B-splines and IGA framework

In this section we recall the definition of B-splines in the univariate and multivariate context. We also introduce the isogeometric spaces, together with their regularity assumptions.

2.2.1 Univariate B-splines

For the definition of B-splines we follow the notations and the guidelines of [13] and [77]. Let us introduce a knot vector $\Xi = \{\xi_1, \ldots, \xi_{n+p+1}\}$, where $\xi_i \in \mathbb{R}$ is the i-th knot, p is the polynomial degree and n is the number of basis functions. We assume that $\xi_1 \leq \xi_2 \leq \cdots \leq \xi_{n+p} \leq \xi_{n+p+1}$, and without loss of generality $\xi_1 = 0$, $\xi_{n+p+1} = 1$. We also admit at most $p+1$ repeated knots. We introduce the B-spline functions $\{\widehat{b}_{i,p}\}_{i=1}^n$ of degree p over the knot vector Ξ following the Cox-DeBoor recursive formula [33, Chapter IX]. We obtain a set of n B-splines with the following properties: non-negativity, partition of unity and local support of each basis function, see [13]. The univariate B-spline space of degree p over the knot vector Ξ is:

$$\widehat{S}_p(\Xi) \coloneqq \mathrm{span}\{\widehat{b}_{i,p} : i = 1, \ldots, n\}.$$

We introduce also the vector $\mathbf{Z} = \{\zeta_1, \ldots, \zeta_z\}$, of knots without repetitions, which are also called breakpoints, and denote with m_j the multiplicity of ζ_j, such that $\sum_{j=1}^z m_j = n + p + 1$, and $\zeta_1 = \xi_1 = \cdots = \xi_{m_1}$, $\zeta_2 = \xi_{m_1+1} = \cdots = \xi_{m_1+m_2}$ and so on. Notice that

$$\Xi = \{\underbrace{\zeta_1, \ldots, \zeta_1}_{m_1 \text{ times}}, \underbrace{\zeta_2, \ldots, \zeta_2}_{m_2 \text{ times}}, \ldots, \underbrace{\zeta_z, \ldots, \zeta_z}_{m_z \text{ times}}\},$$

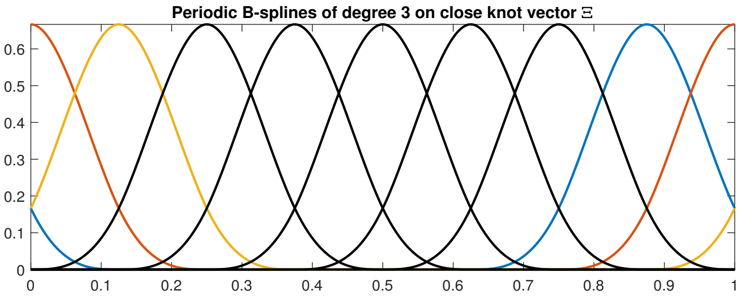


Figure 2.2. B-spline basis of degree 3 in the periodic case. In color the first three B-spline basis. The left hand side matches with regularity C^2 with its right hand side.

with each ζ_j repeated m_j times, and $1 \leq m_j \leq p+1$ for all internal knots. The breakpoints form a partition of the unit interval, and we say the partition is locally quasi-uniform if there exists a constant $\beta \geq 1$ independent of z such that

$$\beta^{-1} \leq \frac{\zeta_{j+1} - \zeta_j}{\zeta_j - \zeta_{j-1}} \leq \beta, \quad \forall j = 2, \ldots, z-1.$$

Assuming that the multiplicity of internal knots is at most p, we can write the derivative of a B-spline as follows

$$\frac{\partial \widehat{b}_{i,p}}{\partial \xi}(\xi) = \widehat{D}_{i-1,p-1}(\xi) - \widehat{D}_{i,p-1}(\xi), \tag{2.2.7}$$

with the Curry-Schoenberg spline basis:

$$\widehat{D}_{i,p-1}(\xi) = \frac{p}{\xi_{i+p+1} - \xi_{i+1}} \widehat{b}_{i+1,p-1}(\xi), \quad \text{for } i = 1, \ldots, n-1, \tag{2.2.8}$$

where we assumed that $\widehat{D}_{1,p-1}(\xi) = \widehat{D}_{n+1,p-1}(\xi) = 0$. Note that the derivative belongs to the spline space $\widehat{S}_{p-1}(\Xi')$ where $\Xi' = \{\xi_2, \ldots, \xi_{n+p}\}$.

In order to handle periodic boundary conditions, we follow the construction of periodic B-spline spaces, see for instance [86]. Consider a uniformly spaced *closed* knot vector Ξ of degree p, that is $\xi_1 < \cdots < \xi_{p+1} = 0$ and $1 = \xi_{n+1} < \cdots < \xi_{n+p+1}$ with at least p internal knots. We can construct a periodic basis on such knot vector, by identifying the basis functions of the B-spline space $\widehat{S}_p(\Xi)$ in this way:

$$\begin{cases} \widehat{b}_{i,p}^{Per} := \widehat{b}_{i,p} + \widehat{b}_{n-p+i,p}, & \text{for } i = 1, \ldots, p; \\ \widehat{b}_{i,p}^{Per} = \widehat{b}_{i,p}, & \text{for } i = p+1, \ldots, n-p. \end{cases}$$

Here we are gluing together the first p basis functions with the last p, this way we get continuity of derivatives up to order $p-1$. In Figure 2.2 it is shown the behavior of this periodic basis at the boundary. We can introduce the periodic B-spline space with highest regularity at the boundary as:

$$\widehat{S}_p^{Per}(\Xi) := \operatorname{span}\{\widehat{b}_{i,p}^{Per} : i = 1, \ldots, n-p\}.$$

It is easy to see that the dimension of the spline space $\widehat{S}_p^{Per}(\Xi)$ is $n-p$, which is the same of $\widehat{S}_{p-1}^{Per}(\Xi')$. Finally we note that equation (2.2.7) holds for periodic B-splines too. For more information and properties on B-splines, we refer the reader to [33].

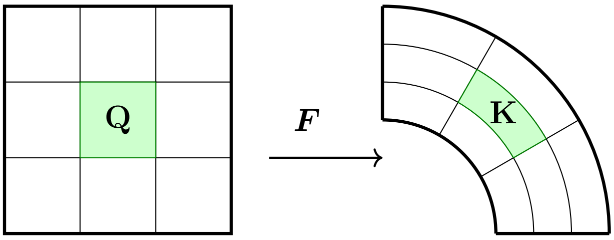


Figure 2.3. Mesh $\widehat{\mathcal{M}}$ in the parametric domain, and its image $\mathcal{M}$ in the physical domain.

2.2.2 Multivariate B-splines

Multivariate B-splines are introduced and defined by tensor product, starting from univariate B-splines on each spatial direction. Let d be the space dimension, usually $d = 2, 3$, and assume $n_l \in \mathbb{N}$ the number of univariate basis functions in direction l, $p_l \in \mathbb{N}$ is the degree, while $\Xi_l = \{\xi_{l,1}, \dots, \xi_{l,n_l+p_l+1}\}$ and $\mathbf{Z}_l = \{\zeta_{l,1}, \dots, \zeta_{l,z_l}\}$ are respectively the knots and breakpoints in direction l. We also set the polynomial degree vector $\boldsymbol{p} = (p_1, \dots, p_d)$ and $\boldsymbol{\Xi} = \{\Xi_1, \dots, \Xi_d\}$. Finally we set $\mathbf{J} = \{\boldsymbol{j} = (j_1, \dots, j_d) \subset \mathbb{N}^d : 1 \le j_l \le n_l\}$. The set of multivariate B-splines basis functions of degree vector $\boldsymbol{p}$ is

$$\{\widehat{B}_{\boldsymbol{i},\boldsymbol{p}}(\boldsymbol{\xi}) = \widehat{b}_{i_1,p_1}(\xi_1) \dots \widehat{b}_{i_d,p_d}(\xi_d), \text{for } \boldsymbol{i} \in \mathbf{J}\},$$

and the B-spline multivariate space of degree vector $\boldsymbol{p}$ over $\boldsymbol{\Xi}$ is

$$\widehat{S}_{\boldsymbol{p}}(\boldsymbol{\Xi}) := \operatorname{span}\{\widehat{B}_{\boldsymbol{i},\boldsymbol{p}}(\boldsymbol{\xi}) : \boldsymbol{i} \in \mathbf{J}\} = \otimes_{l=1}^{d} \widehat{S}_{p_l}(\Xi_l).$$

A similar construction also applies for multivariate periodic B-splines. It is also possible to combine tensor product of standard B-splines in the first directions with periodic ones for the last direction, which is the case of the mixed boundary condition setting for our model problem.

2.2.3 Isogeometric analysis framework

The isogeometric map $\boldsymbol{F} : \widehat{\Omega} \to \Omega$, is a parameterization of the geometry of the physical domain, based on B-splines (or more often in their rational counterpart of NURBS), usually indicated by

$$\boldsymbol{F} := \sum_{\boldsymbol{i} \in \mathbf{J}} \mathbf{c}_{\boldsymbol{i}} \widehat{B}_{\boldsymbol{i},\boldsymbol{p}}.$$

For the analysis of the discrete problem, we need to introduce some assumptions on $\boldsymbol{F}$. First, we introduce the parametric Bézier mesh $\widehat{\mathcal{M}}$, which is

$$\widehat{\mathcal{M}} := \{\mathbf{Q}_{\boldsymbol{j}} = I_{1,j_1} \times \dots \times I_{d,j_d} : I_{l,j_l} = (\zeta_{l,j_l}, \zeta_{l,j_l+1}),\ \text{for } 1 \le j_l \le z_l - 1\}.$$

Given an element $\mathbf{Q} \in \widehat{\mathcal{M}}$, we set $h_{\mathbf{Q}} = \operatorname{diam}(\mathbf{Q}_{\mathbf{j}})$, while $h = \max\{h_{\mathbf{Q}},\ \mathbf{Q} \in \widehat{\mathcal{M}}\}$. Moreover, given $\mathbf{D} \subset \widehat{\Omega}$, we denote by $\widetilde{\mathbf{D}}$ its support extension, that is the interior of the union of the supports of basis functions whose support intersects $\mathbf{D}$.

The Bézier mesh is defined as the image of elements in $\widehat{\mathcal{M}}$ through $\boldsymbol{F}$:

$$\mathcal{M} := \{\mathbf{K} \subset \Omega : \mathbf{K} = \boldsymbol{F}(\mathbf{Q}), \mathbf{Q} \in \widehat{\mathcal{M}}\},$$

see Figure 2.3. We have $\widetilde{\boldsymbol{F}(\mathbf{D})} = \boldsymbol{F}(\widetilde{\mathbf{D}})$, for every subset $\mathbf{D} \subset \widehat{\Omega}$, denoting the support extension in the physical domain. The first assumption is the following.

Assumption 2.1. *We assume that $\boldsymbol{F}$ is a bi-Lipschitz homeomorphism. Moreover, $\boldsymbol{F}|_{\overline{\mathbf{Q}}}$ belongs to $\mathcal{C}^\infty(\overline{\mathbf{Q}})$ for all $\mathbf{Q} \in \widehat{\mathcal{M}}$, where $\overline{\mathbf{Q}}$ denotes the closure of $\mathbf{Q}$, and $\boldsymbol{F}^{-1}|_{\overline{\mathbf{K}}}$ belongs to $\mathcal{C}^\infty(\overline{\mathbf{K}})$, for all $\mathbf{K} \in \mathcal{M}$.*

This prevents the existence of singularities and self-intersections in the parameterization $\boldsymbol{F}$. The second assumption simplifies the dealing with the boundary.

Assumption 2.2. *The boundary region $\Gamma_D \subset \partial\Omega$ is the union of full faces of the boundary. More precisely, $\Gamma_D = \boldsymbol{F}(\widehat{\Gamma}_D)$, with $\widehat{\Gamma}_D$ a collection of full faces of the parametric domain $\widehat{\Omega}$. Finally, if $\Gamma_P \neq \emptyset$, we assume the periodic boundary to be:*

$$\Gamma_P = \boldsymbol{F}\big([0,1]^{d-1} \times \{0\}\big) \cup \boldsymbol{F}\big([0,1]^{d-1} \times \{1\}\big).$$

Finally, we assume local quasi-uniformity of the univariate partitions, thus the parametric Bézier mesh is shape regular, that is, the ratio between the smallest edge of $\mathbf{Q} \in \widehat{\mathcal{M}}$ and its diameter $h_{\mathbf{Q}}$ is bounded uniformly with respect to h and $\mathbf{Q}$. Analogously, the Bézier mesh is shape regular thanks to Assumption 2.1.

2.3 Discretization

In this section, our focus is on introducing the discretization of problem (2.1.6), following the same approach as in [62, 69]. We approximate $\mathcal{T}$ with quasi-interpolant projections that commute with the divergence operator. The Crank-Nicolson method is employed for temporal discretization, chosen for its conservativity. We emphasize that our objective is to study the approximation properties of the proposed method, both theoretically and numerically. To achieve this, we intentionally maintain a highly generic notation. In Section 2.3.1, we present the spatial semi-discretization through the construction of isogeometric discrete spaces and the associated projections. In Section 2.3.2 we discuss a particularization of such projections, which are quasi-interpolant operators based on point evaluation of the functions to be projected. This quasi-interpolant operators will be subjected to numerical study and testing within the proposed discretization framework. Finally, in Section 2.3.3 we recall the Crank-Nicolson semi-discretization in time and the fully discrete problem is presented.

2.3.1 Discretization in space with commutative projections

Here we introduce the discrete spaces for the mixed formulation given by equations (2.1.6a) and (2.1.6b). We present the case $d = 2$, while the case $d = 3$ is completely analogous.

Discrete spline spaces

Let us standardize the notations following [12], and start with the case of Dirichlet boundary conditions, that is $\Gamma_P = \emptyset$, we introduce the symbols:

$$X^1 := \mathbf{H}(c, \mathrm{div}; \Omega), \quad X^2 := L^2(\Omega), \quad \widehat{X}^1 := \mathbf{H}(\widehat{c}, \mathrm{div}; \widehat{\Omega}), \quad \widehat{X}^2 := L^2(\widehat{\Omega}),$$

where we recall $\widehat{c} = c \circ \boldsymbol{F}$. Thanks to Assumption 2.1, which states that both $\boldsymbol{F}$ and its inverse are smooth, we can define the pull-backs that relate these spaces as (see [59, Section 2.2])

$$\begin{aligned} \iota^1(\mathbf{f}) &:= \det(D\boldsymbol{F})(D\boldsymbol{F})^{-1}(\mathbf{f} \circ \boldsymbol{F}), && \mathbf{f} \in X^1, \\ \iota^2(f) &:= \det(D\boldsymbol{F})(f \circ \boldsymbol{F}), && f \in X^2, \end{aligned}$$

where $D\boldsymbol{F}$ is the Jacobian matrix of $\boldsymbol{F}$. Then, due to the divergence preserving property of the map ι^1, see [84, Section 3.9], we have div $\circ\, \iota^1 = \iota^2 \circ$ div. Fixed a polynomial degree vector $\boldsymbol{p} = (p_1, p_2)$ and $\boldsymbol{\Xi} = (\Xi_1, \Xi_2)$, we define the discrete spaces on the parametric domain as:

$$\begin{aligned} \widehat{X}_h^1 &:= \widehat{S}_{p_1,p_2-1}(\Xi_1, \Xi_2') \times \widehat{S}_{p_1-1,p_2}(\Xi_1', \Xi_2), \\ \widehat{X}_h^2 &:= \widehat{S}_{p_1-1,p_2-1}(\Xi_1', \Xi_2'). \end{aligned}$$

The choice of the bases follows from [12, Section 5.2] and [91, Section 4], that is

$$\widehat{X}_h^1 = \operatorname{span}\{\mathcal{J}_1 \cup \mathcal{J}_2\},$$

where we set

$$\begin{aligned} \mathcal{J}_1 &= \{\widehat{b}_{i_1,p_1}(\xi_1)\widehat{D}_{i_2,p_2-1}(\xi_2)\mathbf{e_1} : \ 1 \le i_1 \le n_1, \ 1 \le i_2 \le n_2 - 1\}, \\ \mathcal{J}_2 &= \{\widehat{D}_{i_1,p_1-1}(\xi_1)\widehat{b}_{i_2,p_2}(\xi_2)\mathbf{e_2} : \ 1 \le i_1 \le n_1 - 1, \ 1 \le i_2 \le n_2\}, \end{aligned}$$

and $\{\mathbf{e_1}, \mathbf{e_2}\}$ is the canonical basis of $\mathbb{R}^2$. As regards the second discrete space, we set

$$\widehat{X}_h^2 = \operatorname{span}\{\widehat{D}_{i_1,p_1-1}(\xi_1)\widehat{D}_{i_2,p_2-1}(\xi_2) : \ 1 \le i_l \le n_l - 1, \ l = 1, 2\}.$$

The discrete spaces on the physical domain Ω can be defined from the spaces $\widehat{X}_h^1$ and $\widehat{X}_h^2$ by push-forward, that is, the inverse of the transformations ι^1 and ι^2, that commute with the divergence operator. We have the following definitions:

$$X_h^1 := \{\mathbf{f_h} : \iota^1(\mathbf{f}_h) \in \widehat{X}_h^1\}, \quad X_h^2 := \{f_h : \iota^2(f_h) \in \widehat{X}_h^2\}.$$

For later use, we introduce a suitable notation for the basis of these discrete spaces. We denote by $\{\mathbf{b}_{i,h}\}_{i=1}^N$ and $\{\widehat{\mathbf{b}}_{i,h}\}_{i=1}^N$ the sets of basis functions of X_h^1 and $\widehat{X}_h^1$ respectively, reordered with lexicographic ordering, such that $\iota^1(\mathbf{b}_{i,h}) = \widehat{\mathbf{b}}_{i,h}$ for any $i = 1, \ldots, N$, where $N = n_1(n_2 - 1) + (n_1 - 1)n_2$. Analogously we can introduce the notations $\{b_{i,h}\}_{i=1}^M$ and $\{\widehat{b}_{i,h}\}_{i=1}^M$ for the basis functions of the spaces X_h^2 and $\widehat{X}_h^2$ respectively, such that, for $i = 1, \ldots, M$ with $M = (n_1 - 1)(n_2 - 1)$ we have $\iota^2(b_{i,h}) = \widehat{b}_{i,h}$.

Univariate projections

In order to discretize equations (2.1.6a) and (2.1.6b), we follow the idea in [62] and in [69], and approximate the operator $\mathcal{T}$ with a projector into the discrete spline space. We first introduce commutative projectors in the univariate case, which can be defined by a dual basis, i.e. $\widehat{\pi}_{p,\Xi} : L^2(0,1) \longrightarrow \widehat{S}_p(\Xi)$ such that

$$\widehat{\pi}_{p,\Xi}(f) = \sum_{i=1}^{n} \lambda_{i,p}(f)\widehat{b}_{i,p} \tag{2.3.9}$$

where $\lambda_{i,p}$ are a set of dual functionals verifying $\lambda_{i,p}(\widehat{b}_{j,p}) = \delta_{ij}$, where δ_{ij} is the Kronecker delta. Notice that, with this property, the operator is a projection, that is $\widehat{\pi}_{p,\Xi}(s) = s$, for all $s \in \widehat{S}_p(\Xi)$.

We also recall the construction of univariate projections that commute with the derivative as in [24, Section 3.1.2]. In order to obtain the one-dimensional commuting diagram, we want the commutative projection $\widehat{\pi}^c_{p-1,\Xi'} : L^2(0,1) \to S_{p-1}(\Xi')$ to satisfy:

$$\widehat{\pi}^c_{p-1,\Xi'} \frac{\partial}{\partial \xi} f = \frac{\partial}{\partial \xi} \widehat{\pi}_{p,\Xi} f,$$

for all f in $H^1(0,1)$. In order to satisfy the previous equation, given any projector $\widehat{\pi}_{p,\Xi}$ as in (2.3.9), its commutative projection is defined as

$$\widehat{\pi}^c_{p-1,\Xi'} g(\xi) \coloneqq \frac{\partial}{\partial \xi} \widehat{\pi}_{p,\Xi} \int_0^\xi g(s) \mathrm{d}s, \tag{2.3.10}$$

for all functions g such that $G(\xi) = \int_0^\xi g(s)\mathrm{d}s$ is in the domain of $\widehat{\pi}_{p,\Xi}$.

To apply periodic boundary conditions we need to define the commutative projector in a different way, to ensure that integrating a periodic function in $L^2(0,1)$, we obtain a periodic function in $H^1(0,1)$. Hence, given a periodic projection $\widehat{\pi}^{Per}_{p,\Xi}$, we define its commutative one as follows

$$\widehat{\pi}^{c,Per}_{p-1,\Xi'} g(\xi) \coloneqq \left(\frac{\partial}{\partial \xi} \widehat{\pi}^{Per}_{p,\Xi} \int_0^\xi \tilde{g}(s) \mathrm{d}s \right) \oplus \overline{g}, \tag{2.3.11}$$

for all functions $g = \tilde{g} \oplus \overline{g} \in L^2(0,1)$ such that $\overline{g} \coloneqq \int_0^1 g(s)\mathrm{d}s \in \mathbb{R}$ and $\tilde{g} \in L^2_0(0,1)$, the space of functions in $L^2(0,1)$ with zero average. Notice that $G(\xi) = \int_0^\xi \tilde{g}(s)\mathrm{d}s$ is in the domain of $\widehat{\pi}^{Per}_{p,\Xi}$, and again we have

$$\widehat{\pi}^{c,Per}_{p-1,\Xi'} \frac{\partial}{\partial \xi} f = \frac{\partial}{\partial \xi} \widehat{\pi}^{Per}_{p,\Xi} f, \tag{2.3.12}$$

for all f in the domain of definition of $\widehat{\pi}^{Per}_{p,\Xi}$. It is easy to see that the commutative projections defined in this way preserve splines.

Multivariate construction

The univariate projection operators introduced can be extended to the multidimensional case by tensor product constructions. Given $\widehat{\Omega} = [0,1]^d \subset \mathbb{R}^d$, for $i = 1, 2, \ldots, d$, let us denote with $\widehat{\pi}_{p_i,\Xi_i}$ a generic univariate projection as in (2.3.9). We can define a multivariate projection as

$$\widehat{\Pi}_{\boldsymbol{p},\boldsymbol{\Xi}} \coloneqq \widehat{\pi}_{p_1,\Xi_1} \otimes \widehat{\pi}_{p_2,\Xi_2} \otimes \cdots \otimes \widehat{\pi}_{p_d,\Xi_d}. \tag{2.3.13}$$

It is important to note that it can be expressed as

$$\widehat{\Pi}_{\boldsymbol{p},\boldsymbol{\Xi}}(f) = \sum_{\boldsymbol{i} \in \mathbf{J}} \lambda_{\boldsymbol{i},\boldsymbol{p}}(f) \widehat{B}_{\boldsymbol{i},\boldsymbol{p}}, \tag{2.3.14}$$

where $\boldsymbol{p}$, $\boldsymbol{\Xi}$ and $\mathbf{J}$ are given as in the multivariate B-spline construction, and each dual functional is defined from the univariate dual basis by the expression

$$\lambda_{\boldsymbol{i},\boldsymbol{p}} = \lambda_{i_1,p_1} \otimes \lambda_{i_2,p_2} \otimes \cdots \otimes \lambda_{i_d,p_d}. \tag{2.3.15}$$

It is easy to see that the same constructions hold for the multivariate periodic case, or for combinations with periodic boundary conditions in only one parametric coordinate.

At this point we have all the ingredients to define projections on the spaces $\widehat{X}_h^1$ and $\widehat{X}_h^2$. The choice of the projection follows from the definition of the discrete spaces. More precisely we set

$$\widehat{\Pi}^1 = (\widehat{\pi}_{p_1} \otimes \widehat{\pi}^c_{p_2-1}) \times (\widehat{\pi}^c_{p_1-1} \otimes \widehat{\pi}_{p_2}), \tag{2.3.16}$$

$$\widehat{\Pi}^2 = \widehat{\pi}^c_{p_1-1} \otimes \widehat{\pi}^c_{p_2-1}, \tag{2.3.17}$$

where for simplicity of notation we have omitted the knot vector from the subscript of the univariate projectors. These projectors satisfy the spline preserving properties, see [12, Lemma 5.3].

The projectors into the discrete spline spaces in the physical domain are defined from the ones in the parametric domain (2.3.16) and (2.3.17), and the corresponding pull-backs ι^1 and ι^2, in such a way they are uniquely characterized by the equations

$$\begin{aligned} \iota^1(\Pi^1 \mathbf{f}) &= \widehat{\Pi}^1(\iota^1(\mathbf{f})), \\ \iota^2(\Pi^2 f) &= \widehat{\Pi}^2(\iota^2(f)). \end{aligned} \tag{2.3.18}$$

These projectors satisfy the commutativity property with the divergence operator, as stated in the following lemma:

Lemma 2.1. *Given $\widehat{c}$ and c the coefficients of spaces $\widehat{X}^1$ and X^1, the following equations hold*

$$\text{div } (\widehat{\Pi}^1(\widehat{c}\mathbf{f})) = \widehat{\Pi}^2(\text{div } (\widehat{c}\mathbf{f})), \quad \forall \mathbf{f} \in \widehat{X}^1, \tag{2.3.19}$$

$$\text{div } (\Pi^1(c\mathbf{f})) = \Pi^2(\text{div } (c\mathbf{f})), \quad \forall \mathbf{f} \in X^1. \tag{2.3.20}$$

The proof of (2.3.19) is given in [12, Lemma 5.5]. Equation (2.3.20) is an immediate consequence of the definitions, together with the commutativity property (2.3.19). We conclude this paragraph with the following remark on the periodic case.

Remark 2.1. *If $\Gamma_P \neq \emptyset$, we proceed analogously introducing the discrete spaces*

$$\widehat{X}_h^1 := \left(\widehat{S}_{p_1}(\Xi_1) \otimes \widehat{S}^{Per}_{p_2-1}(\Xi_2')\right) \times \left(\widehat{S}_{p_1-1}(\Xi_1') \otimes \widehat{S}^{Per}_{p_2}(\Xi_2)\right), \tag{2.3.21}$$

$$\widehat{X}_h^2 := \widehat{S}_{p_1-1}(\Xi_1') \otimes \widehat{S}^{Per}_{p_2-1}(\Xi_2'), \tag{2.3.22}$$

and considering the projectors:

$$\widehat{\Pi}^1 = (\widehat{\pi}_{p_1} \otimes \widehat{\pi}^{c,Per}_{p_2-1}) \times (\widehat{\pi}^c_{p_1-1} \otimes \widehat{\pi}^{Per}_{p_2}), \tag{2.3.23}$$

$$\widehat{\Pi}^2 = \widehat{\pi}^c_{p_1-1} \otimes \widehat{\pi}^{c,Per}_{p_2-1}. \tag{2.3.24}$$

Semi-discretization in space

We propose to take as a linear operator $\mathcal{T}$ in (2.1.6a) and (2.1.6b) the generic tensor product commutative projection Π^2. The semi-discrete problem in space reads, find $\mathbf{v}_h \in X_h^1$ and $\phi_h \in X_h^2$, such that the following equations hold:

$$\int_\Omega (\mathbf{v}_h)_t \cdot \mathbf{w}_h \, \mathrm{d}\boldsymbol{x} = -\int_\Omega \Pi^2(\text{div}(c\mathbf{w}_h)) \, \phi_h \, \mathrm{d}\boldsymbol{x}, \qquad \forall \mathbf{w}_h \in X_h^1, \tag{2.3.25a}$$

$$(\phi_h)_t = \Pi^2(\text{div}(c\mathbf{v}_h)). \tag{2.3.25b}$$

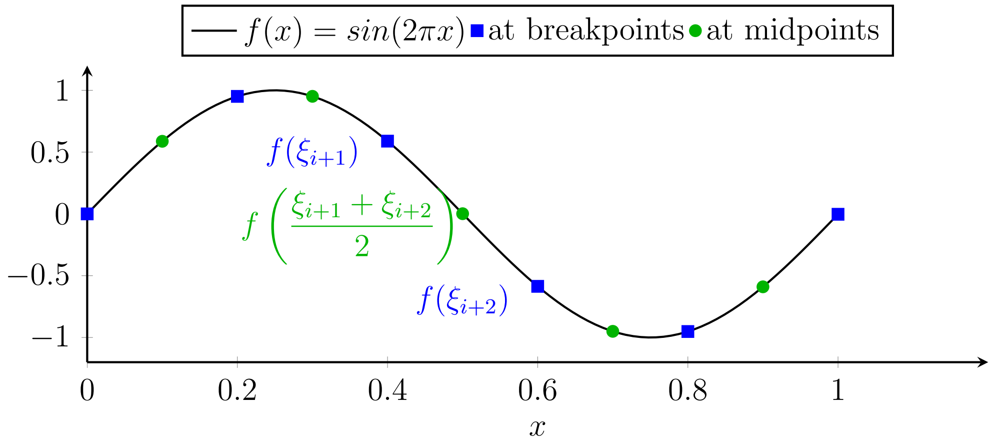


Figure 2.4. Example of pointwise evaluation of $f(x) = sin(2\pi x)$ at breakpoints and midpoints for projection with $\widehat{\mathfrak{L}\pi}_{2,\Xi}$. The three highlighted points are used for the computation of $\lambda_{i,2}(f)$, for $i = 4$, corresponding to the B-spline with support $[0.2, 0.8]$.

In the next section we will exploit a particularization of such discretization, based on quasi-interpolant projections.

2.3.2 Quasi interpolant projections

Here we focus on a particular projection whose construction is given in detail in [73]. We refer to such projection using the notation $\widehat{\mathfrak{L}\pi}_{p,\Xi}$. We deal with open knot vectors with non-repeating internal knots, i.e. $\Xi = \{\xi_1, \ldots, \xi_{n+p+1}\}$ such that the vector $\mathbf{Z} = \{\zeta_1, \ldots, \zeta_z\}$ of the breakpoints has multiplicities $m_1 = m_z = p + 1$ and $m_j = 1$ for $j = 2, \ldots, z-1$. In order to find an approximation $\widehat{\mathfrak{L}\pi}_{p,\Xi}(f)$ of $f : [0, 1] \to \mathbb{R}$, we need to define a dual basis $\lambda_{i,p}$ and compute $\lambda_{i,p}(f)$ for $i = 1, \ldots, n$, and the idea is the following:

1. Choose $\mathcal{I} = [\xi_\mu, \xi_\nu] \subset [\xi_{p+1}, \xi_n]$ such that $\mathcal{I} \cap [\xi_i, \xi_{i+p+1}] \neq \emptyset$. Notice that $[\xi_i, \xi_{i+p+1}]$ is the support of $\widehat{b}_{i,p}$, and denote by $f_{\mathcal{I}}$ the restriction of f to $\mathcal{I}$.

2. Choose a local approximation method $\mathcal{P}_{loc}$ for $f_{\mathcal{I}}$, such that it reproduces splines of degree up to p. This approximation is written analogously to (2.3.9) in this way: $\mathcal{P}_{loc}(f_{\mathcal{I}}) = \sum_{j=\mu-p}^{\nu-1} c_j \widehat{b}_{j,p}$, and we have $\mu - p \leq i \leq \nu - 1$ since the support of $\widehat{b}_{i,p}$ intersects $\mathcal{I}$.

3. Finally set $\lambda_{i,p}(f) = c_i$.

The idea is to use polynomial interpolation as local approximation method. We exploit the construction for spline spaces of degree 2 and 3 given in [73]. Let us now fix $p = 2$, for a given index $i = 1, \ldots, n$, we have $[\xi_{i+1}, \xi_{i+2}] \subset [\xi_i, \xi_{i+3}] = \mathrm{supp}(\widehat{b}_{i,2})$, but $\xi_{i+1} < \xi_{i+2}$ when $i \neq 1, n$. This means that for $i = 2, \ldots, n-1$ we can choose $\mathcal{I} = [\xi_{i+1}, \xi_{i+2}]$, and perform three points interpolation on $\xi_{i+1} < (\xi_{i+1} + \xi_{i+2})/2 < \xi_{i+2}$. Notice that we are interpolating at two consecutive breakpoints and their midpoint in a uniform partition. Due to this particular choice of the interpolation points, we can explicitly write the coefficients of $\widehat{\mathfrak{L}\pi}_{2,\Xi}$, that are:

$$\lambda_{i,2}(f) = -\frac{1}{2}f(\xi_{i+1}) + 2f\Big(\frac{\xi_{i+1}+\xi_{i+2}}{2}\Big) - \frac{1}{2}f(\xi_{i+2}), \quad \text{for } i = 2, \dots, n-1. \tag{2.3.26}$$

We remark this expression is valid whenever $\xi_{i+1} < \xi_{i+2}$, which is not the case for $i = 1$ and $i = n$ in the case of an open knot vector, as used for Dirichlet boundary conditions. In this situation we want the operator to be interpolant at the boundary and hence we fix $\lambda_{1,2} = f(0)$ and $\lambda_{n,2} = f(1)$. As regards closed knot vectors for periodic boundary conditions, the computation is easier, since we can use (2.3.26) for all indices $i = 1, \dots, n$. Notice that we need the pointwise evaluation of the function f we want to approximate, among the breakpoints and their midpoints, as in the example given in Figure 2.4.

We recall now the explicit formulae in the case $p = 3$, that is the quasi-interpolant $\widehat{\mathfrak{L}\pi}_{3,\Xi}$, and refer for details to [73]. Here, as local approximation method, we use five points interpolation over the knots and midpoints of $\mathcal{I} = [\xi_{i+1}, \xi_{i+3}]$. Instead of (2.3.26), we can write now

$$\lambda_{i,3}(f) = \frac{1}{6}f(\xi_{i+1}) - \frac{4}{3}f\Big(\frac{\xi_{i+1}+\xi_{i+2}}{2}\Big) + \frac{10}{3}f(\xi_{i+2}) - \frac{4}{3}f\Big(\frac{\xi_{i+2}+\xi_{i+3}}{2}\Big) + \frac{1}{6}f(\eta_{i+3}). \tag{2.3.27}$$

For open knot vectors, hence Dirichlet boundary conditions, we have the particular cases for $i = 1, 2, n-1, n$, that we recall here:

$$\begin{aligned}
\lambda_{1,3}(f) &= f(0), \\
\lambda_{2,3}(f) &= -\frac{5}{18}f(\xi_4) + \frac{20}{9}f\Big(\frac{\xi_4+\xi_5}{2}\Big) - \frac{4}{3}f(\xi_5) + \frac{4}{9}f\Big(\frac{\xi_5+\xi_6}{2}\Big) - \frac{1}{18}f(\xi_6), \\
\lambda_{n-1,3}(f) &= -\frac{1}{18}f(\xi_{n-1}) + \frac{4}{9}f\Big(\frac{\xi_{n-1}+\xi_n}{2}\Big) - \frac{4}{3}f(\xi_n) \\
&\quad + \frac{20}{9}f\Big(\frac{\xi_n+\xi_{n+1}}{2}\Big) - \frac{5}{18}f(\xi_{n+1}), \\
\lambda_{n,3}(f) &= f(1).
\end{aligned}$$

From the computational point of view, we remember this technique requires evaluation of f over the breakpoints and midpoints. Notice that for projections in periodic spline spaces, it is sufficient to use (2.3.26) or (2.3.27). Finally, for approximation estimates by quasi-interpolant projections based on point evaluation of functions, we refer to [78, Section 5].

Commutative quasi-interpolant projections

In order to compute a projection $\widehat{\mathfrak{L}\pi}^{c}_{p,\Xi'}$ that commutes with $\widehat{\mathfrak{L}\pi}_{p,\Xi}$, we follow the construction in (2.3.10). Since we have to apply $\widehat{\mathfrak{L}\pi}_{p,\Xi}$ to the integral function F, we need its evaluation over all the breakpoints and midpoints. For this reason, instead of using Gaussian quadrature, we prefer to use Cavalieri-Simpson composite quadrature formulae. We want to consider both breakpoints and midpoints in a unified notation. Hence we introduce $\boldsymbol{\eta} = \{\eta_1, \dots, \eta_{2z-1}\}$ such that:

$$\begin{cases} \eta_{2i-1} = \zeta_i, & \text{for } i = 1, \dots, z, \\ \eta_{2i} = \frac{1}{2}(\zeta_i + \zeta_{i+1}), & \text{for } i = 1, \dots, z-1. \end{cases}$$

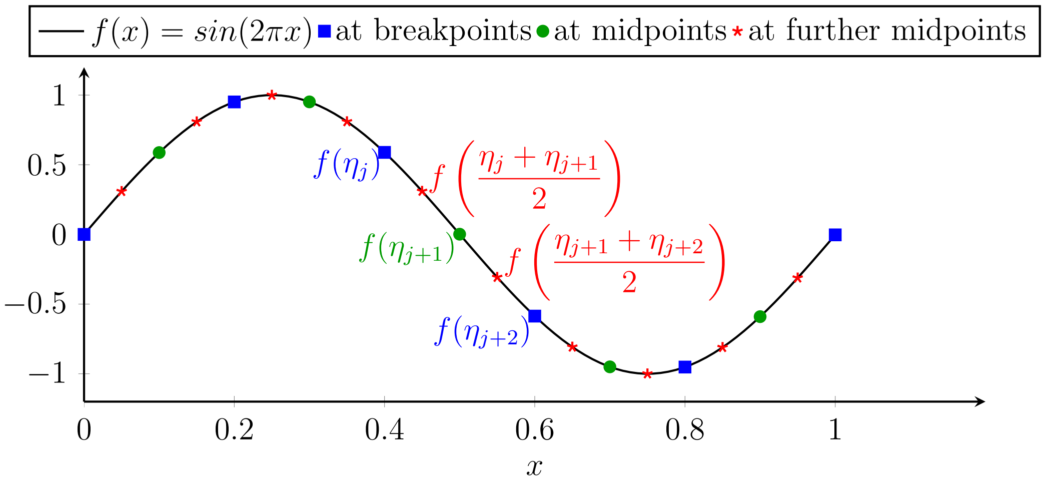


Figure 2.5. Example of evaluation of $f(x) = sin(2\pi x)$ at further midpoints, in order to project with $\widehat{\mathfrak{L}\pi}^{c}_{2,\Xi'}$

With this notation we have $F(\eta_1) = 0$ and for $i = 2, \dots, 2z-1$:

$$
\begin{aligned}
F(\eta_i) = \int_0^{\eta_i} f(s)\mathrm{d}s = \sum_{j=1}^{i-1} \int_{\eta_j}^{\eta_{j+1}} f(s)\mathrm{d}s \\
\approx \sum_{j=1}^{i-1} \frac{h}{6}\left(f(\eta_j) + 4f\left(\frac{\eta_j+\eta_{j+1}}{2}\right) + f(\eta_{j+1})\right),
\end{aligned}
$$

where $h = \eta_{j+1} - \eta_j$. Notice that this quadrature formula is exact for polynomials up to degree 3, but it requires evaluation of the integrating function over further midpoints, i.e. $(\eta_j + \eta_{j+1})/2$, as it is shown in Figure 2.5.

Multivariate quasi-interpolant projections

Finally we define the multivariate quasi-interpolant projections that commute with the divergence operator in the parametric domain, as

$$
\begin{aligned}
\widehat{\mathfrak{L}\Pi}^{1} &= (\widehat{\mathfrak{L}\pi}_{p_1,\Xi_1} \otimes \widehat{\mathfrak{L}\pi}^{c}_{p_2-1,\Xi_2'}) \times (\widehat{\mathfrak{L}\pi}^{c}_{p_1-1,\Xi_1'} \otimes \widehat{\mathfrak{L}\pi}_{p_2,\Xi_2}), \\
\widehat{\mathfrak{L}\Pi}^{2} &= \widehat{\mathfrak{L}\pi}^{c}_{p_1-1,\Xi_1'} \otimes \widehat{\mathfrak{L}\pi}^{c}_{p_2-1,\Xi_2'}.
\end{aligned}
\tag{2.3.28}
$$

The projections in the physical domain, namely $\mathfrak{L}\Pi^1, \mathfrak{L}\Pi^2$, are uniquely determined by (2.3.18). We remark that projections $\widehat{\mathfrak{L}\Pi}^{1}, \widehat{\mathfrak{L}\Pi}^{2}, \mathfrak{L}\Pi^1, \mathfrak{L}\Pi^2$ satisfy lemma 2.1. An analogous construction holds for projections in periodic spline spaces, or in spline spaces with mixed Dirichlet and periodic conditions, as in remark 2.1.

2.3.3 Conservative time discretization

Regarding time discretization, the choice of an energy-preserving method is mandatory since we are interested in preserving the total energy of the system. We detail the discretization with Crank-Nicolson method, which is second order and energy conservative, but other conservative methods could be chosen.

Let us fix $\tau = (t_0, \ldots, t_{N_T})$, a partition of the interval $[0, T]$, such that $t_0 = 0$, $t_{N_T} = T$ and $t_n < t_{n+1}$. For a simplified notation we assume the partition to be uniform, that is $t_{n+1} - t_n = \Delta t$, with a fixed real number $\Delta t > 0$, for $n = 0, \ldots, N_T - 1$. Let us denote by $\mathbf{v}_h^n$ and ϕ_h^n respectively the value of $\mathbf{v}_h$ and ϕ_h at the time instant t_n. Applying Crank-Nicolson method to equations (2.3.25), the fully discrete problem is now, for $n = 0, \ldots, N_T - 1$ find $\mathbf{v}_h^{n+1} \in X_h^1$ and $\phi_h^{n+1} \in X_h^2$, such that

$$\int_\Omega \frac{\mathbf{v}_h^{n+1} - \mathbf{v}_h^n}{\Delta t} \cdot \mathbf{w}_h \mathrm{d}\boldsymbol{x} = -\frac{1}{2} \int_\Omega \Pi^2(\operatorname{div}(c\mathbf{w}_h))(\phi_h^{n+1} + \phi_h^n) \mathrm{d}\boldsymbol{x}, \quad \forall \mathbf{w}_h \in X_h^1, \tag{2.3.29a}$$

$$\frac{\phi_h^{n+1} - \phi_h^n}{\Delta t} = \frac{1}{2}\big(\Pi^2(\operatorname{div}(c\mathbf{v}_h^{n+1})) + \Pi^2(\operatorname{div}(c\mathbf{v}_h^n))\big). \tag{2.3.29b}$$

Notice that in equation (2.3.29b), the left hand side ϕ_h^{n+1} is written as a linear combination of terms that depend on the time instant t_n, except for $\mathbf{v}_h^{n+1}$. Moreover, in view of the definition of the operator Π^2, (2.3.29b) can be seen as a collocation of the equation (2.1.6b). To solve the system, we replace the expression of ϕ_h^{n+1} from the second equation into the first one, and bring to the left all the terms in which the unknown $\mathbf{v}_h^{n+1}$ appears, that is

$$\begin{aligned}&\int_\Omega \mathbf{v}_h^{n+1} \cdot \mathbf{w}_h \mathrm{d}\boldsymbol{x} + \frac{(\Delta t)^2}{4} \int_\Omega \Pi^2(\operatorname{div}(c\mathbf{v}_h^{n+1}))\Pi^2(\operatorname{div}(c\mathbf{w}_h))\mathrm{d}\boldsymbol{x} = \\ &= \int_\Omega \mathbf{v}_h^n \cdot \mathbf{w}_h \mathrm{d}\boldsymbol{x} - \frac{(\Delta t)^2}{4} \int_\Omega \Pi^2(\operatorname{div}(c\mathbf{v}_h^n))\Pi^2(\operatorname{div}(c\mathbf{w}_h))\mathrm{d}\boldsymbol{x} - \Delta t \int_\Omega \phi_h^n \Pi^2(\operatorname{div}(c\mathbf{w}_h))\mathrm{d}\boldsymbol{x},\end{aligned} \tag{2.3.30}$$

which must hold for all $\mathbf{w}_h \in X_h^1$. We solve this equation to compute the unknown $\mathbf{v}_h^{n+1}$, which is then used to update the solution ϕ_h^{n+1} as indicated by (2.3.29b).

In our numerical tests, we will compare the proposed method with the standard Galerkin formulation, that we present for completeness, and which is given by: find $\mathbf{v}_h \in X_h^1$ and $\phi_h \in X_h^2$, such that

$$\int_\Omega \frac{\mathbf{v}_h^{n+1} - \mathbf{v}_h^n}{\Delta t} \cdot \mathbf{w}_h \mathrm{d}\boldsymbol{x} = -\frac{1}{2} \int_\Omega \operatorname{div}(c\mathbf{w}_h)(\phi_h^{n+1} + \phi_h^n) \mathrm{d}\boldsymbol{x}, \quad \forall \mathbf{w}_h \in X_h^1, \tag{2.3.31a}$$

$$\int_\Omega \frac{\phi_h^{n+1} - \phi_h^n}{\Delta t} \psi_h \mathrm{d}\boldsymbol{x} = \frac{1}{2} \int_\Omega \big(\operatorname{div}(c\mathbf{v}_h^{n+1}) + \operatorname{div}(c\mathbf{v}_h^n)\big)\psi_h \mathrm{d}\boldsymbol{x}, \quad \forall \psi_h \in X_h^2, \tag{2.3.31b}$$

Remark 2.2. *Notice that both equations* (2.3.29) *and* (2.3.31) *are energy conservative schemes by construction. On the other hand,* (2.3.31) *can not be reduced to a single equation. Thus, the method* (2.3.29) *is intrinsically cheaper than the method* (2.3.31) *and, at our knowledge, it is the first conservative method that does not require computation of both primal and dual variables at each time step.*

2.4 Error convergence analysis

In this section we analyze the convergence of the proposed method. In Section 2.4.1 we first present a convergence study in an abstract setting, for a generic family of projections as in Section 2.3.1, assuming stability and approximation properties both for the projectors and for their adjoint operators. Unfortunately, the adjoints of the quasi-interpolants of Section 2.3.2 do not satisfy these assumptions. For this reason, we present in Section 2.4.2 the analysis under some relaxed assumptions, for which only linear convergence could be proved. In our analysis, C will denote a generic constant independent of the mesh size h, but it may depend on the polynomial degree, the parameterization of the domain, and the local quasi-uniformity.

It will be useful for the analysis to write the equations of the problem in weak form. Let us denote by $(\mathbf{v}, \phi)$ the classical solution of the problem eq. (2.1.2), while $(\mathbf{v}_h, \phi_h)$ is the solution of the semidiscrete problem in space (2.3.25), which satisfies also the following variational equations:

$$\int_\Omega (\mathbf{v}_h)_t \cdot \mathbf{w}_h \, \mathrm{d}\boldsymbol{x} + \int_\Omega \Pi^2(\mathrm{div}(c\mathbf{w}_h)) \; \phi_h \, \mathrm{d}\boldsymbol{x} = 0, \qquad \forall \mathbf{w}_h \in X_h^1, \tag{2.4.32a}$$

$$\int_\Omega (\phi_h)_t \, \psi_h \, \mathrm{d}\boldsymbol{x} - \int_\Omega \Pi^2(\mathrm{div}(c\mathbf{v}_h)) \; \psi_h \, \mathrm{d}\boldsymbol{x} = 0, \qquad \forall \psi_h \in X_h^2. \tag{2.4.32b}$$

We introduce the energy norm $\|\mathbf{v}, \phi\|_E^2 := \|\mathbf{v}\|_{\mathbf{L}^2(\Omega)}^2 + \|\phi\|_{L^2(\Omega)}^2$, the Sobolev norm $\|\mathbf{v}, \phi\|_{\mathcal{H}^s}^2 := \|\mathbf{v}\|_{\mathbf{H}^s(\mathrm{div};\Omega)}^2 + \|\phi\|_{H^s(\Omega)}^2$, for a positive integer $s \geq 0$, and the following norms in the space-time domain $\Omega \times [0, T]$

$$\|\mathbf{v}, \phi\|_{\infty,E} := \sup_{t\in[0,T]} \|\mathbf{v}, \phi\|_E,$$
$$\|\mathbf{v}, \phi\|_{W^{1,\infty},\mathcal{H}^s} := \max\{ \sup_{t\in[0,T]} \|\mathbf{v}, \phi\|_{\mathcal{H}^s}, \sup_{t\in[0,T]} \|\mathbf{v}_t, \phi_t\|_{\mathcal{H}^s} \}.$$

In order to simplify the notation, we indicate by $\|\phi, \psi\|_E^2 = \|\phi\|_{L^2(\Omega)}^2 + \|\psi\|_{L^2(\Omega)}^2$ the equivalent of the energy norm when $\phi, \psi \in L^2(\Omega)$ are both scalar fields. We are interested in bounding

$$\|\mathbf{v} - \mathbf{v}_h, \phi - \phi_h\|_{\infty,E} = \sup_{t\in[0,T]} \|\mathbf{v} - \mathbf{v}_h, \phi - \phi_h\|_E, \tag{2.4.33}$$

and we will follow the ideas used in [2] for elasticity problems in mixed form with weak symmetry.

2.4.1 Convergence analysis based on the adjoint projections

Let us introduce the adjoint projection $\Pi^{2,*} \in \mathit{End}(X^2)$, which is defined by the following fundamental relation

$${}_{X^2}\langle \Pi^2(\phi), \psi \rangle_{X^{2,*}} = {}_{X^2}\langle \phi, \Pi^{2,*}(\psi) \rangle_{X^{2,*}}, \quad \forall\, \phi \in X^2,\ \psi \in X^{2,*}, \tag{2.4.34}$$

where $X^{2,*}$ is the dual space of X^2, and $\langle \cdot, \cdot \rangle$ denotes the duality pairing. We identify $X^2 = L^2(\Omega)$ with its dual space, as it is usually done, and in particular (2.4.34) can be expressed in terms of the $L^2(\Omega)$ scalar product. Throughout this section, we require that Π^2 and $\Pi^{2,*}$ are L^2-stable, and Π^1, Π^2 and Π^{2*} have good approximation properties, summarizing our request in the following assumption.

Assumption 2.3. *The projections Π^2 and $\Pi^{2,*}$ are L^2-stable, that is*

$$\|\Pi^2(\phi)\|_{L^2(\Omega)} \leq \|\phi\|_{L^2(\Omega)}, \quad \textit{and} \quad \|\Pi^{2,*}(\psi)\|_{L^2(\Omega)} \leq \|\psi\|_{L^2(\Omega)}, \quad \forall \phi,\, \psi \in L^2(\Omega).$$

Moreover, given $\mathbf{w} \in \mathbf{H}^m(\mathrm{div};\Omega)$ *and* $\psi \in H^m(\Omega)$*, for* $0 \leq l \leq m \leq p$*, and* $p = \min_{i=1}^{d}\{p_i\}$*, there exists a constant* C *such that it holds:*

$$\|(Id - \Pi^1)\mathbf{w}\|_{\mathbf{H}^l(\mathrm{div};\Omega)} \leq Ch^{m-l}\|\mathbf{w}\|_{\mathbf{H}^m(\mathrm{div};\Omega)}, \tag{2.4.35a}$$

$$\|(Id - \Pi^2)\psi\|_{H^l(\Omega)} \leq Ch^{m-l}\|\psi\|_{H^m(\Omega)}, \tag{2.4.35b}$$

$$\|(Id - \Pi^{2,*})\psi\|_{H^l(\Omega)} \leq Ch^{m-l}\|\psi\|_{H^m(\Omega)}, \tag{2.4.35c}$$

where Id *is the identity operator, and* h *is the mesh size.*

Projections that commute with the divergence operator and satisfy L^2-stability, together with (2.4.35a) and (2.4.35b), are well known in the literature, see [24, Remark 5.1]. The usefulness of this working hypothesis is thus to ensure (2.4.35c). One option is to explore quasi-interpolant operators constructed with biorthogonal dual bases, the latter having the capability of reproducing polynomials, see [90].

Let us assume from now on that the classical solution is sufficiently regular, for instance $\mathbf{v} \in \mathbf{H}^m(\mathrm{div};\Omega)$ and $\phi \in H^m(\Omega)$, for $0 \leq m \leq p$ and p as in Assumption 2.3. We can start bounding (2.4.33) by using the triangular inequality, and separate the error in the following way:

$$\|\mathbf{v} - \mathbf{v}_h, \phi - \phi_h\|_{\infty,E} \leq \|\mathbf{v} - \mathbf{v}_h^P, \phi - \phi_h^P\|_{\infty,E} + \|\mathbf{v}_h^P - \mathbf{v}_h, \phi_h^P - \phi_h\|_{\infty,E}, \tag{2.4.36}$$

where $\mathbf{v}_h^P$ and ϕ_h^P are suitable approximations of the fields $\mathbf{v}$ and ϕ. Here we fix $\mathbf{v}_h^P := \Pi^1(\mathbf{v})$ and $\phi_h^P := \Pi^2(\phi)$, but other approximation choices that satisfy (2.4.35a) and (2.4.35b) can also be considered. We have the following approximation result.

Lemma 2.2. *There exists a constant* $C > 0$ *such that, for* $0 \leq m \leq p$ *and* p *as in Assumption 2.3, it holds*

$$\|\mathbf{v} - \mathbf{v}_h^P, \phi - \phi_h^P\|_{\infty,E} \leq Ch^m\|\mathbf{v}, \phi\|_{\infty,\mathcal{H}^m}, \tag{2.4.37a}$$

$$\|\phi - \phi_h^P,\ \mathrm{div}(c\mathbf{v} - c\mathbf{v}_h^P)\|_{\infty,E} \leq Ch^m\|\mathbf{v}, \phi\|_{\infty,\mathcal{H}^m}, \tag{2.4.37b}$$

$$\|(\mathbf{v} - \mathbf{v}_h^P)_t, (\phi - \phi_h^P)_t\|_{\infty,E} \leq Ch^m\|\mathbf{v}_t, \phi_t\|_{\infty,\mathcal{H}^m}. \tag{2.4.37c}$$

Proof. In order to bound (2.4.37a) we argue like this:

$$\begin{aligned}\|\mathbf{v} - \mathbf{v}_h^P, \phi - \phi_h^P\|_{\infty,E}^2 &\overset{\mathrm{def}}{=} \sup_{t\in[0,T]} \left(\|\mathbf{v} - \mathbf{v}_h^P\|_{\mathbf{L}^2(\Omega)}^2 + \|\phi - \phi_h^P\|_{L^2(\Omega)}^2\right) \\ &\leq \sup_{t\in[0,T]} \left(Ch^{2m}\|\mathbf{v}\|_{\mathbf{H}^m(\mathrm{div};\Omega)}^2 + Ch^{2m}\|\phi\|_{H^m(\Omega)}^2\right) \\ &= Ch^{2m}\|\mathbf{v}, \phi\|_{\infty,\mathcal{H}^m}^2,\end{aligned}$$

where we used only (2.4.35a) and (2.4.35b), and taking the square roots we proved (2.4.37a). The proof of (2.4.37c) is analogous. As regards (2.4.37b) we have that

$$\|\phi - \phi_h^P, \mathrm{div}(c\mathbf{v} - c\mathbf{v}_h^P)\|_{\infty,E}^2 = \sup_{t\in[0,T]} \left(\|\phi - \phi_h^P\|_{L^2(\Omega)}^2 + \|\mathrm{div}(c\mathbf{v} - c\mathbf{v}_h^P)\|_{L^2(\Omega)}^2\right).$$

By applying the Leibniz rule and Young's inequality, $(a + b)^2 \leq 2a^2 + 2b^2$ for each $a, b > 0$, on the last term of the right hand side we have

$$\begin{aligned}\|\mathrm{div}(c\mathbf{v} - c\mathbf{v}_h^P)\|_{L^2(\Omega)}^2 &= \left(\|c\ \mathrm{div}(\mathbf{v} - \mathbf{v}_h^P)\|_{L^2(\Omega)} + \|\boldsymbol{\nabla}c \cdot (\mathbf{v} - \mathbf{v}_h^P)\|_{L^2(\Omega)}\right)^2 \\ &\leq 2\|c\ \mathrm{div}(\mathbf{v} - \mathbf{v}_h^P)\|_{L^2(\Omega)}^2 + 2\|\boldsymbol{\nabla}c \cdot (\mathbf{v} - \mathbf{v}_h^P)\|_{L^2(\Omega)}^2 \\ &\leq C\|\mathbf{v} - \mathbf{v}_h^P\|_{\mathbf{H}^0(\mathrm{div};\Omega)}^2,\end{aligned}$$

because c is smooth and uniformly bounded. Now again we can apply (2.4.35a) and (2.4.35b), which gives us

$$\begin{aligned}\|\phi - \phi_h^P, \mathrm{div}(c\mathbf{v} - c\mathbf{v}_h^P)\|_{\infty,E}^2 &\leq C \sup_{t\in[0,T]} \left(\|\phi - \phi_h^P\|_{L^2(\Omega)}^2 + \|\mathbf{v} - \mathbf{v}_h^P\|_{\mathbf{H}^0(\mathrm{div};\Omega)}^2 \right) \\ &\leq C \sup_{t\in[0,T]} \left(h^{2m}\|\phi\|_{H^m(\Omega)}^2 + h^{2m}\|\mathbf{v}\|_{\mathbf{H}^m(\mathrm{div};\Omega)}^2 \right) \\ &\leq Ch^{2m}\|\mathbf{v}, \phi\|_{\infty,\mathcal{H}^m}^2.\end{aligned}$$

and again taking the square roots we end the proof of this lemma. □

Next we state the main result of this chapter.

Theorem 2.3. *Under Assumption 2.3, together with the commutativity of the projectors,* $\Pi^2\mathrm{div} = \mathrm{div}\Pi^1$*, given* $\mathbf{v} \in \mathbf{H}^m(\mathrm{div};\Omega)$ *and* $\phi \in H^m(\Omega)$*, it holds that*

$$\|\mathbf{v} - \mathbf{v}_h, \phi - \phi_h\|_{\infty,E} \leq Ch^{m-1}\|\mathbf{v}, \phi\|_{W^{1,\infty},\mathcal{H}^m}. \tag{2.4.38}$$

Proof. Let us consider both the weak formulation (2.1.3) and the semi-discrete problem (2.4.32). With the notation $(\cdot,\cdot)$ indicating the L^2 scalar product over the physical domain Ω we can write:

$$\begin{aligned}((\mathbf{v}_h)_t, \mathbf{w}_h) + \left(\Pi^2\mathrm{div}(c\mathbf{w}_h), \phi_h\right) &= (\mathbf{v}_t, \mathbf{w}_h) + (\mathrm{div}(c\mathbf{w}_h), \phi)\,, \\ ((\phi_h)_t, \psi_h) - \left(\Pi^2\mathrm{div}(c\mathbf{v}_h), \psi_h\right) &= (\phi_t, \psi_h) - (\mathrm{div}(c\mathbf{v}), \psi_h)\,,\end{aligned}$$

for all $\mathbf{w}_h \in X_h^1$ and $\psi_h \in X_h^2$. We can respectively subtract from the previous equations the following quantities:

$$\left((\mathbf{v}_h^P)_t, \mathbf{w}_h\right) + \left(\Pi^2\mathrm{div}(c\mathbf{w}_h), \phi_h^P\right), \quad \text{and} \quad \left((\phi_h^P)_t, \psi_h\right) - \left(\Pi^2\mathrm{div}(c\mathbf{v}_h^P), \psi_h\right),$$

and rearrange the two equations such that it holds that

$$\begin{aligned}&\left((\mathbf{v}_h - \mathbf{v}_h^P)_t, \mathbf{w}_h\right) + \left(\Pi^2\mathrm{div}(c\mathbf{w}_h), (\phi_h - \phi_h^P)\right) \\ &= \left((\mathbf{v} - \mathbf{v}_h^P)_t, \mathbf{w}_h\right) + \left(\Pi^2\mathrm{div}(c\mathbf{w}_h), (\phi - \phi_h^P)\right) + \left((Id - \Pi^2)\mathrm{div}(c\mathbf{w}_h), \phi\right),\end{aligned} \tag{2.4.39}$$

and

$$\begin{aligned}&\left((\phi_h - \phi_h^P)_t, \psi_h\right) - \left(\Pi^2\mathrm{div}(c\mathbf{v}_h - c\mathbf{v}_h^P), \psi_h\right) \\ &\quad = \left((\phi - \phi_h^P)_t, \psi_h\right) - \left(\Pi^2\mathrm{div}(c\mathbf{v} - c\mathbf{v}_h^P), \psi_h\right) - \left((Id - \Pi^2)\mathrm{div}(c\mathbf{v}), \psi_h\right).\end{aligned} \tag{2.4.40}$$

By making the choice $\mathbf{w}_h = \mathbf{v}_h - \mathbf{v}_h^P$ and $\psi_h = \phi_h - \phi_h^P$, and adding together the equations we get

$$\begin{aligned}&\frac{1}{2}\frac{\partial}{\partial t}\|\mathbf{v}_h - \mathbf{v}_h^P, \phi_h - \phi_h^P\|_E^2 \\ &\qquad = \left((\mathbf{v} - \mathbf{v}_h^P)_t, \mathbf{v}_h - \mathbf{v}_h^P\right) + \left((\phi - \phi_h^P)_t, \phi_h - \phi_h^P\right) \\ &\qquad + \left(\Pi^2\mathrm{div}(c\mathbf{v}_h - c\mathbf{v}_h^P), \phi - \phi_h^P\right) - \left(\Pi^2\mathrm{div}(c\mathbf{v} - c\mathbf{v}_h^P), \phi_h - \phi_h^P\right) \\ &\qquad + \left((Id - \Pi^2)\mathrm{div}(c\mathbf{v}_h - c\mathbf{v}_h^P), \phi\right) - \left((Id - \Pi^2)\mathrm{div}(c\mathbf{v}), \phi_h - \phi_h^P\right).\end{aligned} \tag{2.4.41}$$

Here, by construction $\Pi^2\text{div} = \text{div}\,\Pi^1$. Combining this with the definition of the formal adjoint operator, we can rewrite the last two terms on the right hand side as

$$\begin{aligned}\big((Id-\Pi^2)\text{div}(c\mathbf{v}_h - c\mathbf{v}_h^P),\phi\big) &= \big(\text{div}(c\mathbf{v}_h^P - c\mathbf{v}_h),(Id-\Pi^{2,*})\phi\big)\,,\\ \big((Id-\Pi^2)\text{div}(c\mathbf{v}),\phi_h-\phi_h^P\big) &= \big(\text{div}(Id-\Pi^1)(c\mathbf{v}),\phi_h-\phi_h^P\big)\,.\end{aligned} \tag{2.4.42}$$

Now on the right hand side we can apply the Cauchy-Schwarz inequality, which gives

$$\begin{aligned}\frac{1}{2}\frac{\partial}{\partial t}\|\mathbf{v}_h-\mathbf{v}_h^P,\phi_h-\phi_h^P\|_E^2 \le{}& \|(\mathbf{v}-\mathbf{v}_h^P)_t\|_{\mathbf{L}^2(\Omega)}\|\mathbf{v}_h-\mathbf{v}_h^P\|_{\mathbf{L}^2(\Omega)}\\ &+\|(\phi-\phi_h^P)_t\|_{L^2(\Omega)}\|\phi_h-\phi_h^P\|_{L^2(\Omega)}\\ &+\|(\phi-\phi_h^P)\|_{L^2(\Omega)}\|\Pi^2\text{div}\left(c\mathbf{v}_h^P-c\mathbf{v}_h\right)\|_{L^2(\Omega)}\\ &+\|\Pi^2\text{div}(c\mathbf{v}-c\mathbf{v}_h^P)\|_{L^2(\Omega)}\|\phi_h-\phi_h^P\|_{L^2(\Omega)}\\ &+\|(Id-\Pi^{2,*})\phi\|_{L^2(\Omega)}\|\text{div}(c\mathbf{v}_h^P-c\mathbf{v}_h)\|_{L^2(\Omega)}\\ &+\|\text{div}(Id-\Pi^1)(c\mathbf{v})\|_{L^2(\Omega)}\|\phi_h-\phi_h^P\|_{L^2(\Omega)}.\end{aligned}$$

By boundedness of the projections as in Assumption 2.3 and using boundedness of the coefficient c, together with inverse inequalities, there exists a constant $C > 0$ such that:

$$\begin{aligned}\frac{1}{2}\frac{\partial}{\partial t}\|\mathbf{v}_h-\mathbf{v}_h^P,\phi_h-\phi_h^P\|_E^2 \le{}& \|(\mathbf{v}-\mathbf{v}_h^P)_t\|_{\mathbf{L}^2(\Omega)}\|\mathbf{v}_h-\mathbf{v}_h^P\|_{\mathbf{L}^2(\Omega)}\\ &+\|(\phi-\phi_h^P)_t\|_{L^2(\Omega)}\|\phi_h-\phi_h^P\|_{L^2(\Omega)}\\ &+Ch^{-1}\|(\phi-\phi_h^P)\|_{L^2(\Omega)}\|\mathbf{v}_h-\mathbf{v}_h^P\|_{\mathbf{L}^2(\Omega)}\\ &+C\|\text{div}(c\mathbf{v}-c\mathbf{v}_h^P)\|_{L^2(\Omega)}\|\phi_h-\phi_h^P\|_{L^2(\Omega)}\\ &+Ch^{-1}\|(Id-\Pi^{2,*})\phi\|_{L^2(\Omega)}\|\mathbf{v}_h-\mathbf{v}_h^P\|_{\mathbf{L}^2(\Omega)}\\ &+\|\text{div}(Id-\Pi^1)(c\mathbf{v})\|_{L^2(\Omega)}\|\phi_h-\phi_h^P\|_{L^2(\Omega)}.\end{aligned}$$

Again by Cauchy-Schwarz inequality, applied to ordered couples of terms in the right hand side, we have:

$$\begin{aligned}\frac{1}{2}\frac{\partial}{\partial t}\|\mathbf{v}_h-\mathbf{v}_h^P,\ \phi_h-\phi_h^P\|_E^2 \le{}& \|(\mathbf{v}-\mathbf{v}_h^P)_t,\ (\phi-\phi_h^P)_t\|_E\|\mathbf{v}_h-\mathbf{v}_h^P,\ \phi_h-\phi_h^P\|_E\\ &+Ch^{-1}\|(\phi-\phi_h^P),\ \text{div}(c\mathbf{v}-c\mathbf{v}_h^P)\|_E\|\mathbf{v}_h-\mathbf{v}_h^P,\ \phi_h-\phi_h^P\|_E\\ &+Ch^{-1}\|(Id-\Pi^{2,*})\phi,\ \text{div}(Id-\Pi^1)(c\mathbf{v})\|_E\|\mathbf{v}_h-\mathbf{v}_h^P,\ \phi_h-\phi_h^P\|_E.\end{aligned}$$

Dividing both sides by $\|\mathbf{v}_h-\mathbf{v}_h^P,\ \phi_h-\phi_h^P\|_E$ and integrating in time in $[0,t]$, we obtain

$$\begin{aligned}\|\mathbf{v}_h-\mathbf{v}_h^P,\ \phi_h-\phi_h^P\|_E \le{}& \|\mathbf{v}_h(\boldsymbol{x},0)-\mathbf{v}_h^P(\boldsymbol{x},0),\ \phi_h(\boldsymbol{x},0)-\phi_h^P(\boldsymbol{x},0)\|_E\\ &+\int_0^t \|(\mathbf{v}-\mathbf{v}_h^P)_t,\ (\phi-\phi_h^P)_t\|_E\text{ds}\\ &+\int_0^t Ch^{-1}\|(\phi-\phi_h^P),\ \text{div}(c\mathbf{v}-c\mathbf{v}_h^P)\|_E\text{ds}\\ &+\int_0^t Ch^{-1}\|(Id-\Pi^{2,*})\phi,\ \text{div}(Id-\Pi^1)(c\mathbf{v})\|_E\text{ds}.\end{aligned}$$

Specifying the initial condition for problem (2.4.32) to be $\big(\mathbf{v}_h^P(\boldsymbol{x},0), \phi_h^P(\boldsymbol{x},0)\big)$ it holds $\|\mathbf{v}_h(\boldsymbol{x},0) - \mathbf{v}_h^P(\boldsymbol{x},0),\ \phi_h(\boldsymbol{x},0) - \phi_h^P(\boldsymbol{x},0)\|_E = 0$. Since the previous estimate is valid for almost every $t \in [0,T]$ we can take the maximum, and there exists a new constant $C > 0$ such that

$$\begin{aligned}\|\mathbf{v}_h - \mathbf{v}_h^P,\ \phi_h - \phi_h^P\|_{\infty,E} \leq C&\|(\mathbf{v} - \mathbf{v}_h^P)_t,\ (\phi - \phi_h^P)_t\|_{\infty,E}\\ &+Ch^{-1}\|(\phi - \phi_h^P),\ \mathrm{div}(c\mathbf{v} - c\mathbf{v}_h^P)\|_{\infty,E}\\ &+Ch^{-1}\|(Id - \Pi^{2,*})\phi,\ \mathrm{div}(Id - \Pi^1)(c\mathbf{v})\|_{\infty,E}.\end{aligned}$$

Now the left hand side in (2.4.38) splits as in (2.4.36), therefore we have

$$\begin{aligned}\|\mathbf{v} - \mathbf{v}_h,\ \phi - \phi_h\|_{\infty,E} \leq \|\mathbf{v} - \mathbf{v}_h^P,\ &\phi - \phi_h^P\|_{\infty,E}\\ &+C\|(\mathbf{v} - \mathbf{v}_h^P)_t,\ (\phi - \phi_h^P)_t\|_{\infty,E}\\ &+Ch^{-1}\|(\phi - \phi_h^P),\ \mathrm{div}(c\mathbf{v} - c\mathbf{v}_h^P)\|_{\infty,E}\\ &+Ch^{-1}\|(Id - \Pi^{2,*})\phi,\ \mathrm{div}(Id - \Pi^1)(c\mathbf{v})\|_{\infty,E}.\end{aligned}$$

Finally, from Lemma 2.2 and Assumption 2.3 we obtain

$$\|\mathbf{v} - \mathbf{v}_h,\ \phi - \phi_h\|_{\infty,E} \leq C\Big(h^{m-1}\|\mathbf{v},\ \phi\|_{\infty,\mathcal{H}^m} + h^m\|\mathbf{v}_t,\ \phi_t\|_{\infty,\mathcal{H}^m}\Big),$$

which concludes the proof of this theorem. □

Remark 2.3. *The suboptimality of the estimate comes from using inverse inequalities to control the divergence of the error. Optimality can be recovered if we define suitable approximations $\mathbf{v}_h^P$ and ϕ_h^P that control the divergence, such as in [18].*

Remark 2.4. *The projections $\mathfrak{L}\Pi^1$ and $\mathfrak{L}\Pi^2$, that we implement and test numerically, do not satisfy Assumption 2.3, therefore we directly move to next section, dedicated to error estimates for these projections.*

2.4.2 Weaker approximation assumptions

In this section we prove a convergence result for weaker assumptions than the ones in Assumption 2.3. The reason is that the practical interest is on projections in discrete spaces that have good approximation properties and are computationally fast and efficient to compute, see [68, 62]. A very general family of such projections are the quasi-interpolators proposed in [73], that where applied in [62]. Since we do not have good approximation properties of the adjoint projection of such quasi-interpolants, we will require here to relax the assumptions we made in the previous section.

Assumption 2.4. *The projections Π^1 and Π^2 satisfy*

$$\|\Pi^1\|_{\mathbf{L}^\infty(\widetilde{\mathbf{K}})\to\mathbf{L}^\infty(\mathbf{K})} = C_1, \quad \textit{and} \quad \|\Pi^2\|_{L^\infty(\widetilde{\mathbf{K}})\to L^\infty(\mathbf{K})} = C_2.$$

where $\mathbf{K} \in \mathcal{M}$ and $\widetilde{\mathbf{K}}$ is its support extension. Moreover, given $\mathbf{w} \in \mathbf{H}^m(\mathrm{div};\Omega)$ and $\psi \in H^m(\Omega)$, with at least $m \geq 2$, for $0 \leq l < m \leq p$, and $p = \min_{i=1}^d\{p_i\}$, there exists a constant C such that it holds:

$$\|(Id - \Pi^1)\mathbf{w}\|_{\mathbf{H}^l(\mathrm{div};\Omega)} \leq Ch^{m-l}\|\mathbf{w}\|_{\mathbf{H}^m(\mathrm{div};\Omega)}, \tag{2.4.43a}$$

$$\|(Id - \Pi^2)\psi\|_{H^l(\Omega)} \leq Ch^{m-l}\|\psi\|_{H^m(\Omega)}, \tag{2.4.43b}$$

where Id is the identity operator, and h is the mesh size.

Notice that, the projectors $\mathfrak{L}\Pi^1$ and $\mathfrak{L}\Pi^2$ satisfy Assumption 2.4, but not Assumption 2.3. We consider again $\mathbf{v}_h^P = \Pi^1(\mathbf{v})$ and $\phi_h^P = \Pi^2(\phi)$, and we notice that under Assumption 2.4 it still holds lemma 2.2. Therefore we have the following theorem.

Theorem 2.4. *Under Assumption 2.4, together with the commutativity of the projectors,* $\Pi^2\mathrm{div} = \mathrm{div}\Pi^1$*, given* $\mathbf{v} \in \mathbf{H}^m(\mathrm{div};\Omega)$ *and* $\phi \in H^m(\Omega)$*, with at least* $m \geq 2$*, and* $c \in \mathcal{C}^\infty(\Omega)$*, it holds that*

$$\|\mathbf{v} - \mathbf{v}_h, \phi - \phi_h\|_{\infty,E} \leq Ch\|\mathbf{v}, \phi\|_{W^{1,\infty},\mathcal{H}^m}. \tag{2.4.44}$$

Proof. The first steps of the proof are as in Theorem 2.3 until the choice of the test functions, and in particular (2.4.41) is valid. Then, by assumption we have $\Pi^2\mathrm{div} = \mathrm{div}\Pi^1$, and by integrating by parts we obtain

$$\begin{aligned}\left((Id - \Pi^2)\mathrm{div}(c\mathbf{v}_h - c\mathbf{v}_h^P), \phi\right) &= -\left((Id - \Pi^1)(c\mathbf{v}_h - c\mathbf{v}_h^P), \boldsymbol{\nabla}\phi\right)\\ \left((Id - \Pi^2)\mathrm{div}(c\mathbf{v}), \phi_h - \phi_h^P\right) &= \left(\mathrm{div}(Id - \Pi^1)(c\mathbf{v}), \phi_h - \phi_h^P\right),\end{aligned}$$

note the difference in the first equation with respect to (2.4.42). By using Cauchy-Schwarz inequality as in the proof of Theorem 2.3, we have

$$\begin{aligned}\frac{1}{2}\frac{\partial}{\partial t}\|\mathbf{v}_h - \mathbf{v}_h^P, \phi_h - \phi_h^P\|_E^2 \leq{}& \|(\mathbf{v} - \mathbf{v}_h^P)_t\|_{\mathbf{L}^2(\Omega)}\|\mathbf{v}_h - \mathbf{v}_h^P\|_{\mathbf{L}^2(\Omega)}\\ &+\|(\phi - \phi_h^P)_t\|_{L^2(\Omega)}\|\phi_h - \phi_h^P\|_{L^2(\Omega)}\\ &+\|\phi - \phi_h^P\|_{L^2(\Omega)}\|\Pi^2\mathrm{div}(c\mathbf{v}_h - c\mathbf{v}_h^P)\|_{\mathbf{L}^2(\Omega)}\\ &+\|\Pi^2\mathrm{div}(c\mathbf{v} - c\mathbf{v}_h^P)\|_{L^2(\Omega)}\|\phi_h - \phi_h^P\|_{L^2(\Omega)}\\ &+\|\phi\|_{H^1(\Omega)}\|(Id - \Pi^1)(c\mathbf{v}_h - c\mathbf{v}_h^P)\|_{\mathbf{L}^2(\Omega)}\\ &+\|\mathrm{div}(Id - \Pi^1)(c\mathbf{v})\|_{L^2(\Omega)}\|\phi_h - \phi_h^P\|_{L^2(\Omega)},\end{aligned}$$

and the only difference so far with respect to the proof of the previous theorem is in the fifth term of the right hand side. In order to bound this last one, we can apply the superconvergence results stated in [16, Theorem 2.2], that is

$$\|(Id - \Pi^1)(c\mathbf{v}_h - c\mathbf{v}_h^P)\|_{\mathbf{L}^2(\Omega)} \leq Ch\|\mathbf{v}_h - \mathbf{v}_h^P\|_{\mathbf{L}^2(\Omega)}. \tag{2.4.45}$$

The other difference with respect to theorem 2.3 is the stability assumption of Π^2, for which we bound $\|\Pi^2\mathrm{div}(c\mathbf{v}_h - c\mathbf{v}_h^P)\|_{\mathbf{L}^2(\Omega)}$ by local arguments. Given $\mathbf{K} \in \mathcal{M}$, since $|\mathbf{K}| \leq h^d$ where d is the dimension of Ω, we first apply Hölder inequality, then stability of Assumption 2.4 together with uniform boundedness of c, and finally inverse inequalities for splines, that is

$$\begin{aligned}\|\Pi^2\mathrm{div}(c\mathbf{v}_h - c\mathbf{v}_h^P)\|_{\mathbf{L}^2(\mathbf{K})}^2 &\leq |\mathbf{K}|\ \|\Pi^2\mathrm{div}(c\mathbf{v}_h - c\mathbf{v}_h^P)\|_{\mathbf{L}^\infty(\mathbf{K})}^2\\ &\leq h^d C_2^2\|\nabla c\cdot(\mathbf{v}_h - \mathbf{v}_h^P) + c\,\mathrm{div}(\mathbf{v}_h - \mathbf{v}_h^P)\|_{L^\infty(\widetilde{\mathbf{K}})}^2\\ &\leq h^d C\left(\|\mathbf{v}_h - \mathbf{v}_h^P\|_{\mathbf{L}^\infty(\widetilde{\mathbf{K}})} + \|\mathrm{div}(\mathbf{v}_h - \mathbf{v}_h^P)\|_{L^\infty(\widetilde{\mathbf{K}})}\right)^2\\ &\leq h^d C\left(\|\mathbf{v}_h - \mathbf{v}_h^P\|_{\mathbf{L}^\infty(\widetilde{\mathbf{K}})} + h^{-1}\|\mathbf{v}_h - \mathbf{v}_h^P\|_{\mathbf{L}^\infty(\widetilde{\mathbf{K}})}\right)^2\\ &\leq h^d C h^{-2}\|\mathbf{v}_h - \mathbf{v}_h^P\|_{\mathbf{L}^\infty(\widetilde{\mathbf{K}})}^2\\ &\leq Ch^{-2}\|\mathbf{v}_h - \mathbf{v}_h^P\|_{\mathbf{L}^2(\widetilde{\mathbf{K}})}^2.\end{aligned}$$

Notice that in the last inequality it is essential to have a shape regularity assumption of the mesh. Taking the square root and by standard arguments we have immediately the global inequality

$$\|\Pi^2 \mathrm{div}(c\mathbf{v}_h - c\mathbf{v}_h^P)\|_{\mathbf{L}^2(\Omega)} \leq Ch^{-1}\|\mathbf{v}_h - \mathbf{v}_h^P\|_{\mathbf{L}^2(\Omega)}.$$

Recall that $\mathbf{v}_h^P = \Pi^1(\mathbf{v})$ and $\phi_h^P = \Pi^2(\phi)$, together with $\mathrm{div}\Pi^1 = \Pi^2\mathrm{div}$. Again, with the same arguments, and with $|\mathbf{K}| \leq h^d$, we have

$$\begin{aligned}
\|\Pi^2 \mathrm{div}(c\mathbf{v} - c\mathbf{v}_h^P)\|^2_{L^2(\mathbf{K})} &\leq h^d \|\Pi^2 \mathrm{div}(c\mathbf{v} - c\mathbf{v}_h^P)\|^2_{L^\infty(\mathbf{K})} \\
&\leq h^d C_2^2 \|\nabla c \cdot (\mathbf{v} - \mathbf{v}_h^P) + c\,\mathrm{div}(\mathbf{v} - \mathbf{v}_h^P)\|^2_{L^\infty(\widetilde{\mathbf{K}})} \\
&\leq h^d C \left(\|\mathbf{v} - \mathbf{v}_h^P\|_{\mathbf{L}^\infty(\widetilde{\mathbf{K}})} + \|\mathrm{div}(\mathbf{v} - \mathbf{v}_h^P)\|_{L^\infty(\widetilde{\mathbf{K}})} \right)^2 \\
&\leq h^d C \left(\|(Id - \Pi^1)\mathbf{v}\|^2_{\mathbf{L}^\infty(\widetilde{\mathbf{K}})} + \|(Id - \Pi^2)\mathrm{div}(\mathbf{v})\|^2_{L^\infty(\widetilde{\mathbf{K}})} \right) \\
&\leq h^d C \left(h^{2m-d} \|\mathbf{v}\|^2_{\mathbf{H}^m(\widetilde{\widetilde{\mathbf{K}}})} + h^{2m-d} \|\mathrm{div}(\mathbf{v})\|^2_{H^m(\widetilde{\widetilde{\mathbf{K}}})} \right) \\
&\leq C h^{2m} \|\mathbf{v}\|^2_{H^m(\mathrm{div};\widetilde{\widetilde{\mathbf{K}}})},
\end{aligned}$$

where $\widetilde{\widetilde{\mathbf{K}}}$ is the support extension of $\widetilde{\mathbf{K}}$. Notice that we used Cauchy-Schwarz inequality on fourth row, and the local approximation estimates of the kind [78, Theorem 10.2] on fifth row. Taking the square root and by standard arguments this global inequality is straightforward

$$\|\Pi^2 \mathrm{div}(c\mathbf{v} - c\mathbf{v}_h^P)\|_{L^2(\Omega)} \leq Ch^m \|\mathbf{v}\|_{H^m(\mathrm{div};\Omega)}.$$

We bound the remaining of approximation errors as it is done for Lemma 2.2, and together with the above inequalities, we have

$$\begin{aligned}
\frac{1}{2}\frac{\partial}{\partial t}\|\mathbf{v}_h - \mathbf{v}_h^P, \phi_h - \phi_h^P\|^2_E \leq Ch^m \|\mathbf{v}_t\|_{\mathbf{H}^m(\mathrm{div};\Omega)} \|\mathbf{v}_h - \mathbf{v}_h^P\|_{\mathbf{L}^2(\Omega)} \\
+Ch^m \|\phi_t\|_{H^m(\Omega)} \|\phi_h - \phi_h^P\|_{L^2(\Omega)} \\
+Ch^{m-1} \|\phi\|_{H^m(\Omega)} \|\mathbf{v}_h - \mathbf{v}_h^P\|_{\mathbf{L}^2(\Omega)} \\
+Ch^m \|\mathbf{v}\|_{\mathbf{H}^m(\mathrm{div};\Omega)} \|\phi_h - \phi_h^P\|_{L^2(\Omega)} \\
+Ch \|\phi\|_{H^1(\Omega)} \|\mathbf{v}_h - \mathbf{v}_h^P\|_{\mathbf{L}^2(\Omega)} \\
+Ch^m \|\mathbf{v}\|_{\mathbf{H}^m(\mathrm{div};\Omega)} \|\phi_h - \phi_h^P\|_{L^2(\Omega)}.
\end{aligned}$$

By using Cauchy-Schwarz inequality and adding together the similar terms, we have

$$\begin{aligned}
\frac{1}{2}\frac{\partial}{\partial t}\|\mathbf{v}_h - \mathbf{v}_h^P, \phi_h - \phi_h^P\|^2_E \leq Ch^m \|\mathbf{v}_t,\ \phi_t\|_{\mathcal{H}^m} \|\mathbf{v}_h - \mathbf{v}_h^P, \phi_h - \phi_h^P\|_E \\
+ Ch\|\mathbf{v},\ \phi\|_{\mathcal{H}^m} \|\mathbf{v}_h - \mathbf{v}_h^P, \phi_h - \phi_h^P\|_E.
\end{aligned}$$

Dividing by $\|\mathbf{v}_h - \mathbf{v}_h^P, \phi_h - \phi_h^P\|_E$ and integrating in time, we get

$$\begin{aligned}
\|\mathbf{v}_h - \mathbf{v}_h^P, \phi_h - \phi_h^P\|_E \leq \|\mathbf{v}_h(\boldsymbol{x}, 0) - \mathbf{v}_h^P(\boldsymbol{x}, 0), \phi_h(\boldsymbol{x}, 0) - \phi_h^P(\boldsymbol{x}, 0)\|_E \\
+ Ch^m \int_0^t \|\mathbf{v}_t,\ \phi_t\|_{\mathcal{H}^m} \mathrm{ds} + Ch \int_0^t \|\mathbf{v},\ \phi\|_{\mathcal{H}^m} \mathrm{ds}.
\end{aligned}$$

Specifying the initial condition for problem (2.4.32) to be $\big(\mathbf{v}_h^P(\boldsymbol{x},0), \phi_h^P(\boldsymbol{x},0)\big)$, it holds $\|\mathbf{v}_h(\boldsymbol{x},0) - \mathbf{v}_h^P(\boldsymbol{x},0),\ \phi_h(\boldsymbol{x},0) - \phi_h^P(\boldsymbol{x},0)\|_E = 0$. Taking the maximum over $[0,T]$, we have

$$\|\mathbf{v}_h - \mathbf{v}_h^P, \phi_h - \phi_h^P\|_{\infty,E} \leq Ch\|\mathbf{v},\ \phi\|_{W^{1,\infty},\mathcal{H}^m}. \tag{2.4.46}$$

By putting together (2.4.36), Lemma 2.2 and (2.4.46) we end up proving (2.4.44). □

Remark 2.5. *We have proved linear convergence under h refinement for the semi-discretization* (2.3.25) *using projections as in Assumption 2.4. This is the case of the projections $\mathfrak{L}\Pi^1$ and $\mathfrak{L}\Pi^2$. However, in Section 2.6 we investigate numerically this error bound, and show high order rates of convergence. There is numerical evidence that $\big((Id - \Pi^1)(c\mathbf{v}_h - c\mathbf{v}_\mathbf{h}^\mathbf{P}), \nabla\phi\big) \approx h^p$, while $\|(Id - \Pi^1)(c\mathbf{v}_h - c\mathbf{v}_h^P)\|_{\mathbf{L}^2(\Omega)}$ depends linearly on the mesh size h.*

2.5 Implementation

In this section we introduce the matrix form associated to (2.3.30), giving further details about the computation of the projections. Let us start from equation (2.3.30), that must hold for all $\mathbf{w}_h \in X_h^1$. Consider as test functions $\{\mathbf{b}_{i,h}\}_{i=1}^N$, the basis functions of X_h^1, which are the push-forward with the Piola transformation map ι^1 of the B-splines on the parametric domain. To assemble the matrices involved in (2.3.30) we compute the projections of the basis functions

$$\Pi^1(c\mathbf{b}_{i,h}) = \sum_{l=1}^N \theta_l^i \mathbf{b}_{l,h},$$

and we denote by $\boldsymbol{\theta}^i = (\theta_1^i, \dots, \theta_N^i)^T$, the column vectors with the coefficients of the projections $\Pi^1(c\mathbf{b}_{i,h})$ into the space X_h^1. We then define the matrix $\tilde{\mathbf{A}} \in \mathbb{R}^{N\times N}$ that encapsulates the second term of (2.3.30), and with the previous notation each entry of the matrix can be computed as

$$\begin{aligned}[\tilde{\mathbf{A}}]_{i,j} &\coloneqq \int_\Omega \operatorname{div}\big(\Pi^1(c\mathbf{b}_{i,h})\big) \operatorname{div}\big(\Pi^1(c\mathbf{b}_{j,h})\big)\, \mathrm{d}\boldsymbol{x} \\ &= \sum_{l=1}^N \sum_{m=1}^N \theta_l^i \theta_m^j \int_\Omega \operatorname{div}(\mathbf{b}_{l,h})\operatorname{div}(\mathbf{b}_{m,h})\mathrm{d}\boldsymbol{x} = (\boldsymbol{\theta}^i)^T \mathbf{A} \boldsymbol{\theta}^j,\end{aligned}$$

for $i,j = 1,\dots,N$, with $\mathbf{A} \in \mathbb{R}^{N\times N}$ defined as $[\mathbf{A}]_{l,m} \coloneqq \int_\Omega \operatorname{div}(\mathbf{b}_{l,h})\operatorname{div}(\mathbf{b}_{m,h})\mathrm{d}\boldsymbol{x}$. It is therefore convenient to store the coefficients of the projectors in the matrix $\boldsymbol{\Theta} = [\boldsymbol{\theta}^1|\dots|\boldsymbol{\theta}^N] \in \mathbb{R}^{N\times N}$, from which we obtain $\tilde{\mathbf{A}} = \boldsymbol{\Theta}^T\mathbf{A}\boldsymbol{\Theta}$.

For the computation of the projections, we make use of the commutativity property (2.3.18), from which we obtain the two equivalent expressions

$$\iota^1(\Pi^1(c\mathbf{b}_{i,h})) = \widehat{\Pi}^1(\iota^1(c\mathbf{b}_{i,h})) = \widehat{\Pi}^1(\widehat{c\mathbf{b}}_{i,h}) = \sum_{l=1}^N \widehat{\theta}_l^i \widehat{\mathbf{b}}_{l,h},$$

$$\iota^1(\Pi^1(c\mathbf{b}_{i,h})) = \iota^1(\sum_{l=1}^N \theta_l^i \mathbf{b}_{l,h}) = \sum_{l=1}^N \theta_l^i \iota^1(\mathbf{b}_{l,h}) = \sum_{l=1}^N \theta_l^i \widehat{\mathbf{b}}_{l,h},$$

where $\widehat{\theta}_l^i$ are the coefficients of the projection $\widehat{\Pi}^1(\widehat{c}\widehat{\mathbf{b}}_{i,h})$, and $\widehat{c} = c \circ \boldsymbol{F}$. Notice that $\widehat{\theta}_l^i = \theta_l^i$, and therefore the projections can be computed in the parametric domain, exploiting the tensor-product structure. The details on the computation of these projections are given in appendix A. We finally recall that, in order to commute with the divergence operator, the computation of the projection involves (2.3.10), or (2.3.11) for the periodic case, both of them requiring the application of a quadrature formula.

Finally, by introducing the notation $\underline{\mathbf{v}}^n$ and $\underline{\phi}^n$ for the coefficients of the unknown fields $\mathbf{v}_h^n$ and ϕ_h^n respectively, equation (2.3.30) can be written in the following matrix form:

$$\left(\mathbf{M} + \frac{(\Delta t)^2}{4}\boldsymbol{\Theta}^T\mathbf{A}\boldsymbol{\Theta}\right)\underline{\mathbf{v}}^{n+1} = \left(\mathbf{M} - \frac{(\Delta t)^2}{4}\boldsymbol{\Theta}^T\mathbf{A}\boldsymbol{\Theta}\right)\underline{\mathbf{v}}^n - \Delta t\boldsymbol{\Theta}^T\mathbf{B}\underline{\phi}^n, \quad \text{for } n = 0, \dots, N-1,$$

where the matrix $\mathbf{M} \in \mathbb{R}^{N \times N}$ denotes the mass matrix for the space X_h^1, and $\mathbf{B} \in \mathbb{R}^{N \times M}$ is defined as $[\mathbf{B}]_{i,j} = \int_\Omega b_{j,h}\text{div}(\mathbf{b}_{i,h})\text{d}\boldsymbol{x}$.

Remark 2.6. *Notice that the computation of the involved matrices is independent of time, that is,* $\mathbf{M}, \mathbf{A}, \mathbf{B}$ *and* $\boldsymbol{\Theta}$ *can be computed only once at the beginning of our method. If the coefficient* c *is time dependent, it is necessary to recompute only* $\boldsymbol{\Theta}$ *at every time step.*

2.6 Numerical results

In this section, we have numerically investigated the approximation properties of the method by conducting academic tests to determine the convergence order under h-refinements. Furthermore, concerning the employment of the quasi-interpolant in Section 2.3.2 within the described numerical method, even if it is tested on a simplified model problem, we have supplemented the array of numerical tests conducted in [62]. Indeed, their studies observed the favorable approximation properties of the quasi-interpolant but not of the overall global method. Lastly, we have verified the conservation of the total energy for this method. All the numerical tests have been performed in Matlab, with the isogeometric analysis open source package GeoPDEs [107].

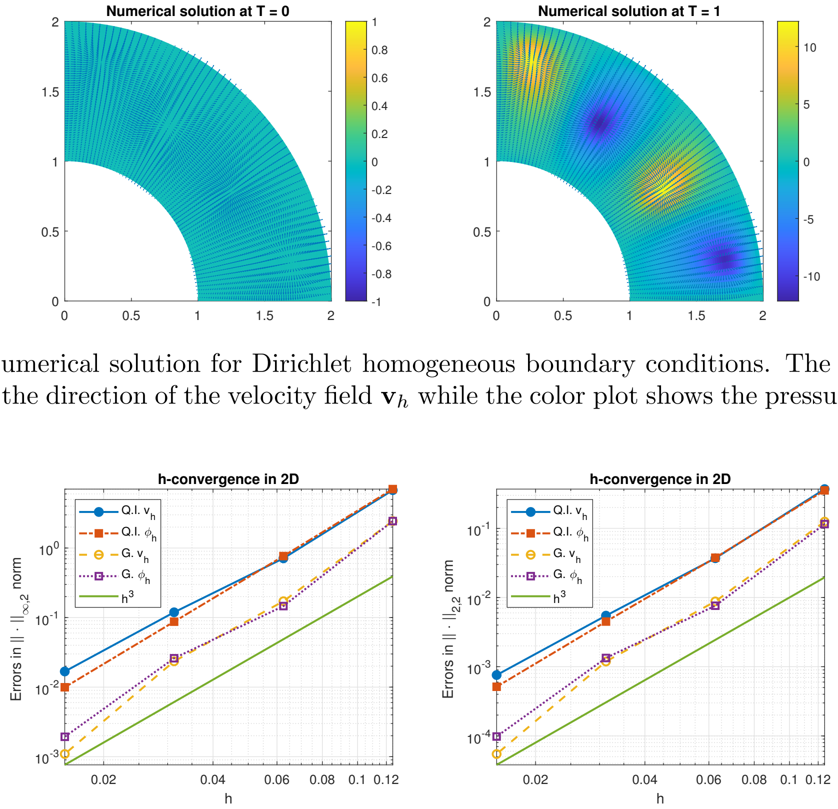
2.6.1 Dirichlet boundary conditions

We consider the wave equation as presented in Section 2.1. The domain is $\Omega = \boldsymbol{F}(\widehat{\Omega})$, where $\widehat{\Omega} = [0,1]^2$ and $\boldsymbol{F}$ is the NURBS map describing the quarter of a ring, as in Figure 2.1. We assume a full Dirichlet boundary with homogeneous conditions, that is $\Gamma_D = \partial\Omega$ and $g = 0$ in (2.1.1). The time interval is $[0, T]$, where we fix $T = 1$. We consider a space dependent coefficient $c(x_1, x_2) := sin(2\pi x_1)sin(2\pi x_2) + 2$, which is smooth, with bounded derivatives and bounded away from zero.

We assume that the solution is of the form $u(\boldsymbol{x}, t) = \chi(\boldsymbol{x})Re(e^{i\omega t})$, with $\chi \in H_0^1(\Omega)$ solution of

$$\int_\Omega c^2\nabla\chi \cdot \nabla\Phi\text{d}\boldsymbol{x} = \omega^2\int_\Omega \chi\Phi\text{d}\boldsymbol{x}, \quad \forall\Phi \in H_0^1(\Omega). \tag{2.6.47}$$

Since χ is not known explicitly, we use as reference solution its approximation with splines of degree $\boldsymbol{p} = (6,6)$ in a uniform mesh with mesh width $h = 1/128$. We

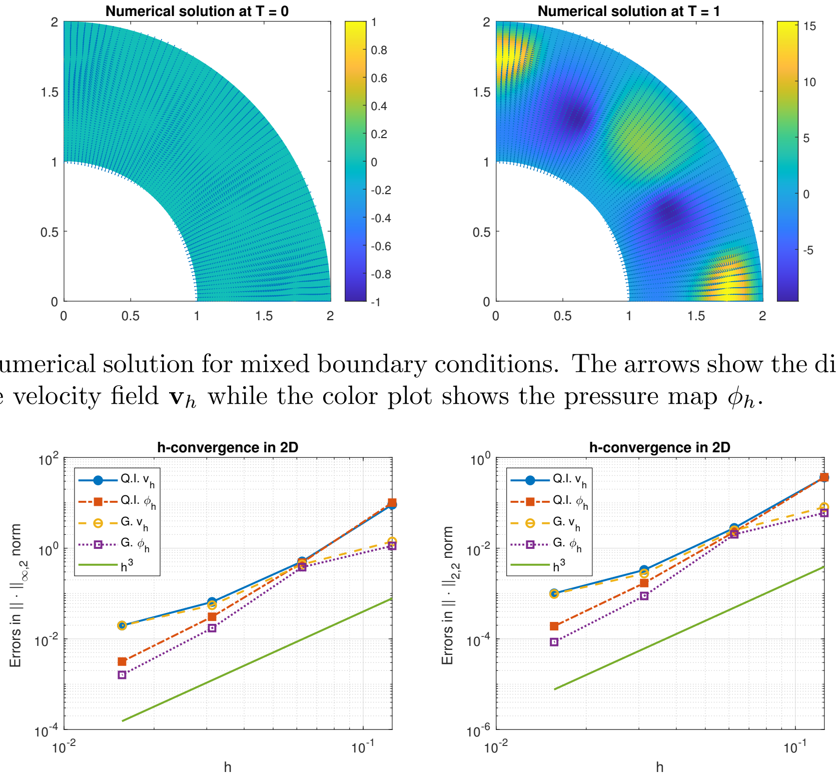


(a) Numerical solution for Dirichlet homogeneous boundary conditions. The arrows show the direction of the velocity field $\mathbf{v}_h$ while the color plot shows the pressure map ϕ_h.

(b) *Left* - Errors in $\|\cdot\|_{\infty,2}$ norm for solutions with quasi-interpolant and Galerkin methods with homogeneous Dirichlet boundary conditions. *Right* - Errors with $\|\cdot\|_{2,2}$ norm for the same problems.

Figure 2.6. (a) Initial data and numerical solutions of (2.3.29) at $T = 1$ with $c = sin(2\pi x_1)sin(2\pi x_2) + 2$ for Dirichlet homogeneous boundary conditions. (b) h-convergence rate estimation for both quasi-interpolant (Q.I.) and Galerkin (G) methods.

compute the reference solution, χ_h, relative to the fourth smallest eigenvalue, that is $\omega = 4\pi$. Our reference solution is $u(\boldsymbol{x}, t) = \chi_h(\boldsymbol{x})Re(e^{i\omega t})$, and its velocity and pressure fields are respectively $\mathbf{v}(\boldsymbol{x}, t) = c(\boldsymbol{x})\nabla\chi_h(\boldsymbol{x})Re(e^{i\omega t})$ and $\phi(\boldsymbol{x}, t) = \chi_h(\boldsymbol{x})Re(i\omega e^{i\omega t})$. In order to study the convergence of our method, we estimate the error between the numerical solution and the reference one for different uniform mesh sizes. We let the space mesh width h vary in $\{1/8, 1/16, 1/32, 1/64\}$. We choose a uniform partition τ of the interval $[0, T]$ with $\Delta t = 5 \times 10^{-4}$. We fix $p = 3$, and we set the initial data from the reference solution, and they can be seen in Figure 2.6a, for $h = 1/64$. The plot on the right of Figure 2.6a shows the solutions obtained at the final time. For each value of h that we are considering, we record $\|\mathbf{v} - \mathbf{v}_h, \phi - \phi_h\|_{\infty,E}$. The plot on the left of Figure 2.6b shows the two components of the computed errors, that are $\|\mathbf{v} - \mathbf{v}_h\|_{\infty,2} := \sup_{t\in\tau} \|\mathbf{v} - \mathbf{v}_h\|_{\mathbf{L}^2(\Omega)}$ and $\|\phi - \phi_h\|_{\infty,2} := \sup_{t\in\tau} \|\phi - \phi_h\|_{L^2(\Omega)}$. We compare these errors with the ones obtained from solving (2.3.31), i.e., the standard Galerkin method. The plot shows third order convergence rates both for our discretization and for Galerkin. In order to investigate also convergence with L^2 norm in time we will measure the errors in the discrete norm $\|f(\boldsymbol{x}, t)\|_{2,2} := \frac{\Delta t}{2} \sum_{t\in\tau\setminus\{T\}} (\|f(\boldsymbol{x}, t)\|^2_{L^2(\Omega)} + \|f(\boldsymbol{x}, t + k)\|^2_{L^2(\Omega)})^{\frac{1}{2}}$. We define the analogous norms for the vector fields. The plot on the right of Figure



(a) Numerical solution for mixed boundary conditions. The arrows show the direction of the velocity field $\mathbf{v}_h$ while the color plot shows the pressure map ϕ_h.

(b) *Left* - Errors in $\|\cdot\|_{\infty,2}$ norm for solutions with quasi-interpolant and Galerkin methods with mixed boundary conditions. *Right* - Errors with $\|\cdot\|_{2,2}$ norm for the same problems.

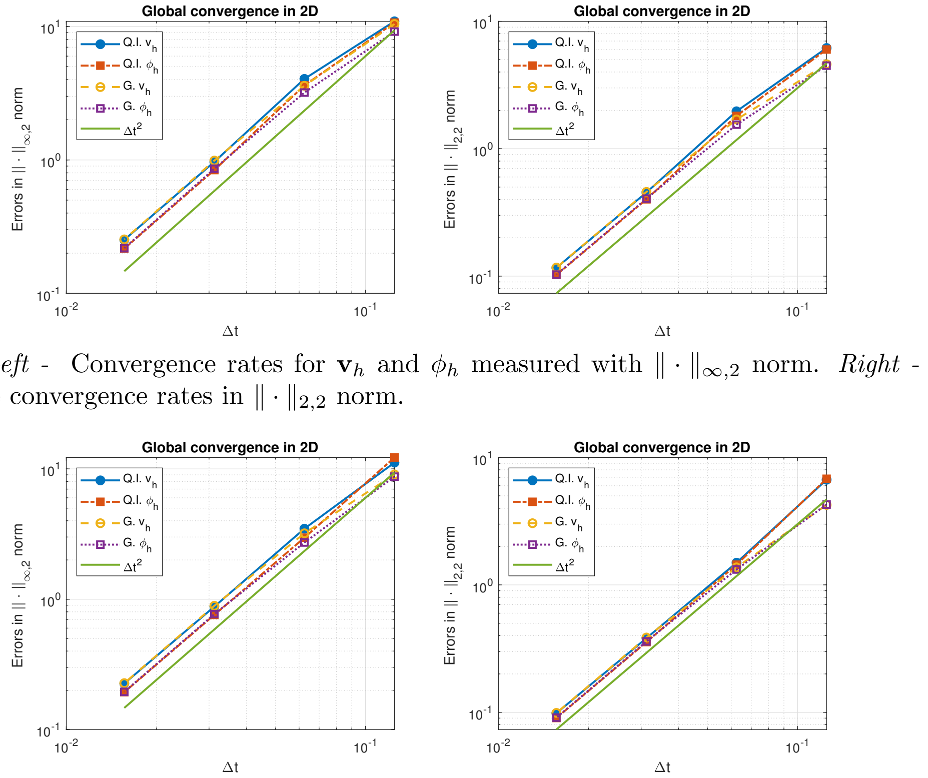
Figure 2.7. (a) Initial data and numerical solutions of (2.3.29) at $T = 1$ with $c = sin(2\pi x_1)sin(2\pi x_2) + 2$ for mixed boundary conditions. (b) h-convergence rate estimation for both quasi-interpolant (Q.I.) and Galerkin (G) methods.

2.6b shows the same error study under h-refinement using the norm $\|\cdot\|_{2,2}$. We obtain the same error convergence rates, that are of order h^p, better then what predicted in the Theorem 2.4.

2.6.2 Mixed boundary conditions

In this second example we consider the domain as above. The boundary of Ω is defined as in Figure 2.1, and we impose homogeneous Dirichlet conditions on Γ_D, and periodic conditions on Γ_P. The time interval is $[0, T]$, with $T = 1$. Here we used the same coefficient for the example in Section 2.6.1, since it is periodic in Γ_P. We use separation of variables to construct the reference solution, as it was done in the example of Section 2.6.1. The reference solution is $u(\boldsymbol{x}, t) = \chi_h(\boldsymbol{x})Re(e^{i\omega t})$, where $\omega = 4\pi$ and χ_h is the approximation with splines of the solution of (2.6.47) with mixed Dirichlet and periodic boundary condition. In order to have a fine approximation, we used $\boldsymbol{p} = (6, 6)$ and a uniform mesh with mesh width $h = 1/128$.

As it was done for the previous example, we study the convergence of the numerical scheme by estimating the error between the numerical solution and the reference one. We discretize the problem as in (2.3.29), choosing $p = 3$ and a uniform mesh. We let the space mesh width h vary in $\{1/8, 1/16, 1/32, 1/64\}$. We choose a uniform partition τ of the interval $[0, T]$ with time step size $\Delta t = 5 \times 10^{-4}$. We compute



(a) *Left* - Convergence rates for $\mathbf{v}_h$ and ϕ_h measured with $\|\cdot\|_{\infty,2}$ norm. *Right* - Same convergence rates in $\|\cdot\|_{2,2}$ norm.

(b) *Left* - convergence rates for $\mathbf{v}_h$ and ϕ_h measured with $\|\cdot\|_{\infty,2}$ norm. *Right* - same convergence rates in $\|\cdot\|_{2,2}$ norm.

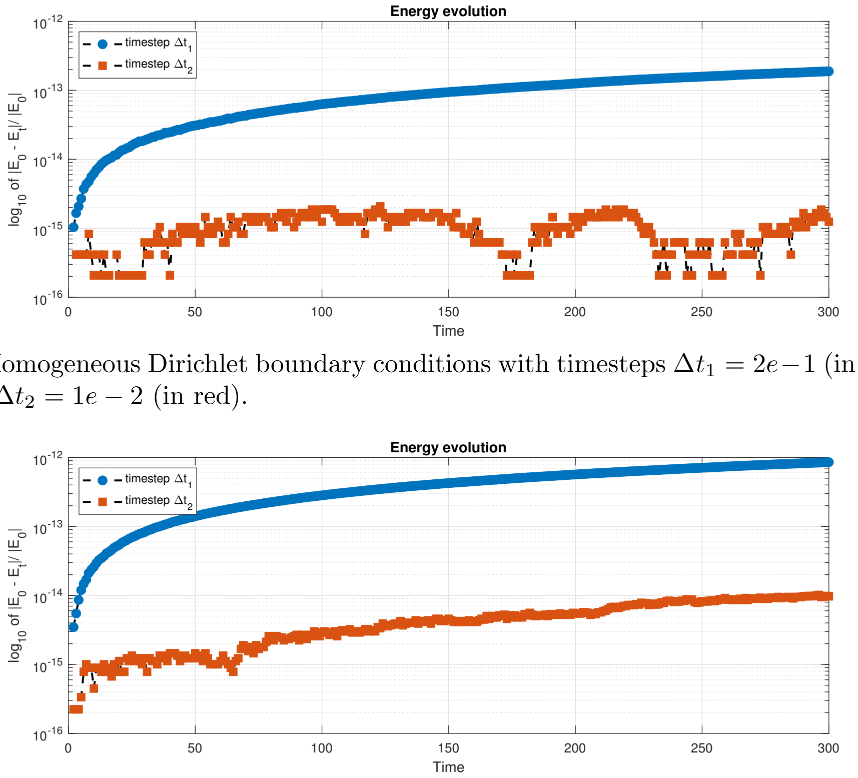
Figure 2.8. Global convergence rate estimation, for quasi-interpolant (Q.I.) and Galerkin (G) methods, for homogeneous Dirichlet boundary conditions (a), while mixed boundary conditions in (b)

the initial data from the reference solution, and they can be seen in Figure 2.7a, for $h = 1/64$. The plot on the right of Figure 2.7a shows the solutions obtained at the final time.

As it was done for the example in Section 2.6.1, for each value of h that we are considering we compute the errors in the energy norm. We also measured the errors with the discrete norm $\|\cdot\|_{2,2}$. Figure 2.7b shows the two components of the computed errors, compared with the ones obtained from solving with the standard Galerkin method. Here it seems the error convergence rate is close to a third order of convergence. Although in the last refinement step the convergence is reduced, the same behavior is observed for the solution with the standard Galerkin method.

2.6.3 Convergence study under time refinement

So long we discussed the h-convergence of the quasi-interpolant method. Here we check the global convergence rate when refining both the mesh size h and the time step Δt. We let vary h as before from $1/8$ to $1/64$, this time taking the time step $\Delta t = h$. We show in Figure 2.8 the convergence of both fields $\mathbf{v}_h$ and ϕ_h in the same norms introduced above, for the example of Section 2.6.1 in Figure 2.8a, and for the example of Section 2.6.2 in Figure 2.8b. The obtained convergence rate is of second order, as expected from the use of Crank-Nicolson method.



(a) Homogeneous Dirichlet boundary conditions with timesteps $\Delta t_1 = 2e{-}1$ (in blue) and $\Delta t_2 = 1e - 2$ (in red).

(b) Mixed boundary conditions with timesteps $\Delta t_1 = 2e{-}1$ (in blue) and $\Delta t_2 = 1e{-}2$ (in red).

Figure 2.9. Energy conservation plots for Dirichlet homogeneous boundary conditions (a) and mixed boundary conditions (b)

2.6.4 Energy conservation

In Remark 2.2 we point out that our numerical scheme is preserving the total energy of the system, as defined in (2.1.4). Here we check the energy conservation for long time simulations, hence we fix $T = 300$ and choose two different uniform partitions τ_1 and τ_2 of the interval $[0, T]$, the first with step $\Delta t_1 = 0.2$, and the second with step size $\Delta t_2 = 0.01$. We also fix the mesh size $h = 1/32$. Recall that we consider as reference solution a stationary wave with a fixed time-frequency $\omega = 4\pi$, and the solution evolves in time as $\Psi(t) = Re(e^{i\omega t})$, therefore it performs around two complete oscillations for each unit interval. We solve problem (2.3.29) and compute the energy as in (2.1.4) for every $t_n \in \tau \cap \mathbb{Z}$. In Figure 2.9 we show the evolution of the relative errors $\frac{|E_0 - E_{t_n}|}{|E_0|}$, where E_{t_n} is the energy evaluated at time t_n. Figure 2.9a shows the evolution of the relative energy errors in semi-logarithmic scale for the solution of the problem with homogeneous Dirichlet boundary conditions, as the example in Section 2.6.1. We can see that for both time steps the error remains at the level of round-off errors, with lower numbers for the finer time grid. The same plots are reproduced in Figure 2.9b for the solutions of the problem with mixed boundary conditions of Section 2.6.2, and we observe a similar behavior. Notice that on the finer grid, for both Dirichlet and mixed boundary conditions, we performed 30000 steps in time without losing energy. The increasing behavior of the errors for the coarse grid solutions seems only due to accumulation of round-off errors. We can conclude that the method preserves the total energy of the system as we expected.

2.7 Conclusions

In this chapter we proposed an isogeometric discretization of the wave equation in mixed form, with mixed Dirichlet and periodic boundary conditions. The method relies on tensor product projections into spline spaces with good approximation properties and that commute with the divergence operator, according to the De Rham complex for splines. The conservation of the total energy of the system is imposed weakly by modifying the differential equations according to the energy constraint and to projections operators that we introduced. Regarding time discretization we employed Crank-Nicolson method, which is of second order and energy conservative, though alternative conservative methods could be selected.

The method employed for energy preservation has been previously proposed in the field of plasma physics and magnetohydrodynamics, see for example the discretization proposed by[68] for the Vlasov-Maxwell equations and [62] for magnetohydrodynamics. The novelty of this thesis, except for the simplified model problem, is the a priori error estimate analysis for the full method, not just for the projection approximation. In cases where the introduced projections, in addition to commutativity with the divergence operator, exhibit L^2-stability and preserve splines, we assume good approximation properties of the adjoint projection operator to establish that the method is of high order – specifically, h^{m-1} assuming the solutions reside in $H^m(\Omega)$ and its vectorial counterpart. However, the stability requirement in L^2-norm, is not always guaranteed, as exemplified by the projections introduced in Section 2.3.2. In light of this, under the less stringent assumption that the projections are locally stable in the uniform norm, without requiring approximation properties of the adjoint projection operators, we prove that the method converges, at least linearly in h.

Numerical tests conducted indicate that, in scenarios where we expect linear convergence, the method exhibits high order convergence in h, matching the order of convergence achieved through a Galerkin approximation without projections. This observation suggests that the proposed error estimate might be improved, representing a potential avenue for future development.

Finally, the conservation of energy is also confirmed by numerical results, where the relative error remains of the order of machine precision at the final time.

Chapter 3

Heat equation

This chapter serves as both a review of preconditioners for isogeometric space-time discretizations of the heat equation and an introduction to novel preconditioning techniques tailored for the aforementioned problem.

Space-time finite element methods originated in the papers [47, 22, 89], where standard finite elements are assigned an extra dimension for the time and, typically, adopt a discontinuous approximation in time, since this produces a time marching algorithm with a traditional step-by-step format (see e.g. [97]).

One of the first work concerning space-time isogeometric discretization is [71], in which a stabilized variational formulation produces a discrete bilinear form that is elliptic with respect to a discrete energy norm. The resulting linear system is then solved through a standard parallel Algebraic MultiGrid (AMG) preconditioned GMRES solver. Other papers in literature propose isogeometric space-time Galerkin methods, favoring a step-by-step structure in time. In [72], the same variational formulation of [71] is used in combination with a space-time domain decomposition into space-time slabs that are sequentially coupled in time by a stabilized discontinuous Galerkin method. In [101] the authors outlined two different space–time computation techniques with continuous representation in time (ST-C methods), respectively, with a successive-projection technique (ST-C-SPT) and with a direct-computation technique (ST-C-DCT).

The first one, analysed in [106], is a way to project a previously computed solution, possibly discontinuous, into isogeometric spaces in order to get a more regular solution and to save memory for its storage.

According to the ST-C-DCT method, the solution with continuous temporal representation is computed sequentially from the space-time variational formulation associated with each slab.

Related multigrid solvers have been proposed in [50, 60] and low-rank approximations in [83]. In [19] the authors consider C^0 coupling between the space-time slabs with a suitable stabilized formulation that also yields a sequential scheme. Finally, the interest in space-time isogeometric analysis for complex real-world simulations is attested by the recent papers [102, 103, 104], where, again, a sequential (discontinuous) approximation in time is adopted.

In this chapter we focus on the heat equation and on its space-time isogeometric discretizations, that allows smooth approximation in both space and time. This originated in [85], focusing on a L^2 least squares formulation, that is obtained minimizing the L^2-norm of the residual in the space-time domain, while in [75], the authors focused on the plain Galerkin space-time formulation, whose well-posedness has been studied, for finite element discretizations and for the heat equation, in the recent papers [99] and [100]. Here we consider both this formulations, focusing on the study of stable preconditioning strategies. Indeed, when adopting smooth approximation in space and in time, the major issue is its computational cost and the key ingredient is an efficient solver for the linear system, which is global in time.

The preconditioners proposed by the authors in [85] and [75], exploit the Kronecker structure of the arising linear systems, extending the original idea in [80]. For the plain Galerkin formulation, and assuming that the spatial domain does not change with time, the linear system has the structure

$$\mathbf{A} := \gamma \mathbf{W}_t \otimes \mathbf{M}_s + \nu \mathbf{M}_t \otimes \mathbf{L}_s, \tag{3.0.1}$$

where $\mathbf{W}_t$ is given by the discretization of the time derivative, $\mathbf{L}_s$ is given by the discretization of the Laplacian in the spatial variables, $\mathbf{M}_t$ and $\mathbf{M}_s$ are *mass* matrices in time and space, that are respectively, the matrix representations of the L^2 scalar product, and $\gamma, \nu > 0$ are constants of the problem. The construction of a preconditioner for (3.0.1) is based on a generalization of the classical Fast Diagonalization (FD) method [79]. Indeed, the FD method cannot be directly applied to (3.0.1), as this would require to compute the eigendecomposition of the pencil $(\mathbf{W}_t, \mathbf{M}_t)$ which is numerically unstable. In [75] the authors circumvent this difficulty by introducing an ad-hoc factorization of the time matrices which allows to design a solver conceptually similar to the FD method.

For the L^2 least squares formulation, instead, the linear system has the structure

$$\mathbf{B} := \gamma^2 \mathbf{L}_t \otimes \mathbf{M}_s + \nu^2 \mathbf{M}_t \otimes \mathbf{J}_s + \gamma\nu \mathbf{R}_t \otimes \mathbf{L}_s, \tag{3.0.2}$$

where $\mathbf{L}_t$ is the matrix discretizing the second derivative in time (with initial conditions), $\mathbf{L}_s$ is the stiffness matrix in the space variable, $\mathbf{R}_t$ is a rank 1 matrix associated to the final time, and $\mathbf{J}_s$ is given by the discretization of the Bi-Laplacian in the spatial variables. Again, $\mathbf{M}_t$ and $\mathbf{M}_s$ are *mass* matrices in time and space, respectively, and $\gamma, \nu > 0$ are constants of the problem. Thus, the problem becomes elliptic and a preconditioner for the linear system associated to (3.0.2) can be easily designed as in [93], again based on the FD method.

Furthermore, in the case of the plain Galerkin formulation, we can consider the Galerkin (or projected) L^2 least squares form associated to the system (3.0.1), that is a linear system with matrix

$$\mathbf{A}^T(\mathbf{M}_t \otimes \mathbf{M}_s)^{-1}\mathbf{A} := \gamma^2 \mathbf{W}_t^T \mathbf{M}_t^{-1} \mathbf{W}_t \otimes \mathbf{M}_s + \nu^2 \mathbf{M}_t \otimes \mathbf{L}_s \mathbf{M}_s^{-1} \mathbf{L}_s + \gamma\nu \mathbf{R}_t \otimes \mathbf{L}_s.$$

This has a structure similar to (3.0.2). In particular both discrete problems are now elliptic, and the third term appearing is the Kronecker product between a rank 1 matrix (in time) and the stiffness matrix (in space).

The main contribution of this chapter, is the design of two new preconditioners, one for the projected L^2 least squares formulation above and one for (3.0.2), both of them relying on FD method and Sherman-Morrison formula for matrix inversion of rank-1 perturbation. The preconditioners introduced here can easily be extended to rank-r perturbations, with $r > 1$, employing the Sherman-Morrison-Woodbury formula, as it is done for finite difference discretizations of the heat-equation in [51].

The computational cost of the setup of all mentioned preconditioners is $O(N_{dof})$ FLoating Point Operations (FLOPs), while the application cost is $O(N_{dof}^{(d+2)/(d+1)})$ FLOPs, where d is the number of spatial dimensions and N_{dof} denotes the total number of degrees-of-freedom (assuming, for simplicity, to have the same number of degrees-of-freedom in time and in each spatial direction). We report in this chapter the numerical benchmarks of [86, 75], and compare them with the performance of the proposed Sherman-Morrison preconditioners. The results show that the computing times (serial and single-core execution) are close to optimality, that is, proportional to N_{dof}, for all the above mentioned strategies. The preconditioners are also robust with respect to the polynomial degree and number of elements. Furthermore, all these approaches are optimal in terms of memory requirement: denoting by N_s the total number of degrees-of-freedom in space, the storage cost is $O(p^d N_s + N_{dof})$. We also remark that global space-time methods in principle facilitate the full parallelization of the solver, see [38, 49, 70].

The outline of this chapter is as follows. In Section 3.1 we present the basics of B-splines based IgA and the main properties of the Kronecker product operation. The space-time formulation is introduced in Section 3.2 while its plain Galerkin isogeometric discretization is given in 3.3, together with the preconditioner introduced in [75] and its application. In Section 3.4 we introduce the Sherman-Morrison preconditioner for the projected L^2 least squares form. The L^2 least squares formulation and its isogeometric discretization are introduced in Section 3.5, while in Section 3.6 we recall the preconditioner introduced in [85] and we discuss its application. In Section 3.7 we introduce the new preconditioner for the least squares formulation. Section 3.8 is devoted to the computational costs of the proposed preconditioners and to memory requirements. We present the numerical results assessing the performance of the proposed preconditioners in Section 3.9. Finally, in the last section we draw some conclusions and we highlight some future research directions.

3.1 B-splines and preliminaries

Given n and p two positive integers, we consider open knot vector Ξ in $[0,1]$, recalling that is a sequence of non-decreasing points $\Xi := \{\xi_1 \leq \cdots \leq \xi_{n+p+1}\}$, such that $\xi_1 = \cdots = \xi_{p+1} = 0$ and $\xi_n = \cdots = \xi_{n+p+1} = 1$. Then, $\widehat{b}_{i,p} : (0,1) \to \mathbb{R}$ are the univariate B-splines for $i = 1, \ldots, n$, defined according to Cox-De Boor recursion formulas (see [34]). In this chapter we denote the univariate spline space by

$$\widehat{\mathcal{S}}_h^p := \operatorname{span}\{\widehat{b}_{i,p} : i = 1, \ldots, n\},$$

where h denotes the mesh-size, i.e. $h := \max\{|\xi_{i+1} - \xi_i|,\ i = 1, \ldots, n+p\}$. The interior knot multiplicity influences the smoothness of the B-splines at the knots (see [34]). For more details on B-splines properties and their use in IgA we refer to [29].

Multivariate B-splines are defined as tensor product of univariate B-splines. We consider functions that depend on d spatial variables and the time variable. Therefore, given positive integers n_l, p_l for $l = 1, \ldots, d$ and n_t, p_t, we introduce $d+1$ univariate knot vectors $\Xi_l := \{\xi_{l,1} \leq \cdots \leq \xi_{l,n_l+p_l+1}\}$ for $l = 1, \ldots, d$ and $\Xi_t := \{\xi_{t,1} \leq \cdots \leq \xi_{t,n_t+p_t+1}\}$. Let h_l be the mesh-size associated to the knot vector Ξ_l for $l = 1, \ldots, d$, let $h_s := \max\{h_l \mid l = 1, \ldots, d\}$ be the maximal mesh-size in all spatial knot vectors and let h_t be the mesh-size of the time knot vector. Let also $\boldsymbol{p}$ be the vector that contains the degree indexes, i.e. $\boldsymbol{p} := (\boldsymbol{p}_s, p_t)$, where $\boldsymbol{p}_s := (p_1, \dots, p_d)$. For simplicity, we assume to have the same polynomial degree in all spatial directions, i.e., with abuse of notations, we set $p_1 = \dots = p_d =: p_s$, but the general case is similar.

We assume that the following quasi-uniformity of the knot vectors holds.

Assumption 3.1. *There exists $0 < \alpha \leq 1$, independent of h_s and h_t, such that each non-empty knot span $(\xi_{l,i}, \xi_{l,i+1})$ of Ξ_l fulfils $\alpha h_s \leq \xi_{l,i+1} - \xi_{l,i} \leq h_s$ for $l = 1, \dots, d$ and each non-empty knot span $(\xi_{t,i}, \xi_{t,i+1})$ of Ξ_t fulfils $\alpha h_t \leq \xi_{t,i+1} - \xi_{t,i} \leq h_t$.*

We recall the definition of multivariate B-splines is

$$\widehat{B}_{\boldsymbol{i},\boldsymbol{p}}(\boldsymbol{\eta}, \tau) := \widehat{B}_{\boldsymbol{i}_s,\boldsymbol{p}_s}(\boldsymbol{\eta}) \widehat{b}_{i_t,p_t}(\tau), \tag{3.1.3}$$

where

$$\widehat{B}_{\boldsymbol{i}_s,\boldsymbol{p}_s}(\boldsymbol{\eta}) := \widehat{b}_{i_1,p_s}(\eta_1) \dots \widehat{b}_{i_d,p_s}(\eta_d), \tag{3.1.4}$$

$\boldsymbol{i}_s := (i_1, \dots, i_d)$, $\boldsymbol{i} := (\boldsymbol{i}_s, i_t)$ and $\boldsymbol{\eta} = (\eta_1, \dots, \eta_d)$. The corresponding spline space is defined as

$$\widehat{\mathcal{S}}_h^{\boldsymbol{p}} := \operatorname{span}\left\{ \widehat{B}_{\boldsymbol{i},\boldsymbol{p}} \;\middle|\; i_l = 1, \dots, n_l \text{ for } l = 1, \dots, d; i_t = 1, \dots, n_t \right\},$$

where $h := \max\{h_s, h_t\}$. We have that $\widehat{\mathcal{S}}_h^{\boldsymbol{p}} = \widehat{\mathcal{S}}_{h_s}^{\boldsymbol{p}_s} \otimes \widehat{\mathcal{S}}_{h_t}^{p_t}$, where

$$\widehat{\mathcal{S}}_{h_s}^{\boldsymbol{p}_s} := \operatorname{span}\left\{ \widehat{B}_{\boldsymbol{i}_s,\boldsymbol{p}_s} \;\middle|\; i_l = 1, \dots, n_l; l = 1, \dots, d \right\} \tag{3.1.5}$$

is the space of tensor-product splines on $\widehat{\Omega} := (0,1)^d$.

Assumption 3.2. *We assume that $p_t, p_s \geq 1$ and that $\widehat{\mathcal{S}}_{h_s}^{\boldsymbol{p}_s} \subset C^0(\widehat{\Omega})$ and $\widehat{\mathcal{S}}_{h_t}^{p_t} \subset C^0\left((0,1)\right)$.*

3.1.1 Space-time isogeometric spaces

The space-time computational domain that we consider is $\Omega \times (0,T)$, where $\Omega \subset \mathbb{R}^d$ and $T > 0$ is the final time. We make the following assumptions.

Assumption 3.3. *We assume that Ω is parametrized by $\boldsymbol{F} : \widehat{\Omega} \to \Omega$, with $\boldsymbol{F} \in \left[\widehat{\mathcal{S}}_{h_s}^{\boldsymbol{p}_s}\right]^d$ on the closure of $\widehat{\Omega}$. Moreover, we assume that $\boldsymbol{F}^{-1}$ has piecewise bounded derivatives of any order.*

We define $\boldsymbol{x} = (x_1, \dots, x_d) := \boldsymbol{F}(\boldsymbol{\eta})$ and $t := T\tau$. Then the space-time domain is given by the parametrization $\boldsymbol{G} : \widehat{\Omega} \times (0,1) \to \Omega \times (0,T)$, such that $\boldsymbol{G}(\boldsymbol{\eta}, \tau) := (\boldsymbol{F}(\boldsymbol{\eta}), T\tau) = (\boldsymbol{x}, t)$.

We introduce the spline space with initial and boundary conditions, in parametric coordinates, as

$$\widehat{\mathcal{X}}_h := \left\{ \widehat{v}_h \in \widehat{\mathcal{S}}_h^{\boldsymbol{p}} \;\middle|\; \widehat{v}_h = 0 \text{ on } \partial\widehat{\Omega} \times (0,1) \text{ and } \widehat{v}_h = 0 \text{ on } \widehat{\Omega} \times \{0\} \right\}. \tag{3.1.6}$$

We also have that $\widehat{\mathcal{X}}_h = \widehat{\mathcal{X}}_{s,h_s} \otimes \widehat{\mathcal{X}}_{t,h_t}$, where

$$\begin{aligned}
\widehat{\mathcal{X}}_{s,h_s} &:= \left\{ \widehat{w}_h \in \widehat{\mathcal{S}}^{\boldsymbol{p}_s}_{h_s} \;\middle|\; \widehat{w}_h = 0 \text{ on } \partial\widehat{\Omega} \right\} \\
&= \ \operatorname{span}\left\{ \widehat{b}_{i_1,p_s} \dots \widehat{b}_{i_d,p_s} \;\middle|\; i_l = 2, \dots, n_l - 1; \ l = 1, \dots, d \ \right\}, \\
\widehat{\mathcal{X}}_{t,h_t} &:= \left\{ \widehat{w}_h \in \widehat{\mathcal{S}}^{p_t}_{h_t} \;\middle|\; \widehat{w}_h(0) = 0 \right\} \ = \ \operatorname{span}\left\{ \widehat{b}_{i_t,p_t} \;\middle|\; i_t = 2, \dots, n_t \ \right\}.
\end{aligned}$$

By introducing a colexicographical reordering of the basis functions, we can write

$$\begin{aligned}
\widehat{\mathcal{X}}_{s,h_s} &= \ \operatorname{span}\left\{ \widehat{b}_{i_1,p_s} \dots \widehat{b}_{i_d,p_s} \;\middle|\; i_l = 1, \dots, N_{s,l}; \ l = 1, \dots, d \ \right\} \\
&= \operatorname{span}\left\{ \widehat{B}_{i,\boldsymbol{p}_s} \;\middle|\; i = 1, \dots, N_s \ \right\}, \\
\widehat{\mathcal{X}}_{t,h_t} &= \ \operatorname{span}\left\{ \widehat{b}_{i,p_t} \;\middle|\; i = 1, \dots, N_t \ \right\},
\end{aligned}$$

and then

$$\widehat{\mathcal{X}}_h = \operatorname{span}\left\{ \widehat{B}_{i,\boldsymbol{p}_s} \;\middle|\; i = 1, \dots, N_{dof} \right\}, \tag{3.1.8}$$

where we defined $N_{s,l} := n_l - 2$ for $l = 1, \dots, d$, $N_s := \prod_{l=1}^d N_{s,l}$, $N_t := n_t - 1$ and $N_{dof} := N_s N_t$.

Finally, the isogeometric space we consider is the isoparametric push-forward of (3.1.8) through the geometric map $\boldsymbol{G}$, i.e.

$$\mathcal{X}_h := \operatorname{span}\left\{ B_{i,\boldsymbol{p}} := \widehat{B}_{i,\boldsymbol{p}} \circ \boldsymbol{G}^{-1} \;\middle|\; i = 1, \dots, N_{dof} \right\}. \tag{3.1.9}$$

We also have that $\mathcal{X}_h = \mathcal{X}_{s,h_s} \otimes \mathcal{X}_{t,h_t}$, where

$$\mathcal{X}_{s,h_s} := \operatorname{span}\left\{ B_{i,\boldsymbol{p}_s} := \widehat{B}_{i,\boldsymbol{p}_s} \circ \boldsymbol{F}^{-1} \;\middle|\; i = 1, \dots, N_s \right\} \tag{3.1.10}$$

and

$$\mathcal{X}_{t,h_t} := \operatorname{span}\left\{ b_{i,p_t} := \widehat{b}_{i,p_t}(\cdot/T) \;\middle|\; i = 1, \dots, N_t \right\}. \tag{3.1.11}$$

3.1.2 Kronecker product

The Kronecker product of two matrices $\mathbf{C} \in \mathbb{C}^{n_1 \times n_2}$ and $\mathbf{D} \in \mathbb{C}^{n_3 \times n_4}$ is defined as

$$\mathbf{C} \otimes \mathbf{D} := \begin{bmatrix} [\mathbf{C}]_{1,1}\mathbf{D} & \dots & [\mathbf{C}]_{1,n_2}\mathbf{D} \\ \vdots & \ddots & \vdots \\ [\mathbf{C}]_{n_1,1}\mathbf{D} & \dots & [\mathbf{C}]_{n_1,n_2}\mathbf{D} \end{bmatrix} \in \mathbb{C}^{n_1 n_3 \times n_2 n_4},$$

where $[\mathbf{C}]_{i,j}$ denotes the ij-th entry of the matrix $\mathbf{C}$. For extensions and properties of the Kronecker product we refer to [67]. In particular, when a matrix has a Kronecker product structure, the matrix-vector product can be efficiently computed. For this purpose, for $m = 1, \dots, d+1$ we introduce the m-mode product of a tensor $\mathfrak{X} \in \mathbb{C}^{n_1 \times \dots \times n_{d+1}}$ with a matrix $\mathbf{J} \in \mathbb{C}^{\ell \times n_m}$, that we denote by $\mathfrak{X} \times_m \mathbf{J}$. This is a tensor of size $n_1 \times \dots \times n_{m-1} \times \ell \times n_{m+1} \times \dots n_{d+1}$, whose elements are defined as

$$[\mathfrak{X} \times_m \mathbf{J}]_{i_1,\dots,i_{d+1}} := \sum_{j=1}^{n_m} [\mathfrak{X}]_{i_1,\dots,i_{m-1},j,i_{m+1},\dots,i_{d+1}} [\mathbf{J}]_{i_m,j}.$$

Then, given $\mathbf{J}_i \in \mathbb{C}^{\ell_i \times n_i}$ for $i = 1, \ldots, d+1$, it holds

$$(\mathbf{J}_{d+1} \otimes \cdots \otimes \mathbf{J}_1) \operatorname{vec}(\mathfrak{X}) = \operatorname{vec}(\mathfrak{X} \times_1 \mathbf{J}_1 \times_2 \cdots \times_{d+1} \mathbf{J}_{d+1}), \tag{3.1.12}$$

where the vectorization operator "vec" applied to a tensor stacks its entries into a column vector as

$$[\operatorname{vec}(\mathfrak{X})]_j = [\mathfrak{X}]_{i_1,\ldots,i_{d+1}} \text{ for } i_l = 1, \ldots, n_l \text{ and for } l = 1, \ldots, d+1,$$

where $j := i_1 + \sum_{k=2}^{d+1} \left[(i_k - 1)\Pi_{l=1}^{k-1} n_l\right]$.

3.2 Space-time variational formulation of the Heat equation

Our model problem is the heat equation with homogeneous boundary and initial conditions: we look for a solution u such that

$$\begin{cases} \gamma \partial_t u - \nabla \cdot (\nu \nabla u) &= f \quad \text{in} \quad \Omega \times (0,T), \\ u &= 0 \quad \text{on} \quad \partial\Omega \times [0,T], \\ u &= 0 \quad \text{in} \quad \Omega \times \{0\}, \end{cases} \tag{3.2.13}$$

where $\Omega \subset \mathbb{R}^d$, T is the final time, $\gamma > 0$ is the heat capacity constant and $\nu > 0$ is the thermal conductivity constant. We assume that $f \in L^2(0,T; H^{-1}(\Omega))$ and we introduce the Hilbert spaces

$$\mathcal{X} := \left\{ v \in L^2(0,T; H_0^1(\Omega)) \cap H^1(0,T; H^{-1}(\Omega)) \mid v(\boldsymbol{x}, 0) = 0 \right\},$$
$$\mathcal{Y} := L^2(0,T; H_0^1(\Omega)),$$

endowed with the following norms

$$\|v\|_{\mathcal{X}}^2 := \frac{\gamma^2}{\nu} \|\partial_t v\|_{L^2(0,T;H^{-1}(\Omega))}^2 + \nu \|v\|_{L^2(0,T;H_0^1(\Omega))}^2 \quad \text{and} \quad \|v\|_{\mathcal{Y}}^2 := \nu \|v\|_{L^2(0,T;H_0^1(\Omega))}^2,$$

respectively. Then, the variational formulation of (3.2.13) reads:

$$\text{Find } u \in \mathcal{X} \text{ such that } \mathcal{A}(u,v) = \mathcal{F}(v) \quad \forall v \in \mathcal{Y}, \tag{3.2.14}$$

where the bilinear form $\mathcal{A}(\cdot,\cdot)$ and the linear form $\mathcal{F}(\cdot)$ are defined $\forall w \in \mathcal{X}$ and $\forall v \in \mathcal{Y}$ as

$$\mathcal{A}(w,v) := \int_0^T \int_\Omega (\gamma \partial_t w \, v + \nu \nabla w \cdot \nabla v) \, \mathrm{d}\Omega \, \mathrm{d}t \quad \text{and} \quad \mathcal{F}(v) := \int_0^T \int_\Omega f \, v \, \mathrm{d}\Omega \, \mathrm{d}t.$$

The well-posedness of the variational formulation above is a classical result, see for example [99].

The previous setting can be generalized to non-homogeneous initial and boundary conditions. For example, suppose that in (3.2.13) we have the initial condition $u = u_0$ in $\Omega \times \{0\}$ with $u_0 \in L^2(\Omega)$. Then, we consider a lifting $\underline{u}_0$ of u_0 such that $\underline{u}_0 \in L^2(0,T; H_0^1(\Omega)) \cap H^1(0,T; H^{-1}(\Omega))$, see e.g. [45]. Finally, we split the solution u as $u = \underline{u} + \underline{u}_0$, where $\underline{u} \in \mathcal{X}$ is the solution of the following heat equation with homogeneous initial and boundary conditions:

$$\begin{cases} \gamma \partial_t \underline{u} - \nabla \cdot (\nu \nabla \underline{u}) &= \underline{f} \quad \text{in} \quad \Omega \times (0,T), \\ \underline{u} &= 0 \quad \text{on} \quad \partial\Omega \times [0,T], \\ \underline{u} &= 0 \quad \text{in} \quad \Omega \times \{0\}, \end{cases}$$

where $\underline{f} := f - \gamma \partial_t \underline{u}_0 + \nabla \cdot (\nu \nabla \underline{u}_0)$.

3.3 Space-time Galerkin method

Let $\mathcal{X}_h \subset \mathcal{X}$ be the isogeometric space defined in (3.1.9). We consider the following Galerkin method for (3.2.14):

$$\text{Find } u_h \in \mathcal{X}_h \text{ such that } \mathcal{A}(u_h, v_h) = \mathcal{F}(v_h) \quad \forall v_h \in \mathcal{X}_h. \tag{3.3.15}$$

Following [99], let $N_h : L^2(0,T;H^{-1}(\Omega)) \to \mathcal{X}_h$ be the discrete Newton potential operator, defined as follows: given $\phi \in L^2(0,T;H^{-1}(\Omega))$ then $N_h\phi \in \mathcal{X}_h$ fulfills

$$\int_0^T \int_\Omega \nu \nabla(N_h\phi) \cdot \nabla v_h \; \mathrm{d}\Omega \, \mathrm{dt} = \gamma \int_0^T \int_\Omega \phi \; v_h \; \mathrm{d}\Omega \, \mathrm{dt} \quad \forall v_h \in \mathcal{X}_h.$$

Thus, we define the norm in $\mathcal{X}_h$ as

$$\|w\|^2_{\mathcal{X}_h} := \nu \|N_h(\partial_t w)\|^2_{L^2(0,T;H^1_0(\Omega))} + \nu \|w\|^2_{L^2(0,T;H^1_0(\Omega))}.$$

The stability and the well-posedness of formulation (3.3.15) are guaranteed by [99, Equation (2.7)] and by a straightforward extension to IgA of [99, Theorem 3.1] and [99, Theorem 3.2]. We summarize these results in the following Proposition 3.1 and Theorem 3.2.

Proposition 3.1. *It holds*

$$\mathcal{A}(w,v) \leq \sqrt{2}\|w\|_{\mathcal{X}}\|v\|_{\mathcal{Y}} \quad \forall w \in \mathcal{X} \textit{ and } \forall v \in \mathcal{Y},$$

and

$$\|w_h\|_{\mathcal{X}_h} \leq 2\sqrt{2} \sup_{v_h \in \mathcal{X}_h} \frac{\mathcal{A}(w_h, v_h)}{\|v_h\|_{\mathcal{Y}}} \quad \forall w_h \in \mathcal{X}_h.$$

Theorem 3.2. *There exists a unique solution $u_h \in \mathcal{X}_h$ to the discrete problem* (3.3.15)*. Moreover, it holds*

$$\|u - u_h\|_{\mathcal{X}_h} \leq 5 \inf_{w_h \in \mathcal{X}_h} \|u - w_h\|_{\mathcal{X}},$$

where $u \in \mathcal{X}$ is the solution of (3.2.14)*.*

We have then the following a-priori estimate for h-refinement.

Theorem 3.3. *Let q be an integer such that $1 < q \leq \min\{p_s, p_t\} + 1$. If $u \in \mathcal{X} \cap H^q(\Omega \times (0,T))$ is the solution of* (3.2.14) *and $u_h \in \mathcal{X}_h$ is the solution of* (3.3.15)*, then it holds*

$$\|u - u_h\|_{\mathcal{X}_h} \leq C\sqrt{\frac{\gamma^2}{\nu} + \nu}\left(h_t^{q-1} + h_s^{q-1}\right)\|u\|_{H^q(\Omega\times(0,T))} \tag{3.3.16}$$

where C is independent of h_s, h_t, γ, ν and u.

The proof of (3.3.16) is given in [75, Theorem 2], we conclude with the following remark.

Remark 3.1. *In Theorem 3.2, the degrees p_t, p_s and the mesh-sizes h_t, h_s play a similar role. This motivates our choice $p_t = p_s =: p$ and $h_t = h_s =: h$ for the numerical tests in Section 3.9. In this case, and if the solution u is smooth,* (3.3.16) *yields h-convergence of order p. A sharper error analysis is possible taking into account a different regularity of the solution u in space and time, in the line of the anisotropic estimates of [14].*

3.3.1 Discrete system

The linear system associated to (3.3.15) is

$$\mathbf{A}\mathbf{u} = \mathbf{f}, \tag{3.3.17}$$

where $[\mathbf{A}]_{i,j} = \mathcal{A}(B_{j,\boldsymbol{p}}, B_{i,\boldsymbol{p}})$ and $[\mathbf{f}]_i = \mathcal{F}(B_{i,\boldsymbol{p}})$. The tensor-product structure of the isogeometric space (3.1.9) allows to write the system matrix $\mathbf{A}$ as sum of Kronecker products of matrices as

$$\mathbf{A} = \gamma \mathbf{W}_t \otimes \mathbf{M}_s + \nu \mathbf{M}_t \otimes \mathbf{L}_s, \tag{3.3.18}$$

where for $i, j = 1, \dots, N_t$

$$[\mathbf{W}_t]_{i,j} = \int_0^T b'_{j,p_t}(t)\, b_{i,p_t}(t)\, \mathrm{d}t \quad \text{and} \quad [\mathbf{M}_t]_{i,j} = \int_0^T b_{j,p_t}(t)\, b_{i,p_t}(t)\, \mathrm{d}t, \tag{3.3.19a}$$

while for $i, j = 1, \dots, N_s$

$$[\mathbf{L}_s]_{i,j} = \int_\Omega \nabla B_{j,\boldsymbol{p}_s}(\boldsymbol{x}) \cdot \nabla B_{i,\boldsymbol{p}_s}(\boldsymbol{x}) \ \mathrm{d}\Omega \quad \text{and} \quad [\mathbf{M}_s]_{i,j} = \int_\Omega B_{j,\boldsymbol{p}_s}(\boldsymbol{x}) \ B_{i,\boldsymbol{p}_s}(\boldsymbol{x}) \ \mathrm{d}\Omega. \tag{3.3.19b}$$

In what follows we define and investigate the application of different preconditioners for solving (3.3.17).

3.3.2 Galerkin preconditioner

As first attempt we introduce, for the system (3.3.17), the preconditioner

$$[\widehat{\mathbf{A}}]_{i,j} := \widehat{\mathcal{A}}(\widehat{B}_{j,\boldsymbol{p}}, \widehat{B}_{i,\boldsymbol{p}}),$$

where

$$\widehat{\mathcal{A}}(\widehat{v}, \widehat{w}) := \int_0^1 \int_{\widehat{\Omega}} (\gamma \partial_t \widehat{v}\, \widehat{w} + T\nu \nabla \widehat{v} \cdot \nabla \widehat{w}) \ \mathrm{d}\widehat{\Omega} \ \mathrm{d}\tau \quad \forall \widehat{v}, \widehat{w} \in \widehat{\mathcal{X}}_h.$$

We have

$$\widehat{\mathbf{A}} = \gamma \mathbf{W}_t \otimes \widehat{\mathbf{M}}_s + \nu \mathbf{M}_t \otimes \widehat{\mathbf{L}}_s, \tag{3.3.20}$$

where $\widehat{\mathbf{L}}_s$ and $\widehat{\mathbf{M}}_s$ are the equivalent of (3.3.19b) in the parametric domain, i.e. we define for $i, j = 1, \dots, N_s$

$$[\widehat{\mathbf{L}}_s]_{i,j} = \int_{\widehat{\Omega}} \nabla \widehat{B}_{j,\boldsymbol{p}_s}(\boldsymbol{\eta}) \cdot \nabla \widehat{B}_{i,\boldsymbol{p}_s}(\boldsymbol{\eta}) \ \mathrm{d}\widehat{\Omega} \quad \text{and} \quad [\widehat{\mathbf{M}}_s]_{i,j} = \int_{\widehat{\Omega}} \widehat{B}_{j,\boldsymbol{p}_s}(\boldsymbol{\eta}) \ \widehat{B}_{i,\boldsymbol{p}_s}(\boldsymbol{\eta}) \ \mathrm{d}\widehat{\Omega}. \tag{3.3.21}$$

We emphasize that the time matrices appearing in (3.3.20) are the same ones appearing in the system matrix (3.3.18). This is because for $i, j = 1, \dots, N_t$ we have

$$[\mathbf{W}_t]_{i,j} = \int_0^T b'_{j,p_t}(t) \ b_{i,p_t}(t) \ \mathrm{d}t = \int_0^1 \widehat{b}'_{j,p_t}(\tau) \ \widehat{b}_{i,p_t}(\tau) \ \mathrm{d}\tau$$

and

$$[\mathbf{M}_t]_{i,j} = \int_0^T b_{j,p_t}(t) \ b_{i,p_t}(t) \ \mathrm{d}t = T \int_0^1 \widehat{b}_{j,p_t}(\tau) \ \widehat{b}_{i,p_t}(\tau) \ \mathrm{d}\tau.$$

Thanks to (3.1.4), the spatial matrices (3.3.21) have the following structure

$$\widehat{\mathbf{L}}_s = \sum_{l=1}^{d} \widehat{\mathbf{M}}_d \otimes \cdots \otimes \widehat{\mathbf{M}}_{l+1} \otimes \widehat{\mathbf{L}}_l \otimes \widehat{\mathbf{M}}_{l-1} \otimes \cdots \otimes \widehat{\mathbf{M}}_1 \quad \text{and} \quad \widehat{\mathbf{M}}_s = \widehat{\mathbf{M}}_d \otimes \cdots \otimes \widehat{\mathbf{M}}_1, \tag{3.3.22}$$

where for $l = 1, \dots, d$ and for $i, j = 1, \dots, N_{s,l}$ we define

$$[\widehat{\mathbf{L}}_l]_{i,j} := \int_0^1 \widehat{b}'_{j,p_s}(\eta_k)\widehat{b}'_{i,p_s}(\eta_k)\ \mathrm{d}\eta_k \quad \text{and} \quad [\widehat{\mathbf{M}}_l]_{i,j} := \int_0^1 \widehat{b}_{j,p_s}(\eta_k)\widehat{b}_{i,p_s}(\eta_k)\ \mathrm{d}\eta_k.$$

The efficient application of the proposed preconditioner, that is, the solution of a linear system with matrix $\widehat{\mathbf{A}}$, should exploit the structure highlighted above. When the pencils $(\mathbf{W}_t, \mathbf{M}_t)$, $(\widehat{\mathbf{L}}_1, \widehat{\mathbf{M}}_1), \dots, (\widehat{\mathbf{L}}_d, \widehat{\mathbf{M}}_d)$ admit a stable generalized eigendecomposition, a possible approach is the Fast Diagonalization (FD) method, see [37] and [79] for details. We will see in Section 3.3.3 that the spatial pencils $(\widehat{\mathbf{L}}_1, \widehat{\mathbf{M}}_1), \dots, (\widehat{\mathbf{L}}_d, \widehat{\mathbf{M}}_d)$ admit a stable diagonalization, but this is not the case of $(\mathbf{W}_t, \mathbf{M}_t)$, that needs a special treatment as explained in Section 3.3.5.

3.3.3 Stable factorization of $(\widehat{\mathbf{L}}_l, \widehat{\mathbf{M}}_l)$ for $l = 1, \dots, d$

The spatial stiffness and mass matrices $\widehat{\mathbf{L}}_l$ and $\widehat{\mathbf{M}}_l$ are symmetric and positive definite for $l = 1, \dots, d$. Thus, the pencils $(\widehat{\mathbf{L}}_l, \widehat{\mathbf{M}}_l)$ for $l = 1, \dots, d$ admit the generalized eigendecomposition

$$\widehat{\mathbf{L}}_l \mathbf{U}_l = \widehat{\mathbf{M}}_l \mathbf{U}_l \mathbf{\Lambda}_l,$$

where the matrices $\mathbf{U}_l$ contain in each column the $\widehat{\mathbf{M}}_l$-orthonormal generalized eigenvectors and $\mathbf{\Lambda}_l$ are diagonal matrices whose entries contain the generalized eigenvalues. Therefore we have for $l = 1, \dots, d$ the factorizations

$$\mathbf{U}_l^T \widehat{\mathbf{L}}_l \mathbf{U}_l = \mathbf{\Lambda}_l \quad \text{and} \quad \mathbf{U}_l^T \widehat{\mathbf{M}}_l \mathbf{U}_l = \mathbb{I}_{N_{s,l}}, \tag{3.3.23}$$

where $\mathbb{I}_{N_{s,l}}$ denotes the identity matrix of dimension $N_{s,l} \times N_{s,l}$. Figure 3.1 shows the shape of the generalized eigenvectors in $\mathbf{U}_l$, with associated eigenvalue in Λ_l, for a fixed univariate direction $l = 1, \dots, d$ discretized with degree $p_s = 3$ B-Splines and uniform partition with $n_{el} = 32$. The stability of the decomposition (3.3.23) is expressed by the condition number of the eigenvector matrix. In particular $\mathbf{U}_l^T \widehat{\mathbf{M}}_l \mathbf{U}_l = \mathbb{I}_{N_{s,l}}$ implies that

$$\kappa_2(\mathbf{U}_l) := \|\mathbf{U}_l\|_2 \|\mathbf{U}_l^{-1}\|_2 = \sqrt{\kappa_2(\widehat{\mathbf{M}}_l)},$$

where $\| \cdot \|_2$ is the norm induced by the Euclidean vector norm. The condition number $\kappa_2(\widehat{\mathbf{M}}_l)$ has been studied in [48] and it does not depend on the mesh-size, but it depends on the polynomial degree. We report in Table 3.1 the behavior of $\kappa_2(\mathbf{U}_l)$ for different values of spline degree p_s and for different uniform discretizations with number of elements denoted by n_{el}. We observe that $\kappa_2(\mathbf{U}_l)$ exhibits a dependence only on p_s, but stays moderately low for all low polynomial degrees that are in the range of interest.

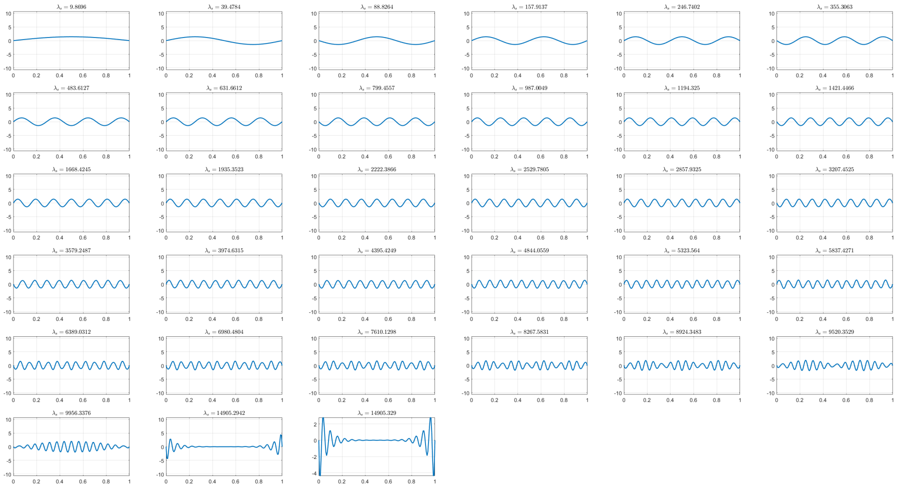


Figure 3.1. Generalized eigenvectors for the pencils $(\widehat{\mathbf{L}}_l, \widehat{\mathbf{M}}_l)$, with associated eigenvalues for $p_s = 3$ and $n_{el} = 32$ elements.

n_{el}	$p_s = 2$	$p_s = 3$	$p_s = 4$	$p_s = 5$	$p_s = 6$	$p_s = 7$	$p_s = 8$
32	$2.7 \cdot 10^0$	$4.5 \cdot 10^0$	$7.6 \cdot 10^0$	$1.3 \cdot 10^1$	$2.1 \cdot 10^1$	$3.5 \cdot 10^1$	$5.7 \cdot 10^1$
64	$2.7 \cdot 10^0$	$4.5 \cdot 10^0$	$7.6 \cdot 10^0$	$1.3 \cdot 10^1$	$2.1 \cdot 10^1$	$3.5 \cdot 10^1$	$5.7 \cdot 10^1$
128	$2.7 \cdot 10^0$	$4.5 \cdot 10^0$	$7.6 \cdot 10^0$	$1.3 \cdot 10^1$	$2.1 \cdot 10^1$	$3.5 \cdot 10^1$	$5.7 \cdot 10^1$
256	$2.7 \cdot 10^0$	$4.5 \cdot 10^0$	$7.6 \cdot 10^0$	$1.3 \cdot 10^1$	$2.1 \cdot 10^1$	$3.5 \cdot 10^1$	$5.7 \cdot 10^1$
512	$2.7 \cdot 10^0$	$4.5 \cdot 10^0$	$7.6 \cdot 10^0$	$1.3 \cdot 10^1$	$2.1 \cdot 10^1$	$3.5 \cdot 10^1$	$5.7 \cdot 10^1$
1024	$2.7 \cdot 10^0$	$4.5 \cdot 10^0$	$7.6 \cdot 10^0$	$1.3 \cdot 10^1$	$2.1 \cdot 10^1$	$3.5 \cdot 10^1$	$5.7 \cdot 10^1$

Table 3.1. $\kappa_2(\mathbf{U}_l)$ for different polynomial degrees p_s and number of elements n_{el}.

3.3.4 Instability of the eigendecomposition of $(\mathbf{W}_t, \mathbf{M}_t)$

As regards the time pencils, our first attempt was to apply the original idea of [80], which suggests to diagonalize among time direction too. Unfortunately, while $\mathbf{M}_t$ is symmetric, $\mathbf{W}_t$ is neither symmetric nor skew-symmetric. Indeed

$$[\mathbf{W}_t]_{i,j} + [\mathbf{W}_t]_{j,i} = \int_0^T b'_{j,p_t}(t)\; b_{i,p_t}(t)\; \mathrm{dt} + \int_0^T b'_{i,p_t}(t)\; b_{j,p_t}(t)\; \mathrm{dt} = b_{i,p_t}(T)\; b_{j,p_t}(T) \tag{3.3.24}$$

where $b_{i,p_t}(T)\; b_{j,p_t}(T)$ vanishes for all $i = 1, \dots, N_t - 1$ or $j = 1, \dots, N_t - 1$. A numerical computation of the generalized eigendecomposition of the pencil $(\mathbf{W}_t, \mathbf{M}_t)$, that is

$$\mathbf{W}_t \mathbf{U}_t = \mathbf{M}_t \mathbf{U}_t \boldsymbol{\Lambda}_t, \tag{3.3.25}$$

where $\boldsymbol{\Lambda}_t$ is the diagonal matrix of the generalized complex eigenvalues and $\mathbf{U}_t$ is the complex matrix whose columns are the generalized eigenvectors normalized w.r.t. the norm induced by $\mathbf{M}_t$, reveals that the eigenvectors are far from $\mathbf{M}_t$-orthogonality, i.e. the matrix ${\mathbf{U}_t}^* \mathbf{M}_t \mathbf{U}_t$ is not diagonal. In Figure 3.2 we set $T = 1$ and plot these generalized eigenvectors with associated eigenvalue, for $p_t = 3$ and uniform

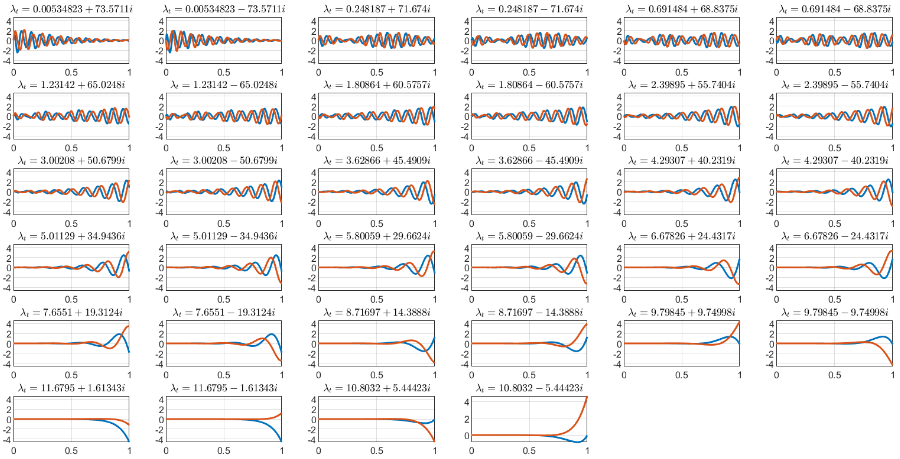


Figure 3.2. Generalized eigenvectors for the pencil $(\mathbf{W}_t, \mathbf{M}_t)$, with associated eigenvalues for $p_t = 3$ and $n_{el} = 32$ elements. The real part is expressed in blue, while the imaginary part is in red.

partition with $n_{el} = 32$. We report in Table 3.2 the condition number $\kappa_2(\mathbf{U}_t)$ for different values of spline degree p_t and for different uniform discretizations with n_{el} number of elements. In contrast to the spatial case (see Section 3.3.3), $\kappa_2(\mathbf{U}_t)$ is large and grows exponentially with respect to the spline degree p_t and the level of mesh refinement. This test clearly indicates a numerical instability when computing the generalized eigendecomposition of $(\mathbf{W}_t, \mathbf{M}_t)$. A similar behavior has also been highlighted in [60] for a SUPG stabilized modification of $\mathbf{W}_t$.

n_{el}	$p_t = 2$	$p_t = 3$	$p_t = 4$	$p_t = 5$	$p_t = 6$	$p_t = 7$	$p_t = 8$
32	$8.9 \cdot 10^2$	$3.0 \cdot 10^4$	$5.0 \cdot 10^4$	$3.4 \cdot 10^5$	$3.1 \cdot 10^6$	$4.2 \cdot 10^7$	$7.0 \cdot 10^8$
64	$4.4 \cdot 10^3$	$2.6 \cdot 10^5$	$5.0 \cdot 10^5$	$5.4 \cdot 10^6$	$8.9 \cdot 10^7$	$3.1 \cdot 10^9$	$2.0 \cdot 10^{10}$
128	$2.3 \cdot 10^4$	$1.2 \cdot 10^6$	$5.8 \cdot 10^6$	$1.0 \cdot 10^8$	$3.0 \cdot 10^9$	$6.4 \cdot 10^{11}$	$1.3 \cdot 10^{12}$
256	$1.2 \cdot 10^5$	$9.4 \cdot 10^6$	$7.6 \cdot 10^7$	$2.1 \cdot 10^9$	$1.2 \cdot 10^{11}$	$1.2 \cdot 10^{13}$	$2.1 \cdot 10^{13}$
512	$7.0 \cdot 10^5$	$8.3 \cdot 10^7$	$1.1 \cdot 10^9$	$4.9 \cdot 10^{10}$	$4.5 \cdot 10^{12}$	$3.6 \cdot 10^{13}$	$4.9 \cdot 10^{12}$
1024	$4.1 \cdot 10^6$	$8.0 \cdot 10^8$	$1.9 \cdot 10^{10}$	$1.3 \cdot 10^{12}$	$9.6 \cdot 10^{12}$	$1.4 \cdot 10^{12}$	$5.6 \cdot 10^{12}$

Table 3.2. $\kappa_2(\mathbf{U}_t)$ in the eigendecomposition for different degrees p_t and number of elements n_{el}.

3.3.5 Stable factorization for $(\mathbf{W}_t, \mathbf{M}_t)$

The analysis above motivates the search of a different but stable factorization of the pencil $(\mathbf{W}_t, \mathbf{M}_t)$. We look now for a factorization of the form

$$\mathbf{W}_t\mathbf{U}_t = \mathbf{M}_t\mathbf{U}_t\mathbf{\Delta}_t, \tag{3.3.26}$$

where $\mathbf{\Delta}_t$ is a complex arrowhead matrix, i.e. with non-zero entries allowed on the diagonal, on the last row and on the last column only. We also require that $\mathbf{U}_t$ fulfils the orthogonality condition

$$\mathbf{U}_t^* \mathbf{M}_t \mathbf{U}_t = \mathbb{I}_{N_t}. \tag{3.3.27}$$

From (3.3.26)–(3.3.27) we then obtain the factorizations

$$\mathbf{U}_t^* \mathbf{W}_t \mathbf{U}_t = \mathbf{\Delta}_t \quad \text{and} \quad \mathbf{U}_t^* \mathbf{M}_t \mathbf{U}_t = \mathbb{I}_{N_t}. \tag{3.3.28}$$

With this aim, we look for $\mathbf{U}_t$ as follows:

$$\mathbf{U}_t := \begin{bmatrix} \mathring{\mathbf{U}}_t & \mathbf{r} \\ \mathbf{0}^T & \rho \end{bmatrix} \tag{3.3.29}$$

where $\mathring{\mathbf{U}}_t \in \mathbb{C}^{(N_t-1)\times(N_t-1)}$, $\mathbf{r} \in \mathbb{C}^{N_t-1}$, $\rho \in \mathbb{C}$ and where $\mathbf{0} \in \mathbb{R}^{N_t-1}$ denotes the null vector. In order to guarantee the non-singularity of $\mathbf{U}_t$, we further impose $\rho \neq 0$. Accordingly, we split the time matrices $\mathbf{W}_t$ and $\mathbf{M}_t$ as

$$\mathbf{W}_t = \begin{bmatrix} \mathring{\mathbf{W}}_t & \mathbf{w} \\ -\mathbf{w}^T & \omega \end{bmatrix} \quad \text{and} \quad \mathbf{M}_t = \begin{bmatrix} \mathring{\mathbf{M}}_t & \mathbf{m} \\ \mathbf{m}^T & \mu \end{bmatrix}, \tag{3.3.30}$$

where we have defined

$$\omega := [\mathbf{W}_t]_{N_t,N_t}, \qquad \mu := [\mathbf{M}_t]_{N_t,N_t},$$

$$[\mathbf{w}]_i = [\mathbf{W}_t]_{i,N_t} \quad \text{and} \quad [\mathbf{m}]_i = [\mathbf{M}_t]_{i,N_t} \quad \text{for} \quad i = 1,\dots,N_t-1,$$

$$[\mathring{\mathbf{W}}_t]_{i,j} = [\mathbf{W}_t]_{i,j} \quad \text{and} \quad [\mathring{\mathbf{M}}_t]_{i,j} = [\mathbf{M}_t]_{i,j} \quad \text{for} \quad i,j = 1,\dots,N_t-1.$$

Recalling (3.3.24), we observe that $\mathring{\mathbf{W}}_t$ is skew-symmetric and, since $\mathring{\mathbf{M}}_t$ is symmetric, we can write the eigendecomposition of the pencils $(\mathring{\mathbf{W}}_t, \mathring{\mathbf{M}}_t)$:

$$\mathring{\mathbf{W}}_t \mathring{\mathbf{U}}_t = \mathring{\mathbf{M}}_t \mathring{\mathbf{U}}_t \mathring{\mathbf{\Lambda}}_t \quad \text{with} \quad \mathring{\mathbf{U}}_t^* \mathring{\mathbf{M}}_t \mathring{\mathbf{U}}_t = \mathbb{I}_{N_t-1}, \tag{3.3.31}$$

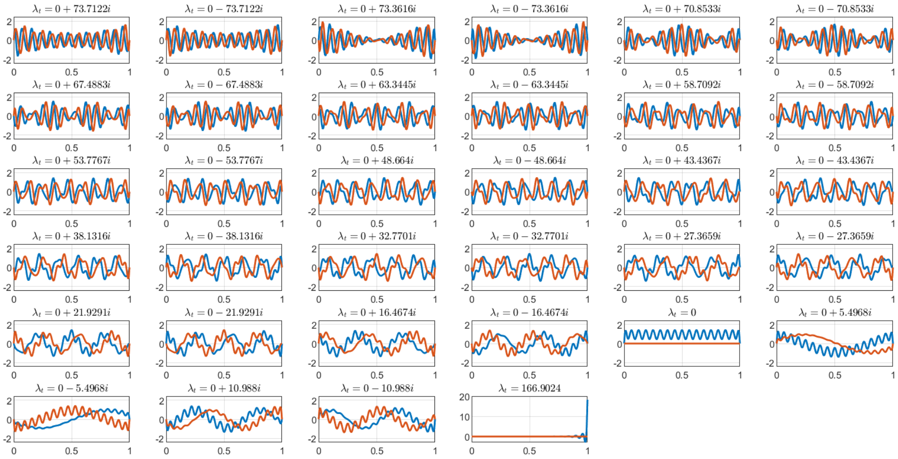
where $\mathring{\mathbf{U}}_t$ contains the complex generalized eigenvectors and $\mathring{\mathbf{\Lambda}}_t$ is the diagonal matrix of the generalized eigenvalues, that are pairs of complex conjugate pure imaginary numbers plus, eventually, the eigenvalue zero. From (3.3.29)–(3.3.30), it follows

$$\mathbf{U}_t^* \mathbf{M}_t \mathbf{U}_t = \begin{bmatrix} \mathbb{I}_{N_t-1} & \mathring{\mathbf{U}}_t^* \mathring{\mathbf{M}}_t \mathbf{r} + \mathring{\mathbf{U}}_t^* \mathbf{m} \rho \\ \mathbf{r}^* \mathring{\mathbf{M}}_t \mathring{\mathbf{U}}_t + \rho^* \mathbf{m}^T \mathring{\mathbf{U}}_t & [\mathbf{r}^* \rho^*]\, \mathbf{M}_t \begin{bmatrix} \mathbf{r} \\ \rho \end{bmatrix} \end{bmatrix},$$

where for the top-left block we have used (3.3.31).

The orthogonality condition in (3.3.27) holds if and only if $\mathbf{r}$ and ρ fulfil the two conditions:

$$\begin{cases} \mathring{\mathbf{U}}_t^* \mathring{\mathbf{M}}_t \mathbf{r} + \mathring{\mathbf{U}}_t^* \mathbf{m} \rho = \mathbf{0}, & \text{(3.3.32a)} \\ [\mathbf{r}^* \rho^*]\, \mathbf{M}_t \begin{bmatrix} \mathbf{r} \\ \rho \end{bmatrix} = 1. & \text{(3.3.32b)} \end{cases}$$



$\lambda_t = 0 + 73.7122i$ $\lambda_t = 0 - 73.7122i$ $\lambda_t = 0 + 73.3616i$ $\lambda_t = 0 - 73.3616i$ $\lambda_t = 0 + 70.8533i$ $\lambda_t = 0 - 70.8533i$

$\lambda_t = 0 + 67.4883i$ $\lambda_t = 0 - 67.4883i$ $\lambda_t = 0 + 63.3445i$ $\lambda_t = 0 - 63.3445i$ $\lambda_t = 0 + 58.7092i$ $\lambda_t = 0 - 58.7092i$

$\lambda_t = 0 + 53.7767i$ $\lambda_t = 0 - 53.7767i$ $\lambda_t = 0 + 48.664i$ $\lambda_t = 0 - 48.664i$ $\lambda_t = 0 + 43.4367i$ $\lambda_t = 0 - 43.4367i$

$\lambda_t = 0 + 38.1316i$ $\lambda_t = 0 - 38.1316i$ $\lambda_t = 0 + 32.7701i$ $\lambda_t = 0 - 32.7701i$ $\lambda_t = 0 + 27.3659i$ $\lambda_t = 0 - 27.3659i$

$\lambda_t = 0 + 21.9291i$ $\lambda_t = 0 - 21.9291i$ $\lambda_t = 0 + 16.4674i$ $\lambda_t = 0 - 16.4674i$ $\lambda_t = 0$ $\lambda_t = 0 + 5.4968i$

$\lambda_t = 0 - 5.4968i$ $\lambda_t = 0 + 10.988i$ $\lambda_t = 0 - 10.988i$ $\lambda_t = 166.9024$

Figure 3.3. Real part (blue) and imaginary part (red) of columns of $\mathbf{U}_t$ with associated diagonal entry in $\mathbf{\Delta}_t$. Discretization with $p_t = 3$ and $n_{el} = 32$.

In order to compute $\mathbf{r}$ and ρ, we first find $\mathbf{v} \in \mathbb{C}^{N_t-1}$ such that

$$\overset{\circ}{\mathbf{M}}_t \mathbf{v} = -\mathbf{m}; \tag{3.3.33}$$

then we normalize the vector $\begin{bmatrix} \mathbf{v} \\ 1 \end{bmatrix}$ w.r.t. the $\|\cdot\|_{\mathbf{M}_t}$-norm to get

$$\begin{bmatrix} \mathbf{r} \\ \rho \end{bmatrix} := \frac{\begin{bmatrix} \mathbf{v} \\ 1 \end{bmatrix}}{\left([\mathbf{v}^* \ 1] \mathbf{M}_t \begin{bmatrix} \mathbf{v} \\ 1 \end{bmatrix} \right)^{\frac{1}{2}}}$$

that fulfils (3.3.32a)–(3.3.32b). Finally, we get (3.3.26) by defining

$$\mathbf{\Delta}_t := \mathbf{U}_t^* \mathbf{W}_t \mathbf{U}_t = \begin{bmatrix} \overset{\circ}{\mathbf{\Lambda}}_t & \mathbf{g} \\ -\mathbf{g}^* & \sigma \end{bmatrix}, \tag{3.3.34}$$

where $\mathbf{g} := \overset{\circ}{\mathbf{U}}{}_t^* \begin{bmatrix} \overset{\circ}{\mathbf{W}}_t & \mathbf{w} \end{bmatrix} \begin{bmatrix} \mathbf{r} \\ \rho \end{bmatrix}$ and $\sigma := [\mathbf{r}^* \rho^*] \, \mathbf{W}_t \begin{bmatrix} \mathbf{r} \\ \rho \end{bmatrix}$. Note that matrix (3.3.34) has an arrowhead structure. Figure 3.3 shows the plot of the columns of $\mathbf{U}_t$ with associated diagonal entry of $\mathbf{\Delta}_t$, for $p_t = 3$ and uniform partition with $n_{el} = 32$.

To assess the stability of the new decomposition (3.3.28), we set $T = 1$ and we compute the condition number $\kappa_2(\mathbf{U}_t)$ for different values of spline degree p_t and for various uniform discretizations with number of elements n_{el}. Thanks to (3.3.27), we have $\kappa_2(\mathbf{U}_t) = \sqrt{\kappa_2(\mathbf{M}_t)}$. The results, reported in Table 3.3, show that the condition numbers $\kappa_2(\mathbf{U}_t)$ are uniformly bounded w.r.t. the mesh refinement, they grow with respect to the polynomial degree but they are moderately small for all the degrees of interest. We conclude that the factorization (3.3.28) for the time pencil $(\mathbf{W}_t, \mathbf{M}_t)$ is stable.

n_{el}	$p_t=2$	$p_t=3$	$p_t=4$	$p_t=5$	$p_t=6$	$p_t=7$	$p_t=8$
32	$3.2\cdot 10^0$	$5.2\cdot 10^0$	$8.3\cdot 10^0$	$1.3\cdot 10^1$	$2.2\cdot 10^1$	$3.6\cdot 10^1$	$5.9\cdot 10^1$
64	$3.3\cdot 10^0$	$5.2\cdot 10^0$	$8.3\cdot 10^0$	$1.3\cdot 10^1$	$2.2\cdot 10^1$	$3.6\cdot 10^1$	$5.9\cdot 10^1$
128	$3.3\cdot 10^0$	$5.2\cdot 10^0$	$8.3\cdot 10^0$	$1.3\cdot 10^1$	$2.2\cdot 10^1$	$3.6\cdot 10^1$	$5.9\cdot 10^1$
256	$3.3\cdot 10^0$	$5.2\cdot 10^0$	$8.3\cdot 10^0$	$1.3\cdot 10^1$	$2.2\cdot 10^1$	$3.6\cdot 10^1$	$5.9\cdot 10^1$
512	$3.3\cdot 10^0$	$5.2\cdot 10^0$	$8.3\cdot 10^0$	$1.3\cdot 10^1$	$2.2\cdot 10^1$	$3.6\cdot 10^1$	$5.9\cdot 10^1$
1024	$3.3\cdot 10^0$	$5.2\cdot 10^0$	$8.3\cdot 10^0$	$1.3\cdot 10^1$	$2.2\cdot 10^1$	$3.6\cdot 10^1$	$5.9\cdot 10^1$

Table 3.3. $\kappa_2(\mathbf{U}_t)$ in arrow decomposition for different degrees p_t and number of elements n_{el}.

3.3.6 Preconditioner application

The application of the preconditioner involves the solution of the linear system

$$\widehat{\mathbf{A}}\mathbf{s} = \mathbf{r}, \tag{3.3.35}$$

where $\widehat{\mathbf{A}}$ has the structure (3.3.20). We are able to efficiently solve system (3.3.35) by extending the FD method. The starting points, that are involved in the setup of the preconditioner, are the following ones:

- for the pencils $(\widehat{\mathbf{L}}_l, \widehat{\mathbf{M}}_l)$ for $l = 1, \dots, d$ we have the factorizations (3.3.23);
- for the pencil $(\mathbf{W}_t, \mathbf{M}_t)$ we have the factorization (3.3.28).

Then, by defining $\mathbf{U}_s := \mathbf{U}_d \otimes \cdots \otimes \mathbf{U}_1$ and $\mathbf{\Lambda}_s := \sum_{l=1}^{d} \mathbb{I}_{N_{s,d}} \otimes \cdots \otimes \mathbb{I}_{N_{s,l+1}} \otimes \mathbf{\Lambda}_l \otimes \mathbb{I}_{N_{s,l-1}} \otimes \cdots \otimes \mathbb{I}_{N_{s,1}}$, we have for the matrix $\widehat{\mathbf{A}}$ the factorization

$$\widehat{\mathbf{A}} = \left(\mathbf{U}_t^* \otimes \mathbf{U}_s^T\right)^{-1} \left(\gamma \mathbf{\Delta}_t \otimes \mathbb{I}_{N_s} + \nu \mathbb{I}_{N_t} \otimes \mathbf{\Lambda}_s\right) \left(\mathbf{U}_t \otimes \mathbf{U}_s\right)^{-1}. \tag{3.3.36}$$

Note that the second factor in (3.3.36) has the block-arrowhead structure

$$\gamma \mathbf{\Delta}_t \otimes \mathbb{I}_{N_s} + \nu \mathbb{I}_{N_t} \otimes \mathbf{\Lambda}_s = \begin{bmatrix} \mathbf{H}_1 & & & \mathbf{B}_1 \\ & \ddots & & \vdots \\ & & \mathbf{H}_{N_t-1} & \mathbf{B}_{N_t-1} \\ -\mathbf{B}_1^* & \dots & -\mathbf{B}_{N_t-1}^* & \mathbf{H}_{N_t} \end{bmatrix} \tag{3.3.37}$$

where $\mathbf{H}_i$ and $\mathbf{B}_i$are diagonal matrices defined as

$$\mathbf{H}_i := \gamma [\mathbf{\Lambda}_t]_{i,i} \mathbb{I}_{N_s} + \nu \mathbf{\Lambda}_s \quad \text{and} \quad \mathbf{B}_i := \gamma [\mathbf{g}]_i \mathbb{I}_{N_s} \quad \text{for} \quad i = 1, \dots, N_t - 1,$$

$$\mathbf{H}_{N_t} := \gamma \sigma \mathbb{I}_{N_s} + \nu \mathbf{\Lambda}_s.$$

The matrix (3.3.37) has the following easy-to-invert block LU decomposition

$$\begin{aligned} &\gamma \mathbf{\Delta}_t \otimes \mathbb{I}_{N_s} + \nu \mathbb{I}_{N_t} \otimes \mathbf{\Lambda}_s \\ &= \begin{bmatrix} \mathbb{I}_{N_s} & & & \\ & \ddots & & \\ & & \mathbb{I}_{N_s} & \\ -\mathbf{B}_1^* \mathbf{H}_1^{-1} & \dots & -\mathbf{B}_{N_t-1}^* \mathbf{H}_{N_t-1}^{-1} & \mathbb{I}_{N_s} \end{bmatrix} \begin{bmatrix} \mathbf{H}_1 & & & \mathbf{B}_1 \\ & \ddots & & \vdots \\ & & \mathbf{H}_{N_t-1} & \mathbf{B}_{N_t-1} \\ & & & \mathbf{S} \end{bmatrix} \end{aligned} \tag{3.3.38}$$

where $\mathbf{S} := \mathbf{H}_{N_t} + \sum_{i=1}^{N_t-1} \mathbf{B}_i^* \mathbf{H}_i^{-1} \mathbf{B}_i$ is a diagonal matrix.

Summarizing, the solution of (3.3.35) can be computed by the following algorithm.

Algorithm 1 Extended FD

```
Compute the factorizations (3.3.23) and (3.3.28).
Compute $\widetilde{\mathbf{q}} = (\mathbf{U}_t^* \otimes \mathbf{U}_s^T)\mathbf{r}$.
Compute $\widetilde{\mathbf{s}} = (\gamma \boldsymbol{\Delta}_t \otimes \mathbb{I}_{N_s} + \nu \mathbb{I}_{N_t} \otimes \boldsymbol{\Lambda}_s)^{-1} \widetilde{\mathbf{q}}$.
Compute $\mathbf{s} = (\mathbf{U}_t \otimes \mathbf{U}_s)\, \widetilde{\mathbf{s}}$.
```

3.3.7 Preconditioner robustness: partial inclusion of the geometry

The preconditioner (3.3.20) does not incorporate any information on the spatial parametrization $\boldsymbol{F}$. Thus, the quality of the preconditioning strategy may depend on the geometry map: we see this trend in the numerical tests presented in the upper tables of [75, Tables 4,6]. However, we can generalize (3.3.20) by including in the univariate spatial matrices $\widehat{\mathbf{L}}_l, \widehat{\mathbf{M}}_l$ for $l = 1, \dots, d$ a suitable approximation of $\boldsymbol{F}$, without increasing the asymptotic computational cost. A similar approach has been used first in [87] for the Stokes problem and in [85] for a least squares formulation of the heat equation. We briefly give an overview of this strategy.

Referring to Section 3.1.1 for the notation of the basis functions, we rewrite the entries of the system matrix (3.3.17) in the parametric domain as

$$\begin{aligned}
[\mathbf{A}]_{i,j} &= \mathcal{A}(B_{j,\boldsymbol{p}}, B_{i,\boldsymbol{p}}) \\
&= \gamma \int_0^1 \int_{\widehat{\Omega}} \tfrac{1}{T} \partial_\tau \widehat{B}_{j,\boldsymbol{p}} \widehat{B}_{i,\boldsymbol{p}} |\mathrm{det}(J_{\boldsymbol{G}})| \ \mathrm{d}\widehat{\Omega}\ \mathrm{d}\tau \\
&+ \int_0^1 \int_{\widehat{\Omega}} \nu (\nabla \widehat{B}_{j,\boldsymbol{p}})^T J_{\boldsymbol{G}}^{-1} J_{\boldsymbol{G}}^{-T} \nabla \widehat{B}_{i,\boldsymbol{p}} |\mathrm{det}(J_{\boldsymbol{G}})| \ \mathrm{d}\widehat{\Omega}\ \mathrm{d}\tau \\
&= \int_0^1 \int_{\widehat{\Omega}} \begin{bmatrix} (\nabla \widehat{B}_{j,\boldsymbol{p}})^T & \partial_\tau \widehat{B}_{j,\boldsymbol{p}} \end{bmatrix} \begin{bmatrix} \nu T\ \mathbb{I}_d & \\ & \gamma \end{bmatrix} \mathfrak{C} \begin{bmatrix} (\nabla \widehat{B}_{i,\boldsymbol{p}})^T & \widehat{B}_{i,\boldsymbol{p}} \end{bmatrix}^T \mathrm{d}\widehat{\Omega}\ \mathrm{d}\tau,
\end{aligned}$$

where

$$\mathfrak{C} := \begin{bmatrix} J_{\boldsymbol{F}}^{-1} J_{\boldsymbol{F}}^{-T} |\,\mathrm{det}(J_{\boldsymbol{F}})| & \\ & |\mathrm{det}(J_{\boldsymbol{F}})| \end{bmatrix}$$

and where we used that $B_{i,\boldsymbol{p}} = \widehat{B}_{i,\boldsymbol{p}} \circ \boldsymbol{G}^{-1}$, $B_{j,\boldsymbol{p}} = \widehat{B}_{j,\boldsymbol{p}} \circ \boldsymbol{G}^{-1}$ and $|\mathrm{det}(J_{\boldsymbol{G}})| = T|\mathrm{det}(J_{\boldsymbol{F}})|$. The construction of the preconditioners is based on the following approximation of the diagonal entries only of $\mathfrak{C}$:

$$[\mathfrak{C}(\boldsymbol{\eta})]_{l,l} \approx [\widetilde{\mathfrak{C}}(\boldsymbol{\eta})]_{l,l} := \varphi_1(\eta_1) \dots \varphi_{l-1}(\eta_{l-1}) \Phi_l(\eta_l) \varphi_{l+1}(\eta_{l+1}) \dots \varphi_d(\eta_d) \text{ for } l = 1, \dots, d, \tag{3.3.39a}$$

$$[\mathfrak{C}(\boldsymbol{\eta})]_{d+1,d+1} \approx [\widetilde{\mathfrak{C}}(\boldsymbol{\eta})]_{d+1,d+1} := \varphi_1(\eta_1) \dots \varphi_d(\eta_d). \tag{3.3.39b}$$

In order to compute such an approximation, we interpolate the functions $[\widetilde{\mathfrak{C}}(\boldsymbol{\eta})]_{l,l}$ in (3.3.39) by piecewise constants in each element and we build the univariate factors φ_l and Φ_l by using the separation of variables algorithm detailed in [85, Appendix C].

The computational cost of the approximation above is proportional to the number of elements in Ω, that, when using smooth B-splines, is almost equal to N_s, independent of p_s and p_t and thus negligible in the whole iterative strategy.

Then we define

$$[\widetilde{\mathbf{A}}]_{i,j} := \int_0^1 \int_{\widehat{\Omega}} \left[(\nabla \widehat{B}_{j,\boldsymbol{p}})^T \quad \partial_\tau \widehat{B}_{j,\boldsymbol{p}}\right] \begin{bmatrix} \nu T \, \mathbb{I}_d & \\ & \gamma \end{bmatrix} \widetilde{\mathfrak{C}} \left[(\nabla \widehat{B}_{i,\boldsymbol{p}})^T \quad \widehat{B}_{i,\boldsymbol{p}}\right]^T \, \mathrm{d}\widehat{\Omega} \, \mathrm{d}\tau.$$

The previous matrix maintains the same Kronecker structure as (3.3.20). Indeed we have that

$$\widetilde{\mathbf{A}} = \gamma \mathbf{W}_t \otimes \widetilde{\mathbf{M}}_s + \nu \mathbf{M}_t \otimes \widetilde{\mathbf{K}}_s, \tag{3.3.40}$$

where

$$\widetilde{\mathbf{K}}_s := \sum_{l=1}^{d} \widetilde{\mathbf{M}}_d \otimes \cdots \otimes \widetilde{\mathbf{M}}_{l+1} \otimes \widetilde{\mathbf{K}}_l \otimes \widetilde{\mathbf{M}}_{l-1} \otimes \cdots \otimes \widetilde{\mathbf{M}}_1, \qquad \widetilde{\mathbf{M}}_s := \widetilde{\mathbf{M}}_d \otimes \cdots \otimes \widetilde{\mathbf{M}}_1,$$

and where for $l = 1, \ldots, d$ and for $i, j = 1, \ldots, N_{s,l}$ we define

$$[\widetilde{\mathbf{K}}_l]_{i,j} := \int_0^1 \Phi_l(\eta_l) \widehat{b}'_{j,p_s}(\eta_l) \widehat{b}'_{i,p_s}(\eta_l) \, \mathrm{d}\eta_l \quad \text{and} \quad [\widetilde{\mathbf{M}}_l]_{i,j} := \int_0^1 \varphi_l(\eta_l) \widehat{b}_{j,p_s}(\eta_l) \widehat{b}_{i,p_s}(\eta_l) \, \mathrm{d}\eta_l.$$

We remark that the application of (3.3.40) can still be performed by Algorithm 1. Finally, we apply a diagonal scaling on $\widetilde{\mathbf{A}}$ and we define the Galerkin preconditioner as

$$\widehat{\mathbf{A}}^{\boldsymbol{G}} := \mathbf{D}^{\frac{1}{2}} \widetilde{\mathbf{A}} \mathbf{D}^{\frac{1}{2}} \tag{3.3.41}$$

where $[\mathbf{D}]_{i,i} := \dfrac{[\mathbf{A}]_{i,i}}{[\widetilde{\mathbf{A}}]_{i,i}}$ for $i = 1, \ldots, N_{dof}$.

3.3.8 The case of non-constant separable coefficients

We briefly discuss a generalization of the preconditioning strategy to the case of non-constant equation coefficients γ and ν. We assume that γ and ν are positive functions defined over $\Omega \times [0, T]$ and that they are separable in space and in time, i.e. we can write

$$\gamma(\boldsymbol{x}, t) = \gamma_s(\boldsymbol{x}) \gamma_t(t), \qquad \nu(\boldsymbol{x}, t) = \nu_s(\boldsymbol{x}) \nu_t(t),$$

with $\gamma_s, \gamma_t, \nu_s$ and ν_t positive functions.

Now, the first equation of (3.2.13) can be written as

$$\gamma_s \partial_t u - \nabla \cdot \left(\frac{\nu_t}{\gamma_t} \nu_s \nabla u \right) = \frac{f}{\gamma_t}.$$

We discretize this equation as described in Section 3.2 and we generalize the definition of the linear system (3.3.18) with

$$\mathbf{A} := \mathbf{W}_t \otimes \underline{\mathbf{M}}_s + \underline{\mathbf{M}}_t \otimes \underline{\mathbf{K}}_s,$$

where $\mathbf{W}_t$ is defined as in (3.3.19a), while for $i, j = 1, \ldots, N_t$

$$[\underline{\mathbf{M}}_t]_{i,j} := \int_0^T \frac{\nu_t(t)}{\gamma_t(t)} b_{i,p_t}(t) \, b_{j,p_t}(t) \, \mathrm{d}t$$

and for $i, j = 1, \dots, N_s$

$$[\underline{\mathbf{M}}_s]_{i,j} := \int_\Omega \gamma_s(\boldsymbol{x}) B_{i,\boldsymbol{p}_s}(\boldsymbol{x}) \; B_{j,\boldsymbol{p}_s}(\boldsymbol{x}) \;\; \mathrm{d}\Omega \quad \text{and} \quad [\underline{\mathbf{K}}_s]_{i,j} := \int_\Omega \nu_s(\boldsymbol{x}) \nabla B_{i,\boldsymbol{p}_s}(\boldsymbol{x}) \cdot \nabla B_{j,\boldsymbol{p}_s}(\boldsymbol{x}) \;\; \mathrm{d}\Omega.$$

Then, the preconditioner that we propose is defined as in (3.3.41)

$$\widehat{\mathbf{A}}^{\boldsymbol{G}} := \mathbf{D}^{\frac{1}{2}} \widetilde{\mathbf{A}} \mathbf{D}^{\frac{1}{2}},$$

but here we generalize (3.3.40) with $\widetilde{\mathbf{A}} := \mathbf{W}_t \otimes \breve{\mathbf{M}}_s + \underline{\mathbf{M}}_t \otimes \breve{\mathbf{K}}_s$, where the matrices $\breve{\mathbf{K}}_s$ and $\breve{\mathbf{M}}_s$ are obtained by using an approximation technique analogous to the one described previously in this section, with γ_s and ν_s included in the coefficient matrix $\mathfrak{C}$. The preconditioner $\widetilde{\mathbf{A}}$ can still be applied as described in Section 3.3.6. Note that, for this purpose, it is crucial that $\mathbf{W}_t$ does not incorporate any time-dependent coefficient, since this would invalidate (3.3.24).

3.4 Galerkin L^2 least squares preconditioner

The second attempt of solving (3.3.15) is by considering its Galerkin L^2 least squares formulation, that is

$$\mathbf{A}^T \mathbf{M}^{-1} \mathbf{A} \mathbf{u} = \mathbf{g}, \tag{3.4.42}$$

where we recall $[\mathbf{A}]_{i,j} = \mathcal{A}(B_{j,\boldsymbol{p}}, B_{i,\boldsymbol{p}})$, $\mathbf{g} := \mathbf{A}^T \mathbf{M}^{-1} \mathbf{f}$ with $[\mathbf{f}]_i = \mathcal{F}(B_{i,\boldsymbol{p}})$ and $\mathbf{M} = \mathbf{M}_t \otimes \mathbf{M}_s$ is the mass matrix with $\mathbf{M}_t$ and $\mathbf{M}_s$ as in (3.3.19). The computation of $\mathbf{g}$ requires to invert the mass $\mathbf{M}$, which can be efficiently performed with different methods, i.e., ad hoc sparse approximations of the inverse [105, 111], or preconditioning with the parametric mass or its extensions proposed in [52, 28, 76] or using low rank approximations as in [81, 82, 61]. Here we iteratively invert the mass with Conjugate Gradients and the preconditioner designed in [76].

The tensor-product structure of the isogeometric space (3.1.9) allows to write the system matrix $\mathbf{A}^T \mathbf{M}^{-1} \mathbf{A}$ as sum of Kronecker products of matrices as

$$\mathbf{A}^T \mathbf{M}^{-1} \mathbf{A} \; = \gamma^2 \mathbf{W}_t^T \mathbf{M}_t^{-1} \mathbf{W}_t \otimes \mathbf{M}_s + \nu^2 \mathbf{M}_t \otimes \mathbf{L}_s \mathbf{M}_s^{-1} \mathbf{L}_s + \gamma\nu \left(\mathbf{W}_t + \mathbf{W}_t^T \right) \otimes \mathbf{L}_s, \tag{3.4.43}$$

where $\mathbf{W}_t$ and $\mathbf{L}_s$ are defined in (3.3.19).

For the problem (3.4.42), we therefore introduce the preconditioner

$$\widehat{\mathbf{P}} := \gamma^2 \mathbf{W}_t^T \mathbf{M}_t^{-1} \mathbf{W}_t \otimes \widehat{\mathbf{M}}_s + \nu^2 \mathbf{M}_t \otimes \widehat{\mathbf{L}}_s \widehat{\mathbf{M}}_s^{-1} \widehat{\mathbf{L}}_s + \gamma\nu \left(\mathbf{W}_t + \mathbf{W}_t^T \right) \otimes \widehat{\mathbf{L}}_s, \tag{3.4.44}$$

where $\widehat{\mathbf{L}}_s$ and $\widehat{\mathbf{M}}_s$ are defined in (3.3.21). Again, the efficient application of the proposed preconditioner, that is, the solution of a linear system with matrix $\widehat{\mathbf{P}}$, should exploit the structure highlighted above.

Recall that the space pencils $(\widehat{\mathbf{L}}_l, \widehat{\mathbf{M}}_l)$ for $l = 1, \dots, d$ admit the stable factorizations described in Section 3.3.3, that is

$$\mathbf{U}_l^T \widehat{\mathbf{L}}_l \mathbf{U}_l = \mathbf{\Lambda}_l \quad \text{and} \quad \mathbf{U}_l^T \widehat{\mathbf{M}}_l \mathbf{U}_l = \mathbb{I}_{N_{s,l}},$$

where $\mathbb{I}_{N_{s,l}}$ denotes the identity matrix of dimension $N_{s,l} \times N_{s,l}$.

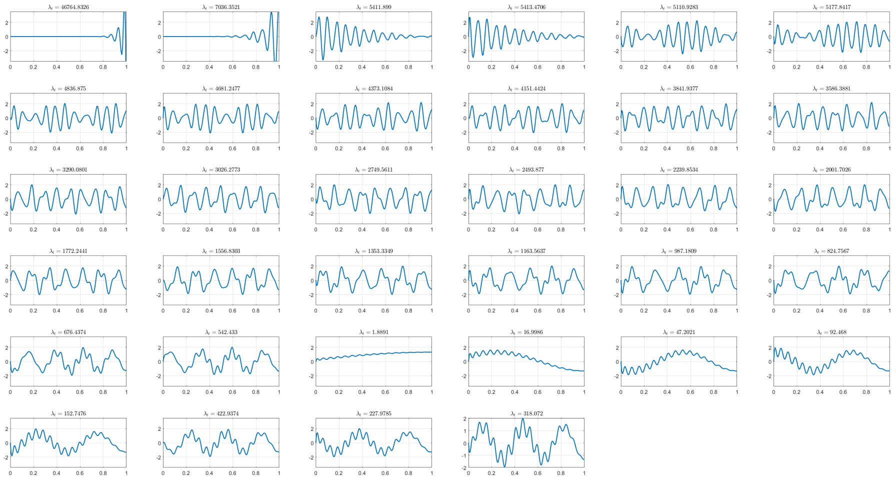


Figure 3.4. Generalized eigenvectors for the pencil $(\mathbf{W}_t^T\mathbf{M}_t^{-1}\mathbf{W}_t,\ \mathbf{M}_t)$, with associated eigenvalues for $p_t = 3$ and $n_{el} = 32$ elements.

3.4.1 Stable factorization in time

The time matrices $\mathbf{W}_t^T\mathbf{M}_t^{-1}\mathbf{W}_t$ and $\mathbf{M}_t$ are symmetric positive definite, therefore they admit a generalized eigendecomposition of the kind

$$\mathbf{W}_t^T\mathbf{M}_t^{-1}\mathbf{W}_t\mathbf{U}_t = \mathbf{\Lambda}_t\mathbf{M}_t\mathbf{U}_t, \tag{3.4.45}$$

where the matrix $\mathbf{U}_t$ contains in each column the $\mathbf{M}_t$-orthonormal generalized eigenvectors and $\mathbf{\Lambda}_t$ is the diagonal matrix whose entries contain the generalized eigenvalues. We have the following factorization

$$\mathbf{U}_t^T\mathbf{W}_t^T\mathbf{M}_t^{-1}\mathbf{W}_t\mathbf{U}_t = \mathbf{\Lambda}_t \quad \text{and} \quad \mathbf{U}_t^T\mathbf{M}_t\mathbf{U}_t = \mathbb{I}_{N_t}. \tag{3.4.46}$$

Figure 3.4 shows the shape of the generalized eigenvectors in $\mathbf{U}_t$, with associated eigenvalue in $\mathbf{\Lambda}_t$, for a fixed univariate direction $l = 1, \ldots, d$ discretized with degree $p_s = 3$ B-Splines and uniform partition. The stability of the decomposition (3.4.45) is again expressed by the condition number of the eigenvector matrix, and since $\mathbf{U}_t^T\mathbf{M}_t\mathbf{U}_t = \mathbb{I}_{N_t}$, it holds $\kappa_2(\mathbf{U}_t) = \sqrt{\kappa_2(\mathbf{M}_t)}$.

We investigate the stability of the diagonalization (3.4.45) by setting $T = 1$ and computing the condition number $\kappa_2(\mathbf{U}_t)$ for different values of spline degree p_t and for various uniform discretizations with number of elements n_{el}. The results, reported in Table 3.4, show that the condition numbers $\kappa_2(\mathbf{U}_t)$ are uniformly bounded w.r.t. the mesh refinement, they grow with respect to the polynomial degree but they are moderately small for all the degrees of interest. We conclude that the generalized diagonalization (3.4.45) for the time pencil $(\mathbf{W}_t^T\mathbf{M}_t^{-1}\mathbf{W}_t, \mathbf{M}_t)$ is stable.

3.4.2 Preconditioner application

The application of the preconditioner involves the solution of the linear system

$$\widehat{\mathbf{P}}\mathbf{s} = \mathbf{r}, \tag{3.4.47}$$

n_{el}	$p_t = 2$	$p_t = 3$	$p_t = 4$	$p_t = 5$	$p_t = 6$	$p_t = 7$	$p_t = 8$
32	$3.2 \cdot 10^0$	$5.2 \cdot 10^0$	$8.3 \cdot 10^0$	$1.3 \cdot 10^1$	$2.2 \cdot 10^1$	$3.6 \cdot 10^1$	$5.9 \cdot 10^1$
64	$3.3 \cdot 10^0$	$5.2 \cdot 10^0$	$8.3 \cdot 10^0$	$1.3 \cdot 10^1$	$2.2 \cdot 10^1$	$3.6 \cdot 10^1$	$5.9 \cdot 10^1$
128	$3.3 \cdot 10^0$	$5.2 \cdot 10^0$	$8.3 \cdot 10^0$	$1.3 \cdot 10^1$	$2.2 \cdot 10^1$	$3.6 \cdot 10^1$	$5.9 \cdot 10^1$
256	$3.3 \cdot 10^0$	$5.2 \cdot 10^0$	$8.3 \cdot 10^0$	$1.3 \cdot 10^1$	$2.2 \cdot 10^1$	$3.6 \cdot 10^1$	$5.9 \cdot 10^1$
512	$3.3 \cdot 10^0$	$5.2 \cdot 10^0$	$8.3 \cdot 10^0$	$1.3 \cdot 10^1$	$2.2 \cdot 10^1$	$3.6 \cdot 10^1$	$5.9 \cdot 10^1$
1024	$3.3 \cdot 10^0$	$5.2 \cdot 10^0$	$8.3 \cdot 10^0$	$1.3 \cdot 10^1$	$2.2 \cdot 10^1$	$3.6 \cdot 10^1$	$5.9 \cdot 10^1$

Table 3.4. $\kappa_2(\mathbf{U}_t)$ in eigendecomposition of the pencil $(\mathbf{W}_t^T \mathbf{M}_t^{-1} \mathbf{W}_t,\ \mathbf{M}_t)$ for different degrees p_t and number of elements n_{el}.

where $\widehat{\mathbf{P}}$ has the structure (3.4.44). We are able to efficiently solve system (3.4.47) by the FD method and the Sherman-Morrison formula. The starting points, that are involved in the setup of the preconditioner, are the following ones:

- for the pencils $(\widehat{\mathbf{L}}_l, \widehat{\mathbf{M}}_l)$ with $l = 1, \dots, d$ we have the factorizations (3.3.23);
- for the pencil $(\mathbf{W}_t^T \mathbf{M}_t^{-1} \mathbf{W}_t, \mathbf{M}_t)$ we have the factorization (3.4.45).

Then, define $\mathbf{U}_s := \mathbf{U}_d \otimes \cdots \otimes \mathbf{U}_1$ and $\mathbf{\Lambda}_s := \sum_{l=1}^d \mathbb{I}_{N_{s,d}} \otimes \cdots \otimes \mathbb{I}_{N_{s,l+1}} \otimes \mathbf{\Lambda}_l \otimes \mathbb{I}_{N_{s,l-1}} \otimes \cdots \otimes \mathbb{I}_{N_{s,1}}$. Notice that $\widehat{\mathbf{M}}_s^{-1} = \mathbf{U}_s \mathbf{U}_s^T$, therefore the matrix $\widehat{\mathbf{L}}_s \widehat{\mathbf{M}}_s^{-1} \widehat{\mathbf{L}}_s$ admits the stable factorization

$$\mathbf{U}_s^T \widehat{\mathbf{L}}_s \widehat{\mathbf{M}}_s^{-1} \widehat{\mathbf{L}}_s \mathbf{U}_s = \mathbf{\Lambda}_s^2. \tag{3.4.48}$$

The preconditioner $\widehat{\mathbf{P}}$ admits the following factorization

$$\widehat{\mathbf{P}} = \left(\mathbb{I}_{N_t} \otimes \mathbf{U}_s^T\right)^{-1} \\ \left(\left(\mathbf{U}_t^T \otimes \mathbb{I}_{N_s}\right)^{-1} \left(\gamma^2 \mathbf{\Lambda}_t \otimes \mathbb{I}_{N_s} + \nu^2 \mathbb{I}_{N_t} \otimes \mathbf{\Lambda}_s^2\right) \left(\mathbf{U}_t \otimes \mathbb{I}_{N_s}\right)^{-1} + \gamma\nu \left(\mathbf{W}_t + \mathbf{W}_t^T\right) \otimes \mathbf{\Lambda}_s\right) \\ \left(\mathbb{I}_{N_t} \otimes \mathbf{U}_s\right)^{-1}. \tag{3.4.49}$$

Note that the second factor in (3.4.49) is sum of two matrices, the second one being

$$\gamma\nu \left(\mathbf{W}_t + \mathbf{W}_t^T\right) \otimes \mathbf{\Lambda}_s \tag{3.4.50}$$

where $\left(\mathbf{W}_t + \mathbf{W}_t^T\right)$ is a rank 1 matrix that can be written as $\left(\mathbf{W}_t + \mathbf{W}_t^T\right) = \boldsymbol{e}_{N_t} \boldsymbol{e}_{N_t}^T$, with $\boldsymbol{e}_{N_t}$ being the last element of the canonical basis of $\mathbb{R}^{N_t}$. Here it is possible to introduce a vector $\boldsymbol{v}_{N_t} \in \mathbb{R}^{N_t}$ such that

$$\boldsymbol{e}_{N_t} \boldsymbol{e}_{N_t}^T = \mathbf{U}_t^{-T} \boldsymbol{v}_{N_t} \boldsymbol{v}_{N_t}^T \mathbf{U}_t^{-1},$$

by defining $\boldsymbol{v}_{N_t} := \mathbf{U}_t^T \boldsymbol{e}_{N_t}$ that is the last column of $\mathbf{U}_t^T$. Thus equation (3.4.49) can be rewritten as

$$\widehat{\mathbf{P}} = \left(\mathbf{U}_t^T \otimes \mathbf{U}_s^T\right)^{-1} (\mathbf{H} + \mathbf{K}) \left(\mathbf{U}_t \otimes \mathbf{U}_s\right)^{-1} \tag{3.4.51}$$

where we have introduced

$$\mathbf{H} := \gamma^2 \mathbf{\Lambda}_t \otimes \mathbb{I}_{N_s} + \nu^2 \mathbb{I}_{N_t} \otimes \mathbf{\Lambda}_s^2,$$

for the full diagonal term, and

$$\mathbf{K} := \gamma\nu \boldsymbol{v}_{N_t}\boldsymbol{v}_{N_t}^T \otimes \boldsymbol{\Lambda}_s.$$

Let us introduce $\mathbf{S}_{N_t,N_s}$, the space and time shuffle matrix, such that $\mathbf{A}\otimes\mathbf{B} = \mathbf{S}_{N_t,N_s}(\mathbf{B}\otimes\mathbf{A})\mathbf{S}_{N_t,N_s}^T$ for all $\mathbf{A}\in\mathbb{R}^{N_t\times N_t}$ and $\mathbf{B}\in\mathbb{R}^{N_s\times N_s}$. Under reshuffling with $\mathbf{S}_{N_t,N_s}$, we have the following block diagonal structure

$$\mathbf{H} = \mathbf{S}_{N_t,N_s}\begin{bmatrix}\mathbf{H}_1 & & \\ & \ddots & \\ & & \mathbf{H}_{N_s}\end{bmatrix}\mathbf{S}_{N_t,N_s}^T, \quad \text{and} \quad \mathbf{K} = \mathbf{S}_{N_t,N_s}\begin{bmatrix}\mathbf{K}_1 & & \\ & \ddots & \\ & & \mathbf{K}_{N_s}\end{bmatrix}\mathbf{S}_{N_t,N_s}^T,$$

where, for $i = 1,\dots,N_s$, the matrices $\mathbf{H}_i$ are diagonal defined as $\mathbf{H}_i := \gamma^2\boldsymbol{\Lambda}_t + \nu^2[\boldsymbol{\Lambda}_s^2]_{i,i}\otimes\mathbb{I}_{N_t}$, while $\mathbf{K}_i = \gamma\nu[\boldsymbol{\Lambda}_s]_{i,i}\otimes\boldsymbol{v}_{N_t}\boldsymbol{v}_{N_t}^T$.

In order to invert $\mathbf{H}+\mathbf{K}$, it is now sufficient to reshuffle the data, and invert the following independent N_s problems of size $N_t\times N_t$:

$$(\mathbf{H}_i+\mathbf{K}_i)\,\mathbf{x}_i = \mathbf{y}_i \quad \text{for } i = 1,\dots,N_s.$$

Notice that, each $\mathbf{K}_i$ is a rank 1 perturbation in the above systems, therefore the Sherman-Morrison formula gives

$$\mathbf{x}_i = \mathbf{H}_i^{-1}\mathbf{y}_i - \gamma\nu[\Lambda_s]_{i,i}\frac{\mathbf{H}_i^{-1}\boldsymbol{v}_{N_t}\boldsymbol{v}_{N_t}^T\mathbf{H}_i^{-1}\mathbf{y}_i}{1+\gamma\nu[\boldsymbol{\Lambda}_s]_{i,i}\boldsymbol{v}_{N_t}^T\mathbf{H}_i^{-1}\boldsymbol{v}_{N_t}} \quad \text{for } i = 1,\dots,N_s. \tag{3.4.52}$$

Summarizing, the solution of (3.4.47) can be computed by the following algorithm.

Algorithm 2 Sherman-Morrison FD

```
1: Compute the factorizations (3.3.23) and (3.4.45).
2: Compute $\widetilde{\mathbf{r}} = (\mathbf{U}_t^T\otimes\mathbf{U}_s^T)\mathbf{r}$.
3: Reshuffle $\mathbf{q} = \mathbf{S}_{N_t,N_s}^T\widetilde{\mathbf{r}}$.
4: Compute $[\widetilde{\mathbf{q}}]_i = (\mathbf{H}_i+\mathbf{K}_i)^{-1}\mathbf{q}_i$    for $i = 1,\dots,N_s$.
5: Reshuffle $\widetilde{\mathbf{s}} = \mathbf{S}_{N_t,N_s}\widetilde{\mathbf{q}}$
6: Compute $\mathbf{s} = (\mathbf{U}_t\otimes\mathbf{U}_s)\,\widetilde{\mathbf{s}}$.
```

3.5 Space-time least squares formulation of the Heat equation

In this section we present the Least Squares space-time formulation introduced in [85], which is an alternative well posed space-time variational formulation, w.r.t. the one presented in Section 3.2. Let us recall the model problem (3.2.13) here. We seek for a solution u such that

$$\begin{cases} \gamma\partial_t u - \nabla\cdot(\nu\nabla u) &= f \quad \text{in} \quad \Omega\times(0,T), \\ u &= 0 \quad \text{on} \quad \partial\Omega\times[0,T], \\ u &= 0 \quad \text{in} \quad \Omega\times\{0\}, \end{cases} \tag{3.5.53}$$

this time assuming $f \in L^2(\Omega \times (0,T))$, while $\gamma > 0$ is the heat capacity constant and $\nu > 0$ is the thermal conductivity constant. We define the space

$$\mathcal{V} := \{v \in [(H_0^1(\Omega) \cap H^2(\Omega)) \otimes L^2(0,T)] \cap [L^2(\Omega) \otimes H^1(0,T)] \text{ s.t. } v = 0 \text{ on } \Omega \times \{0\}\},$$

endowed with the norm

$$\|v\|_{\mathcal{V}}^2 := \int_0^T \|\Delta v(\cdot,t)\|_{L^2(\Omega)}^2 \mathrm{dt} + \int_0^T \|\partial_t v(\cdot,t)\|_{L^2(\Omega)}^2 \mathrm{dt}. \tag{3.5.54}$$

The minimum regularity of the spline spaces that we assume is the following.

Assumption 3.4. *We assume that $p_s \geq 2$, $p_t \geq 1$ and that $\widehat{\mathcal{S}}_{h_s}^{\boldsymbol{p}_s} \subset C^1(\widehat{\Omega})$ and $\widehat{\mathcal{S}}_{h_t}^{p_t} \subset C^0\left((0,1)\right)$.*

Under Assumptions 3.3 and 3.4, $\mathcal{V}$ is a Hilbert space and the $\|\cdot\|_{\mathcal{V}}$-norm is equivalent to

$$|||v|||^2 := \|v\|_{H^2(\Omega)\otimes L^2(0,T)}^2 + \|v\|_{L^2(\Omega)\otimes H^1(0,1)}^2. \tag{3.5.55}$$

The Least Squares space-time variational formulation for system (3.2.13) reads: find $u \in \mathcal{V}$ such that

$$u = \arg\min_{v\in\mathcal{V}} \frac{1}{2}\|\gamma\partial_t v - \nu\Delta v - f\|_{L^2(\Omega\times(0,T))}^2. \tag{3.5.56}$$

Its Euler-Lagrange equation is

$$\mathcal{B}(u,v) = \mathcal{F}(v), \quad \forall v \in \mathcal{V}, \tag{3.5.57}$$

where the bilinear form $\mathcal{B}(\cdot,\cdot)$ and the linear form $\mathcal{F}(\cdot)$ are defined as

$$\begin{aligned} \mathcal{B}(v,w) &:= \int_0^T \int_\Omega (\gamma^2 \partial_t v \partial_t w + \nu^2 \Delta v \Delta w - \gamma\nu\partial_t v \Delta w - \gamma\nu\Delta v \partial_t w)\, \mathrm{d}\Omega\, \mathrm{dt}, \\ \mathcal{F}(w) &:= \int_0^T \int_\Omega f(\gamma\partial_t w - \nu\Delta w)\, \mathrm{d}\Omega\, \mathrm{dt}. \end{aligned} \tag{3.5.58}$$

Notice that $\mathcal{B}$ is a $\mathcal{V}$-elliptic continuous bilinear form and $\mathcal{F}$ is continuous in $\mathcal{V}$. Therefore, the well-posedness of the variational formulation above is a classical result that follows from Lax-Milgram Theorem.

The previous setting can be generalized to non-homogeneous initial and boundary conditions. For example, suppose that in (3.2.13) we have the initial condition $u = u_0 \in \Omega \times \{0\}$, with $u_0 \in H_0^1(\Omega)$, we lift u_0 to $\widetilde{u}_0 \in (H_0^1(\Omega) \cap H^2(\Omega)) \otimes L^2(0,T) \cap L^2(\Omega) \otimes H^1(0,T)$. Then $\widetilde{u} = u - \widetilde{u}_0 \in \mathcal{V}$ is the solution of

$$\left\{ \begin{array}{rcll} \gamma\partial_t\widetilde{u} - \nabla\cdot(\nu\nabla\widetilde{u}) & = & \widetilde{f} & \text{in} \quad \Omega\times(0,T), \\ \widetilde{u} & = & 0 & \text{on} \quad \partial\Omega\times[0,T], \\ \widetilde{u} & = & 0 & \text{in} \quad \Omega\times\{0\}, \end{array} \right. \tag{3.5.59}$$

where $\widetilde{f} := f - \gamma\partial_t\widetilde{u}_0 + \nabla\cdot(\nu\nabla\widetilde{u}_0)$. For a detailed description of the variational formulation of problems (3.5.53)-(3.5.59) and their well-posedness see, for example, [45, 96].

3.5.1 Space-time Galerkin method

Denote by $\mathcal{V}_h := \mathcal{X}_h$ endowed with the $\|\cdot\|_{\mathcal{V}}$-norm. Thanks to Assumption 3.4, it holds

$$\mathcal{V}_h \subset (H_0^1(\Omega) \cap H^2(\Omega)) \otimes H^1(0,T) \subset \mathcal{V}. \tag{3.5.60}$$

Therefore, we consider a Galerkin method for (3.5.57), that is, the least squares approximation of the system (3.5.53): find $u_h \in \mathcal{X}_h$ such that

$$u_h = \arg\min_{v_h \in \mathcal{V}_h} \frac{1}{2} \|\gamma \partial_t v_h - \nu \Delta v_h - f\|^2_{L^2(\Omega \times (0,T))}. \tag{3.5.61}$$

Its Euler-Lagrange equation is

$$\mathcal{B}(u_h, v_h) = \mathcal{F}(v_h), \quad \forall v_h \in \mathcal{V}_h, \tag{3.5.62}$$

Well-posedness and quasi optimality follow from standard arguments.

Theorem 3.4. *The minimization problem* (3.5.61) *and the variational problem* (3.5.62) *are equivalent and they admit a unique solution $u_h \in \mathcal{V}_h$. It also holds:*

$$\|u - u_h\|_{\mathcal{V}} \leq \sqrt{2} \inf_{v_h \in \mathcal{V}_h} \|u - v_h\|_{\mathcal{V}} \tag{3.5.63}$$

Proof. The proof of the equivalence and the existence and uniqueness of the solution follow by using Lax-Milgram Theorem, while the proof of (3.5.63) is a consequence of the Ceà Lemma and the symmetry of the bilinear form $\mathcal{B}$. □

We have then the following a-priori estimate for h-refinement.

Theorem 3.5. *Let q_s and q_t be two positive integers such that $q_s \geq 2$ and $q_t \geq 1$. If $u \in \mathcal{V} \cap (H^{q_s}(\Omega) \otimes H^1(0,T)) \cap (H^2(\Omega) \otimes H^{q_t}(0,T))$ is the solution of* (3.5.53) *and $u_h \in \mathcal{V}_h$ is the solution of* (3.5.62)*, then*

$$\|u - u_h\|_{\mathcal{V}} \leq C(h_s^{k_s - 2} \|u\|_{H^{k_s}(\Omega) \otimes H^1(0,T)} + h_t^{k_t - 1} \|u\|_{H^2(\Omega) \otimes H^{k_t}(0,T)}), \tag{3.5.64}$$

where $k_s := \min\{q_s, p_s + 1\}$, $k_t := \min\{q_t, p_t + 1\}$,$C$ is a constant that depends only on p_s,p_t,α and the parameterization $\boldsymbol{G}$

The result follows from the anisotropic approximation estimates that are developed in [14]. An overview of the proof is given in [85].

3.5.2 Discrete system

Before introducing the discrete system, we rewrite the bilinear form $\mathcal{B}(\cdot,\cdot)$ in an equivalent way, through the following Lemma.

Lemma 3.6. *The bilinear form $\mathcal{B}(\cdot,\cdot)$ can be written as*

$$\mathcal{B}(v_h, w_h) = \gamma^2 \int_0^T \int_\Omega \partial_t v_h \partial_t w_h \, \mathrm{d}\Omega \, \mathrm{dt} + \nu^2 \int_0^T \int_\Omega \Delta v_h \Delta w_h \, \mathrm{d}\Omega \, \mathrm{dt} + \gamma\nu \int_\Omega \nabla v_h(\boldsymbol{x}, T) \cdot \nabla w_h(\boldsymbol{x}, T) \, \mathrm{d}\Omega, \tag{3.5.65}$$

for all $v_h, w_h \in \mathcal{V}_h$.

Proof. Let $v_h, w_h \in \mathcal{V}_h$. First note that $\partial_t v_h, \partial_t w_h \in (H_0^1(\Omega) \cap H^2(\Omega)) \otimes L^2(0,T)$, from (3.5.60), and $\partial_t v_h, \partial_t w_h = 0$ on $\partial\Omega \times [0,T]$. Using Green's formula and integrating by parts yields to

$$
\begin{aligned}
&-\int_0^T \int_\Omega (\partial_t v_h \Delta w_h + \Delta v_h \partial_t w_h)\, \mathrm{d}\Omega\, \mathrm{dt} = \\
-\int_0^T \int_{\partial\Omega} (\partial_t v_h \nabla w_h + \nabla v_h \partial_t w_h) \cdot \boldsymbol{n}\, \mathrm{d}\Omega\, \mathrm{dt} + &\int_0^T \int_\Omega [\nabla(\partial_t v_h) \nabla w_h + \nabla v_h \nabla(\partial_t w_h)]\, \mathrm{d}\Omega\, \mathrm{dt} = \\
&+\int_0^T \partial_t \left[\int_\Omega \nabla v_h \cdot \nabla w_h \, \mathrm{d}\Omega\right] \mathrm{dt} = \\
&+\int_\Omega [\nabla v_h(\boldsymbol{x},T) \cdot \nabla w_h(\boldsymbol{x},T) - \nabla v_h(\boldsymbol{x},0) \cdot \nabla w_h(\boldsymbol{x},0)]\, \mathrm{d}\Omega = \\
&+\int_\Omega [\nabla v_h(\boldsymbol{x},T) \cdot \nabla w_h(\boldsymbol{x},T)]\, \mathrm{d}\Omega,
\end{aligned}
$$

where $\boldsymbol{n} \in \mathbb{R}^d$ is the external normal unit vector to $\partial\Omega$. Then (3.5.65) follows. □

Remark 3.2. *Note that the identity* (3.5.65) *holds also in the continuous setting (see [85, Appendix B]).*

The linear system associate to (3.5.62) is

$$\mathbf{Bu} = \mathbf{f}, \tag{3.5.66}$$

where $[\mathbf{B}]_{i,j} := \mathcal{B}(B_{j,\boldsymbol{p}}, B_{i,\boldsymbol{p}})$ and $[\mathbf{f}]_i := \mathcal{F}(B_{i,\boldsymbol{p}})$. The discrete system matrix $\mathbf{B}$ can be written as sum of Kronecker product matrices

$$\mathbf{B} = \gamma^2 \mathbf{L}_t \otimes \mathbf{M}_s + \nu^2 \mathbf{M}_t \otimes \mathbf{J}_s + \gamma\nu \mathbf{R}_t \otimes \mathbf{L}_s, \tag{3.5.67}$$

where the time matrices involved are, for $i,j = 1,\dots,N_t$

$$[\mathbf{L}_t]_{i,j} := \int_0^T b'_{j,p_t}(t)\, b'_{i,p_t}(t)\, \mathrm{dt}, \quad [\mathbf{M}_t]_{i,j} := \int_0^T b_{j,p_t}(t)\, b_{i,p_t}(t)\, \mathrm{dt}, \quad [\mathbf{R}_t]_{i,j} := b_{j,p_t}(T)\, b_{i,p_t}(T), \tag{3.5.68a}$$

and the space matrices are $i,j = 1,\dots,N_s$

$$
\begin{gathered}
[\mathbf{J}_s]_{i,j} := \int_\Omega \Delta B_{j,\boldsymbol{p}_s}(\boldsymbol{x}) \Delta B_{i,\boldsymbol{p}_s}(\boldsymbol{x}) \;\, \mathrm{d}\Omega, \quad [\mathbf{M}_s]_{i,j} := \int_\Omega B_{j,\boldsymbol{p}_s}(\boldsymbol{x}) \; B_{i,\boldsymbol{p}_s}(\boldsymbol{x}) \;\, \mathrm{d}\Omega, \\
[\mathbf{L}_s]_{i,j} := \int_\Omega \nabla B_{j,\boldsymbol{p}_s}(\boldsymbol{x}) \cdot \nabla B_{i,\boldsymbol{p}_s}(\boldsymbol{x}) \;\, \mathrm{d}\Omega.
\end{gathered}
\tag{3.5.68b}
$$

3.6 Diagonalizable preconditioner

The matrix $\mathbf{B}$ in (3.5.66) is symmetric and positive definite. Thus, we design and analyze a suitable symmetric positive definite preconditioner to be used for a preconditioned Conjugate Gradients method. Recall $\widehat{\mathcal{X}}_h$ is the spline space defined in Section 3.1.1, satisfying the regularity condition of Assumption 3.4.

The definition of the preconditioner is associated with the bilinear form $\widehat{\mathcal{B}} : \widehat{\mathcal{X}}_h \times \widehat{\mathcal{X}}_h \to \mathbb{R}$ defined as

$$\widehat{\mathcal{B}}(\widehat{v}_h, \widehat{w}_h) := \gamma^2 \int_0^1 \int_{\widehat{\Omega}} \partial_\tau \widehat{v}_h \partial_\tau \widehat{w}_h \, \mathrm{d}\widehat{\Omega} \ \mathrm{d}\tau + \nu^2 \sum_{l=1}^{d} \int_0^1 \int_{\widehat{\Omega}} \frac{\partial^2 \widehat{v}_h}{\partial \eta_l^2} \frac{\partial^2 \widehat{w}_h}{\partial \eta_l^2} \, \mathrm{d}\widehat{\Omega} \ \mathrm{d}\tau \quad (3.6.69)$$

and with the corresponding norm

$$\|\widehat{v}_h\|^2_{\widehat{\mathcal{B}}} := \widehat{\mathcal{B}}(\widehat{v}_h, \widehat{v}_h). \quad (3.6.70)$$

Thus, the definition of the preconditioner is

$$[\widehat{\mathbf{B}}]_{i,j} := \widehat{\mathcal{B}}(\widehat{B}_{j,\boldsymbol{p}}, \widehat{B}_{i,\boldsymbol{p}}), \quad i, j = 1, \ldots, N_{dof}, \quad (3.6.71)$$

and has the following structure

$$\widehat{\mathbf{B}} = \gamma^2 \widehat{\mathbf{L}}_t \otimes \widehat{\mathbf{M}}_s + \nu^2 \widehat{\mathbf{M}}_t \otimes \widetilde{\mathbf{J}}_s, \quad (3.6.72)$$

where for $i, j = 1, \ldots, N_t$

$$[\widehat{\mathbf{L}}_t]_{i,j} := \int_0^1 \widehat{b}'_{j,p_t}(\tau) \widehat{b}'_{i,p_t}(\tau) \ \mathrm{d}\tau, \quad [\widehat{\mathbf{M}}_t]_{i,j} := \int_0^1 \widehat{b}_{j,p_t}(\tau) \widehat{b}_{i,p_t}(\tau) \ \mathrm{d}\tau,$$

while for $i, j = 1, \ldots, N_s$

$$[\widetilde{\mathbf{J}}_s]_{i,j} := \sum_{l=1}^{d} \int_0^1 \frac{\partial^2 \widehat{B}_{j,\boldsymbol{p}_s}(\boldsymbol{\eta})}{\partial \eta_l^2} \frac{\partial^2 \widehat{B}_{i,\boldsymbol{p}_s}(\boldsymbol{\eta})}{\partial \eta_l^2} \ \mathrm{d}\widehat{\Omega}, \quad [\widehat{\mathbf{M}}_s]_{i,j} := \int_0^1 \widehat{B}_{j,\boldsymbol{p}_s}(\boldsymbol{\eta}) \widehat{B}_{i,\boldsymbol{p}_s}(\boldsymbol{\eta}) \ \mathrm{d}\widehat{\Omega}$$

Notice that $\widehat{\mathbf{L}}_t, \widehat{\mathbf{M}}_t$ and $\widehat{\mathbf{M}}_s$ correspond to $\mathbf{L}_t, \mathbf{M}_t$ and $\mathbf{M}_s$, respectively, where the integration is performed on the parametric domain $\widehat{\Omega}$. Moreover, the matrices $\widetilde{\mathbf{J}}_s$ and $\widehat{\mathbf{M}}_s$ can be further factorized as sum of Kronecker products as

$$[\widetilde{\mathbf{J}}_s]_{i,j} = \sum_{l=1}^{d} \widehat{\mathbf{M}}_d \otimes \ldots \otimes \widehat{\mathbf{M}}_{l-1} \otimes \widehat{\mathbf{J}}_l \otimes \widehat{\mathbf{M}}_{l-1} \otimes \ldots \otimes \widehat{\mathbf{M}}_1, \quad \widehat{\mathbf{M}}_s = \widehat{\mathbf{M}}_d \otimes \ldots \otimes \widehat{\mathbf{M}}_1,$$

where for $l = 1, \ldots, d$ and for $i, j = 1, \ldots, N_{s,l}$

$$[\widehat{\mathbf{J}}_l]_{i,j} := \int_0^1 \widehat{b}''_{j,p_s}(\eta_l) \widehat{b}''_{i,p_s}(\eta_l) \ \mathrm{d}\eta_l, \quad [\widehat{\mathbf{M}}_l]_{i,j} := \int_0^1 \widehat{b}_{j,p_s}(\eta_l) \widehat{b}_{i,p_s}(\eta_l) \ \mathrm{d}\eta_l.$$

The efficient application of the proposed preconditioner, that is, the solution of a liner system with matrix $\widehat{\mathbf{B}}$, should exploit the structure highlighted above. Again a possible approach is Fast Diagonalization method.

Finally, the following spectral stability of the preconditioned matrix $\widehat{\mathbf{B}}^{-1}\mathbf{B}$ holds true.

Theorem 3.7. *Under Assumptions 3.13.3 and 3.4, it holds*

$$\theta \leq \lambda_{min}(\widehat{\mathbf{B}}^{-1}\mathbf{B}), \quad \lambda_{max}(\widehat{\mathbf{B}}^{-1}\mathbf{B}) \leq \Theta,$$

where θ and Θ are positive constants that depend on $\boldsymbol{G}$, but do not depend on h_s, h_t, p_s and p_t.

For the proof we refer to [85, Theroem 4].

3.6.1 Stable factorization of $(\widehat{\mathbf{J}}_l, \widehat{\mathbf{M}}_l)$ for $l = 1, \ldots, d$

The spatial matrices $\widehat{\mathbf{J}}_l$ and $\widehat{\mathbf{M}}_l$ are symmetric and positive definite for $l = 1, \ldots, d$. Thus, the pencils $(\widehat{\mathbf{J}}_l, \widehat{\mathbf{M}}_l)$ for $l = 1, \ldots, d$ admit the generalized eigendecomposition

$$\widehat{\mathbf{J}}_l \mathbf{V}_l = \widehat{\mathbf{M}}_l \mathbf{V}_l \mathbf{\Lambda}_l,$$

where the matrices $\mathbf{V}_l$ contain in each column the $\widehat{\mathbf{M}}_l$-orthonormal generalized eigenvectors and $\mathbf{\Lambda}_l$ are diagonal matrices whose entries contain the generalized eigenvalues. Therefore we have for $l = 1, \ldots, d$ the factorizations

$$\mathbf{V}_l^T \widehat{\mathbf{J}}_l \mathbf{V}_l = \mathbf{\Lambda}_l \quad \text{and} \quad \mathbf{V}_l^T \widehat{\mathbf{M}}_l \mathbf{V}_l = \mathbb{I}_{N_{s,l}}, \tag{3.6.73}$$

where $\mathbb{I}_{N_{s,l}}$ denotes the identity matrix of dimension $N_{s,l} \times N_{s,l}$. The stability of the decomposition (3.6.73) is expressed by the condition number of the eigenvector matrix. In particular $\mathbf{V}_l^T \widehat{\mathbf{M}}_l \mathbf{V}_l = \mathbb{I}_{N_{s,l}}$ implies that $\kappa_2(\mathbf{V}_l) = \sqrt{\kappa_2(\widehat{\mathbf{M}}_l)}$ and it does not depend on the mesh-size, but it depends on the polynomial degree. Indeed, we report in Table 3.5 the behavior of $\kappa_2(\mathbf{V}_l)$ for different values of spline degree p_s and for different uniform discretizations with number of elements denoted by n_{el}. We observe that $\kappa_2(\mathbf{V}_l)$ exhibits a dependence only on p_s, but stays moderately low for all low polynomial degrees that are in the range of interest.

n_{el}	$p_s = 2$	$p_s = 3$	$p_s = 4$	$p_s = 5$	$p_s = 6$	$p_s = 7$	$p_s = 8$
32	$2.7 \cdot 10^0$	$4.5 \cdot 10^0$	$7.6 \cdot 10^0$	$1.3 \cdot 10^1$	$2.1 \cdot 10^1$	$3.5 \cdot 10^1$	$5.7 \cdot 10^1$
64	$2.7 \cdot 10^0$	$4.5 \cdot 10^0$	$7.6 \cdot 10^0$	$1.3 \cdot 10^1$	$2.1 \cdot 10^1$	$3.5 \cdot 10^1$	$5.7 \cdot 10^1$
128	$2.7 \cdot 10^0$	$4.5 \cdot 10^0$	$7.6 \cdot 10^0$	$1.3 \cdot 10^1$	$2.1 \cdot 10^1$	$3.5 \cdot 10^1$	$5.7 \cdot 10^1$
256	$2.7 \cdot 10^0$	$4.5 \cdot 10^0$	$7.6 \cdot 10^0$	$1.3 \cdot 10^1$	$2.1 \cdot 10^1$	$3.5 \cdot 10^1$	$5.7 \cdot 10^1$
512	$2.7 \cdot 10^0$	$4.5 \cdot 10^0$	$7.6 \cdot 10^0$	$1.3 \cdot 10^1$	$2.1 \cdot 10^1$	$3.5 \cdot 10^1$	$5.7 \cdot 10^1$
1024	$2.7 \cdot 10^0$	$4.5 \cdot 10^0$	$7.6 \cdot 10^0$	$1.3 \cdot 10^1$	$2.1 \cdot 10^1$	$3.5 \cdot 10^1$	$5.7 \cdot 10^1$

Table 3.5. $\kappa_2(\mathbf{V}_l)$ for different polynomial degrees p_s and number of elements n_{el}.

3.6.2 Stable factorization in time

The time matrices $\widehat{\mathbf{L}}_t$ and $\widehat{\mathbf{M}}_t$ are symmetric positive definite, therefore they admit a generalized eigendecomposition of the kind

$$\widehat{\mathbf{L}}_t \mathbf{V}_t = \mathbf{\Lambda}_t \widehat{\mathbf{M}}_t \mathbf{V}_t, \tag{3.6.74}$$

where the matrix $\mathbf{V}_t$ contains in each column the $\widehat{\mathbf{M}}_t$-orthonormal generalized eigenvectors and $\mathbf{\Lambda}_t$ is the diagonal matrix whose entries contain the generalized eigenvalues. We have the following factorization

$$\mathbf{V}_t^T \widehat{\mathbf{L}}_t \mathbf{V}_t = \mathbf{\Lambda}_t \quad \text{and} \quad \mathbf{V}_t^T \widehat{\mathbf{M}}_t \mathbf{V}_t = \mathbb{I}_{N_t}. \tag{3.6.75}$$

Figure 3.5 shows the shape of the generalized eigenvectors in $\mathbf{V}_t$, with associated eigenvalue in $\mathbf{\Lambda}_t$, for a fixed univariate direction $l = 1, \ldots, d$ discretized with degree $p_s = 3$ B-Splines and uniform partition. The stability of the decomposition (3.6.74)

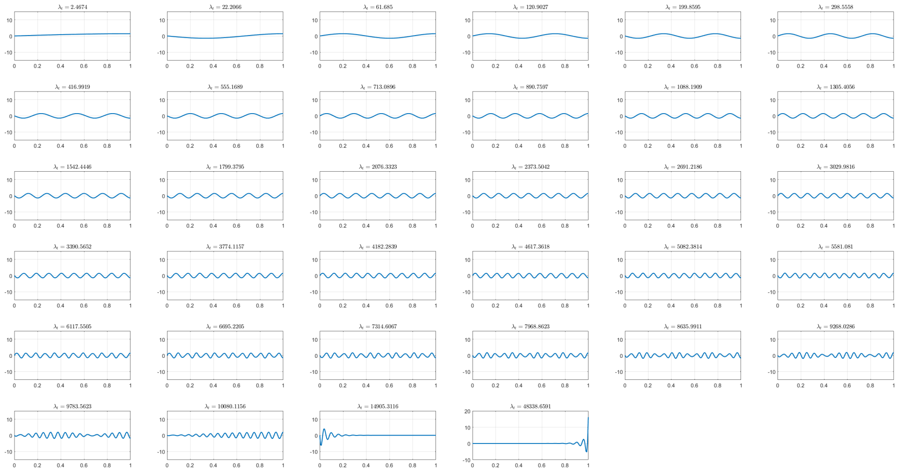


Figure 3.5. Generalized eigenvectors for the pencil $(\widehat{\mathbf{L}}_t,\ \widehat{\mathbf{M}}_t)$, with associated eigenvalues for $p_t = 3$ and $n_{el} = 32$ elements.

is again expressed by the condition number of the eigenvector matrix, and since $\mathbf{V}_t^T \widehat{\mathbf{M}}_t \mathbf{V}_t = \mathbb{I}_{N_t}$, it holds $\kappa_2(\mathbf{V}_t) = \sqrt{\kappa_2(\widehat{\mathbf{M}}_t)}$.

We investigate the stability of the diagonalization (3.6.74) by computing the condition number $\kappa_2(\mathbf{V}_t)$ for different values of spline degree p_t and for various uniform discretizations with number of elements n_{el}. The results, reported in Table 3.6, show that the condition numbers $\kappa_2(\mathbf{V}_t)$ are uniformly bounded w.r.t. the mesh refinement, they grow with respect to the polynomial degree but they are moderately small for all the degrees of interest. We conclude that the generalized diagonalization (3.6.74) for the time pencil $(\widehat{\mathbf{L}}_t, \widehat{\mathbf{M}}_t)$ is stable.

n_{el}	$p_t = 2$	$p_t = 3$	$p_t = 4$	$p_t = 5$	$p_t = 6$	$p_t = 7$	$p_t = 8$
32	$3.2 \cdot 10^0$	$5.2 \cdot 10^0$	$8.3 \cdot 10^0$	$1.3 \cdot 10^1$	$2.2 \cdot 10^1$	$3.6 \cdot 10^1$	$5.9 \cdot 10^1$
64	$3.3 \cdot 10^0$	$5.2 \cdot 10^0$	$8.3 \cdot 10^0$	$1.3 \cdot 10^1$	$2.2 \cdot 10^1$	$3.6 \cdot 10^1$	$5.9 \cdot 10^1$
128	$3.3 \cdot 10^0$	$5.2 \cdot 10^0$	$8.3 \cdot 10^0$	$1.3 \cdot 10^1$	$2.2 \cdot 10^1$	$3.6 \cdot 10^1$	$5.9 \cdot 10^1$
256	$3.3 \cdot 10^0$	$5.2 \cdot 10^0$	$8.3 \cdot 10^0$	$1.3 \cdot 10^1$	$2.2 \cdot 10^1$	$3.6 \cdot 10^1$	$5.9 \cdot 10^1$
512	$3.3 \cdot 10^0$	$5.2 \cdot 10^0$	$8.3 \cdot 10^0$	$1.3 \cdot 10^1$	$2.2 \cdot 10^1$	$3.6 \cdot 10^1$	$5.9 \cdot 10^1$
1024	$3.3 \cdot 10^0$	$5.2 \cdot 10^0$	$8.3 \cdot 10^0$	$1.3 \cdot 10^1$	$2.2 \cdot 10^1$	$3.6 \cdot 10^1$	$5.9 \cdot 10^1$

Table 3.6. $\kappa_2(\mathbf{V}_t)$ for different degrees p_t and number of elements n_{el}.

3.6.3 Preconditioner application

The application of the preconditioner involves the solution of the linear system

$$\widehat{\mathbf{B}}^{-1}\mathbf{s} = \mathbf{r}, \tag{3.6.76}$$

where $\widehat{\mathbf{B}}$ has the structure (3.6.72). We are able to efficiently solve system (3.6.76) by Fast Diagonalization (FD) method. The starting point, is the setup of the preconditioner, that is

- for the pencils $(\widehat{\mathbf{J}}_l, \widehat{\mathbf{M}}_l)$, with $l = 1, \dots, d$ we have the factorization (3.6.73).

- for the pencil $(\widehat{\mathbf{L}}_t, \widehat{\mathbf{M}}_t)$ we have the factorization (3.6.75).

Then, by defining $\mathbf{V}_s := \mathbf{V}_d \otimes \cdots \otimes \mathbf{V}_1$ and $\mathbf{\Lambda}_s := \sum_{l=1}^d \mathbb{I}_{N_{s,d}} \otimes \cdots \otimes \mathbb{I}_{N_{s,l+1}} \otimes \mathbf{\Lambda}_l \otimes \mathbb{I}_{N_{s,l-1}} \otimes \cdots \otimes \mathbb{I}_{N_{s,1}}$, we have for the matrix $\widehat{\mathbf{B}}$ the factorization

$$\widehat{\mathbf{B}} = \left(\mathbf{V}_t^T \otimes \mathbf{V}_s^T\right)^{-1} \left(\gamma^2 \mathbf{\Lambda}_t \otimes \mathbb{I}_{N_s} + \nu^2 \mathbb{I}_{N_t} \otimes \mathbf{\Lambda}_s\right) \left(\mathbf{V}_t \otimes \mathbf{V}_s\right)^{-1}. \tag{3.6.77}$$

Notice that, the second factor in (3.6.77) is diagonal. Therefore, the solution of (3.6.76) can be computed by the following algorithm.

Algorithm 3 Fast Diagonalization

```
1: Compute the factorizations (3.6.73) and (3.6.75).
2: Compute $\widetilde{\mathbf{q}} = (\mathbf{V}_t^T \otimes \mathbf{V}_s^T)\mathbf{r}$.
3: Compute $\widetilde{\mathbf{s}} = (\gamma^2 \mathbf{\Lambda}_t \otimes \mathbb{I}_{N_s} + \nu^2 \mathbb{I}_{N_t} \otimes \mathbf{\Lambda}_s)^{-1} \widetilde{\mathbf{q}}$.
4: Compute $\mathbf{s} = (\mathbf{V}_t \otimes \mathbf{V}_s)\, \widetilde{\mathbf{s}}$.
```

3.6.4 Preconditioner robustness: partial inclusion of the geometry

The spectral estimate in Theorem 3.7 show the dependence on $\boldsymbol{G}$, that is, the geometry parametrization affects the performance of our preconditioner (3.6.71), as it is confirmed by the numerical tests in Section 3.9. In this section, we present a strategy to partially incorporate $\boldsymbol{G}$ in the preconditioner, without increasing its computational cost. The same idea has been used in [87] for the Stokes problem.

Let us split the bilinear form $\mathcal{B}(\cdot, \cdot)$ as

$$\mathcal{B}(v_h, w_h) = \mathcal{B}_t(v_h, w_h) + \mathcal{B}_s(v_h, w_h) - \mathcal{R}(v_h, w_h), \quad \forall v_h, w_h \in \mathcal{V}_h,$$

where

$$\mathcal{B}_t(v_h, w_h) := \gamma^2 \int_0^T \int_\Omega \partial_t v_h \partial_t w_h \, \mathrm{d}\Omega \, \mathrm{dt}, \quad \mathcal{B}_s(v_h, w_h) := \nu^2 \int_0^T \int_\Omega \Delta v_h \Delta w_h \, \mathrm{d}\Omega \, \mathrm{dt},$$
$$\mathcal{R}(v_h, w_h) := \gamma\nu \int_0^T \int_\Omega \partial_t v_h \Delta w_h + \Delta v_h \partial_t w_h \, \mathrm{d}\Omega \, \mathrm{dt}.$$

Using that $v_h = \widehat{v}_h \circ \boldsymbol{G}^{-1}$, $w_h = \widehat{w}_h \circ \boldsymbol{G}^{-1}$ and for $i = 1, \dots, d$,

$$\frac{\partial^2 v_h}{\partial x_i^2} = \sum_{j,k=1}^d \frac{\partial^2 \widehat{v}_h \circ \boldsymbol{G}^{-1}}{\partial \eta_j \partial \eta_k} [J_{\boldsymbol{F}}^{-1}]_{k,i} [J_{\boldsymbol{F}}^{-1}]_{j,i} + \sum_{j=1}^d \frac{\partial \widehat{v}_h \circ \boldsymbol{G}^{-1}}{\partial \eta_j} \frac{\partial [J_{\boldsymbol{F}}^{-1}]_{j,i}}{\partial \eta_i},$$

we can write $\mathcal{B}_t$ and $\mathcal{B}_s$ as

$$\mathcal{B}_t(v_h, w_h) = \int_0^1 \int_{\widehat{\Omega}} c_{d+1} \partial_\tau \widehat{v}_h \partial_\tau \widehat{w}_h \, \mathrm{d}\widehat{\Omega} \, \mathrm{d}\tau, \quad \mathcal{B}_s(v_h, w_h) = \mathcal{B}_{s,1}(\widehat{v}_h, \widehat{w}_h) + \mathcal{B}_{s,2}(\widehat{v}_h, \widehat{w}_h),$$

with

$$
\begin{aligned}
\mathcal{B}_{s,1}(\widehat{v}_h, \widehat{w}_h) &:= \sum_{l=1}^{d} \int_0^1 \int_{\widehat{\Omega}} c_l \frac{\partial^2 \widehat{v}_h}{\partial \eta_l^2} \frac{\partial^2 \widehat{w}_h}{\partial \eta_l^2} \ \mathrm{d}\widehat{\Omega} \ \mathrm{d}\tau, \\
\mathcal{B}_{s,2}(\widehat{v}_h, \widehat{w}_h) &:= \sum_{\substack{r,s=1 \\ r \neq s}}^{d} \sum_{\substack{j,k=1 \\ j \neq k}}^{d} \int_0^1 \int_{\widehat{\Omega}} g^1_{rsjk} \frac{\partial^2 \widehat{v}_h}{\partial \eta_k \partial \eta_j} \frac{\partial^2 \widehat{w}_h}{\partial \eta_r \partial \eta_s} \ \mathrm{d}\widehat{\Omega} \ \mathrm{d}\tau + \\
&+ \sum_{j,k=1}^{d} \int_0^1 \int_{\widehat{\Omega}} g^2_{kj} \frac{\partial \widehat{v}_h}{\partial \eta_k} \frac{\partial \widehat{w}_h}{\partial \eta_j} \ \mathrm{d}\widehat{\Omega} \ \mathrm{d}\tau + \\
&+ \sum_{r=1}^{d} \sum_{j,k=1}^{d} \int_0^1 \int_{\widehat{\Omega}} g^3_{rjk} \frac{\partial^2 \widehat{v}_h}{\partial \eta_k \partial \eta_j} \frac{\partial \widehat{w}_h}{\partial \eta_r} + \frac{\partial \widehat{v}_h}{\partial \eta_r} \frac{\partial^2 \widehat{w}_h}{\partial \eta_k \partial \eta_j} \ \mathrm{d}\widehat{\Omega} \ \mathrm{d}\tau,
\end{aligned}
$$

and where we have defined

$$
c_l := \nu^2 \left(\| [J_{\boldsymbol{F}}^{-1}]_{\cdot,l} \|_2 \right)^4 |\det(J_{\boldsymbol{F}})| T, \quad \text{for } l = 1, \dots, d, \quad c_{d+1} := \gamma^2 |\det(J_{\boldsymbol{F}})| T^{-1},
$$

while $g^1_{rsjk}, g^2_{j,k}, g^3_{rjk}$ are functions that depend on the parameterization $\boldsymbol{G}$. The preconditioner will be based on an approximation of $\mathcal{B}_t, \mathcal{B}_{s,1}$ only. In particular we approximate c_l for $l = 1, \dots, d+1$ as

$$
c_l(\boldsymbol{\eta}, \tau) \approx \mu_1(\eta_1) \dots \mu_{l-1}(\eta_{l-1}) \omega_l(\eta_l) \mu_{l+1}(\eta_{l+1}) \dots \mu_d(\eta_d) \mu_{d+1}(\tau), \tag{3.6.78a}
$$

$$
c_{d+1}(\boldsymbol{\eta}, \tau) \approx \mu_1(\eta_1) \dots \mu_d(\eta_d) \omega_{d+1}(\tau). \tag{3.6.78b}
$$

The functions c_l in (3.6.78) are first interpolated by constants in each element and then the construction of the univariate factors μ_l and ω_l is performed by the separation of variables algorithm detailed in [85, Appendix C]. The resulting computational cost is proportional to the number of elements, which for smooth splines i roughly equal to N_{dof}, independent of the degrees p_s and p_t, and therefore negligible in the whole iterative strategy.

This first step leads to a matrix of this form

$$
\widetilde{\mathbf{B}} = \widehat{\mathbf{L}}_t^{\boldsymbol{G}} \otimes \widehat{\mathbf{M}}_s^{\boldsymbol{G}} + \widehat{\mathbf{M}}_t^{\boldsymbol{G}} \otimes \widetilde{\mathbf{J}}_s^{\boldsymbol{G}}
$$

where, with the notation of the basis functions in Section 3.1.1, for $i, j = 1, \dots, N_t$

$$
[\widehat{\mathbf{L}}_t]^{\boldsymbol{G}}_{i,j} := \int_0^1 \omega_{d+1}(\tau) \widehat{b}'_{j,p_t}(\tau) \widehat{b}'_{i,p_t}(\tau) \ \mathrm{d}\tau, \quad [\widehat{\mathbf{M}}_t]^{\boldsymbol{G}}_{i,j} := \int_0^1 \mu_{d+1}(\tau) \widehat{b}_{j,p_t}(\tau) \widehat{b}_{i,p_t}(\tau) \ \mathrm{d}\tau,
$$

while

$$
\widetilde{\mathbf{J}}_s^{\boldsymbol{G}} = \sum_{l=1}^{d} \widehat{\mathbf{M}}_d^{\boldsymbol{G}} \otimes \ldots \otimes \widehat{\mathbf{M}}_{l+1}^{\boldsymbol{G}} \otimes \widehat{\mathbf{J}}_l^{\boldsymbol{G}} \otimes \widehat{\mathbf{M}}_{l-1}^{\boldsymbol{G}} \otimes \ldots \otimes \widehat{\mathbf{M}}_1^{\boldsymbol{G}}, \quad \widehat{\mathbf{M}}_s^{\boldsymbol{G}} = \widehat{\mathbf{M}}_d^{\boldsymbol{G}} \otimes \ldots \otimes \widehat{\mathbf{M}}_1^{\boldsymbol{G}},
$$

and for $l = 1, \dots, d$ and for $i, j = 1, \dots, N_{s,l}$

$$
[\widehat{\mathbf{J}}_l]^{\boldsymbol{G}}_{i,j} := \int_0^1 \omega_l(\eta_l) \widehat{b}''_{j,p_s}(\eta_l) \widehat{b}''_{i,p_s}(\eta_l) \ \mathrm{d}\eta_l, \quad [\widehat{\mathbf{M}}_l]^{\boldsymbol{G}}_{i,j} := \int_0^1 \mu_l(\eta_l) \widehat{b}_{j,p_s}(\eta_l) \widehat{b}_{i,p_s}(\eta_l) \ \mathrm{d}\eta_l.
$$

The matrix $\widetilde{\mathbf{B}}$ maintains the Kronecker structure of (3.6.71) and Algorithm 3 can still be used to compute its application.

Finally, we apply a diagonal scaling and define the preconditioner as

$$\widehat{\mathbf{B}}^{\boldsymbol{G}} := \mathbf{D}^{1/2}\widetilde{\mathbf{B}}\mathbf{D}^{1/2} \tag{3.6.79}$$

where $\mathbf{D}$ is the diagonal matrix whose entries are $[\mathbf{D}]_{i,i} := \dfrac{[\mathbf{B}]_{i,i}}{[\widehat{\mathbf{B}}^{\boldsymbol{G}}]_{i,i}}$ for $i = 1, \dots, N_{dof}$.

Remark 3.3. *For the model problem considered, the approximation of the geometry parametrization in the time direction is trivial. Notice that the coefficients in* (3.6.78) *do not depend on τ. Indeed, in our case it holds*

$$\mathbf{L}_t = \frac{1}{T}\widehat{\mathbf{L}}_t, \quad \mathbf{M}_t = T\widehat{\mathbf{M}}_t,$$

and hence we could set explicitly $\widehat{\mathbf{L}}_t^{\boldsymbol{G}} = \mathbf{L}_t$ and $\widehat{\mathbf{M}}_t^{\boldsymbol{G}} = \mathbf{M}_t$, which is exact. However, we want to present the more general approximating strategy above which could be used also when the spatial geometry or equation's coefficients depend on time.

3.7 L^2 least squares preconditioner

In this section, we propose a second preconditioner for problem (3.5.66), which exploits the Kronecker structure of (3.5.67). The definition of the preconditioner is

$$\widehat{\mathbf{Q}} := \gamma^2\mathbf{L}_t \otimes \widehat{\mathbf{M}}_s + \nu^2\mathbf{M}_t \otimes \widehat{\mathbf{L}}_s\widehat{\mathbf{M}}_s^{-1}\widehat{\mathbf{L}}_s + \gamma\nu\mathbf{R}_t \otimes \widehat{\mathbf{L}}_s, \tag{3.7.80}$$

where $\widehat{\mathbf{L}}_s$ and $\widehat{\mathbf{M}}_s$ are defined in (3.3.21), while $\mathbf{L}_t, \mathbf{M}_{\mathbf{t}}$ and $\mathbf{R}_t$ are defined in (3.5.68a). We recall that, by Remark 3.3, $\mathbf{L}_t = \dfrac{1}{T}\widehat{\mathbf{L}}_t$ and $\mathbf{M}_t = T\widehat{\mathbf{M}}_t$. Therefore, the preconditioner $\widehat{\mathbf{Q}}$ can be written as

$$\widehat{\mathbf{Q}} = \frac{\gamma^2}{T}\widehat{\mathbf{L}}_t \otimes \widehat{\mathbf{M}}_s + \nu^2 T\widehat{\mathbf{M}}_t \otimes \widehat{\mathbf{L}}_s\widehat{\mathbf{M}}_s^{-1}\widehat{\mathbf{L}}_s + \gamma\nu\mathbf{R}_t \otimes \widehat{\mathbf{L}}_s. \tag{3.7.81}$$

Notice that,instead of the matrix $\widetilde{\mathbf{J}}_s$ of the previous section, we consider the matrix $\widehat{\mathbf{L}}_s\widehat{\mathbf{M}}_s^{-1}\widehat{\mathbf{L}}_s$ in the spacial factor of the middle term. This choice is motivated by the spectral equivalence between the two matrices, see [58, Proposition 4.1]. We investigate numerically the stability of such spectral equivalence, and report in Figure 3.6 the eigenvalues of $(\widehat{\mathbf{L}}_s\widehat{\mathbf{M}}_s^{-1}\widehat{\mathbf{L}}_s)^{-1}\mathbf{J}_s$, which are clustered close to 1, for different mesh sizes h and different polynomial degrees p. In conclusion the spectral equivalence is stable under h-refinement and p-refinement.

Finally, the efficient application of the preconditioner exploits the Kronecker structure in (3.7.80), and again the implementation involves Fast Diagonalization method, together with Sherman-Morrison formula.

3.7.1 Stable factorization in space and time

The space pencils $(\widehat{\mathbf{L}}_l, \widehat{\mathbf{M}}_l)$ for $l = 1, \dots, d$ admit the stable factorizations described in Section 3.3.3, that is

$$\mathbf{U}_l^T\widehat{\mathbf{L}}_l\mathbf{U}_l = \mathbf{\Lambda}_l \quad \text{and} \quad \mathbf{U}_l^T\widehat{\mathbf{M}}_l\mathbf{U}_l = \mathbb{I}_{N_{s,l}},$$

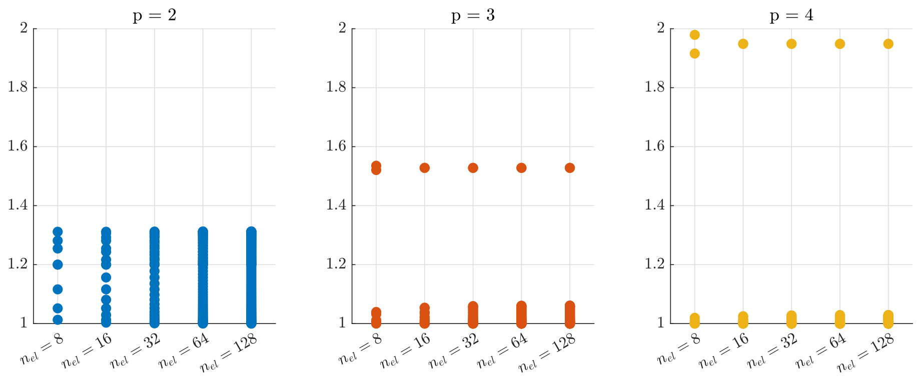


Figure 3.6. Eigenvalues of $(\widehat{\mathbf{L}}_s^T\widehat{\mathbf{M}}_s^{-1}\widehat{\mathbf{L}}_s)^{-1}\widetilde{\mathbf{J}}_s$ for different degrees p and number of elements n_{el}.

where $\mathbb{I}_{N_{s,l}}$ denotes the identity matrix of dimension $N_{s,l} \times N_{s,l}$.

Whereas, the time pencil $(\widehat{\mathbf{L}}_t, \widehat{\mathbf{M}}_t)$ admit the stable factorization described in Section 3.6.2, that is

$$\mathbf{V}_t^T\widehat{\mathbf{L}}_t\mathbf{V}_t = \mathbf{\Lambda}_t \quad \text{and} \quad \mathbf{V}_t^T\widehat{\mathbf{M}}_t\mathbf{U}_t = \mathbb{I}_{N_t},$$

where $\mathbb{I}_{N_t}$ denotes the identity matrix of dimension $N_t \times N_t$.

3.7.2 Preconditioner application

The application of the preconditioner involves the solution of the linear system

$$\widehat{\mathbf{Q}}\mathbf{s} = \mathbf{r}, \tag{3.7.82}$$

where $\widehat{\mathbf{Q}}$ has the structure (3.7.81). We are able to efficiently solve system (3.7.82) by the FD method and the Sherman-Morrison formula. The starting points, that are involved in the setup of the preconditioner, are the following ones:

- for the pencils $(\widehat{\mathbf{L}}_l, \widehat{\mathbf{M}}_l)$ for $l = 1, \ldots, d$ we have the factorizations (3.3.23);
- for the pencil $(\widehat{\mathbf{L}}_t, \widehat{\mathbf{M}}_t)$ we have the factorization (3.6.75).

Then, define $\mathbf{U}_s := \mathbf{U}_d \otimes \cdots \otimes \mathbf{U}_1$ and $\mathbf{\Lambda}_s := \sum_{l=1}^d \mathbb{I}_{N_{s,d}} \otimes \cdots \otimes \mathbb{I}_{N_{s,l+1}} \otimes \mathbf{\Lambda}_l \otimes \mathbb{I}_{N_{s,l-1}} \otimes \cdots \otimes \mathbb{I}_{N_{s,1}}$. Notice that $\widehat{\mathbf{M}}_s^{-1} = \mathbf{U}_s\mathbf{U}_s^T$, therefore the matrix $\widehat{\mathbf{L}}_s\widehat{\mathbf{M}}_s^{-1}\widehat{\mathbf{L}}_s$ admits the stable factorization (3.4.48), that is

$$\mathbf{U}_s^T\widehat{\mathbf{L}}_s\widehat{\mathbf{M}}_s^{-1}\widehat{\mathbf{L}}_s\mathbf{U}_s = \mathbf{\Lambda}_s^2.$$

The preconditioner $\widehat{\mathbf{Q}}$ admits the following factorization

$$\begin{aligned}\widehat{\mathbf{Q}} = {} & \left(\mathbb{I}_{N_t} \otimes \mathbf{U}_s^T\right)^{-1} \\ & \left(\left(\mathbf{V}_t^T \otimes \mathbb{I}_{N_s}\right)^{-1}\left(\frac{\gamma^2}{T}\mathbf{\Lambda}_t \otimes \mathbb{I}_{N_s} + \nu^2 T\mathbb{I}_{N_t} \otimes \mathbf{\Lambda}_s^2\right)\left(\mathbf{V}_t \otimes \mathbb{I}_{N_s}\right)^{-1} + \gamma\nu\mathbf{R}_t \otimes \mathbf{\Lambda}_s\right) \\ & \left(\mathbb{I}_{N_t} \otimes \mathbf{U}_s\right)^{-1}.\end{aligned} \tag{3.7.83}$$

Note that the second factor in (3.7.83) is sum of two matrices, the second one being

$$\gamma\nu \mathbf{R}_t \otimes \mathbf{\Lambda}_s \tag{3.7.84}$$

where $\mathbf{R}_t$ is a rank 1 matrix that can be written as $\mathbf{R}_t = \boldsymbol{e}_{N_t}\boldsymbol{e}_{N_t}^T$, with $\boldsymbol{e}_{N_t}$ being the last element of the canonical basis of $\mathbb{R}^{N_t}$. Here it is possible to introduce a vector $\boldsymbol{v}_{N_t} \in \mathbb{R}^{N_t}$ such that

$$\boldsymbol{e}_{N_t}\boldsymbol{e}_{N_t}^T = \mathbf{V}_t^{-T}\boldsymbol{v}_{N_t}\boldsymbol{v}_{N_t}^T\mathbf{V}_t^{-1},$$

by defining $\boldsymbol{v}_{N_t} := \mathbf{V}_t^T\boldsymbol{e}_{N_t}$ that is the last column of $\mathbf{V}_t^T$. Thus equation (3.7.83) can be rewritten as

$$\widehat{\mathbf{Q}} = \left(\mathbf{V}_t^T \otimes \mathbf{U}_s^T\right)^{-1} (\mathbf{H} + \mathbf{K}) \left(\mathbf{V}_t \otimes \mathbf{U}_s\right)^{-1} \tag{3.7.85}$$

where we have introduced

$$\mathbf{H} := \frac{\gamma^2}{T}\mathbf{\Lambda}_t \otimes \mathbb{I}_{N_s} + \nu^2 T \mathbb{I}_{N_t} \otimes \mathbf{\Lambda}_s^2$$

for the full diagonal term, and

$$\mathbf{K} := \gamma\nu \boldsymbol{v}_{N_t}\boldsymbol{v}_{N_t}^T \otimes \mathbf{\Lambda}_s. \tag{3.7.86}$$

Introducing $\mathbf{S}_{N_t,N_s}$, the space and time shuffle matrix, such that $\mathbf{A} \otimes \mathbf{B} = \mathbf{S}_{N_t,N_s}(\mathbf{B} \otimes \mathbf{A})\mathbf{S}_{N_t,N_s}^T$ for all $\mathbf{A} \in \mathbb{R}^{N_t \times N_t}$ and $\mathbf{B} \in \mathbb{R}^{N_s \times N_s}$. Under reshuffling with $\mathbf{S}_{N_t,N_s}$, we have the following block diagonal structure

$$\mathbf{H} = \mathbf{S}_{N_t,N_s}\begin{bmatrix}\mathbf{H}_1 & & \\ & \ddots & \\ & & \mathbf{H}_{N_s}\end{bmatrix}\mathbf{S}_{N_t,N_s}^T, \quad \text{and} \quad \mathbf{K} = \mathbf{S}_{N_t,N_s}\begin{bmatrix}\mathbf{K}_1 & & \\ & \ddots & \\ & & \mathbf{K}_{N_s}\end{bmatrix}\mathbf{S}_{N_t,N_s}^T,$$

where, for $i = 1, \dots, N_s$, the matrices $\mathbf{H}_i$ are diagonal matrices defined as $\mathbf{H}_i := \frac{\gamma^2}{T}\mathbf{\Lambda}_t + \nu^2 T[\mathbf{\Lambda}_s^2]_{i,i} \otimes \mathbb{I}_{N_t}$, while $\mathbf{K}_i = \gamma\nu[\mathbf{\Lambda}_s]_{i,i} \otimes \boldsymbol{v}_{N_t}\boldsymbol{v}_{N_t}^T$.

In order to invert $\mathbf{H} + \mathbf{K}$, it is now sufficient to reshuffle the data, and invert the following independent N_s problems of size $N_t \times N_t$:

$$(\mathbf{H}_i + \mathbf{K}_i)\,\mathbf{x}_i = \mathbf{y}_i \quad \text{for } i = 1, \dots, N_s.$$

Notice that, each $\mathbf{K}_i$ is a rank 1 perturbation in the above systems, therefore the Sherman-Morrison formula gives

$$\mathbf{x}_i = \mathbf{H}_i^{-1}\mathbf{y}_i - \gamma\nu[\mathbf{\Lambda}_s]_{i,i}\frac{\mathbf{H}_i^{-1}\boldsymbol{v}_{N_t}\boldsymbol{v}_{N_t}^T\mathbf{H}_i^{-1}\mathbf{y}_i}{1 + \gamma\nu[\Lambda_s]_{i,i}\boldsymbol{v}_{N_t}^T\mathbf{H}_i^{-1}\boldsymbol{e}_{N_t}} \quad \text{for } i = 1, \dots, N_s. \tag{3.7.87}$$

Summarizing, the solution of (3.7.82) can be computed by Algorithm 2, with the proper notation for the involved matrices.

3.8 Computational cost and memory requirement

In this section we discuss the computational costs and memory requirements in the implementation of Algorithms 1,2 and 3. First, notice that the matrix $\mathbf{A}$ in (3.3.18) is neither positive definite nor symmetric and we choose GMRES as linear solver for the system (3.3.17). Whereas, the matrix $\mathbf{A}^T\mathbf{M}^{-1}\mathbf{A}$ in (3.4.43) and $\mathbf{B}$ in (3.5.67) are symmetric positive definite, therefore we choose Conjugate Gradients (CG) as linear solver for solving (3.4.42) and (3.5.66). Clearly, the computational cost of each iteration of the CG solver, or GMRES solver, depends on both the preconditioner setup and application cost, and on the residual computation cost.

We assume for simplicity that, for each univariate direction $l = 1, \dots, d$, the space matrices have dimension $n_s \times n_s$, while the time matrices involved in the preconditioners, have dimension $N_t \times N_t$. Thus the total number of degrees-of-freedom is $N_{dof} = N_s N_t = n_s^d N_t$.

We remark that the setup of the preconditioners has to be performed only once, since the matrices involved do not change during the iterative procedure.

3.8.1 Setup and application cost of Algorithm 1

The setup of $\widehat{\mathbf{A}}$ and $\widehat{\mathbf{A}}^{\boldsymbol{G}}$ includes the operations performed in Step 1 of Algorithm 1, i.e. d spatial eigendecompositions, that have a total cost of $O(dn_s^3)$ FLOPs, and the factorization of the time matrices. The computational cost of the latter, that is the sum of the cost of the eigendecomposition (3.3.31) and of the cost to compute the solution $\mathbf{v}$ of the linear system (3.3.33), yields a cost of $O(N_t^3)$ FLOPs. Then, the total cost of the spatial and time factorizations is $O(dn_s^3 + N_t^3)$ FLOPs. Note that, if $N_t = O(n_s)$, this cost is optimal for $d = 2$ and negligible for $d = 3$. The setup cost of $\widehat{\mathbf{A}}^{\boldsymbol{G}}$ includes also the the construction of the diagonal matrix $\mathbf{D}$, that has a negligible cost, and the computation of the $2d$ approximations $\varphi_1, \dots, \varphi_d$ and $\Phi_1, \dots, \Phi_d$ in (3.3.39), whose cost is negligible too, as mentioned in Section 3.3.7.

The application of the preconditioner is performed by Steps 2-4 of Algorithm 1. Exploiting (3.1.12), Step 2 and Step 4 costs $4(dn_s^{d+1}N_t + N_t^2 n_s^d) = 4N_{dof}(dn_s + N_t)$ FLOPs. The use of the block LU decomposition (3.3.38) makes the cost for Step 3 equal to $O(N_{dof})$ FLOPs.

In conclusion, the total cost of Algorithm 1 is $4N_{dof}(dn_s + N_t) + O(N_{dof})$ FLOPs. The non-optimal dominant cost of Step 2 and Step 4 is determined by the dense matrix-matrix products. However, these operations are usually implemented on modern computers in a very efficient way. For this reason, in our numerical tests, the overall serial computational time grows almost as $O(N_{dof})$, as reported in [75, Figure 5].

3.8.2 Setup and application cost of Algorithm 2

We apply Algorithm 2 both for the preconditioner $\widehat{\mathbf{P}}$ of Section 3.4 and for the preconditioner $\widehat{\mathbf{Q}}$ of Section 3.7. We discuss in detail the first strategy, since the second is analogous.

First notice that, for the projected least squares formulation of Section 3.4, we have to consider also the setup of the right hand side $\mathbf{g} = \mathbf{A}^T\mathbf{M}^{-1}\mathbf{f}$, which requires to invert the mass matrix $\mathbf{M} = \mathbf{M}_t \otimes \mathbf{M}_s$. Here we follow the preconditioned iterative technique proposed in [76], whose computational cost of each iteration for inverting the mass is $O(pN_{dof})$ FLOPs, assuming $p = p_s \approx p_t$. We recall no inversion of the mass is need for the strategy presented in 3.7.

The setup of $\widehat{\mathbf{P}}$ includes the operations performed in Step 1 of Algorithm 2, i.e. d spatial eigendecompositions, that have a total cost of $O(dn_s^3)$ FLOPs, and one time eigendecomposition, that have a total cost of $O(N_t^3)$ FLOPs. Then, the total cost of the spatial and time factorizations is $O(dn_s^3 + N_t^3)$ FLOPs. Again, if $N_t = O(n_s)$, this cost is optimal for $d = 2$ and negligible for $d = 3$.

The application of the preconditioner is performed by Steps 2-6 of Algorithm 2. Exploiting again (3.1.12), Step 2 and Step 6 costs $4N_{dof}(dn_s + N_t)$ FLOPs. The cost of the reshuffling in Step 3 and Step 5 is negligible. Step 4 requires to solve N_s independent problems of size $N_t \times N_t$. The Sherman-Morrison formula together with the time decomposition, allow to compute each solution by first inverting the diagonal matrices $\mathbf{H}_i$ for $i = 1, \ldots, N_s$, and then computing the correction term in (3.4.52). Both of these steps cost $O(N_tN_s) = O(N_{dof})$ FLOPs. In conclusion, the total cost of Algorithm 2 is $4N_{dof}O(dn_s + N_t) + O(N_{dof})$.

3.8.3 Setup and application cost of Algorithm 3

The setup of the preconditioners $\widehat{\mathbf{B}}$ and $\widehat{\mathbf{B}}^{\boldsymbol{G}}$ include again the eigendecomposition of the pencils, that is Step 1 of Algorithm 3. Therefore, the cost of the factorizations is $O(dn_s^3 + N_t^3)$ FLOPs. We remark this cost is optimal for $d = 2$ and negligible for $d = 3$, provided that $N_t \approx n_s$. The setup cost of $\widehat{\mathbf{B}}^{\boldsymbol{G}}$ includes also the the construction of the diagonal matrix $\mathbf{D}$, that has a negligible cost, and the computation of the $2(d + 1)$ univariate approximations $\mu_1, \ldots, \mu_{d+1}$ and $\omega_1, \ldots, \omega_{d+1}$, that are used to incorporate the information of the geometry into the preconditioner. As mentioned above, this cost is negligible.

The application of the preconditioner is performed by Steps 2-4 of Algorithm (3). Exploiting (3.1.12), Step 2 and Step 4 costs $4N_{dof}(dn_s + N_t)$ FLOPs, while Step 3 has an optimal cost, as it requires $O(N_{dof})$ FLOPs. Thus, the total cost of Algorithm 3 is $4N_{dof}(dn_s + N_t) + O(N_{dof})$ FLOPs. We remark that the non-optimal dominant cost is given by the dense matrix-matrix products of Step 2 and Step 4, which, however, are usually implemented on modern computers in a high efficient way, as they are BLAS level 3 operations.

3.8.4 Computational cost of the residuals

The other dominant computational cost in a CG or GMRES iteration is the cost of the residual computation. In Algorithm 1, this involves the multiplication of the matrix $\mathbf{A}$ with a vector. This multiplication is done by exploiting the special structure (3.3.18), that allows a matrix-free approach and the use of formula (3.1.12). As mentioned above, the computational cost of a single matrix-vector product is $O(N_{dof}p^d)$ FLOPs, if we assume $p = p_s \approx p_t$.

In Algorithm 2, the multiplication by $\mathbf{A}^T\mathbf{M}^{-1}\mathbf{A}$ is done by exploiting the kronecker structure in (3.3.18) together with the above mentioned iterative technique for the inversion of the mass matrix. Notice that, this allows a matrix free approach and uses the formula (3.1.12). In particular we do not need to compute and to store the whole matrix $\mathbf{A}^T\mathbf{M}^{-1}\mathbf{A}$, but only the time and spatial factors of the matrix $\mathbf{A}$ are enough. The time matrices $\mathbf{M}_t$ and $\mathbf{W}_t$ are banded with a band of width $2p_t + 1$ and the spatial matrices $\mathbf{L}_s$ and $\mathbf{M}_s$ have roughly $N_s(2p_s + 1)^d$ nonzero entries. Assuming $p = p_s \approx p_t$, the computational cost of a single matrix-vector product is given by:

- the multiplication by $\mathbf{A}$, whose cost in FLOPs is
$$O\left((2p_s+1)^d N_s N_t + (2p_t+1)N_t N_s\right) = O(p^d N_{dof})$$
- the inversion of the mass $\mathbf{M}$, which costs $O(pN_{dof})$ FLOPs;
- the multiplication by $\mathbf{A}^T$, that costs again $O(p^d N_{dof})$ FLOPs.

The overall cost is thus $O(p^d N_{dof})$ FLOPs. Notice that the number of iterations required in the CG solver for inverting the mass $\mathbf{M}_s$ over 3D objects, with a tolerance of 10^{-8}, may be around 5, see [76, Tables 1]. This factor, together with the possible higher number of iterations required by the iterative strategy preconditioned by $\widehat{\mathbf{P}}$, may cause lack of performance for this preconditioning strategy w.r.t the previous one.

In Algorithm 3, the residual computation consists in the multiplication between $\mathbf{B}$ and a vector. This multiplication can be computed by exploiting the special structure (3.5.67) and the formula (3.1.12). Again, we do not need to compute and store the whole matrix $\mathbf{B}$ but only its factors $\mathbf{L}_t, \mathbf{M}_t, \mathbf{R}_t, \mathbf{J}_s, \mathbf{L}_s$and $\mathbf{M}_s$. With this matrix-free approach, noting again that the time matrices $\mathbf{L}_t, \mathbf{M}_t, \mathbf{R}_t$ are banded matrices with a band of width $2p_t + 1$ and the spatial matrices $\mathbf{J}_s, \mathbf{L}_s, \mathbf{M}_s$ have a number of non-zeros per row approximately equal to $(2p_s + 1)^d$, the computational cost of a single matrix-vector product is $O(N_{dof}p^d)$ FLOPs, if we assume $p = p_s \approx p_t$.

The numerical experiments reported in [75, Table 5] and [85, Table 3] show that the dominant cost in the iterative solver is represented by the residual computation. This is a typical behavior of the FD-based preconditioning strategies, see [75, 85, 87, 93].

3.8.5 Memory requirements

We now investigate the memory consumption of the preconditioning strategies proposed, giving the details for the preconditioner $\widehat{\mathbf{A}}$, since the other cases are analogous. For the preconditioner, we have to store the eigenvector spatial matrices, $\mathbf{U}_1, \dots, \mathbf{U}_d$, the time matrix $\mathbf{U}_t$ and the block-arrowhead matrix (3.3.37). The memory required is roughly
$$N_t^2 + dn_s^2 + 2N_{dof}.$$
This extends analogously to the preconditioners proposed in Sections 3.4, 3.6 and 3.7.

For the system matrix $\mathbf{A}$, we have to store the time factors $\mathbf{M}_t$ and $\mathbf{W}_t$ and the spatial factors $\mathbf{M}_s$ and $\mathbf{L}_s$. Thus the memory required is roughly
$$2(2p_t+1)N_t + 2(2p_s+1)^d N_s \approx 4p_t N_t + 2^{d+1} p_s^d N_s.$$

The projected least squares approach of Section 3.4, further requires the application of $\mathbf{M}^{-1}$. Using the approach of [76], requires a further spatial eigendecomposition of the mass matrix $\mathbf{M}$, thus the memory required is again $N_t^2 + dn_s^2 + 2N_{dof}$, plus a diagonal scaling $\mathbf{D}$ whose memory consumption is N_{dof}.

These numbers show that memory-wise our space-time strategies are very appealing when compared to other approaches, even when space and time variables are discretized separately, e.g., with finite differences in time or other time-stepping schemes. For example if we assume $d = 3$, $p_t \approx p_s = p$ and $N_t^2 \leq Cp^3 N_s$, then the total memory consumption is $O(p^3 N_s + N_{dof})$, that is equal to the sum of the memory needed to store the Galerkin matrices associated to spatial variables and the memory needed to store the solution of the problem.

3.9 Numerical Results

In this section we present the numerical experiments that assess the performance of the preconditioners. As regards the orders of convergence of the discretizations of Section 3.2 and Section 3.5, we refer respectively to [75] and [85].

We consider only sequential executions and we force the use of a single computational thread in a Intel Core i7-5820K processor, running at 3.30 GHz and with 64 GB of RAM.

The tests are performed with Matlab R2023a and GeoPDEs toolbox [107]. We use the `eig` Matlab function to compute the generalized eigendecompositions present in Step 1 of Algorithms 1,2 and 3 , while Tensorlab toolbox [98] is employed to perform the multiplications with Kronecker matrices. The solution of the linear system (3.3.33) is performed by Matlab direct solver (backslash operator "\"). The linear system (3.3.17) is solved by GMRES (Matlab routine `gmres`), while the linear systems 3.4.42 and 3.5.66 are solved with CG (Matlab routine `pcg`). The tolerance in the iterative solvers is set equal to 10^{-8} and the null vector is the initial guess in all tests. We remark that GMRES computes and stores a full orthonormal basis for the Krylov space, and this might be unfeasible if the number of iterations is too large. This issue could be addressed by switching to a different solver for nonsymmetric systems, like e.g. BiCGStab, or using the restarted version of GMRES.

We use the same mesh-size in space and in time $h_s = h_t =: h$, and use splines of maximal continuity and same degree in space and in time $p_t = p_s =: p$. For the sake of simplicity, we also consider uniform knot vectors, and denote the number of elements in each parametric direction by $n_{el} := \frac{1}{h}$.

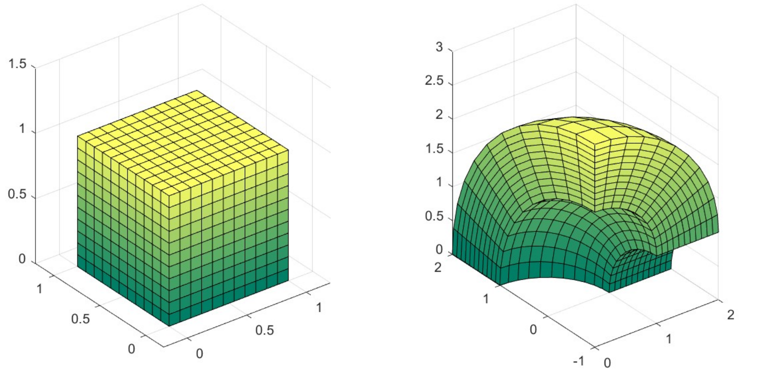
To assess the performance of our preconditioning strategies, we set $T = 1$ and we focus on two 3D spatial domains $\Omega \subset \mathbb{R}^3$, represented in Figure 3.7a and Figure 3.7b: the cube and the rotated quarter of annulus, respectively.

In out tables, the symbol "$*$" denotes that the construction of the matrix factors of $\mathbf{A}$ (see (3.3.18)) goes out of memory, while the symbol "$**$" indicates that the dimension of the Krylov subspace is too high and there is not enough memory to store all the GMRES iterates. We remark that in all the tables the total solving time of the iterative strategies includes also the setup time of the considered preconditioner.

3.9.1 Performance of the preconditioners: cube domain

In the cube domain, we set homogeneous Dirichlet and zero initial boundary conditions and we fix f such that the exact solution is $u = \sin(\pi x)\sin(\pi y)\sin(\pi z)\sin(t)$. Clearly, in this computational domain, the preconditioners $\widehat{\mathbf{A}}$ and $\widehat{\mathbf{P}}$ are direct solvers. Moreover, since we are solving on the parametric domain, we have $\widehat{\mathbf{A}}^{\boldsymbol{G}} = \widehat{\mathbf{A}}$ and $\widehat{\mathbf{B}}^{\boldsymbol{G}} = \widehat{\mathbf{B}}$.

In view of this observations, we analyze the performance of $\widehat{\mathbf{B}}$ and $\widehat{\mathbf{Q}}$. As a comparison we also consider as preconditioner for CG the Incomplete Cholesky with



(a) Cube. (b) Rotated quarter of annulus.

Figure 3.7. Computational domains.

zero fill-in (IC(0)) factorization of $\mathbf{B}$, that is executed with Matlab routine `ichol`. Table 3.7 report the number of iterations and the total solving time, that includes the setup time of the preconditioner. The results for $\widehat{\mathbf{B}}$ (upper table) and IC(0) (lower table) are taken from [85, Table 1]. The matrix-vector products of CG are computed in a matrix-free way using its time and spacial factors. Matrix $\mathbf{B}$ is assembled when we want to use the IC(0) preconditioner. In any case, the assembly times are never included in the reported times. The number of iterations obtained with $\widehat{\mathbf{B}}$ and $\widehat{\mathbf{Q}}$ are stable with respect to the polynomial degree $p := p_t = p_s$. The performance of $\widehat{\mathbf{B}}$ is stable also with respect to the number of elements n_{el}, while $\widehat{\mathbf{Q}}$ seems to converge to a direct solver. Even in the case when the number of iterations of $\widehat{\mathbf{B}}$ might be larger than that of IC(0), the overall computational time is significantly lower, up to two orders of magnitude for the problems considered. This is due to the higher setup and application cost of the IC(0) preconditioner.

3.9.2 Performance of the preconditioners: rotated quarter of annulus

The second computational domain Ω is a quarter of annulus with center in the origin, internal radius 1 and external radius 2, rotated by $\pi/2$ along the axis $y = -1$. Boundary data and forcing function are set such that the exact solution is $u = -(x^2+y^2-1)(x^2+y^2-4)xy^2 \sin(z)\sin(t)$.

We analyze the performance of $\widehat{\mathbf{A}}$, $\widehat{\mathbf{A}}^{\boldsymbol{G}}$, $\widehat{\mathbf{P}}$, $\widehat{\mathbf{B}}$, $\widehat{\mathbf{B}}^{\boldsymbol{G}}$ and $\widehat{\mathbf{Q}}$. In the GMRES solver, the maximum dimension of the Krylov subspace is set equal to 100 for both the preconditioners $\widehat{\mathbf{A}}$ and $\widehat{\mathbf{A}}^{\boldsymbol{G}}$, up to $n_{el} = 64$. We are able to reach convergence and to perform the tests with $\widehat{\mathbf{A}}^{\boldsymbol{G}}$, $n_{el} = 128$ and $p = 1, 2, 3$ by setting the maximum Krylov subspace dimension equal to 25.

In Table 3.8 we first report the number of iterations and the total solving time of GMRES preconditioned with $\widehat{\mathbf{A}}$ (upper table) and $\widehat{\mathbf{A}}^{\boldsymbol{G}}$ (middle table), taken from [75, Table 4]. The non-trivial geometry clearly affects the performance of $\widehat{\mathbf{A}}$, but, when we include some information on the parametrization by using $\widehat{\mathbf{A}}^{\boldsymbol{G}}$, the number of iterations is more than halved and it is stable w.r.t. p and n_{el}. Moreover, the computational times are one order of magnitude lower for the highest degrees p and

	$\widehat{\mathbf{B}}$ + CG Iterations / Time			
n_{el}	$p = 2$	$p = 3$	$p = 4$	$p = 5$
8	9 / 0.06	11 / 0.07	11 / 0.18	11 / 0.28
16	11 / 0.27	11 / 0.69	12 / 1.80	12 / 3.80
32	12 / 5.10	12 / 13.17	12 / 27.31	12 / 52.95
64	13 / 100.09	13 / 227.93	13 / 458.86	13 / 924.44
128	13 / 2012.94	13 / 4235.96	$*$	$*$

	$\widehat{\mathbf{Q}}$ + CG Iterations / Time			
n_{el}	$p = 2$	$p = 3$	$p = 4$	$p = 5$
8	3 / 0.03	5 / 0.08	5 / 0.19	4 / 0.26
16	3 / 0.13	5 / 0.78	5 / 2.18	3 / 2.61
32	2 / 1.79	4 / 7.17	3 / 12.70	2 / 19.60
64	2 / 32.36	4 / 113.51	3 / 186.92	2 / 356.48
128	2 / 468.76	3 / 1639.75	$*$	$*$

	IC(0) + CG Iterations / Time			
n_{el}	$p = 2$	$p = 3$	$p = 4$	$p = 5$
8	9 / 0.18	7 / 1.69	6 / 14.04	6 / 80.39
16	22 / 5.01	16 / 45.54	12 / 355.99	10 / 1913.90
32	64 / 157.05	$*$	$*$	$*$

Table 3.7. Cube domain. Performance of $\widehat{\mathbf{B}}$, $\widehat{\mathbf{Q}}$ and IC(0).

	$\widehat{\mathbf{A}}$ Iterations / Time				
n_{el}	$p = 1$	$p = 2$	$p = 3$	$p = 4$	$p = 5$
8	34 / 0.20	37 / 0.21	42 / 0.42	46 / 0.63	50 / 1.13
16	43 / 1.15	46 / 1.65	50 / 3.42	54 / 5.80	57 / 11.87
32	50 / 22.75	53 / 31.10	57 / 54.02	61 / 96.06	64 / 184.84
64	57 / 586.73	60 / 764.26	67 / 1254.81	67 / 1858.55	71 / 3188.51
128	$**$	$**$	$**$	$*$	$*$

	$\widehat{\mathbf{A}}^{\boldsymbol{G}}$ Iterations / Time				
n_{el}	$p = 1$	$p = 2$	$p = 3$	$p = 4$	$p = 5$
8	11 / 0.06	12 / 0.09	12 / 0.11	13 / 0.18	14 / 0.29
16	13 / 0.26	14 / 0.52	14 / 1.18	14 / 1.44	15 / 3.85
32	15 / 4.73	15 / 6.76	15 / 12.67	15 / 21.47	16 / 40.54
64	16 / 107.24	16 / 135.74	18 / 249.27	16 / 370.31	17 / 695.44
128	17 / 2623.57	17 / 3105.76	17 / 5614.10	$*$	$*$

	$\widehat{\mathbf{P}}$ Iterations / Time				
n_{el}	$p = 1$	$p = 2$	$p = 3$	$p = 4$	$p = 5$
8	93 / 1.90	95 / 3.15	98 / 2.84	104 / 6.28	107 / 19.14
16	119 / 5.60	118 / 16.24	117 / 38.84	119 / 112.74	117 / 232.96
32	129 / 178.08	127 / 285.19	125 / 645.68	125 / 1711.45	124 / 4038.55
64	133 / 1607.33	130 / 3834.92	129 / 8981.30	144 / 19868.87	165 / 43930.63
128	136 / 29764.56	133 / 74537.02	131 /157773.58	$*$	$*$

Table 3.8. Rotated quarter domain. Performance of $\widehat{\mathbf{A}}$, $\widehat{\mathbf{A}}^{\boldsymbol{G}}$ and $\widehat{\mathbf{P}}$.

numbers of elements n_{el}.

Next, in the lower table of Table 3.8 we report the results for the preconditioner $\widehat{\mathbf{P}}$ obtained by solving the projected least squares problem (3.4.42). Recall in this case the iterative solver is the preconditioned conjugate gradient method, with tolerance 10^{-8} and initial guess the null vector. In this case the number of iterations with respect to the preconditioner $\widehat{\mathbf{A}}$ is more than doubled, although they are stable with respect to the degrees and number of elements, we suggest to use $\widehat{\mathbf{P}}$ as a last resort.

In Table 3.9, we report the results obtained in [85, Table 2], for $\widehat{\mathbf{B}}$ (top section of the table) and $\widehat{\mathbf{B}}^{\boldsymbol{G}}$ (middle section of the table), applied to the same problem in the least squares formulation framework. For the preconditioner $\widehat{\mathbf{B}}$, the numbers of iterations have more than doubled, while still remaining stable with respect to degrees and numbers of elements. Finally, in the lower section of Table 3.9 we present the results obtained with the preconditioner $\widehat{\mathbf{Q}}$, which is stable w.r.t. the

	$\widehat{\mathbf{B}}$ + CG Iterations / Time			
n_{el}	$p = 2$	$p = 3$	$p = 4$	$p = 5$
8	107 / 0.21	107 / 0.48	114 / 1.17	123 / 2.73
16	126 / 2.56	128 / 6.90	133 / 17.04	135 / 35.17
32	142 / 52.77	143 / 132.24	148 / 292.53	151 / 572.84
64	153 / 1056.21	155 / 2415.23	156 / 4956.68	159 / 9906.33
128	164 / 22106.01	166 / 47539.02	$*$	$*$

	$\widehat{\mathbf{B}}^{\boldsymbol{G}}$ + CG Iterations / Time			
n_{el}	$p = 2$	$p = 3$	$p = 4$	$p = 5$
8	24 / 0.09	24 / 0.13	26 / 0.37	26 / 0.60
16	35 / 0.77	34 / 1.96	33 / 4.62	33 / 9.35
32	42 / 17.03	41 / 39.57	40 / 82.35	41 / 161.73
64	46 / 333.20	44 / 716.03	49 / 1577.55	53 / 3384.08
128	48 / 6767.08	50 / 14814.09	$*$	$*$

	$\widehat{\mathbf{Q}}$ + CG Iterations / Time			
n_{el}	$p = 2$	$p = 3$	$p = 4$	$p = 5$
8	101 / 0.35	111 / 1.27	136 / 3.02	172 / 6.78
16	132 / 3.08	140 / 14.98	161 / 41.14	175 / 92.71
32	139 / 72.11	151 / 190.92	169 / 444.41	186 / 938.47
64	142 / 1424.39	156 / 3197.97	172 / 6781.49	189 / 14399.77
128	145 / 19993.49	154 / 54146.41	$*$	$*$

Table 3.9. Rotated quarter domain. Performance of $\widehat{\mathbf{B}}$, $\widehat{\mathbf{B}}^{\boldsymbol{G}}$ and $\widehat{\mathbf{Q}}$.

degrees and numbers of elements, and behaves like $\widehat{\mathbf{B}}$, suggesting that including the information of the geometry may boost its performance.

3.10 Conclusions

In this chapter we proposed a review of several preconditioners suited for space-time Galerkin isogeometric discretizations of the heat equation. Our preconditioners are represented by a suitable sum of Kronecker products of matrices, that makes the computational cost of their construction (setup) and application, as well as the storage cost, very appealing. In particular, inspired by the FD technique, the application of the preconditioner $\widehat{\mathbf{A}}$ exploits an ad-hoc factorization of the time matrices. The preconditioners $\widehat{\mathbf{P}}$ and $\widehat{\mathbf{Q}}$ factorize in time direction as sum of diagonal blocks and rank-1 block perturbations, while in space have a diagonal structure.

Lastly, the preconditioner $\widehat{\mathbf{B}}$ admits a full diagonal factorization.

The application cost of all the preconditioners is almost equal to $O(N_{dof})$ and does not depend on the polynomial degree.

At the same time, the storage cost is roughly the same that we would have by discretizing separately in space and in time, if we assume $N_t \leq Cp^d N_s$. Indeed, in this case the memory used for the whole iterative solver is $O(p^d N_s + N_{dof})$.

In this review, we have compared the performance of the four preconditioners $\widehat{\mathbf{A}}, \widehat{\mathbf{B}}, \widehat{\mathbf{P}}$ and $\widehat{\mathbf{Q}}$, first on the parametric 3D cube spacial domain, and then on a rotated quarter of ring. In the first computational domain $\widehat{\mathbf{A}}$ and $\widehat{\mathbf{P}}$ are direct solvers and $\widehat{\mathbf{Q}}$ seem to converge, for finer meshes and higher degrees, to a direct solver. The preconditioner $\widehat{\mathbf{B}}$ is although stable under mesh refinement and degree elevation.

On the rotated quarter of ring geometry, the performances of $\widehat{\mathbf{A}}^{\boldsymbol{G}}$ and $\widehat{\mathbf{B}}^{\boldsymbol{G}}$, that are the versions of $\widehat{\mathbf{A}}$ and $\widehat{\mathbf{B}}$ with a partial inclusion of the geometry's information, are outstanding when compared to their parametric versions. Thus the inclusion of information for the preconditioners $\widehat{\mathbf{P}}$ and $\widehat{\mathbf{Q}}$ seems to be a natural way forward as a future development of this work.

As a final comment, we mention that our methods has a strong potential for parallelization, and this will be an interesting future direction of study.

Chapter 4

Schrödinger equation

Space-time finite element methods originated in the papers [47, 22, 88, 89], where standard finite elements are ascribed an extra dimension for the time and, typically, adopt a discontinuous approximation in time, since this produces a time marching algorithm with a traditional step-by-step format (see e.g. [97]). Over the years, the theory of space-time methods has been developed mainly for evolutionary equations of the parabolic type and hyperbolic type, whereas, for quantum mechanics, and more precisely for Schrödinger's equation, there are few contributions and the methods are still in development.

To our knowledge, one of the first works concerning space-time variational formulations for the (nonlinear) Schrödinger equation, is [64], in which Karakashian and Makridakis proposed a space–time method combining a conforming Galerkin discretization in space and an upwind DG time-stepping. This method reduces to a Radau IIA Runge-Kutta time discretization in the case of constant potentials. In [36], for the linear Schrödinger equation the authors propose two variational formulations that are proved to be well posed: a strong formulation, with no relaxation of the original equation, and an ultraweak formulation, that transfers all derivatives onto test functions. The proposed discretization for the ultraweak form is based on a discontinuous Petrov-Galerkin (DPG) method, that addresses optimal stability, and quasi-optimal error rates in L^2-norm. In [56] a space–time ultraweak Trefftz discontinuous Galerkin (DG) method for the Schrödinger equation with piecewise-constant potential is proposed and analyzed, proving well-posedness and stability of the method, and optimal high-order h-convergence error estimates in a skeleton norm, for the one and two dimensional cases. Recently, in [57], Hain and Urban proposed a space–time ultraweak variational formulation with optimal inf-sup constant. The formulation in [57] is related to the ultraweak DPG method in [36], but differs in the choice of the test and trial spaces. Hain and Urban first fix a conforming test space, and then construct an optimal trial space, while Demkowicz et al. first constructs a trial space and then a suitable test space. The discretization proposed in [57] uses high order B-splines with maximum regularity and can be extended to the Isogeometric Analysis (IgA) framework.

Introduced in [63], see also the book [29], IgA, is an evolution of the classical finite element methods. In IgA, both the approximation of the solution of the partial differential equation that models the problem, and the representation of the computational domain, are accomplished using B-spline functions, or their generalizations (NURBS). This is meant to simplify the interoperability between computer aided design and numerical simulations. IgA also benefits from the approximation properties of splines, whose high-continuity yields higher accuracy when compared to C^0 piecewise polynomials, see e.g., [41, 20, 94].

In this chapter we focus on the linear time dependent Schrödinger equation without potential. Starting from the well posed space-time strong formulations in [36], we derive a well posed space-time isogeometric Petrov-Galerkin discretization, that is essentially a Galerkin approximation of the space-time least squares variational formulation of the model problem. The matrix associated to the discrete linear system can be written as sum of Kronecker products, and has the same structure of the one arising from [57]. The main contribution of this chapter, is the development of a stable preconditioner that leads to a fast solver for the problem modeled in the parametric domain. As it was done in [85, 75] for parabolic problems, our preconditioner exploits the Kronecker structure of the linear system, and makes use of Fast Diagonalization method (FD) [79]. In this work, FD is applied among the space direction only. Although, the computational cost of the setup of the resulting preconditioner is $O(N_{dof})$ FLoating-Point Operations (FLOPs), while its application is $O(N_{dof}^{(1+2)/(d+1)})$ FLOPs, where d is the number of spatial dimensions and N_{dof} denotes the total number of degrees-of-freedom (assuming, for simplicity, to have the same number of degrees-of-freedom in time and in each spatial direction). We remark that global space-time methods, in principle, facilitate the full parallelization of the solver, see [38, 49, 70].

The outline of the chapter is as follows. The model problem is introduced in Section 4.1. In Section 4.2 we present the basics of B-splines based IgA and the best approximation properties. The isogeometric least squares discretization is introduced in Section 4.3 and compared to the ultraweak form of [57], while in Section 4.4 we define the preconditioner for the parametric domain and we discuss its application. We present the numerical results assessing the performance of the proposed preconditioner in Section 4.5. Finally, in the last section we draw some conclusions and we highlight some future research directions.

4.1 Model problem

We consider a bounded domain $\Omega \subset \mathbb{R}^d$, usually $d = 1, 2, 3$, with Lipschitz boundary, and a time interval $(0, T)$, where $T > 0$ is the final time. The space-time domain is denoted by $\mathcal{Q} := (0, T) \times \Omega$. Assuming Dirichlet boundary conditions, denote by $\Gamma_D := (0, T) \times \partial\Omega$ the Dirichlet boundary of the space-time cylinder $\mathcal{Q}$, while $\mathcal{Q}_0 = \{0\} \times \Omega$ is the initial side. Our model problem is the Schrödinger equation with homogeneous boundary and initial conditions: we look for a solution u such that

$$\left\{\begin{array}{rcll} \mathrm{i}\partial_t u - \nu\Delta u &=& f & \text{in} \quad \mathcal{Q}, \\ u &=& 0 & \text{on} \quad \Gamma_D, \\ u &=& 0 & \text{in} \quad \mathcal{Q}_0, \end{array}\right. \tag{4.1.1}$$

where i is the imaginary unit and $\nu > 0$ is a constant coefficient usually depending on Planck's constant $\hbar$ and the mass of the modeled physical particle. We assume that $f \in L^2(\mathcal{Q})$ and denote by $\mathbb{S} := \mathrm{i}\partial_t - \nu\Delta$ the Schrödinger operator, $\mathbb{S}^*$ its adjoint operator, and $(\cdot, \cdot)$ the complex scalar product in $L^2(\mathcal{Q})$.

The previous setting can be generalized to non-homogeneous initial and boundary conditions. For example, suppose that in (4.1.1) we have the initial condition $u = u_0 \in H_0^1(\Omega)$ in $\mathcal{Q}_0$. Then, we consider a lifting $\underline{u}_0 \in H^1\left((0,T); H_0^1(\Omega)\right)$ of u_0 such that the solution u can be split as $u = \underline{u} + \underline{u}_0$, where $\underline{u} \in \mathcal{V}$ is the solution of the following Schrödinger equation with homogeneous initial and boundary conditions:

$$\begin{cases} \mathrm{i}\partial_t \underline{u} - \nu\Delta\underline{u} &= \underline{f} \quad \text{in} \quad \mathcal{Q}, \\ \underline{u} &= 0 \quad \text{on} \quad \Gamma_D, \\ \underline{u} &= 0 \quad \text{in} \quad \mathcal{Q}_0, \end{cases}$$

where $\underline{f} := f - \mathbb{S}\underline{u}_0$.

4.1.1 Space-time variational formulation

Let us introduce the Hilbert spaces

$$\mathcal{V} := \left\{ v \in L^2(\mathcal{Q}) : \mathbb{S}v \in L^2(\mathcal{Q}) \text{ and } (\mathbb{S}^* w, v) - (w, \mathbb{S}v) = 0 \ \forall w \in \mathcal{C}_0^\infty(\mathbb{R}^{d+1}) : w|_{\Gamma_D \cup (\{T\}\times\Omega)} = 0 \right\},$$

$$\mathcal{W} := L^2(\mathcal{Q}),$$

endowed with the following norms

$$\|v\|_{\mathcal{V}}^2 := \|v\|_{L^2(\mathcal{Q})}^2 + \|\mathbb{S}v\|_{L^2(\mathcal{Q})}^2 \quad \text{and} \quad \|w\|_{\mathcal{W}} := \|w\|_{L^2(\mathcal{Q})},$$

respectively. Then, the space-time variational formulation of (4.1.1) reads:

$$\text{Find } u \in \mathcal{V} \text{ such that } \mathcal{A}(u,v) = \mathcal{F}(v) \quad \forall v \in \mathcal{W}, \tag{4.1.2}$$

where the sesquilinear form $\mathcal{A}(\cdot,\cdot)$ and the linear form $\mathcal{F}(\cdot)$ are defined $\forall v \in \mathcal{V}$ and $\forall w \in \mathcal{W}$ as

$$\mathcal{A}(v,w) := \int_\Omega \int_0^T (\mathbb{S}v)\,\overline{w}\,\mathrm{dt}\ \mathrm{d}\Omega \quad \text{and} \quad \mathcal{F}(w) := \int_\Omega \int_0^T f\,\overline{w}\,\mathrm{dt}\ \mathrm{d}\Omega.$$

The well-posedness of Problem (4.1.2) can be reduced to the density of smooth functions in $\mathcal{V}$ and in another Hilbert space. This depends on the domain and it holds when $\mathcal{Q}$ is smoothly diffeomorphic to an hypercube. The details are in Appendix B and the main result is the following.

Theorem 4.1. *Under Assumption B.1, the linear Schrödinger operator* $\mathbb{S} : \mathcal{V} \to L^2(\mathcal{Q})$ *is a continuous bijections, that is problem* (4.1.2) *is well posed.*

4.2 Isogeometric framework and preliminaries

With the same notations introduced in Section 3.1, let n and p be two positive integers, and let Ξ be an open knot vector in $[0,1]$. Denote by $Z = \{\zeta_1, \ldots, \zeta_r\}$ the vector of breakpoints, that is the vector of knots without repetition. The univariate spline space is $\widehat{\mathcal{S}}_h^p := \operatorname{span}\{\widehat{b}_{i,p}\}_{i=1}^m$, where $\widehat{b}_{i,p} : (0,1) \to \mathbb{R}$ are the univariate B-spline basis functions, and $h := \max\{|\xi_{i+1} - \xi_i|,\ i = 1, \ldots, n+p\}$.

Multivariate B-splines are defined as tensor product of univariate B-splines. Thus, we introduce $d+1$ univariate knot vectors Ξ_l and Ξ_t, with associated breakpoints Z_l and Z_t, for $l = 1, \ldots, d$. Let $h_s := \max\{h_l \mid l = 1, \ldots, d\}$ with h_l the mesh-size in direction l, and denote by h_t the mesh-size in time direction. Assume that the following local quasi-uniformity of the knot vectors holds.

Assumption 4.1. *There exists $\theta \geq 1$, independent of h_s and h_t, such that $\theta^{-1} \leq \zeta_{l,i}/\zeta_{l,i+1} \leq \theta$ for $i = 1, \ldots, r_l$, $l = 1, \ldots, d$ and $\theta^{-1} \leq \zeta_{t,i}/\zeta_{t,i+1} \leq \theta$ for $i = 1, \ldots, r_t$.*

Given $\boldsymbol{p} := (p_t, \boldsymbol{p}_s)$, where $\boldsymbol{p}_s := (p_1, \ldots, p_d)$, the multivariate spline space is $\widehat{\mathcal{S}}_h^{\boldsymbol{p}} = \widehat{\mathcal{S}}_{h_t}^{p_t} \otimes \widehat{\mathcal{S}}_{h_s}^{\boldsymbol{p}_s}$ with $\widehat{\mathcal{S}}_{h_s}^{\boldsymbol{p}_s}$ as in 3.1.5. From now on we consider $p_1 = \cdots = p_d =: p_s$, but the general case is similar. Finally, let us make the following regularity assumption.

Assumption 4.2. *We assume that $p_t \geq 1$, $p_s \geq 2$ and that $\widehat{\mathcal{S}}_{h_t}^{p_t} \subset C^0\left((0,1)\right)$ and $\widehat{\mathcal{S}}_{h_s}^{\boldsymbol{p}_s} \subset C^1(\widehat{\Omega})$.*

4.2.1 Isogeometric spaces

The space-time computational domain that we consider is $(0, T) \times \Omega$, where $T > 0$ is the final time and $\Omega \subset \mathbb{R}^d$ is the space domain. The choice of considering the time as first variable will be clarified in Section 4.4.4. The following assumptions asserts the regularity of the parametrization.

Assumption 4.3. *We assume that Ω is parametrized by a smooth diffeomorphism $\boldsymbol{F} : \widehat{\Omega} \to \Omega$.*

Denote by $\boldsymbol{x} = (x_1, \ldots, x_d) := \boldsymbol{F}(\boldsymbol{\eta})$ and $t := T\tau$. Then the space-time domain is given by the parametrization $\boldsymbol{G} : (0,1) \times \widehat{\Omega} \to (0,T) \times \Omega$, such that $\boldsymbol{G}(\tau, \boldsymbol{\eta}) := (T\tau, \boldsymbol{F}(\boldsymbol{\eta})) = (t, \boldsymbol{x})$.

We denote by $\widehat{\mathcal{X}}_{h,0}$ the spline space with initial and boundary conditions, in parametric coordinates, defined in 3.1.6. Analogously, the spline space with homogeneous Dirichlet final and boundary conditions, in parametric coordinates, is

$$\widehat{\mathcal{X}}_{h,T} := \left\{ \widehat{v}_h \in \widehat{\mathcal{S}}_h^{\boldsymbol{p}} \;\middle|\; \widehat{v}_h = 0 \text{ on } \{T\} \times \widehat{\Omega} \text{ and } \widehat{v}_h = 0 \text{ on } (0,1) \times \partial\widehat{\Omega} \right\}. \tag{4.2.3}$$

Recall that, by reordering the basis functions, it holds

$$\widehat{\mathcal{X}}_{h,0} = \operatorname{span}\left\{ \widehat{B}_{i,\boldsymbol{p}} \;\middle|\; i = 1, \ldots, N_{dof} \right\}, \tag{4.2.4}$$

where $N_{dof} = N_t N_s$, with $N_t := n_t - 1$, $N_s := \prod_{l=1}^d N_{s,l}$ and $N_{s,l} := n_l - 2$ for $l = 1, \ldots, d$. We can proceed analogously with the space with final conditions. Finally, we denote by $\mathcal{X}_{h,0}$ the isogeometric space defined in 3.1.9 as the isoparametric push-forward of (4.2.4) through the geometric map $\boldsymbol{G}$, that is

$$\mathcal{X}_{h,0} := \operatorname{span}\left\{ B_{i,\boldsymbol{p}} := \widehat{B}_{i,\boldsymbol{p}} \circ \boldsymbol{G}^{-1} \;\middle|\; i = 1, \ldots, N_{dof} \right\}. \tag{4.2.5}$$

Recall that, the following tensor-product structure holds true,

$$\mathcal{X}_{h,0} = \mathcal{X}_{t,h_t,0} \otimes \mathcal{X}_{s,h_s},$$

where $\mathcal{X}_{t,h_t,0}$ is given as in 3.1.11, while $\mathcal{X}_{s,h_s}$ is defined in 3.1.10. Analogously, we define $\mathcal{X}_{h,T}$, the isogeometric space with homogeneous Dirichlet and final conditions.

$$\mathcal{X}_{h,T} := \operatorname{span}\left\{ B_{i,\boldsymbol{p}} := \widehat{B}_{i,\boldsymbol{p}} \circ \boldsymbol{G}^{-1} \;\middle|\; i = 1, \ldots, N_{dof} \right\}, \tag{4.2.6}$$

4.2.2 Best approximation in $\mathcal{V}$ norm

In this section we recall the approximation properties of splines. For the best approximation estimate of isogeometric spaces in Sobolev norms, we refer to [4, 14]. The following results states the orders of approximation with $\mathcal{X}_{h,0}$ in the $\mathcal{V}$ norm.

Theorem 4.2. *For all $u \in (H^{q_t}(0,T) \otimes H^2_0(\Omega)) \cap (H^1_0(0,T) \otimes H^{q_s}(\Omega))$, where $q_t \geq 1$ and $q_s \geq 2$ the best approximation in $\mathcal{X}_{h,0}$ satisfies*

$$\inf_{u_h \in \mathcal{X}_{h,0}} \|u - u_h\|_{\mathcal{V}} \leq C \left(h_t^{k_t - 1} \|u\|_{H^{k_t}(0,T)\otimes H^2(\Omega)} + h_s^{k_s-2} \|u\|_{H^1(0,T)\otimes H^{k_s}(\Omega)} \right), \quad (4.2.7)$$

where $k_t := \min\{q_t, p_t + 1\}$, $k_s := \min\{q_s, p_s + 1\}$, and C is a constant that depends only on p_t, p_s, θ and the parametrization $\boldsymbol{G}$.

Proof. The result follows from the anisotropic error estimates developed in [14]. We report here only the main steps, since the proof is similar to the one of [85, Proposition 4]. The generalization of [14, Theorem 5.1] to the $d+1$ dimensional case, gives the existence of a projection $\Pi_h : (H^{q_t}(0,T) \otimes H^2_0(\Omega)) \cap (H^1_0(0,T) \otimes H^{q_s}(\Omega)) \to \mathcal{X}_{h,0}$, such that

$$\begin{aligned}
\|u - \Pi_h u\|_{L^2(0,T)\otimes L^2(\Omega)} &\leq C_1 \left(h_t^{k_t-1} \|u\|_{H^{k_t-1}(0,T)\otimes L^2(\Omega)} + h_s^{k_s-2} \|u\|_{L^2(0,T)\otimes H^{k_s-2}(\Omega)} \right), \\
\|u - \Pi_h u\|_{H^1(0,T)\otimes L^2(\Omega)} &\leq C_2 \left(h_t^{k_t-1} \|u\|_{H^{k_t}(0,T)\otimes L^2(\Omega)} + h_s^{k_s-2} \|u\|_{H^1(0,T)\otimes H^{k_s-2}(\Omega)} \right), \\
\|u - \Pi_h u\|_{L^2(0,T)\otimes H^2(\Omega)} &\leq C_3 \left(h_t^{k_t-1} \|u\|_{H^{k_t-1}(0,T)\otimes H^2(\Omega)} + h_s^{k_s-2} \|u\|_{L^2(0,T)\otimes H^{k_s}(\Omega)} \right).
\end{aligned} \quad (4.2.8)$$

From the following inequality

$$\begin{aligned}
\|u - v_h\|^2_{\mathcal{V}} &= \|u - v_h\|^2_{L^2(\mathcal{Q})} + \|\mathbb{S}(u - v_h)\|^2_{L^2(\mathcal{Q})} \\
&\leq \|u - v_h\|^2_{L^2(\mathcal{Q})} + 2\|\partial_t(u - v_h)\|^2_{L^2(\mathcal{Q})} + 2\nu\|\Delta(u - v_h)\|^2_{L^2(\mathcal{Q})} \\
&\leq \|u - v_h\|^2_{L^2(0,T)\otimes L^2(\Omega)} + 2\|u - v_h\|^2_{H^1(0,T)\otimes L^2(\Omega)} + 2\nu\|u - v_h\|^2_{L^2(0,T)\otimes H^2(\Omega)},
\end{aligned}$$

with the choice $v_h = \Pi_h u$, and by (4.2.8) with obvious upper bounds on the right hand side, it holds

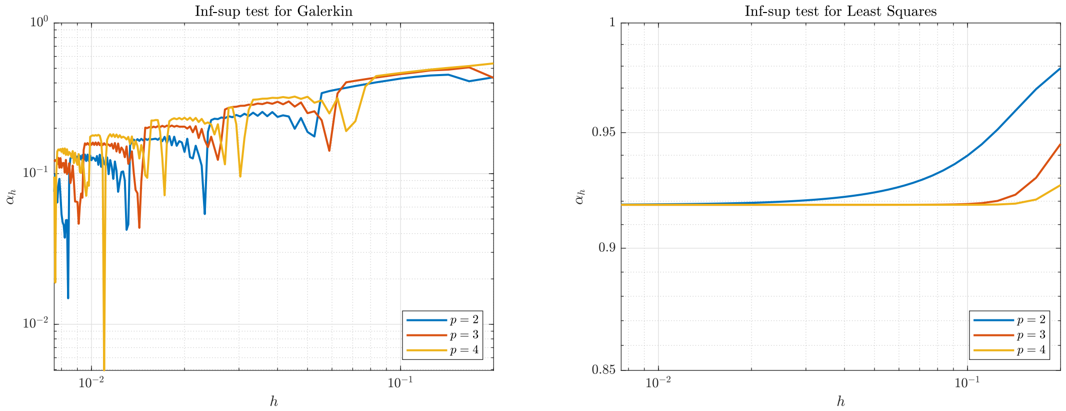
$$\|u - \Pi_h u\|_{\mathcal{V}} \leq C \left(h_t^{k_t-1} \|u\|_{H^{k_t}(0,T)\otimes H^2(\Omega)} + h_s^{k_s-2} \|u\|_{H^1(0,T)\otimes H^{k_s}(\Omega)} \right),$$

therefore, (4.2.7) follows immediatly. □

Remark 4.1. *From Theorem 4.2, when u is smooth or $q_t = q_s$, the order of convergence is dominated by the space direction. This motivates our choice $p_t = p_s =: p$ and $h_t = h_s =: h$ for our numerical tests in Section 4.5. In this case, and if u is smooth,* (4.2.7) *yields h-convergence of order $p - 1$.*

4.3 Space-time discretizations of the Schrödinger equation

In this section we compare three different space-time spline discretizations of problem 4.1.2. The well posedness of the discrete problems relies on the *inf-sup* theory. More precisely, given $\mathcal{V}_h \subset \mathcal{V}$ and $\mathcal{W}_h \subset \mathcal{W}$ two discrete spaces, parametrized by



(a) Instability of space-time Galerkin method. (b) Stability of space-time least squares method.

Figure 4.1. Inf-sup test for the space-time discretizations.

the mesh size h, with $\dim(\mathcal{V}_h) = \dim(\mathcal{W}_h)$, consider the following Petrov-Galerkin approximation problem:

$$\text{Find } u_h \in \mathcal{V}_h \text{ such that } \mathcal{A}(u_h, w_h) = \mathcal{F}(w_h) \quad \forall w_h \in \mathcal{W}_h. \tag{4.3.9}$$

It is well known that: it exists a unique solution $u_h \in \mathcal{V}_h$, if and only if,

$$\exists\, \alpha > 0 \text{ such that } 0 < \alpha \leq \alpha_h := \inf_{v_h \in \mathcal{V}_h} \sup_{w_h \in \mathcal{W}_h} \frac{|\mathcal{A}(v_h, w_h)|}{\|v_h\|_{\mathcal{V}} \|w_h\|_{\mathcal{W}}}, \tag{4.3.10}$$

Moreover, the existence of the inf-sup constant $\alpha > 0$ ensures the following quasi-optimality result.

Theorem 4.3. *If $u \in \mathcal{V}$ is the solution of* (4.1.2)*, and $u_h \in \mathcal{V}_h$ is the solution of* (4.3.9)*, it holds*

$$\|u - u_h\|_{\mathcal{V}} \leq \frac{1}{\alpha} \inf_{v_h \in \mathcal{V}_h} \|u - v_h\|_{\mathcal{V}}. \tag{4.3.11}$$

4.3.1 Instability of the space-time Galerkin method

Let $\mathcal{V}_h := \mathcal{X}_{h,0}$ be the isogeometric space defined in (4.2.5) endowed with the $\|\cdot\|_{\mathcal{V}}$-norm, and choose $\mathcal{W}_h := \mathcal{X}_{h,0}$ endowed with the $\|\cdot\|_{\mathcal{W}}$-norm. Consider the following Galerkin formulation of (4.1.2):

$$\text{Find } u_h \in \mathcal{V}_h \text{ such that } \mathcal{A}(u_h, w_h) = \mathcal{F}(w_h) \quad \forall w_h \in \mathcal{W}_h. \tag{4.3.12}$$

The stability and the well-posedness of formulation (4.3.12) are not guaranteed. Indeed the discrete inf-sup constant α_h depends on the mesh size h and degenerates under mesh refinement, as shown in Figure 4.1a.

4.3.2 Least squares space-time method

In order to retrieve a well posed space-time discretization to (4.1.2), given the quadratic functional $\mathcal{J} : \mathcal{V} \to \mathbb{R}$, defined as

$$\mathcal{J}(u) := \frac{1}{2} \|\mathbb{S}u - f\|^2_{L^2(\mathcal{Q})},$$

we can write the least squares space-time formulation of (4.1.1): find $u \in \mathcal{V}$ such that

$$u = \arg\min_{v \in \mathcal{V}} \mathcal{J}(v),$$

which is a well posed problem and its Euler-Lagrange equations are

$$(\mathbb{S}u, \mathbb{S}v) = (f, \mathbb{S}v), \quad \forall v \in \mathcal{V}.$$

This suggests to consider the following least squares discretization method for problem (4.1.2). Let $\mathcal{V}_h := \mathcal{X}_{h,0}$ be the isogeometric space defined in (4.2.5) endowed with the $\|\cdot\|_{\mathcal{V}}$-norm, and choose $\mathcal{W}_h := \mathbb{S}(\mathcal{V}_h)$ endowed with the $\|\cdot\|_{\mathcal{W}}$-norm. Consider the following discrete problem:

$$\text{Find } u_h \in \mathcal{V}_h \text{ such that } \mathcal{A}(u_h, w_h) = \mathcal{F}(w_h) \quad \forall w_h \in \mathcal{W}_h. \tag{4.3.13}$$

$\mathbb{S}$ is a bijection between $\mathcal{V}_h$ and $\mathcal{W}_h$, which means, for any $h > 0$ it exists the discrete inf-sup constant $\alpha_h > 0$. Consider $0 < \alpha := \|\mathbb{S}^{-1}\|^{-1}_{\mathcal{W}\to\mathcal{V}}$, and for all $w_h \in \mathcal{W}_h$ take $v_h = \mathbb{S}^{-1}(w_h)$, and it holds

$$0 < \alpha\|v_h\|_{\mathcal{V}}\|w_h\|_{\mathcal{W}} \leq \|w_h\|^2_{\mathcal{W}} = (\mathbb{S}v_h, w_h) = \mathcal{A}(v_h, w_h),$$

that is,

$$0 < \alpha \leq \sup_{w_h \in \mathcal{W}_h} \frac{|\mathcal{A}(v_h, w_h)|}{\|v_h\|_{\mathcal{V}}\|w_h\|_{\mathcal{W}}}.$$

Taking the infimum for $v_h \in \mathcal{V}_h$ proves (4.3.10), that is the discrete inf-sup α_h is uniformly bounded from below by a positive constant $\alpha > 0$. This is investigated numerically in Figure 4.1b. The following a-priori error estimate for h-refinements is a direct consequence of quasi-optimality Theorem 4.3 and best approximation Theorem 4.2.

Corollary 4.3.1. *Given $u \in \mathcal{V} \cap (H^{q_t}(0,T) \otimes H^2_0(\Omega)) \cap (H^1_0(0,T) \otimes H^{q_s}(\Omega))$, where where $q_t \geq 1$ and $q_s \geq 2$, and $u_h \in \mathcal{V}_h$ the solution of* (4.3.13)*, it holds*

$$\|u - u_h\|_{\mathcal{V}} \leq C\left(h_t^{k_t-1}\|u\|_{H^{k_t}(0,T)\otimes H^2(\Omega)} + h_s^{k_s-2}\|u\|_{H^1(0,T)\otimes H^{k_s}(\Omega)}\right) \tag{4.3.14}$$

where $k_t := \min\{q_t, p_t + 1\}$ and $k_s := \min\{q_s, p_s + 1\}$.

4.3.3 Ultraweak space-time method

Here we recall also the following ultraweak discretization that has been proposed in [57]. Let $\mathcal{W}_h := \mathcal{X}_{h,T}$ be the isogeometric space with final conditions endowed with the $\|\cdot\|_{\mathcal{W}}$-norm, and fix $\mathcal{V}_h := \mathbb{S}(\mathcal{W}_h)$ endowed with the $\|\cdot\|_{L^2(\mathcal{Q})}$-norm. Notice that, $\forall v_h \in \mathcal{V}_h$, $w_h \in \mathcal{W}_h$, it holds

$$\mathcal{A}(v_h, w_h) = (\mathbb{S}(v_h), w_h) = (v_h, \mathbb{S}(w_h)) - \mathrm{i}\,(v_h(\cdot,0), w_h(\cdot,0))_{L^2(\Omega)},$$

with $(\cdot,\cdot)_{L^2(\Omega)}$ denoting the complex scalar product in $L^2(\Omega)$. Therefore, introducing the sesquilinear form

$$\mathcal{A}_{\mathtt{uw}}(v_h, w_h) := (v_h, \mathbb{S}(w_h)), \quad \forall v_h \in \mathcal{V}_h,\ w_h \in \mathcal{W}_h, \tag{4.3.15}$$

we have the following ultraweak formulation of (4.1.2):

$$\text{Find } u_h \in \mathcal{V}_h \text{ such that } \mathcal{A}_{\mathtt{uw}}(u_h, w_h) = \mathcal{F}(w_h) + \mathrm{i}\,(u_0, w_h(\cdot, 0))_{L^2(\Omega)} \quad \forall w_h \in \mathcal{W}_h, \tag{4.3.16}$$

where now the right hand side contains eventually the initial data u_0. As regards the well posedness and stability of (4.3.16) we refer to [57].

We conclude this section by noting that the discrete ultraweak operator is the same operator that arises from the least squares discretization. The difference is clearly based on the initial or final conditions included in the discretization spaces. For least squares the discrete spaces include initial conditions, while for ultraweak they include final conditions.

Remark 4.2. *For the least squares form with $u_h = \mathbb{S}(w_h)$ and $v_h, w_h \in \mathcal{X}_{h,0}$, it holds*

$$\mathcal{A}(v_h, u_h) = (\mathbb{S}(v_h), u_h) = (\mathbb{S}(v_h), \mathbb{S}(w_h))\,.$$

For the ultraweak form with $u_h = \mathbb{S}(v_h)$ and $v_h, w_h \in \mathcal{X}_{h,T}$, it holds

$$\mathcal{A}_{\mathtt{uw}}(u_h, w_h) = (u_h, \mathbb{S}(w_h)) = (\mathbb{S}(v_h), \mathbb{S}(w_h))\,.$$

4.4 Fast solver for the parametric domain

In this section, first we consider the matrix representation of the least squares discretization, then we focus on the case $\mathcal{Q} = (0, T) \times (0, 1)^d$ That is the parametric domain in space times a finite interval in time direction. In this framework, the isogeometric map $\boldsymbol{F}$ is the identity operator, and we are able to introduce a stable and fast solver for problem (4.3.13).

4.4.1 Matrix structure

The least squares space-time discretization (4.3.13) can be written as:

$$\text{Find } u_h \in \mathcal{V}_h \text{ such that } \mathcal{A}(u_h, \mathbb{S}v_h) = \mathcal{F}(\mathbb{S}v_h) \quad \forall v_h \in \mathcal{V}_h,$$

and in particular, for all $v_h \in \mathcal{V}_h$, we point out that

$$\begin{aligned}\mathcal{A}(u_h, \mathbb{S}v_h) &= \int_\Omega \int_0^T (\mathbb{S}u_h)\,\overline{(\mathbb{S}v_h)}\,\mathrm{dt}\;\;\mathrm{d}\Omega \\ &= \int_\Omega \int_0^T \partial_t u_h \overline{\partial_t v_h} + \nu^2 \Delta u_h \overline{\Delta v_h} + i\nu \partial_t \nabla u_h \cdot \overline{\nabla v_h} - i\nu \nabla u_h \cdot \overline{\partial_t \nabla v_h}\,\mathrm{dt}\;\;\mathrm{d}\Omega,\end{aligned}$$

and

$$\mathcal{F}(\mathbb{S}v_h) = \int_\Omega \int_0^T f\,\overline{(\mathbb{S}v_h)}\,\mathrm{dt}\;\;\mathrm{d}\Omega = \int_\Omega \int_0^T f\,\overline{(i\partial_t v_h - \Delta v_h)}\,\mathrm{dt}\;\;\mathrm{d}\Omega.$$

Therefore, the linear system associated to (4.3.13) is

$$\mathbf{A}\mathbf{u} = \mathbf{f}, \tag{4.4.17}$$

where $[\mathbf{A}]_{i,j} = \mathcal{A}\left(B_{j,\boldsymbol{p}}, \mathbb{S}(B_{i,\boldsymbol{p}})\right)$ and $[\mathbf{f}]_i = \mathcal{F}\left(\mathbb{S}(B_{i,\boldsymbol{p}})\right)$. The tensor-product structure of the isogeometric space (4.2.5) allows to write the system matrix $\mathbf{A}$ as sum of Kronecker products of matrices as

$$\mathbf{A} \;= \mathbf{M}_s \otimes \mathbf{L}_t + \nu^2 \mathbf{B}_s \otimes \mathbf{M}_t + \nu \mathbf{L}_s \otimes \left(\mathbf{W}_t + \mathbf{W}_t^*\right), \tag{4.4.18}$$

where for $i, j = 1, \dots, N_s$

$$[\mathbf{L}_s]_{i,j} = \int_\Omega \nabla B_{j,\boldsymbol{p}_s}(\boldsymbol{x}) \cdot \nabla B_{i,\boldsymbol{p}_s}(\boldsymbol{x}) \;\, \mathrm{d}\Omega, \quad [\mathbf{M}_s]_{i,j} = \int_\Omega B_{j,\boldsymbol{p}_s}(\boldsymbol{x}) \; B_{i,\boldsymbol{p}_s}(\boldsymbol{x}) \;\, \mathrm{d}\Omega,$$
$$[\mathbf{B}_s]_{i,j} = \int_\Omega \Delta B_{j,\boldsymbol{p}_s}(\boldsymbol{x}) \Delta B_{i,\boldsymbol{p}_s}(\boldsymbol{x}) \;\, \mathrm{d}\Omega. \tag{4.4.19a}$$

while for $i, j = 1, \dots, N_t$

$$[\mathbf{L}_t]_{i,j} = \int_0^T b'_{j,p_t}(t)\, b'_{i,p_t}(t)\, \mathrm{d}t, \quad [\mathbf{M}_t]_{i,j} = \int_0^T b_{j,p_t}(t)\, b_{i,p_t}(t)\, \mathrm{d}t,$$
$$[\mathbf{W}_t]_{i,j} = \mathrm{i} \int_0^T b'_{j,p_t}(t)\, b_{i,p_t}(t)\, \mathrm{d}t. \tag{4.4.19b}$$

4.4.2 Preconditioner definition

We introduce, for the system (4.4.17), the preconditioner

$$\widehat{\mathbf{P}} := \widehat{\mathbf{M}}_s \otimes \mathbf{L}_t + \nu^2 \widehat{\mathbf{L}}_s^T \widehat{\mathbf{M}}_s^{-1} \widehat{\mathbf{L}}_s \otimes \mathbf{M}_t + \nu \widehat{\mathbf{L}}_s \otimes (\mathbf{W}_t + \mathbf{W}_t^*), \tag{4.4.20}$$

where the matrices $\mathbf{L}_t, \mathbf{M}_t$ and $\mathbf{W}_t$ are defined in (4.4.19b), while $\widehat{\mathbf{L}}_s$ and $\widehat{\mathbf{M}}_s$ are

$$\widehat{\mathbf{L}}_s = \sum_{l=1}^d \widehat{\mathbf{M}}_d \otimes \cdots \otimes \widehat{\mathbf{M}}_{l+1} \otimes \widehat{\mathbf{L}}_l \otimes \widehat{\mathbf{M}}_{l-1} \otimes \cdots \otimes \widehat{\mathbf{M}}_1, \quad \text{and} \quad \widehat{\mathbf{M}}_s = \widehat{\mathbf{M}}_d \otimes \cdots \otimes \widehat{\mathbf{M}}_1,$$

and for $l = 1, \dots, d$, with indexes $i, j = 1, \dots, N_{s,l}$, it holds

$$[\widehat{\mathbf{L}}_l]_{i,j} := \int\limits_0^1 \widehat{b}'_{j,p}(x_l) \widehat{b}'_{i,p}(x_l) \mathrm{d}x_l, \qquad \text{and} \qquad [\widehat{\mathbf{M}}_l]_{i,j} := \int\limits_0^1 \widehat{b}_{j,p}(x_l) \widehat{b}_{i,p}(x_l) \mathrm{d}x_l.$$

The efficient application of the proposed preconditioner, that is, the solution of a linear system with matrix $\widehat{\mathbf{P}}$, should exploit the structure highlighted above. When the pencils $(\widehat{\mathbf{L}}_1, \widehat{\mathbf{M}}_1), \dots, (\widehat{\mathbf{L}}_d, \widehat{\mathbf{M}}_d)$ admit a stable generalized eigendecomposition, a possible approach is the Fast Diagonalization (FD) method, see [37, 79] for details.

4.4.3 Stable factorization of $(\widehat{\mathbf{L}}_l, \widehat{\mathbf{M}}_l)$ for $l = 1, \dots, d$

The spatial stiffness and mass matrices $\widehat{\mathbf{L}}_l$ and $\widehat{\mathbf{M}}_l$ are symmetric and positive definite for $l = 1, \dots, d$. Thus, the pencils $(\widehat{\mathbf{L}}_l, \widehat{\mathbf{M}}_l)$ for $l = 1, \dots, d$ admit the generalized eigendecomposition

$$\widehat{\mathbf{L}}_l \mathbf{U}_l = \widehat{\mathbf{M}}_l \mathbf{U}_l \mathbf{\Lambda}_l,$$

where the matrices $\mathbf{U}_l$ contain in each column the $\widehat{\mathbf{M}}_l$-orthonormal generalized eigenvectors and $\mathbf{\Lambda}_l$ are diagonal matrices whose entries contain the generalized eigenvalues. Therefore we have for $l = 1, \dots, d$ the factorizations

$$\mathbf{U}_l^T \widehat{\mathbf{L}}_l \mathbf{U}_l = \mathbf{\Lambda}_l \quad \text{and} \quad \mathbf{U}_l^T \widehat{\mathbf{M}}_l \mathbf{U}_l = \mathbb{I}_{N_{s,l}}, \tag{4.4.21}$$

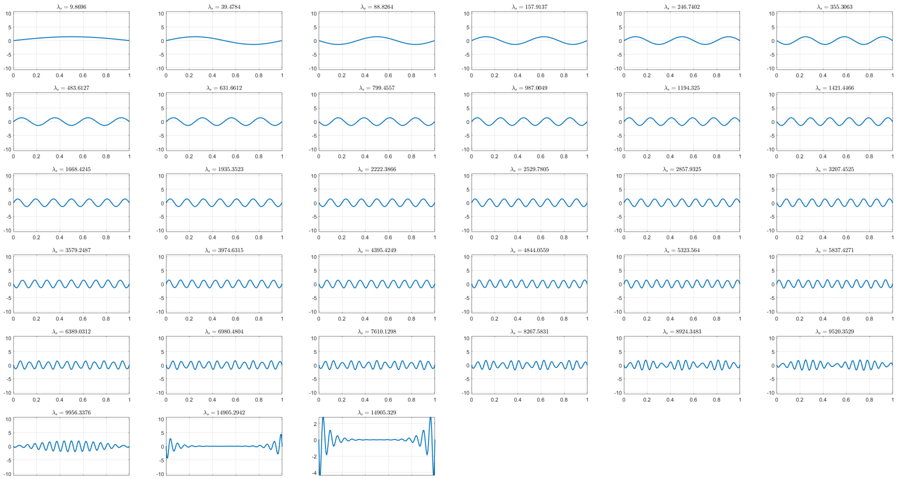


Figure 4.2. Generalized eigenvectors of the pencils $(\widehat{\mathbf{L}}_l, \widehat{\mathbf{M}}_l)$ with associated eigenvalues for $p_s = 3$ and $n_{el} = 32$ elements.

where $\mathbb{I}_{N_{s,l}}$ denotes the identity matrix of dimension $N_{s,l} \times N_{s,l}$.

Figure 4.2 shows the shape of the generalized eigenvectors in $\mathbf{U}_l$, with associated eigenvalue in Λ_l, for a fixed univariate direction $l = 1, \dots, d$ discretized with degree $p_s = 3$ B-Splines and uniform partition. The stability of the decomposition is expressed by the condition number of the eigenvector matrix. In particular $\mathbf{U}_l^T \widehat{\mathbf{M}}_l \mathbf{U}_l = \mathbb{I}_{N_{s,l}}$ implies that

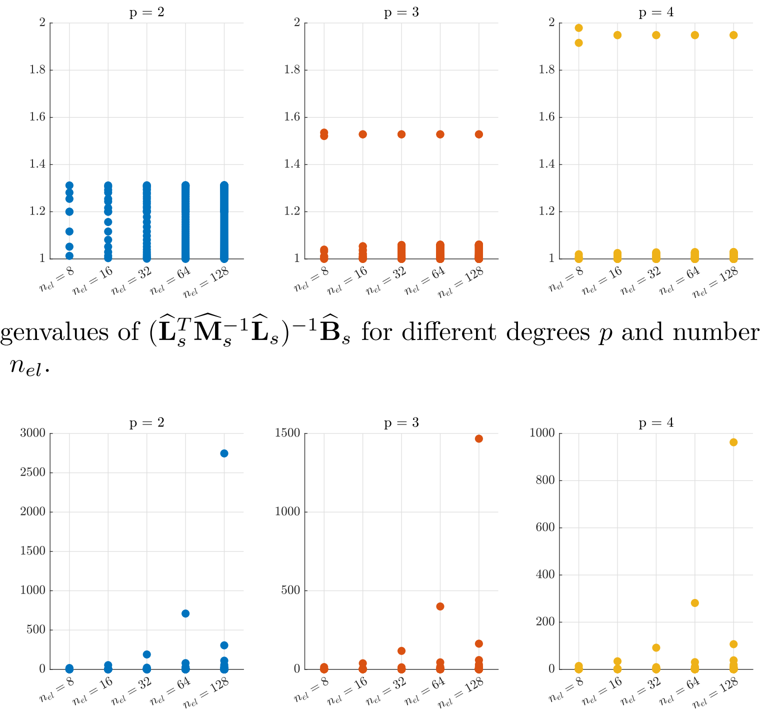
$$\kappa_2(\mathbf{U}_l) := \|\mathbf{U}_l\|_2 \|\mathbf{U}_l^{-1}\|_2 = \sqrt{\kappa_2(\widehat{\mathbf{M}}_l)},$$

where $\|\cdot\|_2$ is the norm induced by the Euclidean vector norm. The condition number $\kappa_2(\widehat{\mathbf{M}}_l)$ has been studied theoretically in [48] and numerically in [85] and it does not depend on the mesh-size, but it depends on the polynomial degree. Indeed, we report in Table 4.1 the behavior of $\kappa_2(\mathbf{U}_l)$ for different values of spline degree p_s and for different uniform discretizations with number of elements denoted by n_{el}. We observe that $\kappa_2(\mathbf{U}_l)$ exhibits a dependence only on p_s, but stays moderately low for all low polynomial degrees that are in the range of interest.

n_{el}	$p_s = 2$	$p_s = 3$	$p_s = 4$	$p_s = 5$	$p_s = 6$	$p_s = 7$	$p_s = 8$
32	$2.7 \cdot 10^0$	$4.5 \cdot 10^0$	$7.6 \cdot 10^0$	$1.3 \cdot 10^1$	$2.1 \cdot 10^1$	$3.5 \cdot 10^1$	$5.7 \cdot 10^1$
64	$2.7 \cdot 10^0$	$4.5 \cdot 10^0$	$7.6 \cdot 10^0$	$1.3 \cdot 10^1$	$2.1 \cdot 10^1$	$3.5 \cdot 10^1$	$5.7 \cdot 10^1$
128	$2.7 \cdot 10^0$	$4.5 \cdot 10^0$	$7.6 \cdot 10^0$	$1.3 \cdot 10^1$	$2.1 \cdot 10^1$	$3.5 \cdot 10^1$	$5.7 \cdot 10^1$
256	$2.7 \cdot 10^0$	$4.5 \cdot 10^0$	$7.6 \cdot 10^0$	$1.3 \cdot 10^1$	$2.1 \cdot 10^1$	$3.5 \cdot 10^1$	$5.7 \cdot 10^1$
512	$2.7 \cdot 10^0$	$4.5 \cdot 10^0$	$7.6 \cdot 10^0$	$1.3 \cdot 10^1$	$2.1 \cdot 10^1$	$3.5 \cdot 10^1$	$5.7 \cdot 10^1$
1024	$2.7 \cdot 10^0$	$4.5 \cdot 10^0$	$7.6 \cdot 10^0$	$1.3 \cdot 10^1$	$2.1 \cdot 10^1$	$3.5 \cdot 10^1$	$5.7 \cdot 10^1$

Table 4.1. $\kappa_2(\mathbf{U}_l)$ for different polynomial degrees p_s and number of elements n_{el}.

Moreover, in [58] it is shown that there is spectral equivalence between $\widehat{\mathbf{B}}_s$ and $\widehat{\mathbf{L}}_s^T \widehat{\mathbf{M}}_s^{-1} \widehat{\mathbf{L}}_s$. We investigate numerically this spectral equivalence, and Figure 4.3a



(a) Eigenvalues of $(\widehat{\mathbf{L}}_s^T\widehat{\mathbf{M}}_s^{-1}\widehat{\mathbf{L}}_s)^{-1}\widehat{\mathbf{B}}_s$ for different degrees p and number of elements n_{el}.

(b) Eigenvalues of $(\widehat{\mathbf{W}}_t^*\widehat{\mathbf{M}}_t^{-1}\widehat{\mathbf{W}}_t)^{-1}\widehat{\mathbf{L}}_t$ for different degrees p and number of elements n_{el}.

Figure 4.3. Numerical investigation of spectral equivalence.

shows the eigenvalues of $(\widehat{\mathbf{L}}_s^T\widehat{\mathbf{M}}_s^{-1}\widehat{\mathbf{L}}_s)^{-1}\widehat{\mathbf{B}}_s$ are clustered and close to 1, for splines of degree $p = 2, 3, 4$ and different uniform partitionas with $n_{el} = 8, 16, 32, 64, 128$. In conclusion the spectral equivalence is stable under mesh refinement.

As regards the time pencils, the spectral equivalence between $\widehat{\mathbf{L}}_t$ and $\widehat{\mathbf{W}}_t^*\widehat{\mathbf{M}}_t^{-1}\widehat{\mathbf{W}}_t$ is unstable under mesh refinement, see Figure 4.3b where we performed the analogous test, therefore we kept the full structure of the time pencils in the preconditioner.

4.4.4 Application of the preconditioner

The application of the preconditioner involves the solution of the linear system

$$\widehat{\mathbf{P}}\mathbf{s} = \mathbf{r}, \tag{4.4.22}$$

where $\widehat{\mathbf{P}}$ has the structure (4.4.20). We are able to efficiently solve system (4.4.22) by the Fast Diagonalization method. The starting point, is the setup of the preconditioner, that is the factorizations (4.4.21) of the pencils $(\widehat{\mathbf{L}}_l, \widehat{\mathbf{M}}_l)$ for $l = 1, \ldots, d$.

Then, define $\mathbf{U}_s := \mathbf{U}_d \otimes \cdots \otimes \mathbf{U}_1$ and $\mathbf{\Lambda}_s := \sum_{l=1}^d \mathbb{I}_{N_{s,d}} \otimes \cdots \otimes \mathbb{I}_{N_{s,l+1}} \otimes \mathbf{\Lambda}_l \otimes \mathbb{I}_{N_{s,l-1}} \otimes \cdots \otimes \mathbb{I}_{N_{s,1}}$. Notice that $\widehat{\mathbf{M}}_s^{-1} = \mathbf{U}_s\mathbf{U}_s^T$, therefore the matrix $\widehat{\mathbf{L}}_s\widehat{\mathbf{M}}_s^{-1}\widehat{\mathbf{L}}_s$ admits the stable factorization

$$\mathbf{U}_s^T\widehat{\mathbf{L}}_s\widehat{\mathbf{M}}_s^{-1}\widehat{\mathbf{L}}_s\mathbf{U}_s = \mathbf{\Lambda}_s^2.$$

The preconditioner $\widehat{\mathbf{P}}$ admits the following factorization

$$\widehat{\mathbf{P}} = \left(\mathbf{U}_s^T \otimes \mathbb{I}_{N_t}\right)^{-1}\left(\mathbb{I}_{N_s} \otimes \mathbf{L}_t + \nu^2\mathbf{\Lambda}_s^2 \otimes \mathbf{M}_t + \nu\mathbf{\Lambda}_s \otimes (\mathbf{W}_t + \mathbf{W}_t^*)\right)(\mathbf{U}_s \otimes \mathbb{I}_{N_t})^{-1}. \tag{4.4.23}$$

Note that the second factor in (4.4.23) that is

$$\mathbf{H} := \left(\mathbb{I}_{N_s} \otimes \mathbf{L}_t + \nu^2 \boldsymbol{\Lambda}_s^2 \otimes \mathbf{M}_t + \nu \boldsymbol{\Lambda}_s \otimes (\mathbf{W}_t + \mathbf{W}_t^*)\right)$$

is sum of three Kronecker matrices, whose space factors are diagonal matrices. We have the following block diagonal structure

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_1 & & \\ & \ddots & \\ & & \mathbf{H}_{N_s} \end{bmatrix},$$

where $\mathbf{H}_i$, for $i = 1, \dots, N_s$, are banded matrices with bandwidth $2p_t + 1$ defined as

$$\mathbf{H}_i := \mathbf{L}_t + \nu^2 [\boldsymbol{\Lambda}_s^2]_{i,i} \mathbf{M}_{N_t} + \nu [\boldsymbol{\Lambda}_s]_{i,i} \left(\mathbf{W}_t + \mathbf{W}_t^*\right).$$

In order to invert $\mathbf{H}$, it is now sufficient to invert the following independent N_s problems of size $N_t \times N_t$:

$$\mathbf{H}_i \mathbf{x}_i = \mathbf{y}_i \quad \text{for } i = 1, \dots, N_s. \tag{4.4.24}$$

Summarizing, the solution of (4.4.22) can be computed by the following algorithm.

Algorithm 4 Fast Diagonalization

```
1: Compute the factorizations (4.4.21).
2: Compute $\mathbf{y} = (\mathbf{U}_s^T \otimes \mathbb{I}_{N_t})\mathbf{r}$.
3: Compute $\mathbf{x}_i = \mathbf{H}_i^{-1}\mathbf{y}_i \quad$ for $i = 1, \dots, N_s$.
4: Compute $\mathbf{s} = (\mathbf{U}_s \otimes \mathbb{I}_{N_t})\, \widetilde{\mathbf{s}}$.
```

We conclude with the following remark for a possible parallel implementation of Algorithm 4.

Remark 4.3. *The decision to consider time as the first variable allows us to write the matrix* $\mathbf{H}$ *in a block diagonal form. In view of an efficient parallel implementation, this natural diagonal block structure does not require data shuffling, reducing the communication cost between nodes.*

4.4.5 Computational cost and memory requirements

In this section we discuss the computational costs and memory requirements in the implementation of Algorithm 4. First, notice that the matrix $\mathbf{A}$ in (4.4.18) is symmetric positive definite therefore we choose Conjugate Gradients (CG) as linear solver for solving the system (4.4.17). Clearly, the global computational cost of the iterative CG solver depends on both the preconditioner setup and application cost. The setup has to be performed only once, while the application of the preconditioner is performed at each iteration. Denoting by N_{iter} the number of iterations required to get convergence, the global computational cost is:

$$\texttt{COST} = \texttt{SETUP} + N_{iter} * \texttt{APPLICATION}.$$

We assume for simplicity that $N_{s,l} = n_s$ for each univariate direction $l = 1, \dots, d$, that is the space matrices have dimension $n_s \times n_s$, while the time matrices involved in the preconditioners have dimension $N_t \times N_t$. Thus the total number of degrees-of-freedom is $N_{dof} = N_s N_t = n_s^d N_t$.

The setup of $\widehat{\mathbf{P}}$ includes the operations performed in Step 1 of Algorithm 4, i.e. d spatial eigendecompositions, that have a total cost of $O(dn_s^3)$ FLOPs, and the construction of the block diagonal matrix $\mathbf{H}$, which costs $O(p_t N_t N_s) = O(p_t N_{dof})$. We remark that the setup of the preconditioners has to be performed only once, since the matrices involved do not change during the iterative procedure.

The application of the preconditioner is performed by Steps 2-4 of Algorithm 4. Exploiting the properties of the Kronecker product, Step 2 and Step 4 costs $O(dn_s^{d+1} N_t) = O(dn_s N_{dof})$ FLOPs. The cost of solving each sparse problem (4.4.24) makes the cost for Step 3 equal to $O(p_t^2 N_t N_s) = O(p_t^2 N_{dof})$ FLOPs. The non-optimal dominant cost of Step 2 and Step 4 is determined by the dense matrix-matrix products. However, these operations are usually implemented on modern computers in a very efficient way and the overall serial computational time grows almost as $O(N_{dof})$, see i.e. [85, 75]

The other dominant computational cost in a CG iteration is the cost of the residual computation. In Algorithm 4, this involves the multiplication of the matrix $\mathbf{A}$ with a vector. This multiplication is done by exploiting the special structure (4.4.18), that allows a matrix-free approach. With the matrix-free approach, noting that the time matrices $\mathbf{L}_t, \mathbf{M}_t, \mathbf{W}_t$ are banded matrices with bandwidth $2p_t + 1$, and the spatial matrices $\mathbf{J}_s, \mathbf{L}_s, \mathbf{M}_s$ have a number of non-zeros per row equal to $(2p_s + 1)^d$, the computational cost of a single matrix-vector product is $O(N_{dof} p^d)$ FLOPs, if we assume $p = p_s \approx p_t$.

Considering also $n_s = N_t = N_{dof}^{1/(d+1)}$, we conclude that the total cost, in FLOPs, of Algorithm 4 is

$$O(dN_{dof}^{3/(d+1)}) + N_{iter} * \left(O(dN_{dof}^{(d+2)/(d+1)}) + O(p^2 N_{dof}) + O(p^d N_{dof}) \right).$$

The dominant cost in the iterative solver is therefore represented by the residual computation. This is a typical behaviour of the FD-based preconditioning strategies, see [85, 93, 87].

We now investigate the memory consumption of the preconditioning strategy proposed. For the preconditioner, we have to store the eigenvector spatial matrices, $\mathbf{U}_1, \dots, \mathbf{U}_d$, the diagonal matrices $\mathbf{\Lambda}_1, \dots, \mathbf{\Lambda}_d$ and the banded time pencils $\mathbf{L}_t, \mathbf{M}_t$ and $\mathbf{W}_t$ of size $N_t \times N_t$. The memory required is roughly

$$O(d(n_s^2 + n_s)) + O(p_t N_{dof}) + O(p_t N_t).$$

For the system matrix $\mathbf{A}$, in addition to the time factors $\mathbf{L}_t, \mathbf{M}_t$ and $\mathbf{W}_t$, we need to store the spatial factors $\mathbf{M}_s, \mathbf{B}_s$ and $\mathbf{L}_s$. Thus the memory further required is roughly

$$O(p_s^d N_s).$$

These numbers show that memory-wise our space-time strategy is very appealing when compared to other approaches, even when space and time variables are discretized separately, e.g., with finite differences in time or other time-stepping schemes. For example if we assume $d = 3$, $p_t \approx p_s = p$ and $N_t^2 \leq Cp^3 N_s$, then the total memory consumption is $O(p^3 N_s + N_{dof})$, that is equal to the sum of the memory needed to store the Galerkin matrices associated to spatial variables and the memory needed to store the solution of the problem.

We remark that we could avoid storing the factors of $\mathbf{A}$ by using the matrix-free approach of [94]. The memory and the computational cost of the iterative solver would significantly improve, both for the setup and the matrix-vector multiplications. However, we do not pursue this strategy, as it is beyond the scope of this work.

Remark 4.4. *In view of Remark 4.2, the matrices associated with the resulting linear systems (least squares and ultraweak) are distinct submatrices extracted from the representative matrix of the operator without boundary conditions. Therefore, the preconditioner proposed in this section for the least squares form, together with its factorization and application, can be extended to the ultraweak form, with the same computational costs and memory requirements.*

4.5 Numerical Results

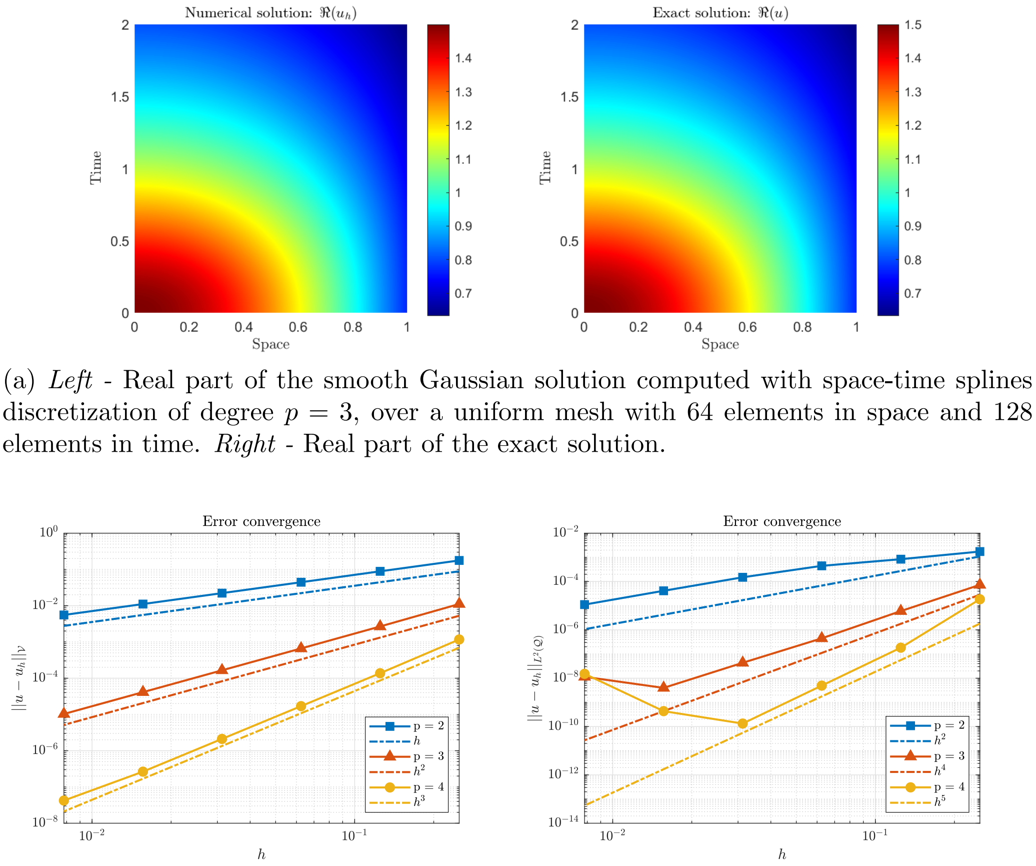
This section is devoted to the computation of the solution of Schrödinger problem (4.1.1), and to its extension to non-homogeneous conditions, with the discretization proposed in (4.3.13). We first present the numerical experiments that assess the convergence behavior of the least squares Petrov-Galerkin approximation and then we analyze the performance of the preconditioners.

We consider only sequential executions and we force the use of a single computational thread in a Intel Core i5-1035G1 processor, running at 1 GHz and with 16 GB of RAM.

The tests are performed with Matlab R2023a and GeoPDEs toolbox [107]. We use the `eig` Matlab function to compute the generalized eigendecompositions present in Step 1 of Algorithm 4,while Tensorlab toolbox [98] is employed to perform the multiplications with Kronecker matrices occurring in Step 2 and Step 4. The solution of the linear systems (4.4.24) in Step 3 is performed pagewise by Matlab direct solver (pagewise backslash operator `pagemldivide`). The linear system is solved by CG, with tolerance equal to 10^{-8} and with the null vector as initial guess in all tests.

According to Remark 4.1, we use the same mesh-size in space and in time $h_s = h_t =: h$, and use splines of maximal continuity and same degree in space and in time $p_t = p_s =: p$. For the sake of simplicity, we also consider uniform knot vectors, and denote the number of elements in each parametric direction by $n_{el} := \frac{1}{h}$.

In our tables, the symbol "$**$" denotes that the invertion of the matrix $\mathbf{A}$ in (4.4.18), by Matlab direct solver backslash operator "\", requires more than 2 hours of computational time, while the symbol "$*$" indicates that the number of iterations in the CG solver exceeds the upper bound set to 200 iterates. We remark that in all the tables the total solving time of the iterative strategies includes also the setup time of the considered preconditioner.

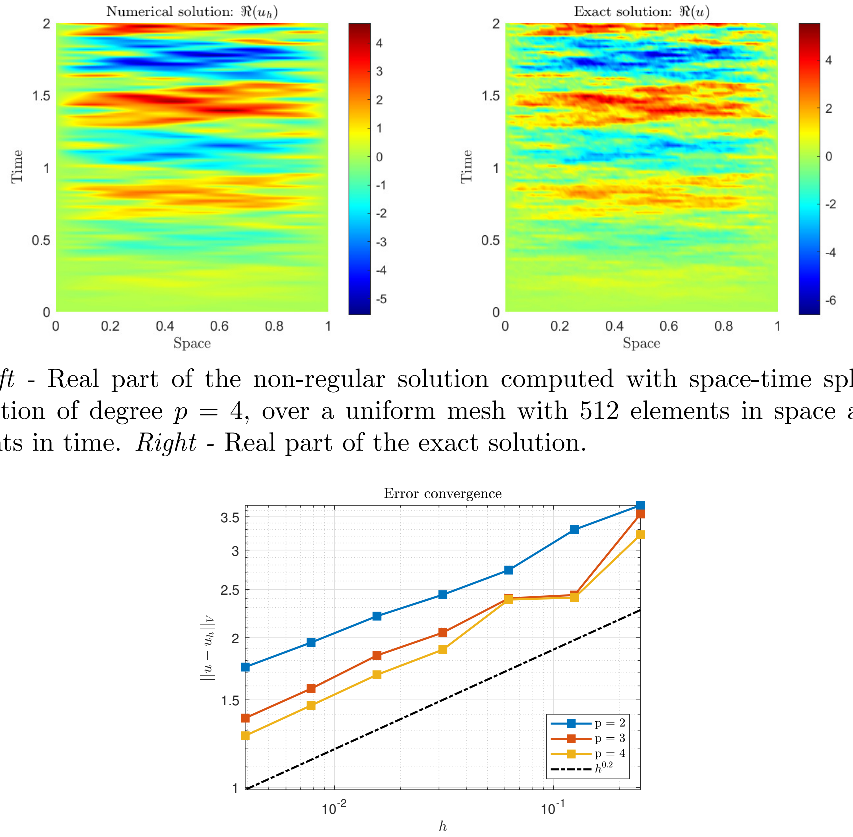


(a) *Left* - Real part of the smooth Gaussian solution computed with space-time splines discretization of degree $p = 3$, over a uniform mesh with 64 elements in space and 128 elements in time. *Right* - Real part of the exact solution.

(b) Error convergence in $\mathcal{V}$-norm.

(c) Error convergence in $L^2(\mathcal{Q})$-norm.

Figure 4.4. Smooth solution and error convergence of the space-time least squares discretization.



(a) *Left* - Real part of the non-regular solution computed with space-time splines discretization of degree $p = 4$, over a uniform mesh with 512 elements in space and 1024 elements in time. *Right* - Real part of the exact solution.

(b) Error convergence in $\mathcal{V}$-norm.

Figure 4.5. Non-regular solution and error convergence of the space-time discretization.

4.5.1 Orders of convergence

Consider the Schrödinger equation as modeled in (4.1.1), for the space-time domain $\mathcal{Q} = (0,T) \times (0,1)$, with $T = 2$. The reference solution is the complex Gaussian

$$u(t,x) = \frac{\alpha\beta}{\sqrt{\beta^2 - \mathrm{i}\gamma t}} \exp\left\{-\frac{x^2}{\beta^2 - \mathrm{i}\gamma t}\right\}, \tag{4.5.25}$$

where $\alpha = \beta = 1.5$ and $\gamma = 2.5$. Here the Dirichlet boundary condition is $u|_{\Gamma_D}$, the initial condition is $u(0,x)$ and the right hand side is $f = Au$. The problem is discretized with a uniform mesh in both space and time directions. The solution for $p = 3$ is shown in Figure 4.4a. In Figure 4.4b it is shown the convergence analysis of the error under h-refinement and for different polynomial degrees $p = 2,3,4$. The errors are computed both with $\|\cdot\|_{\mathcal{V}}$-norm, for which the convergence Theorem 4.3 holds, and with $\|\cdot\|_{L^2(\mathcal{Q})}$-norm, even if this case is not covered by theoretical results. For this smooth solution, the error study reveals optimal convergence under h-refinement in $\mathcal{V}$-norm. The numerical results seem to exclude superconvergence in the $L^2(\mathcal{Q})$ norm, see Figure 4.4c

The second test considers the following example from [36]. Consider the space domain $\Omega = (0,1)$, the final time $T = 2$, and the space-time domain $\mathcal{Q} = (0,T) \times (0,1)$. Homogeneous Dirichlet boundary conditions are considered on Γ_D. Let us denote by e_k and ω_k^2, which is, for $k = 1,2,\dots,$ an eigenpair of

$$-\Delta e_k = \omega_k^2 e_k, \quad \text{a.e. in } \Omega.$$

By normalizing e_k such that $\|e_k\|_{L^2(\Omega)} = 1$, we consider $f(t,x) = \sum_{k=1}^{+\infty} f_k(t) e_k(x)$, where $f_k(t)$ are the Fourier coefficients of f decomposed in the orthonormal basis e_k at a given time t. By the following specific choice of coefficients

$$f_k(t) = \frac{1}{k} \exp\left\{i\omega_k^2 t\right\} \quad \text{for } k = 1,2,\dots,$$

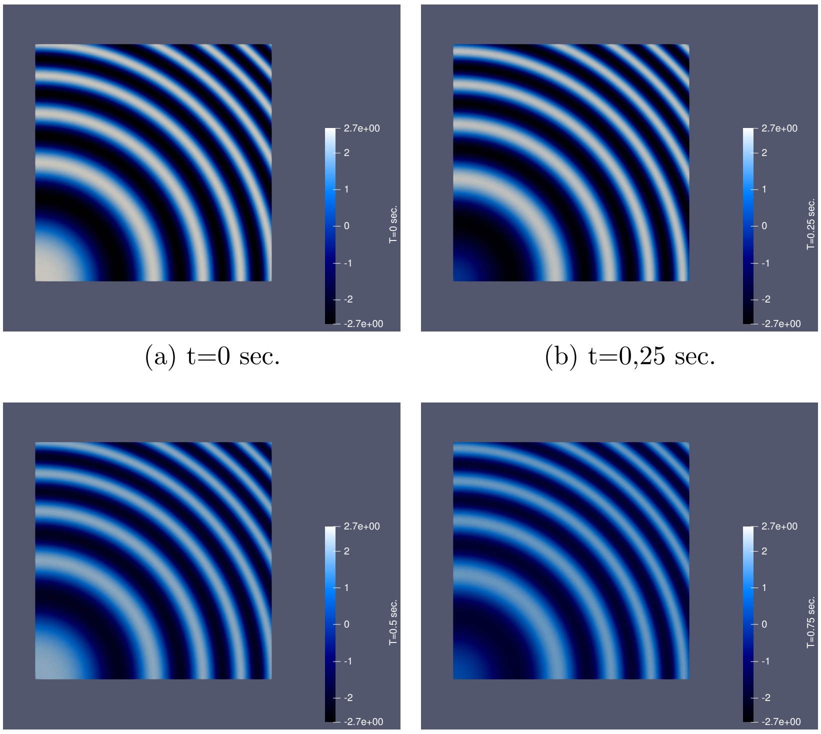
we considered as right hand side in (4.1.1) the following high mode truncated expansion

$$f_M = \sum_{k=1}^{M} \frac{1}{k} \exp\left\{i\omega_k^2 t\right\} e_k(x),$$

with $M \gg 0$. Notice that, the solution to (4.1.1) with this specific right hand side, is

$$u(t,x) = \sum_{k=1}^{M} \frac{-it}{k} \exp\left\{i\omega_k^2 t\right\} e_k(x).$$

We computed the solution for different polynomial degrees, on a uniform mesh, for an high mode right hand side f_M, with $M = 625$. In 4.5a it is plotted the real part of the numerical solution for splines with degree $p = 4$, together with the real part of the explicit solution. The solution of such a problem is non-regular and it can be shown that $u(t,\cdot) \in H^{1/2}(\Omega)$, while $u(\cdot,x) \in H^{1/4}(0,T)$. Figure 4.5b shows the error convergence for the high mode right hand side f_M, with $M = 652$ modes, that is optimal for each polynomial degree $p = 2,3,4$.



(a) t=0 sec.

(b) t=0,25 sec.

(c) t=0,5 sec.

(d) t=0,75 sec.

Figure 4.6. Real part of the numerical approximation of (4.5.26) at four different time frames.

4.5.2 Performance of the preconditioner in the parametric domain

The computational space domain is $\Omega = (0,1)^2$ and the space-time domain is $\mathcal{Q} = (0,T) \times \Omega$ with $T = 1$. The reference solution is a traveling wave, that is

$$u(t,\boldsymbol{x}) = a \exp\left\{-i\frac{|\boldsymbol{x}|^2 + t^2}{\omega^2}\right\}, \tag{4.5.26}$$

with wave number $\omega = 0.2$ and amplitude $a = \sqrt[4]{2/\omega^2}$. Here the Dirichlet boundary conditions are $u|_{\Gamma_D}$. The right hand side is the following

$$f(t,\boldsymbol{x}) = a\left(\frac{4i}{\omega^2} + \frac{4|\boldsymbol{x}|^2}{\omega^4} + \frac{2t}{\omega^2}\right)\exp\left\{-i\frac{|\boldsymbol{x}|^2 + t^2}{\omega^2}\right\}.$$

The numerical solution on a mesh of 64 elements per univariate direction is shown in Figure 4.6 for different time frames.

We analyze the performance of the proposed preconditioner $\widehat{\mathbf{P}}$ for a variety of uniform partitions up to $n_{el} = 64$ per univariate directions, and polynomial degree $p = 2, 3, 4$. In Table 4.2 it is reported the computational clock time cost of solving the linear system both directly using Matlab backslash and iteratively by CG solver, reporting also the number of iterations for this latter case. When solving with CG we investigate the performance of the solver with preconditioner $\widehat{\mathbf{P}}$, and compare it with a classical algebraic preconditioner as incomplete Cholesky factorization (ICHOL). Matlab backslash is clearly inefficient, since performing Gaussian

Table 4.2. Parametric domain. Performance of $\widehat{\mathbf{P}}$.

Performance of preconditioner				
n_{el}	N_{dof}	`backslash` (time)	$\widehat{\mathbf{P}}$ (iter / time)	ICHOL (iter / time)
Degree $p = 2$				
8	1000	0.0306	7 / 0.0426	12 / 0.0256
16	5832	0.7974	7 / 0.0659	38 / 0.1338
32	39304	27.46	7 / 0.2992	174 / 3.9153
64	287496	1148	7 / 2.7304	*
Degree $p = 3$				
8	1331	0.0618	8 / 0.0302	10 / 0.0172
16	6859	1.6415	9 / 0.0982	32 / 0.2884
32	42875	118	8 / 0.4703	148 / 8.1482
64	300763	**	8 / 4.7142	*
Degree $p = 4$				
8	1728	0.1587	10 / 0.0427	10 / 0.0548
16	8000	2.6433	10 / 0.2289	28 / 0.4363
32	46656	419	10 / 1.2329	127 / 14.1742
64	314432	**	10 / 9.0219	191 / 176

elimination requires N_{dof}^3 FLOPs. Using classical preconditioners in CG iterative solvers is a reasonable approach for small size problem, but the number of iterations grows with the size of the problem. The performance of the preconditioner $\widehat{\mathbf{P}}$ with CG, is identical among h-refined meshes, and seams reasonably p-robust. The number of iterations never exceeds 10, and the total amount of time required to solve the discrete problem, is always cheaper than the other approaches we tested.

4.6 Conclusions

In this chapter we proposed and studied a space-time least square method for the Schrödinger equation in the framework of isogeometric analysis. Our scheme is based on smooth spline in space and time, that allows, in the particular case of the parametric domain, to introduce a suitable preconditioner for the arising linear system. Our preconditioner $\widehat{\mathbf{P}}$ is represented by a sum of Kronecker products of matrices, that makes the computational cost of its construction (setup) and application, as well as the storage cost, very appealing. In particular the construction of the preconditioner exploits a spectral equivalence between the space matrices $\mathbb{B}_s$ and $\mathbf{L}_s^T\mathbf{M}_s^{-1}\mathbf{L}_s$ that, thanks to the FD technique, admits a stable block-diagonal factorization.

The application cost for a serial execution is almost equal to $O(N_{dof})$, and the block-diagonal structure is suitable for parallel implementation on distributed memory machines, and this will be an interesting future direction of study.

At the same time, the storage cost is roughly the same that we would have by discretizing separately in space and in time, if we assume $N_t \leq Cp^dN_s$. Indeed, in this case the memory used for the whole iterative solver is $O(p^dN_s+N_{dof})$. Although, our approach could be coupled with a matrix-free idea, and this is expected to further improve the efficiency of the overall method.

As a final comment, it would be interesting to further exploits the structure of time pencils, in order to achieve a full factorization of the proposed preconditioner. This may also give a hint in proposing an ad-hoc preconditioner for the isogeometric framework, which we are still working on.

Chapter 5

Final conclusions

In this thesis we focused on two topics, an energy conserving isogeometric discretization for the wave equation in mixed-form and fast preconditioners for space-time formulations of the heat and Schrödinger equation.

For the first part, the proposed semi-discretization in space relies on tensor product projections into spline spaces, with good approximation properties, and that commute with the divergence operator, according to the De Rham complex for splines. The fully discrete problem is obtained applying Crank-Nicolson time stepping method to the semi-discrete form, and is proved to be energy preservative. We proved convergence estimates for the semi-discretization in two working hypothesis: firstly for rigorous assumptions on the projections, and then for more relaxed and practical conditions. The theoretical convergence analysis is covered by numerical results, and energy conservation is confirmed as a key property of the proposed approach.

For the second part of this thesis, we proposed several preconditioners for the fast solution of linear systems arising from space-time isogeometric discretizations of evolutionary equations. The preconditioners are represented by a suitable sum of Kronecker products of matrices, which is essential for a fast application. The main idea relies on the factorization of univariate pencils in the Kronecker structure. This is primarily achieved through the Fast Diagonalization method, except for the time direction where it proves to be unstable. Different stable factorizations in time have been proposed, leading to stable fast solvers for the above mentioned problems.

As a conclusive remark, the innovations of this work have to be investigated in more complicated geometries or multi-patch domains, which is still an open problem, and this certainly represents one of the future directions of development.

Appendix A

Computation of quasi-interpolants

In this section we want to discuss the implementation of $\widehat{\mathfrak{L}\Pi}^1$, as given in (2.3.28). The construction of the projection $\mathfrak{L}\Pi^1$, and the projections in the periodic spaces, are analogous. Let us recall that

$$\widehat{\mathfrak{L}\Pi}^1 = (\widehat{\mathfrak{L}\pi}_{p_1,\Xi_1} \otimes \widehat{\mathfrak{L}\pi}^c_{p_2-1,\Xi_2'}) \times (\widehat{\mathfrak{L}\pi}^c_{p_1-1,\Xi_1'} \otimes \widehat{\mathfrak{L}\pi}_{p_2,\Xi_2})$$

This is applied to project bivariate vector functions $\mathbf{f} = (f_1, f_2)$, that in our practical case are the functions $\widehat{c\mathbf{b}}_{i,h}$ as in Section 2.5. This means that we need to compute $\widehat{\mathfrak{L}\pi}_{p_1,\Xi_1} \otimes \widehat{\mathfrak{L}\pi}^c_{p_2-1,\Xi_2'}(f_1)$ and $\widehat{\mathfrak{L}\pi}^c_{p_1-1,\Xi_1'} \otimes \widehat{\mathfrak{L}\pi}_{p_2,\Xi_2}(f_2)$. With some abuse of notation, these tensorizations must be interpreted as commuting compositions of the kind

$$\widehat{\mathfrak{L}\pi}_{p_1,\Xi_1} \otimes \widehat{\mathfrak{L}\pi}^c_{p_2-1,\Xi_2'}(f_1) = \widehat{\mathfrak{L}\pi}_{p_1,\Xi_1}\big(\widehat{\mathfrak{L}\pi}^c_{p_2,\Xi_2'}(f_1)\big),$$

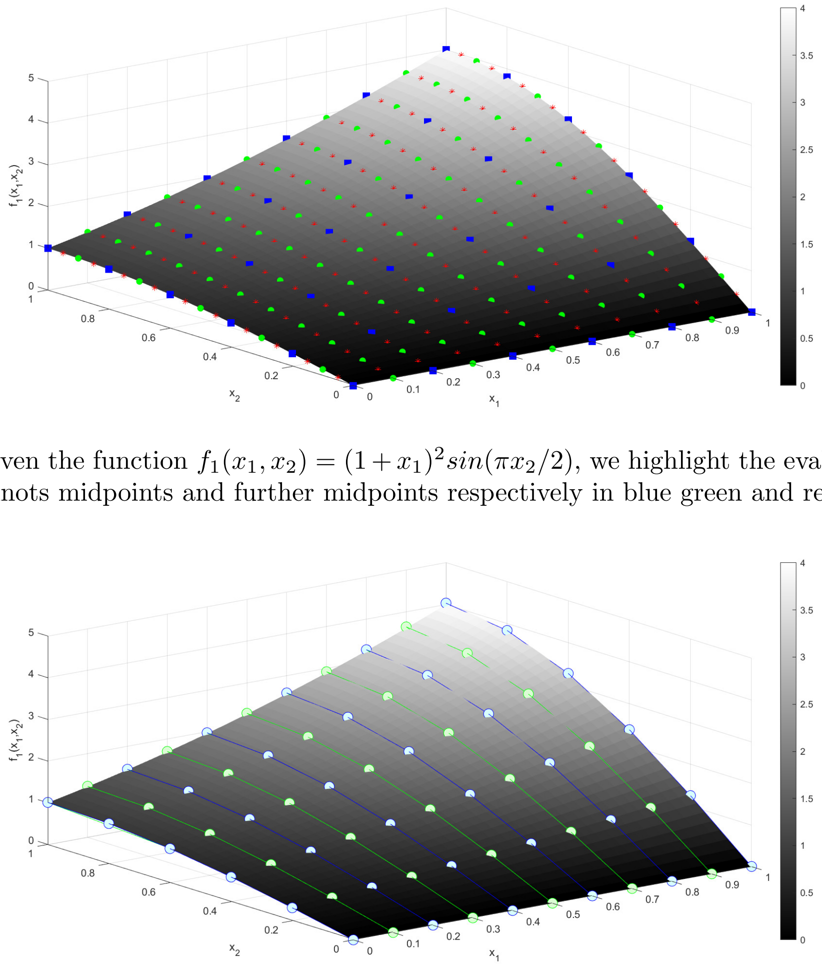
Notice that $\widehat{\mathfrak{L}\pi}^c_{p_2,\Xi_2'}$ projection is applied to f_1 seen as a function of x_2 for a fixed x_1, while the $\widehat{\mathfrak{L}\pi}_{p_1,\Xi_1}$ projection is applied at the result of the previous computation by seeing it as a function of x_1 for any fixed x_2, as it is specified in [12, Section 2.2.2]. We focus on applying $\widehat{\mathfrak{L}\pi}_{p_1,\Xi_1} \otimes \widehat{\mathfrak{L}\pi}^c_{p_2-1,\Xi_2'}(f_1)$, that is:

$$\begin{aligned}
\widehat{\mathfrak{L}\pi}_{p_1,\Xi_1}\left(\widehat{\mathfrak{L}\pi}^c_{p_2,\Xi_2'}\big(f_1(x_1,x_2)\big)\right) &= \widehat{\mathfrak{L}\pi}_{p_1,\Xi_1}\left(\sum_{i_2=1}^{n_2-1} \mu_{i_2}(x_1)\widehat{D}_{i_2}(x_2)\right) \\
&= \sum_{i_2=1}^{n_2-1} \widehat{\mathfrak{L}\pi}_{p_1,\Xi_1}\big(\mu_{i_2}(x_1)\big)\widehat{D}_{i_2}(x_2) \\
&= \sum_{i_2=1}^{n_2-1}\sum_{i_1=1}^{n_1} \mu_{i_1,i_2}\widehat{b}_{i_1}(x_1)\widehat{D}_{i_2}(x_2),
\end{aligned}$$

and the coefficients μ_{i_1,i_2} will depend only on f_1. In order to compute these coefficients μ_{i_1,i_2} we may perform the following steps:

1. First evaluate f_1 on the Cartesian grid given by breakpoints and midpoints of the first univariate direction, and breakpoints, midpoints and further midpoints of the second univariate direction, see Figure A.1a.

2. Use these pointwise evaluations to project $f_1(x_1,\cdot)$ with $\widehat{\mathfrak{L}\pi}^c_{p_2,\Xi_2'}$. This means to perform $\widehat{\mathfrak{L}\pi}^c_{p_2,\Xi_2'}(f_1(x_1,\cdot))$ for every x_1 in the set of breakpoints and midpoint



(a) Given the function $f_1(x_1, x_2) = (1 + x_1)^2 sin(\pi x_2/2)$, we highlight the evaluation over knots midpoints and further midpoints respectively in blue green and red.

(b) Here we see the coefficients $\mu_{i_2}(x_1)_{i_2=1}^{n_2-1}$ in blue and green for knots and midpoint.

Figure A.1. Graphical visualization of intermediate steps in the computation of μ_{i_1,i_2}.

of the first univariate direction. We recall that in our implementation this requires to use Cavalieri-Simpson composite quadrature formula. We end up with a set of coefficients $\{\mu_{i_2}(x_1)\}_{i_2=1}^{n_2-1}$, for every fixed x_1. In Figure A.1b, these coefficients are highlighted in blue and green circles, respectively for breakpoints and midpoints of x_1 direction, for the case $p = 2$.

3. Due to the choice of step 1, for a fixed index i_2, the function $\mu_{i_2}(x_1)$ is already evaluated over the breakpoints and midpoints of x_1 direction. Therefore we can perform a projection $\widehat{\mathfrak{L}\pi}_{p_1,\Xi_1}(\mu_{i_2})$ for every $i_2 = 1, \dots, n_2 - 1$. We end up with a set of coefficients $\{\mu_{i_1,i_2}\}_{i_1=1,i_2=1}^{n_1,n_2-1}$ over the parametric domain that uniquely identify the bivariate spline that approximates f_1.

The procedure to compute $\widehat{\mathfrak{L}\pi}^{c}_{p_1-1,\Xi_1'} \otimes \widehat{\mathfrak{L}\pi}_{p_2,\Xi_2}(f_2)$ is analogous, we just need to swap the evaluation points required per univariate direction. After projecting the two scalar components f_1 and f_2, we have two sets of coefficients whose union corresponds to the degrees of freedom of the spline function that approximates $\mathbf{f}$ in the discrete space $\widehat{X}_h^1$.

Appendix B

Well-posedness of the space-time variational formulation

Here we extend the results presented in [36] on the well posedness of (4.1.2). First we introduce a suitable notation, such that this appendix can be read independently from the previous chapters. Let us recall $\mathcal{Q} = (0,T) \times \Omega$, with $\Omega \subset \mathbb{R}^d$, $d = 1, 2, 3$, while $\Gamma_D = (0,T) \times \partial\Omega$. Consider $\Gamma_0 = \Gamma_D \cup (\{0\} \times \Omega)$ and $\Gamma_T = \Gamma_D \cup (\{T\} \times \Omega)$ and let us define $\mathcal{D}_0 := \left\{\phi \in C_0^\infty(\mathbb{R}^{d+1}) : \phi|_{\Gamma_0} = 0\right\}$, which is the space of smooth functions of $\mathbb{R}^{d+1}$ with compact support such that restricted to $\mathcal{Q}$ satisfy both homogeneous Dirichlet and initial conditions. Analogously define $\mathcal{D}_T := \left\{\psi \in C_0^\infty(\mathbb{R}^{d+1}) : \psi|_{\Gamma_T} = 0\right\}$, that instead satisfies homogeneous Dirichlet and final conditions. Recall $\mathbb{S} := \mathrm{i}\partial_t - \nu\Delta$, and notice that integration by parts gives:

$$\int_\Omega \int_0^T (\mathbb{S}\phi)\,\overline{\psi}\,\mathrm{dt}\ \ \mathrm{d}\Omega = \int_\Omega \int_0^T \phi\,\left(\overline{\mathbb{S}\psi}\right)\ \mathrm{dt}\ \ \mathrm{d}\Omega, \quad \forall\phi \in \mathcal{D}_0,\ \text{and}\ \forall\psi \in \mathcal{D}_T.$$

The space $\mathcal{V}$ in (4.1.2) is the domain of $\mathbb{S} : \mathcal{V} \subset L^2(\mathcal{Q}) \to L^2(\mathcal{Q})$, that can be written as:

$$\mathcal{V} := \left\{v \in L^2(\mathcal{Q}) : \mathbb{S}v \in L^2(\mathcal{Q}) \text{ and } (\mathbb{S}\psi, v) - (\psi, \mathbb{S}v) = 0\ \forall\psi \in \mathcal{D}_T\right\}, \qquad (2.0.1)$$

and we have $\mathcal{C}_0^\infty(\mathcal{Q}) \subset \mathcal{V} \subset L^2(\mathcal{Q})$, that is $\mathbb{S}$ is densely defined. Denoting by $\mathbb{S}^* : \mathcal{V}^* \subset L^2(\mathcal{Q}) \to L^2(\mathcal{Q})$ the adjoint operator, whose domain is given by

$$\mathcal{V}^* := \left\{w \in L^2(\mathcal{Q}) : \exists g \in L^2(\mathcal{Q}) \text{ such that } (\mathbb{S}v, w) = (v, g)\ \forall v \in \mathcal{V}\right\}, \qquad (2.0.2)$$

we have $\mathbb{S}^*w := g$, with $\mathbb{S}^* = \mathbb{S}$, and $\mathcal{C}_0^\infty(\mathcal{Q}) \subset \mathcal{V}^* \subset L^2(\mathcal{Q})$. Notice that we are identifying $L^2(\mathcal{Q})' \equiv L^2(\mathcal{Q})$ through Riesz isomorphism. We endow both $\mathcal{V}$ and $\mathcal{V}^*$ with the norms $\|\cdot\|_{\mathcal{V}}$ and $\|\cdot\|_{\mathcal{V}^*}$ respectively, such that

$$\|v\|_{\mathcal{V}}^2 := \|v\|_{L^2(\mathcal{Q})}^2 + \|\mathbb{S}v\|_{L^2(\mathcal{Q})}^2, \quad \text{and} \quad \|v\|_{\mathcal{V}^*}^2 := \|v\|_{L^2(\mathcal{Q})}^2 + \|\mathbb{S}^*v\|_{L^2(\mathcal{Q})}^2.$$

Define the boundary operators $B : \mathcal{V} \to (\mathcal{V}^*)'$ and $B^* : \mathcal{V}^* \to (\mathcal{V})'$, such that

$$\langle Bv, w\rangle := (\mathbb{S}v, w)_{L^2(\mathcal{Q})} - (v, \mathbb{S}^*w)_{L^2(\mathcal{Q})}, \qquad (2.0.3a)$$

$$\langle B^*w, v\rangle := (\mathbb{S}^*w, v)_{L^2(\mathcal{Q})} - (w, \mathbb{S}v)_{L^2(\mathcal{Q})}, \qquad (2.0.3b)$$

hold true for all $v, w \in L^2(\mathcal{Q})$ such that $\mathbb{S}v, \mathbb{S}^*w \in L^2(\mathcal{Q})$. From [36, Lemma A.2], we have

$$\mathcal{V}^* = (B(\mathcal{V}))^\perp, \tag{2.0.4}$$

and, from (2.0.1), it holds

$$\mathcal{V} = (B^*(\mathcal{D}_T))^\perp. \tag{2.0.5}$$

In particular, from [36, Lemma 2.1], it holds $\mathcal{D}_0 \subset \mathcal{V}$ and $\mathcal{D}_T \subset \mathcal{V}^*$, and in addition we make the following density assumption.

Assumption B.1. *We assume that* $\overline{\mathcal{D}_0}^{\|\cdot\|_\mathcal{V}} = \mathcal{V}$ *and* $\overline{\mathcal{D}_T}^{\|\cdot\|_{\mathcal{V}^*}} = \mathcal{V}^*$.

Under Assumption B.1, from [36, Lemma 2.2], it holds

$$\mathcal{V}^* = (B(\mathcal{V}))^\perp = (B(\mathcal{D}_0))^\perp, \tag{2.0.6}$$

and

$$\mathcal{V} = (B^*(\mathcal{V}^*))^\perp = (B^*(\mathcal{D}_T))^\perp. \tag{2.0.7}$$

The proof of well posedness of Theorem 4.1 is given in [36, Theorem 2.4]. Assumption B.1 is needed to prove injectivity of $\mathbb{S}$. In [36, Theorem 3.1], the verification of the density assumption B.1 has been proved for the case $d = 1$.

We now prove that, Assumption (B.1) is verified for every d-dimensional hypercube Ω, with integer $d \geq 1$.

Lemma B.1. *Given* $\mathcal{Q} = (0, T) \times \Omega$, *with* $\Omega = [0, 1]^d$ *and integer* $d \geq 1$, *then Assumption* (B.1) *holds true.*

Proof. We prove that $\mathcal{D}_0$ is dense in $\mathcal{V}$, the other stated density result is analogous. The case $d = 1$ is in [36, Theorem 3.1], thus we fix an integer $d > 1$. Consider $v \in \mathcal{V}$, first we extend v to the whole $\mathbb{R}^{d+1}$ domain.

1. *Extending along space directions:* Let us extend the space-time domain among the space directions as follows. Denote by $\mathcal{Q}_{1,c} = \mathcal{Q}$, $\mathcal{Q}_{1,l} := [0, T] \times [-1, 0] \times [0, 1]^{d-1}$ and $\mathcal{Q}_{1,r} := [0, T] \times [1, 2] \times [0, 1]^{d-1}$. Analogously, for $i = 2, \dots, d$, we introduce $\mathcal{Q}_{i,c} := \bigcup_{j \in \{l,c,r\}} \mathcal{Q}_{i-1,j}$, then $\mathcal{Q}_{i,l} := [0, T] \times [-1, 2]^{i-1} \times [-1, 0] \times [0, 1]^{d-i}$ and $\mathcal{Q}_{i,r} := [0, T] \times [-1, 2]^{i-1} \times [1, 2] \times [0, 1]^{d-i}$, considering $[0, 1]^0 = \emptyset$. Finally let us call $\Omega_E := \bigcup_{j \in \{l,c,r\}} \mathcal{Q}_{d,j}$ the enlarged space-time cylinder. Then, we introduce the intermediate extension operators, $E_i : \mathcal{Q}_{i,c} \to \mathcal{Q}_{i+1,c}$ for $i = 1, \dots, d-1$, and $E_d : \mathcal{Q}_{d,c} \to \mathcal{Q}_E$, such that

$$E_i f(t, \boldsymbol{x}) := \begin{cases} -f(t, x_1, \dots, -x_i, \dots, x_d) & (t, \boldsymbol{x}) \in \mathcal{Q}_{i,l}, \\ f(t, x_1, \dots, x_i, \dots, x_d) & (t, \boldsymbol{x}) \in \mathcal{Q}_{i,c}, \\ -f(t, x_1, \dots, 2 - x_i, \dots, x_d) & (t, \boldsymbol{x}) \in \mathcal{Q}_{i,r}. \end{cases}$$

We denote the reverse operators by E_i', defined as $E_i' g(t, \boldsymbol{x}) = g(t, \boldsymbol{x}) - g(t, x_1, \dots, -x_i, \dots, x_d) - g(t, x_1, \dots, 2 - x_i, \dots, x_d)$, for $(t, \boldsymbol{x}) \in \mathcal{Q}_{i,c}$, and for $i = 1, \dots, d$. The definitions are to be interpreted almost everywere, and finally our extension operator from $\mathcal{Q}$ to $\mathcal{Q}_E$ is $E := E_d \circ \cdots \circ E_1$, while its reverse operator from $\mathcal{Q}_E$ to $\mathcal{Q}$ is $E' = E_1' \circ \cdots \circ E_d'$. It is easy to see by a change of variable that

$$(Ef, g)_{L^2(\mathcal{Q}_E)} = (f, E'g)_{L^2(\mathcal{Q})}, \quad \forall f \in L^2(\mathcal{Q}), \forall g \in L^2(\mathcal{Q}_E).$$

Next, we claim that

$$\mathbb{S}Ev = E\mathbb{S}v, \quad \forall v \in \mathcal{V}.$$

Clearly, Ev is in $L^2(\mathcal{Q}_E)$ and notice that $E'\mathbb{S}\varphi = \mathbb{S}E'\varphi$ for all $\varphi \in \mathcal{C}_0^\infty(\mathcal{Q}_E)$. Therefore, it holds

$$\begin{aligned}\langle \mathbb{S}Ev, \varphi\rangle_{\mathcal{C}_0^\infty(\mathcal{Q}_E)} &= (Ev, \mathbb{S}\varphi)_{L^2(\mathcal{Q}_E)} = (v, E'\mathbb{S}\varphi)_{L^2(\mathcal{Q})} = \\ &= (v, \mathbb{S}E'\varphi)_{L^2(\mathcal{Q})} = (\mathbb{S}v, E'\varphi)_{L^2(\mathcal{Q})} - \langle Bv, E'\varphi\rangle.\end{aligned}$$

Now, since $E'\varphi|_{\Gamma_T} = 0$, we have $E'\varphi \in \mathcal{V}^*$, and thus $\langle Bv, E'\varphi\rangle = 0$ by (2.0.4). It follows that

$$\langle \mathbb{S}Ev, \varphi\rangle_{\mathcal{C}_0^\infty(\mathcal{Q}_E)} = (\mathbb{S}v, E'\varphi)_{L^2(\mathcal{Q})} = (E\mathbb{S}v, E\varphi)_{L^2(\mathcal{Q}_E)},$$

completing the proof of the claim. We also conclude that $\mathbb{S}Ev$ is in $L^2(\mathcal{Q}_E)$ whenever $v \in \mathcal{V}$.

2. *Extending along time direction:* Let $\widetilde{E}$ denote the extension of E by zero to $\mathbb{R}^{d+1}$, and τ_δ be the translation operator in t direction, i.e., $\tau_\delta w(t, \boldsymbol{x}) = w(t-\delta, \boldsymbol{x})$. From [21] it holds

$$\lim_{\delta\to 0} \|\tau_\delta w - w\|_{L^2(\mathbb{R}^{d+1})} = 0, \quad \forall w \in L^2(\mathbb{R}^{d+1}). \tag{2.0.8}$$

Introducing $\mathcal{Q}_{E,\delta} = (-\delta, T+\delta) \times (-1,2)^d$, by a change of variables, it holds

$$(\tau_\delta \widetilde{E}f, g)_{L^2(\mathcal{Q}_{E,\delta})} = (Ef, \tau_{-\delta}g)_{L^2(\mathcal{Q}_E)}, \quad \forall f \in L^2(\mathcal{Q}), \forall g \in L^2(\mathcal{Q}_{E,\delta}).$$

Denoting by R_δ the restriction operator of function on $\mathbb{R}^{d+1}$ to $\mathcal{Q}_{E,\delta}$, we now claim that

$$\mathbb{S}R_\delta\tau_\delta\widetilde{E}v = R_\delta\tau_\delta\widetilde{E}\mathbb{S}v, \quad \forall v \in \mathcal{V}.$$

The proof is analogous to the one in Step 1. Given $\varphi \in \mathcal{C}_0^\infty(\mathcal{Q}_{E,\delta})$ it holds

$$\begin{aligned}\langle \mathbb{S}R_\delta\tau_\delta\widetilde{E}v, \varphi\rangle_{\mathcal{C}_0^\infty(\mathcal{Q}_{E,\delta})} &= (\tau_\delta\widetilde{E}v, \mathbb{S}\varphi)_{L^2(\mathcal{Q}_{E,\delta})} = (Ev, \mathbb{S}\tau_{-\delta}\varphi)_{L^2(\mathcal{Q}_E)} \\ &= (v, E'\mathbb{S}\tau_{-\delta}\varphi)_{L^2(\mathcal{Q})} = (v, \mathbb{S}E'\tau_{-\delta}\varphi)_{L^2(\mathcal{Q})} \\ &= (\mathbb{S}v, E'\tau_{-\delta}\varphi)_{L^2(\mathcal{Q})} - \langle Bv, E'\tau_{-\delta}\varphi\rangle.\end{aligned}$$

Now, since $E'\tau_{-\delta}\varphi|_{\Gamma_T} = 0$, we have $E'\tau_{-\delta}\varphi \in \mathcal{V}^*$, and thus $\langle Bv, E'\tau_{-\delta}\varphi\rangle = 0$ by (2.0.4). It follows that

$$\langle \mathbb{S}R_\delta\tau_\delta\widetilde{E}v, \varphi\rangle_{\mathcal{C}_0^\infty(\mathcal{Q}_{E,\delta})} = (\widetilde{E}\mathbb{S}v, \tau_{-\delta}\varphi)_{L^2(\mathcal{Q}_E)} = (\tau_{-\delta}\widetilde{E}\mathbb{S}v, \varphi)_{L^2(\mathcal{Q}_{E,\delta})},$$

which proves the claim.

3. *Mollify:* Consider the mollifier $\rho_\epsilon \in \mathcal{C}_0^\infty(\mathbb{R}^{d+1})$, defined by

$$\rho_\epsilon(t, \boldsymbol{x}) := \epsilon^{-d-1}\rho_1(\epsilon^{-1}t, \epsilon^{-1}x_1, \ldots, \epsilon^{-1}x_d), \quad \text{for} \epsilon > 0,$$

where

$$\rho_1(t, \boldsymbol{x}) := \begin{cases} ke^{-1/(1-|(t,\boldsymbol{x})|^2)}, & \text{if } |(t,\boldsymbol{x})|^2 < 1, \\ 0 & \text{if } |(t,\boldsymbol{x})|^2 \geq 1, \end{cases}$$

with $|\cdot|$ denoting the Euclidean norm in $\mathbb{R}^{d+1}$, and k is a constant chosen such that $\int_{\mathbb{R}^{d+1}} \rho_1 = 1$. Notice that, given $\delta > 0$ small enough, i.e., $\delta < \min\{T/2, 1/2\}$, the convolutions $v_\epsilon := \rho_\epsilon * \tau_\delta \widetilde{E} v$ and $s_\epsilon := \rho_\epsilon * \tau_\delta \widetilde{E} \mathbb{S} v$ are smooth functions that satisfy

$$\lim_{\epsilon\to 0} \|v_\epsilon - \tau_\delta \widetilde{E} v\|_{L^2(\mathbb{R}^{d+1})} = 0, \quad \text{and} \quad \lim_{\epsilon\to 0} \|s_\epsilon - \tau_\delta \widetilde{E}\mathbb{S} v\|_{L^2(\mathbb{R}^{d+1})} = 0. \tag{2.0.9}$$

Moreover, the smooth function $\mathbb{S}v_\epsilon$ need not coincide to s_ϵ everywere, but they coincide on $\mathcal{Q}$ whenever $\epsilon < \delta/2$. Thus, consider $\delta = 3\epsilon$, and let $\epsilon < \min\{T/6, 1/6\}$ go to zero. We have

$$\|\mathbb{S}v_\epsilon - \mathbb{S}v\|_{L^2(\mathcal{Q})} = \|s_\epsilon - \mathbb{S}v\|_{L^2(\mathcal{Q})} \leq \|s_\epsilon - \tau_\delta \widetilde{E}\mathbb{S}v\|_{L^2(\mathbb{R}^{d+1})} + \|\tau_\delta \widetilde{E}\mathbb{S}v - \widetilde{E}\mathbb{S}v\|_{L^2(\mathbb{R}^{d+1})}$$

and

$$\|v_\epsilon - v\|_{L^2(\mathcal{Q})} \leq \|v_\epsilon - \tau_\delta \widetilde{E} v\|_{L^2(\mathcal{Q})} + \|\tau_\delta \widetilde{E} v - \widetilde{E} v\|_{L^2(\mathcal{Q})}$$

Using (2.0.8) and (2.0.9), it follows that

$$\lim_{\epsilon\to 0} \|v_\epsilon - v\|_{\mathcal{V}} = 0.$$

To conclude, we examine the value of v_ϵ at the edges of the space-time cylinder. We have

$$v_\epsilon(t, 0, x_2, \ldots, x_d) = \int_{\mathbb{R}^{d+1}} \rho_\epsilon(t-\sigma, -r_1, x_2 - r_2, \ldots, x_d - r_d) \tau_\delta \widetilde{E} v(\sigma, r_1, \ldots, r_d)\, \mathrm{d}r_1 \ldots\, \mathrm{d}r_d\, \mathrm{d}\sigma,$$

with the integrand in the inner integral being the product of an even function ρ_ϵ, with respect to r_1, and an odd function $\tau_\delta \widetilde{E} v$ of r_1. Thus $v_\epsilon(t, 0, x_2, \ldots, x_d) = 0$ and the same holds for $v_\epsilon(t, 1, x_2, \ldots, x_d)$ and the other univariate space directions. Moreover since $\tau_\delta \widetilde{E} v$ is identically zero in a neighborhood of $(0, \boldsymbol{x})$, we conclude that $v_\epsilon|_{\Gamma_0} = 0$.

□

Next we extend this result to smooth parameterizations of Ω.

Theorem B.2. *Given $\mathcal{Q} = (0, T) \times \Omega$, with $\Omega = \boldsymbol{F}([0,1]^d)$, $\boldsymbol{F} : [0,1]^d \to \Omega$ smooth diffeomorphism, and integer $d \geq 1$, then Assumption* (B.1) *holds true.*

Proof. We prove that $\mathcal{D}_0$ is dense in $\mathcal{V}$, the other stated density result is analogous. Given $v \in \mathcal{V}$, recall $\boldsymbol{G} : [0,1]^{d+1} \to \mathcal{Q}$ is the parameterization of the space-time cilinder, such that $\boldsymbol{G}(\tau, \boldsymbol{\eta}) := (T\tau, \boldsymbol{F}(\boldsymbol{\eta})) = (t, \boldsymbol{x})$. Define $\widehat{v} := v \circ \boldsymbol{G}$. Clearly $\widehat{v} \in L^2([0,1]^{d+1})$, and $\mathbb{S}\widehat{v} \in L^2([0,1]^{d+1})$. Moreover $(\mathbb{S}v, \phi)_{L^2([0,1]^{d+1})} - (v, \mathbb{S}\phi)_{L^2([0,1]^{d+1})} = 0$, for all $\phi \in \mathcal{C}_0^\infty(\mathbb{R}^{d+1})$ such that $\phi|_{\boldsymbol{G}^{-1}(\Gamma_T)} = 0$. By applying Lemma 1, it exists $\{\widehat{v}_\epsilon\}_{\epsilon>0} \subset \mathcal{C}_0^\infty(\mathbb{R}^{d+1})$ such that $\widehat{v}_\epsilon|_{\boldsymbol{G}^{-1}(\Gamma_0)} = 0$, that satisfies

$$\lim_{\epsilon\to 0} \|\widehat{v}_\epsilon - \widehat{v}\|^2_{L^2([0,1]^{d+1})} + \|\mathbb{S}\widehat{v}_\epsilon - \mathbb{S}\widehat{v}\|^2_{L^2([0,1]^{d+1})} = 0.$$

Therefore define $v_\epsilon := \widehat{v}_\epsilon \circ \boldsymbol{G}^{-1}$ and notice that $\{v_\epsilon\}_{\epsilon>0} \subset \mathcal{D}_0$. Moreover, by a change of variable,

$$\lim_{\epsilon\to 0} \|v_\epsilon - v\|^2_{\mathcal{V}} \leq C \lim_{\epsilon\to 0} \|\widehat{v}_\epsilon - \widehat{v}\|^2_{L^2([0,1]^{d+1})} + \|\mathbb{S}\widehat{v}_\epsilon - \mathbb{S}\widehat{v}\|^2_{L^2([0,1]^{d+1})} = 0.$$

This completes the proof. □

Bibliography

[1] Douglas N. Arnold, Richard S. Falk, and Ragnar Winther. Finite element exterior calculus, homological techniques, and applications. *Acta Numerica*, 15:1–155, 2006.

[2] Douglas N. Arnold and Jeonghun J. Lee. Mixed methods for elastodynamics with weak symmetry. *SIAM Journal on Numerical Analysis*, 52(6):2743–2769, 2014.

[3] Ferdinando Auricchio, Lourenço Beirão da Veiga, Carlo Lovadina, and Alessandro Reali. The importance of the exact satisfaction of the incompressibility constraint in nonlinear elasticity: mixed FEMs versus NURBS-based approximations. *Computer Methods in Applied Mechanics and Engineering*, 199(5-8):314–323, 2010.

[4] Yuri Bazilevs, Lourenço Beirão da Veiga, John A. Cottrell, Thomas J.R. Hughes, and Giancarlo Sangalli. Isogeometric analysis: approximation, stability and error estimates for h-refined meshes. *Mathematical Models and Methods in Applied Sciences*, 16(07):1031–1090, 2006.

[5] Yuri Bazilevs, Victor M. Calo, John A. Cottrell, Thomas J. R. Hughes, Alessandro Reali, and Guglielmo Scovazzi. Variational multiscale residual-based turbulence modeling for large eddy simulation of incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 197(1-4):173–201, 2007.

[6] Yuri Bazilevs, Victor M. Calo, Thomas J. R. Hughes, and Yongjie Zhang. Isogeometric fluid-structure interaction: theory, algorithms, and computations. *Computational mechanics*, 43:3–37, 2008.

[7] Yuri Bazilevs, Ming-Chen Hsu, Josef Kiendl, Roland Wüchner, and Kai-Uwe Bletzinger. 3D simulation of wind turbine rotors at full scale. Part II: Fluid–structure interaction modeling with composite blades. *International Journal for Numerical Methods in Fluids*, 65(1-3):236–253, 2011.

[8] Yuri Bazilevs and Thomas J. R. Hughes. NURBS-based isogeometric analysis for the computation of flows about rotating components. *Computational Mechanics*, 43:143–150, 2008.

[9] Yuri Bazilevs, Christian Michler, Victor M. Calo, and Thomas J. R. Hughes. Isogeometric variational multiscale modeling of wall-bounded turbulent flows with weakly enforced boundary conditions on unstretched meshes. *Computer Methods in Applied Mechanics and Engineering*, 199(13):780–790, 2010.

[10] Eliane Bécache, Patrick Joly, and Chrysoula Tsogka. An analysis of new mixed finite elements for the approximation of wave propagation problems. *SIAM Journal on Numerical Analysis*, 37(4):1053–1084, 2000.

[11] Lourenço Beirão da Veiga, Annalisa Buffa, Carlo Lovadina, Massimiliano Martinelli, and Giancarlo Sangalli. An isogeometric method for the Reissner–Mindlin plate bending problem. *Computer Methods in Applied Mechanics and Engineering*, 209:45–53, 2012.

[12] Lourenço Beirão da Veiga, Annalisa Buffa, Giancarlo Sangalli, and Rafael Vázquez. Mathematical analysis of variational isogeometric methods. *Acta Numerica*, 23:157, 2014.

[13] Lourenço Beirão da Veiga, Annalisa Buffa, Giancarlo Sangalli, and Rafael Vázquez. *An introduction to the numerical analysis of isogeometric methods*, volume 9, pages 3–69. SEMA SIMAI Springer Series, 07 2016.

[14] Lourenço Beirão da Veiga, Durkbin Cho, and Giancarlo Sangalli. Anisotropic NURBS approximation in isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 209:1–11, 2012.

[15] David J. Benson, Yuri Bazilevs, Ming-Chen Hsu, and Thomas J. R. Hughes. A large deformation, rotation-free, isogeometric shell. *Computer Methods in Applied Mechanics and Engineering*, 200(13-16):1367–1378, 2011.

[16] Silvia Bertoluzza. The discrete commutator property of approximation spaces. *Comptes Rendus de l'Académie des Sciences-Series I-Mathematics*, 329(12):1097–1102, 1999.

[17] Daniele Boffi, Franco Brezzi, and Michel Fortin. *Mixed finite element methods and applications*, volume 44. Springer, 2013.

[18] Daniele Boffi, Annalisa Buffa, and Lucia Gastaldi. Convergence analysis for hyperbolic evolution problems in mixed form. *Numerical Linear Algebra with Applications*, 20(4):541–556, 2013.

[19] Jesús Bonilla and Santiago Badia. Maximum-principle preserving space–time isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 354:422–440, 2019.

[20] Andrea Bressan and Espen Sande. Approximation in FEM, DG and IGA: a theoretical comparison. *Numerische Mathematik*, 2019.

[21] Haim Brezis. *Functional analysis, Sobolev spaces and partial differential equations*, volume 2. Springer, 2011.

[22] John C. Bruch Jr. and George Zyvoloski. Transient two-dimensional heat conduction problems solved by the finite element method. *International Journal for Numerical Methods in Engineering*, 8(3):481–494, 1974.

[23] Annalisa Buffa, Carlo De Falco, and Giancarlo Sangalli. Isogeometric analysis: stable elements for the 2D Stokes equation. *International Journal for Numerical Methods in Fluids*, 65(11-12):1407–1422, 2011.

[24] Annalisa Buffa, Judith Rivas, Giancarlo Sangalli, and Rafael Vázquez. Isogeometric discrete differential forms in three dimensions. *SIAM Journal on Numerical Analysis*, 49(2):818–844, 2011.

[25] Annalisa Buffa, Giancarlo Sangalli, and Christoph Schwab. Exponential convergence of the *hp* version of isogeometric analysis in 1D. In *Spectral and High Order Methods for Partial Differential Equations-ICOSAHOM 2012: Selected papers from the ICOSAHOM conference, June 25-29, 2012, Gammarth, Tunisia*, pages 191–203. Springer, 2013.

[26] Annalisa Buffa, Giancarlo Sangalli, and Rafael Vázquez. Isogeometric analysis in electromagnetics: B-splines approximation. *Computer Methods in Applied Mechanics and Engineering*, 199(17-20):1143–1152, 2010.

[27] Annalisa Buffa, Giancarlo Sangalli, and Rafael Vázquez. Isogeometric methods for computational electromagnetics: B-spline and T-spline discretizations. *Journal of Computational Physics*, 257:1291–1320, 2014.

[28] Jesse Chan and John A. Evans. Multi-patch discontinuous Galerkin isogeometric analysis for wave propagation: Explicit time-stepping and efficient mass matrix inversion. *Computer Methods in Applied Mechanics and Engineering*, 333:22–54, 2018.

[29] John A. Cottrell, Thomas J. R. Hughes, and Yuri Bazilevs. *Isogeometric analysis: Toward integration of CAD and FEA*. John Wiley & Sons, 2009.

[30] John A. Cottrell, Alessandro Reali, Yuri Bazilevs, and Thomas J. R. Hughes. Isogeometric analysis of structural vibrations. *Computer Methods in Applied Mechanics and Engineering*, 195(41-43):5257–5296, 2006.

[31] Nicolas Crouseilles, Ahmed Ratnani, and Eric Sonnendrücker. An isogeometric analysis approach for the study of the gyrokinetic quasi-neutrality equation. *Journal of Computational Physics*, 231(2):373–393, 2012.

[32] Robert Dautray and Jacques L. Lions. *Evolution Problems 1, Mathematical Analysis and Numerical Methods for Science and Technology, vol. 5*. Springer, Berlin, 2000.

[33] Carl De Boor. *A practical guide to splines*, volume 27. Springer-Verlag New York, 1978.

[34] Carl De Boor. *A practical guide to splines (revised edition)*. Applied Mathematical Sciences. Springer, Berlin, 2001.

[35] Carl De Boor and George J. Fix. Spline approximation by quasiinterpolants. *Journal of Approximation Theory*, 8(1):19–45, 1973.

[36] Leszek Demkowicz, Jayadeep Gopalakrishnan, Sriram Nagaraj, and Paulina Sepulveda. A spacetime DPG method for the Schrödinger equation. *SIAM Journal on Numerical Analysis*, 55(4):1740–1759, 2017.

[37] Michael O. Deville, Paul F. Fischer, and Ernest H. Mund. *High-order methods for incompressible fluid flow*. Cambridge University Press, 2002.

[38] Carlos A. Dorao and Hugo A. Jakobsen. A parallel time–space least-squares spectral element solver for incompressible flow problems. *Applied Mathematics and Computation*, 185(1):45–58, 2007.

[39] Ralph Echter and Manfred Bischoff. Numerical efficiency, locking and unlocking of NURBS finite elements. *Computer Methods in Applied Mechanics and Engineering*, 199(5-8):374–382, 2010.

[40] Thomas Elguedj, Yuri Bazilevs, Victor M. Calo, and Thomas J. R. Hughes. B and F projection methods for nearly incompressible linear and non-linear elasticity and plasticity using higher-order NURBS elements. *Computer Methods in Applied Mechanics and Engineering*, 197(33-40):2732–2762, 2008.

[41] John A. Evans, Yuri Bazilevs, Ivo Babuška, and Thomas J. R. Hughes. n-widths, sup-infs, and optimality ratios for the k-version of the isogeometic finite element method. *Computer Methods in Applied Mechanics and Engineering*, 198:1726–1741, 2009.

[42] John A. Evans and Thomas J. R. Hughes. Isogeometric divergence-conforming B-splines for the Darcy–Stokes–Brinkman equations. *Mathematical Models and Methods in Applied Sciences*, 23(04):671–741, 2013.

[43] John A. Evans and Thomas J. R. Hughes. Isogeometric divergence-conforming B-splines for the steady Navier–Stokes equations. *Mathematical Models and Methods in Applied Sciences*, 23(08):1421–1478, 2013.

[44] John A. Evans and Thomas J. R. Hughes. Isogeometric divergence-conforming B-splines for the unsteady Navier–Stokes equations. *Journal of Computational Physics*, 241:141–167, 2013.

[45] Lawrence C. Evans. *Partial Differential equations*. American Mathematical Society, Berlin, 2010.

[46] Sara Fraschini, Gabriele Loli, Andrea Moiola, and Giancarlo Sangalli. An unconditionally stable space-time isogeometric method for the acoustic wave equation. *arXiv preprint arXiv:2303.07268*, 2023.

[47] Isaac Fried. Finite-element analysis of time-dependent phenomena. *AIAA Journal*, 7(6):1170–1173, 1969.

[48] Krishan P. S. Gahalaut, Satyendra K. Tomar, and Craig C. Douglas. Condition number estimates for matrices arising in NURBS based isogeometric discretizations of elliptic partial differential equations. *arXiv preprint arXiv:1406.6808*, 2014.

[49] Martin J. Gander. 50 years of time parallel time integration. In *Multiple Shooting and Time Domain Decomposition Methods*, pages 69–113. Springer, 2015.

[50] Martin J. Gander and Martin Neumüller. Analysis of a new space-time parallel multigrid algorithm for parabolic problems. *SIAM Journal on Scientific Computing*, 38(4):A2173–A2208, 2016.

[51] Martin J. Gander and Davide Palitta. A new paradiag time-parallel time integration method. *SIAM Journal on Scientific Computing*, 46(2):A697–A718, 2024.

[52] Longfei Gao and Victor M. Calo. Fast isogeometric solvers for explicit dynamics. *Computer Methods in Applied Mechanics and Engineering*, 274:19–41, 2014.

[53] Roland Glowinski and Serguei Lapin. Solution of a wave equation by a mixed finite element-fictitious domain method. *Computational Methods in Applied Mathematics*, 4(4):431–444, 2004.

[54] Héctor Gómez, Victor M. Calo, Yuri Bazilevs, and Thomas J. R. Hughes. Isogeometric analysis of the Cahn–Hilliard phase-field model. *Computer Methods in Applied Mechanics and Engineering*, 197(49-50):4333–4352, 2008.

[55] Héctor Gómez, Thomas J. R. Hughes, Xesús Nogueira, and Victor M. Calo. Isogeometric analysis of the isothermal Navier–Stokes–Korteweg equations. *Computer Methods in Applied Mechanics and Engineering*, 199(25-28):1828–1840, 2010.

[56] Sergio Gómez and Andrea Moiola. A space-time Trefftz discontinuous Galerkin method for the linear Schrödinger equation. *SIAM Journal on Numerical Analysis*, 60(2):688–714, 2022.

[57] Stefan Hain and Karsten Urban. An ultra-weak space-time variational formulation for the Schrödinger equation. *arXiv preprint arXiv:2212.14398*, 2022.

[58] Julian Henning, Davide Palitta, Valeria Simoncini, and Karsten Urban. An ultraweak space-time variational formulation for the wave equation: Analysis and efficient numerical solution. *ESAIM: Mathematical Modelling and Numerical Analysis*, 56(4):1173–1198, 2022.

[59] Ralf Hiptmair. Finite elements in computational electromagnetism. *Acta Numerica*, 11:237–339, 2002.

[60] Christoph Hofer, Ulrich Langer, Martin Neumüller, and Rainer Schneckenleitner. Parallel and robust preconditioning for space-time isogeometric analysis of parabolic evolution problems. *SIAM Journal on Scientific Computing*, 41(3):A1793–A1821, 2019.

[61] Clemens Hofreither. A black-box low-rank approximation algorithm for fast matrix assembly in isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 333:311–330, 2018.

[62] Florian Holderied, Stefan Possanner, and Xin Wang. MHD-kinetic hybrid code based on structure-preserving finite elements with particles-in-cell. *Journal of Computational Physics*, 433:110143, 2021.

[63] Thomas J. R. Hughes, John A. Cottrell, and Yuri Bazilevs. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering*, 194(39):4135–4195, 2005.

[64] Ohannes Karakashian and Charalambos Makridakis. A space-time finite element method for the nonlinear Schrödinger equation: the discontinuous Galerkin method. *Mathematics of computation*, 67(222):479–499, 1998.

[65] Josef Kiendl, Yuri Bazilevs, Ming-Chen Hsu, Roland Wüchner, and Kai-Uwe Bletzinger. The bending strip method for isogeometric analysis of Kirchhoff–Love shell structures comprised of multiple patches. *Computer Methods in Applied Mechanics and Engineering*, 199(37-40):2403–2416, 2010.

[66] Josef Kiendl, Kai-Uwe Bletzinger, Johannes Linhard, and Roland Wüchner. Isogeometric shell analysis with Kirchhoff–Love elements. *Computer Methods in Applied Mechanics and Engineering*, 198(49-52):3902–3914, 2009.

[67] Tamara G. Kolda and Brett W. Bader. Tensor decompositions and applications. *SIAM review*, 51(3):455–500, 2009.

[68] Michael Kraus, Katharina Kormann, Philip J. Morrison, and Eric Sonnendrücker. GEMPIC: Geometric electromagnetic particle-in-cell methods. *Journal of Plasma Physics*, 83(4), 2017.

[69] Alen Kushova. Energy conservative isogeometric techniques for the wave equation. Master's thesis, Università degli studi di Pavia, September 2020.

[70] Arne M. Kvarving and Einar M. Rønquist. A fast tensor-product solver for incompressible fluid flow in partially deformed three-dimensional domains: Parallel implementation. *Computers & Fluids*, 52:22–32, 2011.

[71] Ulrich Langer, Stephen E. Moore, and Martin Neumüller. Space–time isogeometric analysis of parabolic evolution problems. *Computer Methods in Applied Mechanics and Engineering*, 306:342 – 363, 2016.

[72] Ulrich Langer, Martin Neumüller, and Ioannis Toulopoulos. Multipatch space-time isogeometric analysis of parabolic diffusion problems. In *International Conference on Large-Scale Scientific Computing*, pages 21–32. Springer, 2017.

[73] Byung-Gook Lee, Tom Lyche, and Knut Mørken. Some examples of quasi-interpolants constructed from local spline projectors. *Mathematical Methods for Curves and Surfaces: Oslo*, pages 243–252, 2000.

[74] Scott Lipton, John A. Evans, Yuri Bazilevs, Thomas Elguedj, and Thomas J. R. Hughes. Robustness of isogeometric structural discretizations under severe mesh distortion. *Computer Methods in Applied Mechanics and Engineering*, 199(5-8):357–373, 2010.

[75] Gabriele Loli, Monica Montardini, Giancarlo Sangalli, and Mattia Tani. An efficient solver for space–time isogeometric Galerkin methods for parabolic problems. *Computers & Mathematics with Applications*, 80(11):2586–2603, 2020.

[76] Gabriele Loli, Giancarlo Sangalli, and Mattia Tani. Easy and efficient preconditioning of the isogeometric mass matrix. *Computers & Mathematics with Applications*, 116:245–264, 2022.

[77] Tom Lyche and Knut Mørken. Spline methods. Draft. *Department of Informatics, Center of Mathematics for Applications, University of Oslo, Oslo*, 2008.

[78] Tom Lyche and Larry L. Schumaker. Local spline approximation methods. *Journal of Approximation Theory*, 15(4):294–325, 1975.

[79] Robert E. Lynch, John R. Rice, and Donald H. Thomas. Direct solution of partial difference equations by tensor product methods. *Numerische Mathematik*, 6(1):185–199, 1964.

[80] Yvon Maday and Einar M. Rønquist. Parallelization in time through tensor-product space–time solvers. *Comptes Rendus. Mathématique*, 346(1-2):113–118, 2008.

[81] Angelos Mantzaflaris, Bert Jüttler, Boris N. Khoromskij, and Ulrich Langer. Matrix generation in isogeometric analysis by low rank tensor approximation. In *Curves and Surfaces: 8th International Conference, Paris, France, June 12-18, 2014, Revised Selected Papers 8*, pages 321–340. Springer, 2015.

[82] Angelos Mantzaflaris, Bert Jüttler, Boris N. Khoromskij, and Ulrich Langer. Low rank tensor methods in Galerkin-based isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 316:1062–1085, 2017.

[83] Angelos Mantzaflaris, Felix Scholz, and Ioannis Toulopoulos. Low-rank space-time decoupled isogeometric analysis for parabolic problems with varying coefficients. *Computational Methods in Applied Mathematics*, 19(1):123–136, 2019.

[84] Peter Monk. *Finite element methods for Maxwell's equations.* Oxford University Press, 2003.

[85] Monica Montardini, Matteo Negri, Giancarlo Sangalli, and Mattia Tani. Space–time least–squares isogeometric method and efficient solver for parabolic problems. *Mathematics of Computation*, 89(323):1193–1227, 2020.

[86] Monica Montardini, Filippo Remonato, and Giancarlo Sangalli. Isogeometric methods for free boundary problems. In *Conference on Isogeometric Analysis and Applications*, pages 131–155. Springer, 2018.

[87] Monica Montardini, Giancarlo Sangalli, and Mattia Tani. Robust isogeometric preconditioners for the Stokes system based on the Fast Diagonalization method. *Computer Methods in Applied Mechanics and Engineering*, 338:162 – 185, 2018.

[88] John Tinsley Oden. A general theory of finite elements. I. Topological considerations. *International Journal for Numerical Methods in Engineering*, 1(2):205–221, 1969.

[89] John Tinsley Oden. A general theory of finite elements. II. Applications. *International Journal for Numerical Methods in Engineering*, 1(3):247–259, 1969.

[90] Peter Oswald and Barbara Wohlmuth. On polynomial reproduction of dual FE bases. In *Thirteenth international conference on domain decomposition methods*, pages 85–96, 2001.

[91] Ahmed Ratnani and Eric Sonnendrücker. An arbitrary high-order spline finite element solver for the time domain maxwell equations. *Journal of Scientific Computing*, 51(1):87–106, 2012.

[92] Christelle Saadé, Stéphane Lejeunes, Dominique Eyheramendy, and Roy Saad. Space-time isogeometric analysis for linear and non-linear elastodynamics. *Computers & Structures*, 254:106594, 2021.

[93] Giancarlo Sangalli and Mattia Tani. Isogeometric preconditioners based on fast solvers for the Sylvester equation. *SIAM Journal on Scientific Computing*, 38(6):A3644–A3671, 2016.

[94] Giancarlo Sangalli and Mattia Tani. Matrix-free weighted quadrature for a computationally efficient isogeometric k-method. *Computer Methods in Applied Mechanics and Engineering*, 338:117 – 133, 2018.

[95] Larry L. Schumaker. *Spline functions: basic theory.* Cambridge university press, 2007.

[96] Christoph Schwab and Rob Stevenson. Space-time adaptive wavelet methods for parabolic evolution problems. *Mathematics of Computation*, 78(267):1293–1318, 2009.

[97] Farzin Shakib and Thomas J. R. Hughes. A new finite element formulation for computational fluid dynamics: IX. Fourier analysis of space-time Galerkin/least-squares algorithms. *Computer Methods in Applied Mechanics and Engineering*, 87(1):35–58, 1991.

[98] Laurent Sorber, Marc Van Barel, and Lieven De Lathauwer. Tensorlab v2. 0. *Available online, URL: www.tensorlab.net*, 2014.

[99] Olaf Steinbach. Space-time finite element methods for parabolic problems. *Computational Methods in Applied Mathematics*, 15(4):551–566, 2015.

[100] Rob Stevenson and Jan Westerdiep. Stability of Galerkin discretizations of a mixed space-time variational formulation of parabolic evolution equations. *arXiv:1902.06279*, 2019.

[101] Kenji Takizawa and Tayfun E. Tezduyar. Space–time computation techniques with continuous representation in time (ST-C). *Computational Mechanics*, 53(1):91–99, 2014.

[102] Kenji Takizawa, Tayfun E. Tezduyar, Yuto Otoguro, Takuya Terahara, Takashi Kuraishi, and Hitoshi Hattori. Turbocharger flow computations with the space–time isogeometric analysis (ST-IGA). *Computers & Fluids*, 142:15–20, 2017.

[103] Kenji Takizawa, Tayfun E. Tezduyar, and Takuya Terahara. Ram-air parachute structural and fluid mechanics computations with the space–time isogeometric analysis (ST-IGA). *Computers & Fluids*, 141:191–200, 2016.

[104] Kenji Takizawa, Tayfun E. Tezduyar, Takuya Terahara, and Takafumi Sasaki. Heart valve flow computation with the space–time slip interface topology change (ST-SI-TC) method and isogeometric analysis (IGA). In *Biomedical Technology*, pages 77–99. Springer, 2018.

[105] Anton Tkachuk and Manfred Bischoff. Direct and sparse construction of consistent inverse mass matrices: general variational formulation and application to selective mass scaling. *International Journal for Numerical Methods in Engineering*, 101(6):435–469, 2015.

[106] Yuki Ueda and Norikazu Saito. Stability and error estimates for the successive-projection technique with B-splines in time. *Journal of Computational and Applied Mathematics*, 358:266 – 278, 2019.

[107] Rafael Vázquez. A new design for the implementation of isogeometric analysis in Octave and Matlab: GeoPDEs 3.0. *Computers & Mathematics with Applications*, 72(3):523–554, 2016.

[108] Rafael Vázquez, Annalisa Buffa, and Luca Di Rienzo. NURBS-based BEM implementation of high-order surface impedance boundary conditions. *IEEE transactions on magnetics*, 48(12):4757–4766, 2012.

[109] Rafael Vázquez, Annalisa Buffa, and Luca Di Rienzo. Isogeometric FEM implementation of high-order surface impedance boundary conditions. *IEEE transactions on magnetics*, 50(6):1–8, 2014.

[110] Wolfgang A. Wall, Moritz A. Frenzel, and Christian Cyron. Isogeometric structural shape optimization. *Computer Methods in Applied Mechanics and Engineering*, 197(33-40):2976–2988, 2008.

[111] Linus Wunderlich, Alexander Seitz, Mert D. Alaydın, Barbara Wohlmuth, and Alexander Popp. Biorthogonal splines for optimal weak patch-coupling in isogeometric analysis with applications to finite deformation elasticity. *Computer Methods in Applied Mechanics and Engineering*, 346:197–215, 2019.