Thesis for the degree of Doctor of Philosophy

Cut Finite Element Methods on Overlapping Meshes: Analysis and Applications

Carl Lundholm

Department of Mathematical Sciences
Chalmers University of Technology and University of Gothenburg
Gothenburg, Sweden 2021

Cut Finite Element Methods on Overlapping Meshes: Analysis and Applications
Carl Lundholm

© Carl Lundholm, 2021

ISBN 978-91-7905-519-6
Doktorsavhandlingar vid Chalmers tekniska högskola, Ny serie nr 4986
ISSN 0346-718X

Department of Mathematical Sciences
Chalmers University of Technology and University of Gothenburg
SE-412 96 Gothenburg, Sweden
Tel: +46 (0)31 772 1000

Author e-mail: carlun@chalmers.se

Printed by Chalmers digitaltryck
Gothenburg, Sweden 2021

Cover image: Collage of images from the various works presented in the thesis.

Cut Finite Element Methods on Overlapping Meshes: Analysis and Applications

Carl Lundholm

Department of Mathematical Sciences
Chalmers University of Technology and University of Gothenburg

Abstract

This thesis deals with both analysis and applications of cut finite element methods (CutFEMs) on overlapping meshes. By overlapping meshes we mean a mesh hierarchy with a background mesh at the bottom and one or more overlapping meshes that are stacked on top of it. Overlapping meshes can be used as an alternative to costly remeshing for problems with changing geometry. The main content of the thesis is the five appended papers. The thesis consists of an analysis part and an applications part.

In the analysis part (Paper I and Paper II), we consider cut finite element methods on overlapping meshes for a *time-dependent* parabolic model problem: the heat equation on two overlapping meshes, where one mesh is allowed to move around on top of the other. In Paper I, the overlapping mesh is prescribed a cG(1) movement, meaning that its location as a function of time is *continuous* and *piecewise linear*. The cG(1) mesh movement results in a space-time discretization for which existing analysis methodologies either fail or are unsuitable. We therefore propose, to the best of our knowledge, a new energy analysis framework that is general enough to be applicable to the current setting. In Paper II, the overlapping mesh is prescribed a dG(0) movement, meaning that its location as a function of time is *discontinuous* and *piecewise constant*. The dG(0) mesh movement results in a space-time discretization for which existing analysis methodologies work with some modifications to handle the shift in the overlapping mesh's location at discrete times.

The applications part (Paper III, IV, and V) concerns cut finite element methods on overlapping meshes for *stationary* PDE-problems. We consider two potential applications for CutFEM on overlapping meshes. The first application, presented in Paper III, presents methodology for evaluating configurations of buildings based on wind and view. The wind model is based on a CutFEM on overlapping meshes for Stokes equations. The second application, presented in Paper IV and Paper V, concerns a software application (app). The app lets a user define and solve physical problems governed by PDEs in an immersive and interactive augmented reality environment.

Keywords: CutFEM, overlapping meshes, multi-mesh, moving meshes, parabolic problem, energy analysis, multi-objective optimization, augmented reality

List of papers

Appended papers

Paper I Mats G. Larson, Anders Logg, **Carl Lundholm**
A cut finite element method for the heat equation on overlapping meshes: Energy analysis for cG(1) mesh movement
Manuscript

Paper II Mats G. Larson, **Carl Lundholm**
A cut finite element method for the heat equation on overlapping meshes: L^2-analysis for dG(0) mesh movement
Manuscript

Paper III Anders Logg, Christian Valdemar Lorenzen, **Carl Lundholm**
Multi-mesh multi-objective optimization with application to a model problem in urban design
Preprint

Paper IV Anders Logg, **Carl Lundholm**, Magne Nordaas
Solving Poisson's equation on the Microsoft HoloLens
Published in Proceedings of VRST'17

Paper V Anders Logg, **Carl Lundholm**, Magne Nordaas
Finite element simulation of physical systems in augmented reality
Published in Advances in Engineering Software

Paper not part of this thesis

Johannes Borgqvist, Adam Malik, **Carl Lundholm**, Anders Logg, Philip Gerlee, Marija Cvijovic
Cell polarisation in a bulk-surface model can be driven by both classic and non-classic Turing instability
Published in npj Systems Biology and Applications

The thesis author's contribution to the appended papers:

Paper I: Most of the analysis and implementation, which both were developed in close collaboration with the co-authors. Writer of paper.

Paper II: All, except some essential details and results about the shift operator.

Paper III: Part of modeling. Part of implementation and simulation of wind model. All implementation and simulation of view 2D model and optimization. Main writer of paper.

Paper IV: Part of modeling. Implementation of treatment of problem data (except meshing) and assembly of the linear system. Design and implementation of visualization of holograms representing problem data and solution. Part writer of paper.

Paper V: Part of modeling. All extension of the existing implementation from Poisson problems to time-dependent advection–diffusion problems. Main writer of paper.

Acknowledgements

I like to thank my supervisor Anders Logg and my co-supervisor Mats G. Larson. I also wish to thank Magne Nordaas, Christian Valdemar Lorenzen, Johannes Borgqvist, and Adam Malik for collaboration. Thanks also to members of the FEniCS-community: August Johansson, Benjamin Kehlet, Jørgen Dokken, Simon Funke, and many more. I like to thank CREAM Architects, Michael Patriksson, Ann-Brith Strömberg, and Stig Larsson for valuable input in discussions about the work presented in this thesis. Thanks to other faculty members, colleagues, and friends at the department for lectures, seminars, discussions, problem-solving sessions, lunches, parties, and other fun stuff.

Thanks also to friends outside of the department for a wide range of activities, warmth and friendship. Final thanks to my family for support and love, and for canine countryside weekends on Orust.

Carl Lundholm
Gothenburg, May 2021

Contents

1 Introduction

The finite element method (FEM) is a well-known tool for computing approximate solutions of partial differential equations (PDEs). It is particularly suitable for PDE-problems with complicated geometry since it allows for unstructured domain-fitted meshes. *Unstructured* meshes are more computationally expensive to generate and memory demanding to store than *structured* meshes since there is no underlying structure that may be used. Cut finite element methods (CutFEMs) enable the use of structured meshes in problems with complicated geometry. This may be done by using an underlying structured mesh together with an interface that represents the boundary of the solution domain. There is no relation between the locations of the interface and the mesh cells, thus some of the mesh cells will be "cut" by the interface, hence the name CutFEM. The cost of using CutFEM is the extra treatment that is needed to handle functions on these cut mesh cells.

CutFEM may also make costly remeshing redundant for problems with changing or evolving geometries or for other situations involving meshing such as adaptive mesh refinement. Using standard FEM for such problems usually means that a new mesh has to be generated when the geometry has changed too much. With CutFEM the geometry may be represented by an interface whose location in relation to the mesh may be arbitrary, thus allowing the same mesh to be used for different or changing interfaces.

A common type of problem with changing geometry is one where there is an object in the solution domain that moves relatively to the domain boundary. An advantageous CutFEM approach to such problems is to use *overlapping meshes*, meaning two or more meshes ordered in a mesh hierarchy. This is also called composite grids/meshes and multimesh in the literature but the meaning is the same. The idea is to first remove the object from the domain and to generate a stationary background mesh in the empty solution domain. The background mesh may thus be a nicely structured mesh. A second mesh is then generated around the object. The mesh containing the object is then placed "on top" of the background mesh, creating a mesh hierarchy. The movement of the object will thus also cause its encapsulating mesh to move.

A number of other methodologies have been proposed to circumvent the limitations of domain-fitted discretizations. Notable examples are the fictitious domain method by [1] and the extended finite element method (XFEM) by Belytschko et al. [2]. Both methods have been successful in extending the range of problems that can be simulated, but both suffer from limitations in that the conditioning of the discretization cannot be guaranteed, and a theoretical framework for convergence analysis and error estimation is lacking. In particular, time-dependent multiphysics problems on evolving geometries are typically

discretized using *ad hoc* low order discretization methods, which cannot easily be analyzed, nor extended to higher order.

Over the past two decades, a theoretical foundation for the formulation of stabilized CutFEM has been developed by extending the ideas of Nitsche, presented in [3], to a general weak formulation of the interface conditions, thereby removing the need for domain-fitted meshes. The foundations of CutFEM were presented in [4] and then extended to overlapping meshes in [5]. The CutFEM methodology has since been developed and applied to a number of important multiphysics problems. See for example [6–11]. For CutFEM on overlapping meshes in particular, see for example [12–15]. So far, only CutFEM for *stationary* PDE-problems on overlapping meshes have been developed and analysed to a satisfactory degree, thus leaving analogous work for *time-dependent* PDE-problems to be desired.

This thesis deals with both analysis and applications of CutFEM on overlapping meshes. The premise for both parts is the aforementioned state of such methods. Thus, in the analysis part we consider CutFEM on overlapping meshes for a time-dependent parabolic model problem: The heat equation on two overlapping meshes. The applications part only concern CutFEM on overlapping meshes for stationary PDE-problems.

The outline of the thesis is:

- In Section 2, we present the concept CutFEM on overlapping meshes. We start with the standard finite element method and successively present ideas and theory that make us arrive at the aimed concept.
- In Section 3, we summarize the quite long and explicit analysis manuscripts presented as Paper I and Paper II. We start by presenting general analysis methodology for finite element methods. First for elliptic, then parabolic problems. We do this to elucidate the analysis framework at our disposal as a means to create a context for the two analysis manuscripts. In both Section 2 and Section 3, we try to be relatively brief, meaning that we sometimes cut back on explicit detail and rigour, and that we try to not dive too deep into existing theory. The reason for this of course being that we want to focus on the novelties presented in this thesis. For details and rigour we instead refer to the appended papers and the literature.
- In Section 4, we discuss both potential and existing applications of CutFEM on overlapping meshes. Some implementation aspects are also discussed involving software and computational gain in using CutFEM on overlapping meshes.
- In Section 5, we summarize the five appended papers.
- The last part of the thesis is the five appended papers.

2 Cut finite element methods

In this section, we present background, underlying ideas and basic theory for the concept cut finite element methods on overlapping meshes. To put it into context, we will consider elliptic model problems. First we present the standard finite element method for such a problem. Then, using Nitsche's method, we present a *cut* finite element method on a fictitious domain. Lastly we introduce the concept overlapping meshes and present a cut finite element method for an elliptic model problem on overlapping meshes.

2.1 Standard finite element method

The finite element method is a well-known concept in applied mathematics and a valuable tool for science and engineering. There is plenty of literature about it; see for example [16–18] for introductory books. The finite element method is a recipe for obtaining approximate solutions of problems governed by partial differential equations. This is done by discretizing the solution domain and turning the *differential* equation into several *algebraic* equations. The general procedure of FEM may be summarized in the pipeline:

$$\text{PDE-problem} \xrightarrow{1} \begin{array}{c}\textit{Continuous}\\ \text{variational}\\ \text{problem}\end{array} \xrightarrow{2} \begin{array}{c}\textit{Discrete}\\ \text{variational}\\ \text{problem}\end{array} \xrightarrow{3} \begin{array}{c}\text{System of}\\ \text{algebraic}\\ \text{equations}\end{array}$$

where the numbered steps are:

1. Multiply the differential equation with a suitable test function, integrate over the solution domain, use integration by parts to decrease the highest order derivative, and define trial and test spaces.
2. Discretize the solution domain to obtain a finite element mesh, and use it to define a discrete function space.
3. *Test* with the basis functions of the discrete subspace to obtain one algebraic equation for every basis function.

We apply the above steps to the following elliptic model problem.

PDE-problem

For $d = 1, 2$, or 3, let $\Omega \subset \mathbb{R}^d$ be a bounded *domain*, i.e., connected open set, with polygonal boundary $\partial\Omega$. We consider Poisson's equation on Ω with given source function $f \in L^2(\Omega)$ and homogeneous Dirichlet boundary conditions. The problem is: Find $u \in H^2(\Omega)$ such that

$$\begin{cases} -\Delta u = f & \text{in } \Omega, \\ u = 0 & \text{on } \partial\Omega. \end{cases} \tag{2.1}$$

The PDE-problem (2.1) is thus the starting point.

Step 1
We take as a suitable test function $v \in H_0^1(\Omega)$, multiply Poisson's equation by it, and integrate over the solution domain, and thus get

$$\int_\Omega -\Delta u v \,\mathrm{d}x = \int_\Omega f v \,\mathrm{d}x. \tag{2.2}$$

By integration by parts in this context we mean using Green's first identity. Applying it to the left-hand side of (2.2), we get

$$\int_\Omega -\Delta u v \,\mathrm{d}x = \int_\Omega \nabla u \cdot \nabla v \,\mathrm{d}x - \int_{\partial\Omega} n \cdot \nabla u v \,\mathrm{d}s = \int_\Omega \nabla u \cdot \nabla v \,\mathrm{d}x, \tag{2.3}$$

since $v|_{\partial\Omega} = 0$. We now have the equation

$$\int_\Omega \nabla u \cdot \nabla v \,\mathrm{d}x = \int_\Omega f v \,\mathrm{d}x. \tag{2.4}$$

For this equation to mean anything, we see that u and v must indeed belong to $H^1(\Omega)$. Taking the boundary conditions into account, we choose as both trial space (the space where we will look for u) and test space (the space of the functions which we will "test" against) the Sobolev space $H_0^1(\Omega)$.

Continuous variational problem
From the conclusion of Step 1, we formulate the continuous variational problem: Find $u \in H_0^1(\Omega)$ such that

$$\int_\Omega \nabla u \cdot \nabla v \,\mathrm{d}x = \int_\Omega f v \,\mathrm{d}x, \quad \forall v \in H_0^1(\Omega). \tag{2.5}$$

Note that in the continuous variational problem (2.5), the highest order derivative of the solution u is of the *first* order, whereas in the original PDE-problem (2.1) it is of the *second* order. The regularity assumptions on u have thus been relaxed, which allows us to search for a solution in a much larger class of functions. The variational formulation (2.5) is therefore also referred to as the *weak* formulation of (2.1). We define the bilinear form a on $H^1(\Omega)$ and linear form l on $L^2(\Omega)$ by

$$a(w, v) := (\nabla w, \nabla v), \qquad l(v) := (f, v), \tag{2.6}$$

where $(\cdot, \cdot)$ is the $L^2(\Omega)$-inner product. By using a and l, we give (2.5) the more abstract and consice form: Find $u \in H_0^1(\Omega)$ such that

$$a(u, v) = l(v), \quad \forall v \in H_0^1(\Omega). \tag{2.7}$$

Step 2

We discretize the domain Ω by tessellating it into d-simplices. This is done in such a way that every two adjacent simplices share the exact same sub-simplices (simplices of dimension $< d$) belonging to the intersection of their boundaries. The tessellation is the finite element mesh $\mathcal{T}$. We denote by h the mesh size parameter, taken to be the largest diameter of a simplex $K \in \mathcal{T}$. We define the finite dimensional discrete function space V_h by

$$V_h := \{v \in C(\Omega) : v|_K \in \mathcal{P}^p(K), \forall K \in \mathcal{T}, v|_{\partial\Omega} = 0\}, \tag{2.8}$$

i.e., all continuous functions on Ω, that are polynomials of at most degree p on every simplex $K \in \mathcal{T}$, and that are zero on the boundary of Ω.

Discrete variational problem

Using the above, we formulate the discrete variational problem: Find $u_h \in V_h$ such that

$$a(u_h, v) = l(v), \quad \forall v \in V_h. \tag{2.9}$$

This is also the standard finite element formulation of problem (2.1). A discrete variational problem of the form (2.9) is also referred to as a finite element method. Noting that $V_h \subset H_0^1(\Omega)$, we have that (2.7) also holds for all $v \in V_h$. We thus obtain the Galerkin orthogonality

$$a(u - u_h, v) = 0, \quad \forall v \in V_h, \tag{2.10}$$

which means that the finite element solution u_h is the a-projection of the analytic solution u onto V_h.

Step 3

With $M = \dim(V_h)$, we let $\{\varphi_j\}_{j=1}^M$ denote a basis for V_h. Every function $v \in V_h$ may thus be represented as

$$v(x) = \sum_{j=1}^{M} V_j \varphi_j(x), \tag{2.11}$$

where the V_j's are the degrees of freedom. Using this representation for u_h and testing with $v = \varphi_i$ in (2.9) gives

$$a(\sum_{j=1}^{M} U_j \varphi_j, \varphi_i) = l(\varphi_i) \quad \Longrightarrow \quad \sum_{j=1}^{M} U_j \int_\Omega \nabla\varphi_j \cdot \nabla\varphi_j \,\mathrm{d}x = \int_\Omega f\varphi_i \,\mathrm{d}x. \tag{2.12}$$

We thus get one equation for every φ_i in the basis.

System of algebraic equations
We define the matrix A and the vector b by

$$A_{ij} := \int_\Omega \nabla\varphi_j \cdot \nabla\varphi_j \,\mathrm{d}x, \qquad b_i := \int_\Omega f\varphi_i \,\mathrm{d}x, \tag{2.13}$$

where i and j denote the row and column index, respectively. Using A, b, and (2.12), we formulate a linear system of equations on matrix-vector form: Find $U = (U_1, \ldots, U_M)^T$ such that

$$AU = b. \tag{2.14}$$

Final remark
The resulting linear system of equations (2.14) is typically solved by a computer. As already mentioned, the discrete variational problem is the finite element formulation corresponding to the original PDE-problem with the given discretization and choice of discrete subspace. From such a finite element formulation it is straightforward, albeit not always trivial, to derive the system of algebraic equations. In the remainder of this text we will therefore not go further along the finite element pipeline than the discrete variational problem.

2.2 Nitsche's method

The basis for CutFEMs is Nitsche's method for weakly imposing boundary conditions; see [3]. We will therefore start by applying Nitsche's method to an elliptic model problem. In the next section, we will use that result to derive and formulate a basic cut finite element method on a fictitious domain.

Consider the elliptic model problem (2.1) and its corresponding *standard* weak formulation (2.7). Note that the boundary condition in (2.1), $u = 0$ on $\partial\Omega$, is represented in (2.7) by the choice of trial space, $u \in H_0^1(\Omega)$. Incorporating boundary conditions like this, in the *trial space* of the weak formulation, is known as *strongly* imposed boundary conditions. An alternative is to incorporate the boundary conditions in the *equation* of the weak formulation by the addition of suitable terms. This is then called *weakly* imposed boundary conditions. To illustrate this, we consider the problem (2.1) but with *inhomogeneous* Dirichlet boundary conditions given by $g \in L^2(\partial\Omega)$. The problem is: Find $u \in H^2(\Omega)$ such that

$$\begin{cases} -\Delta u = f & \text{in } \Omega, \\ \quad u = g & \text{on } \partial\Omega. \end{cases} \tag{2.15}$$

With the tools from the previous section, a corresponding continuous variational problem with strongly imposed boundary condition is: Find $u \in H^1_g(\Omega)$ such that

$$a(u, v) = l(v), \quad \forall v \in H^1_0(\Omega). \tag{2.16}$$

Note that we still test with functions v that are zero on the boundary, just as in the case with homogeneous boundary conditions. This is because in both cases we do not need to test for u on the boundary since its value there is known.

Now if we instead would like to impose the boundary conditions weakly, we could add the term $\gamma(u - g, v)_{\partial\Omega}$ to the left-hand side of (2.16), where γ is a penalty parameter and $(\cdot, \cdot)_{\partial\Omega}$ denotes the $L^2(\partial\Omega)$-inner product. Typically all the u-v-product terms are gathered on the left-hand side, and all the data-v terms on the right-hand side. Thus the additional term would be split to give

$$a(u, v) + \gamma(u, v)_{\partial\Omega} = l(v) + \gamma(g, v)_{\partial\Omega}. \tag{2.17}$$

Note that the requirement that $v|_{\partial\Omega} = 0$ must be removed for two reasons: so that the additional term is not always zero; and because the requirement that $u|_{\partial\Omega} = g$ would no longer be imposed on the trial space, thus resulting in a need to test for u on the boundary. Removing the requirement $v|_{\partial\Omega} = 0$ also has other consequences. In the derivation of the variational equation in Step 1 in the preceding section, the fact that $v|_{\partial\Omega} = 0$ led to that the term $-(n \cdot \nabla u, v)_{\partial\Omega}$ vanished. Thus, if we plug in the strong solution of the original PDE-problem in (2.17) and use integration by parts in the other direction, we would get that the strong solution does not solve the weak problem. This means that the method is inconsistent and a remedy is needed. Adding $-(n \cdot \nabla u, v)_{\partial\Omega}$ to the left-hand side of (2.17) makes the method consistent, hence this term is usually called the *consistency* term. Using the notation $\partial_n u = n \cdot \nabla u$, we get

$$a(u, v) - (\partial_n u, v)_{\partial\Omega} + \gamma(u, v)_{\partial\Omega} = l(v) + \gamma(g, v)_{\partial\Omega}. \tag{2.18}$$

But we are still not done. Note that the left-hand sides of all the variational equations presented so far have been symmetric, i.e., switching places of u and v still gives the same left-hand side. This is a desirable property to have and to preserve this symmetry, we add the so called *symmetry* term $-(\partial_n v, u)_{\partial\Omega}$ to the left-hand side of (2.18). This in turn means that to preserve consistency we also need to add $-(\partial_n v, g)_{\partial\Omega}$ to the right-hand side of (2.18). Thus

$$\begin{aligned} &a(u, v) - (\partial_n u, v)_{\partial\Omega} - (\partial_n v, u)_{\partial\Omega} + \gamma(u, v)_{\partial\Omega} \\ &= l(v) - (\partial_n v, g)_{\partial\Omega} + \gamma(g, v)_{\partial\Omega}. \end{aligned} \tag{2.19}$$

We define the bilinear form A and linear form L by

$$A(w, v) := a(w, v) - (\partial_n w, v)_{\partial\Omega} - (\partial_n v, w)_{\partial\Omega} + \gamma(w, v)_{\partial\Omega}, \tag{2.20}$$

$$L(v) := l(v) - (\partial_n v, g)_{\partial\Omega} + \gamma(g, v)_{\partial\Omega}. \tag{2.21}$$

The continuous variational problem with weakly imposed boundary condition is: Find $u \in H^1(\Omega)$ such that

$$A(u, v) = L(v), \quad \forall v \in H^1(\Omega). \tag{2.22}$$

This is Nitsche's method of imposing boundary conditions weakly and the boundary terms in (2.20) are referred to as the Nitsche terms.

2.3 A cut finite element method on a fictitious domain

The first cut finite element method was presented in the influential paper [4] which lay the foundation for CutFEM. In that paper a CutFEM for an elliptic model problem with an interface was considered. Here, as our basic CutFEM, we will instead consider an elliptic model problem on a fictitious domain, which is more in line with the previous section about Nitsche's method.

The key here is to realize the unlocked potential in freedom of discretization that comes from the relaxation of the imposition of the boundary conditions, i.e., imposing them weakly instead of strongly.

If the boundary conditions are imposed strongly, this has to be taken into account when discretizing the domain to create the finite element mesh. This is so since in the discrete subspace, there typically have to be degrees of freedom on the boundary that directly can represent the strongly imposed boundary conditions. The location of the degrees of freedom usually depends on the mesh, meaning that the domain boundary needs to influence the discretization.

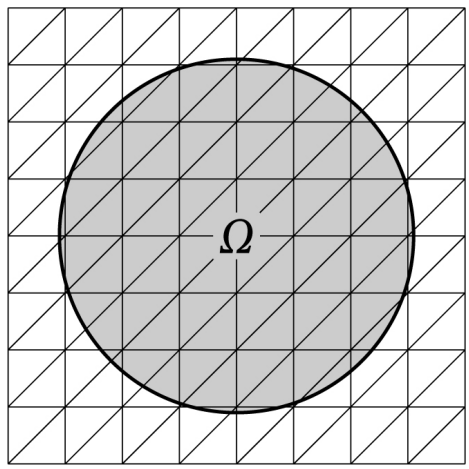

Figure 1: 2D-example of an unfitted mesh with a domain Ω. (Image from [19].)

With weakly imposed boundary conditions, there is no need for degrees of freedom on the boundary, since the functions in the discrete subspace do not need to satisfy the boundary conditions. This means that the domain boundary does not need to be considered for the discretization. An *unfitted* mesh may therefore be used; see Figure 1. A benefit of using an unfitted mesh is that it can be *structured* as opposed to a *boundary-fitted* mesh that in general is *unstructured*. Generating and storing unstructured

meshes is more computationally costly and memory demanding since there is no underlying structure that may be used.

For the CutFEM, we need a mesh-dependent discrete function space. Letting $\mathcal{T}$ denote an unfitted mesh, we define its active part by

$$\mathcal{T}_\Omega := \{K \in \mathcal{T} : K \cap \Omega \neq \emptyset\}. \tag{2.23}$$

The active mesh $\mathcal{T}_\Omega$ thus consists of all simplices in $\mathcal{T}$ that either lie completely inside Ω or are cut by its boundary. We use $\mathcal{T}_\Omega$ to define the eponymous fictitious domain $\Omega_h \supset \Omega$ by

$$\Omega_h := \bigcup_{K \in \mathcal{T}_\Omega} \bar{K}. \tag{2.24}$$

It is simply the domain defined by the extent of the active mesh. Using these objects and based on the spaces used in the continuous variational formulation (2.22), we define the discrete subspace by

$$V_h := \{v \in C(\Omega_h) : v|_K \in \mathcal{P}^p(K), \forall K \in \mathcal{T}_\Omega\}. \tag{2.25}$$

For the discrete variational problem, we modify the bilinear form A and the linear form L by including the mesh size parameter $h_K = h_K(x) = \text{diam}(K)$ for $x \in K$ in the penalty parameter as

$$A_h(w, v) := a(w, v) - (\partial_n w, v)_{\partial\Omega} - (\partial_n v, w)_{\partial\Omega} + \gamma(h_K^{-1} w, v)_{\partial\Omega}, \tag{2.26}$$

$$L_h(v) := l(v) - (\partial_n v, g)_{\partial\Omega} + \gamma(h_K^{-1} g, v)_{\partial\Omega}. \tag{2.27}$$

The mesh parameter h_K is included to make the bilinear form A_h *coercive* on V_h. The Nitsche term in A_h with the h_K-penalty parameter is therefore also referred to as the *coercivity* term. The fictitious domain cut finite element formulation is: Find $u_h \in V_h$ such that

$$A_h(u_h, v) = L_h(v), \quad \forall v \in V_h. \tag{2.28}$$

Functions in V_h live on the larger fictitious domain Ω_h, but the solution domain of interest is still the smaller domain Ω. This is reflected in the discrete variational formulation by the integrals in the forms A_h and L_h. Thus when deriving a linear system of equations from (2.28), only the Ω-part of the boundary cut simplices in $\mathcal{T}$ will contribute. It is of course these cut simplices and their impact on the resulting finite element method that gives CutFEM its name. The CutFEM solution is sometimes considered to be the restriction to Ω of the actual discrete solution $u_h \in V_h$. This is often the case when visualizing the solution.

2.4 Overlapping meshes

An essential step in the finite element method is to discretize the solution domain to form a computational mesh. Let us here refer to a mesh of the underlying solution domain as a *background* mesh. Now, consider another mesh that is allowed to move around "on top" of the background mesh. Let us refer to such a mesh as an *overlapping* mesh. By introducing one or more overlapping meshes over a background mesh, we arrive at the concept overlapping meshes.

We motivate this concept by an application example. Consider an object in the solution domain that has a movement relative to the outer domain boundary. For example, imagine that we would like to resolve the fluid flow around a rotating propeller, i.e., solving the Navier-Stokes equations. Taking our moving object to be the rotating propeller, the usual way of creating a computational mesh would be to discretize the space between the domain walls and the propeller. An example of this can be seen in the left illustration of Figure 2. If the propeller starts to rotate, the triangles that have nodes in both the solution domain and on the propeller boundary will start to get deformed. This can be seen in the middle illustration of Figure 2. Eventually, these triangles will get too deformed and ruin the mesh so that it cannot be used for FEM. This is shown in the right illustration of Figure 2.

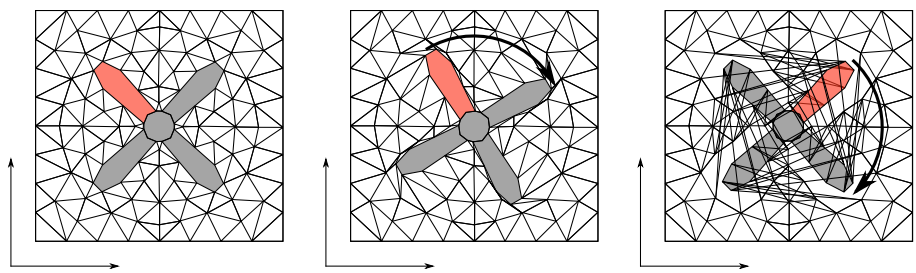

Figure 2: 2D example of how a moving object can ruin the computational mesh. The moving object in this example is a rotating propeller. One of its rotor blades is colored differently for reference. *Left*: Initial setting with a propeller in the solution domain and a triangular mesh. *Middle*: The propeller is rotated by a small angle. Triangles in front of a rotor blade are compressed and triangles behind a rotor blade are streched out. *Right*: The propeller is rotated by a large angle and ruins the mesh.

One way of dealing with this inconvenience is to generate a new mesh when the old mesh starts to get messed up. Mesh generation can however be a time consuming procedure, especially in industrial applications, where there are often

several dimensions and complicated geometries. A mesh might then also consist of several million simplices.

By instead using overlapping meshes, costly remeshing may be avoided altogether. To apply this concept, the object is first removed from the solution domain. A background mesh is then generated in the empty solution domain, and an overlapping mesh is generated around the object. The mesh encapsulated object is reintroduced into the solution domain, where it can then move around freely on top of the background mesh. This procedure for the rotating propeller example is shown in Figure 3. Using overlapping meshes transfers the issue of dealing with the object's movement from the object boundary to the joint boundary between the meshes. This is where CutFEM and Nitsche's method come in which is demonstrated in the next section.

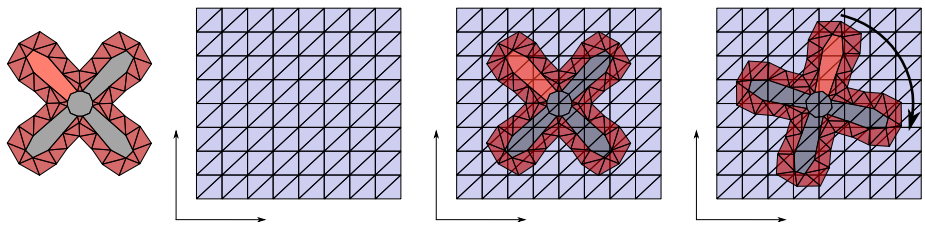

Figure 3: 2D example of how overlapping meshes can be used to avoid remeshing for problems with moving objects in the solution domain. The moving object is again the rotating propeller. *Left*: The propeller is removed from the solution domain and encapsulated in an overlapping mesh with red triangles. The remaining solution domain is triangulated to form a background mesh with blue triangles. *Middle*: The mesh encapsulated propeller is reintroduced on top of the background mesh. *Right*: The overlapping mesh with the propeller is rotated on top of the background mesh.

2.5 A cut finite element method on overlapping meshes

Here we derive and present a cut finite element method on overlapping meshes. This was first done by [5] for an elliptic model problem, which we will also consider. For simplicity and focus on the concept itself, we will not consider a problem with an object. Thus, the overlapping mesh will be empty as opposed to the introductory example. For the same reasons, we will also let the overlapping mesh be fully immersed in the background mesh, i.e., it does not protrude outside the solution domain.

Problem formulation

For $d = 1, 2$, or 3, let $\Omega_0 \subset \mathbb{R}^d$ and $G \subset \Omega_0 \subset \mathbb{R}^d$ be bounded domains with polygonal boundaries $\partial\Omega_0$ and ∂G, respectively. From Ω_0 and G, we define the following two domains:

$$\Omega_1 := \Omega_0 \setminus (G \cup \partial G), \quad \Omega_2 := \Omega_0 \cap G, \tag{2.29}$$

with boundaries $\partial\Omega_1$ and $\partial\Omega_2$, respectively. Let the common boundary between Ω_1 and Ω_2 be

$$\Gamma := \partial\Omega_1 \cap \partial\Omega_2, \tag{2.30}$$

Note that we have the partition

$$\Omega_0 = \Omega_1 \cup \Gamma \cup \Omega_2. \tag{2.31}$$

We will use the jump and average of a function v on the interface Γ, defined by

$$[v] := v_1 - v_2, \tag{2.32}$$

$$\langle v \rangle := \omega_1 v_1 + \omega_2 v_2, \tag{2.33}$$

where v_i denotes the Ω_i-limit of v on Γ, and where the ω_i's are convex weights. We also denote the outward pointing unit normal vector to $\partial\Omega_i$ by n_i. We consider Poisson's equation on Ω_0 with given source function $f \in L^2(\Omega_0)$ and for simplicity homogeneous Dirichlet boundary conditions. We also assume that the derivative of the solution in the normal direction to Γ is continuous. The problem is: Find $u \in H^2(\Omega_0)$ such that

$$\begin{cases} -\Delta u = f & \text{in } \Omega_0, \\ u = 0 & \text{on } \partial\Omega_0, \\ [\partial_n u] = 0 & \text{on } \Gamma. \end{cases} \tag{2.34}$$

where $[\partial_n u] = n \cdot (\nabla u)_1 - n \cdot (\nabla u)_2$, where n can be either n_1 or n_2.

Nitsche's method for interfaces

To perform Step 1 in the FEM-pipeline with the coming overlapping meshes in mind, we define the broken Sobolev space

$$H_0^1(\Omega_1, \Omega_2) := \{v \in L^2(\Omega_0) : v|_{\Omega_i} \in H^1(\Omega_i), \text{ for } i = 1, 2, v|_{\partial\Omega_0} = 0\}. \tag{2.35}$$

Another equivalent notation for this space is $H_0^1(\cup_i \Omega_i)$. Taking $v \in H_0^1(\Omega_1, \Omega_2)$, and following the instructions of Step 1 gives

$$\int_{\Omega_0} -\Delta u v \, \mathrm{d}x = \int_{\Omega_0} f v \, \mathrm{d}x. \tag{2.36}$$

When partially integrating the left-hand side, we need to consider the partition (2.31) since we do not have H^1-regularity of v on the whole of Ω_0. Thus

$$\begin{aligned}\int_{\Omega_0} -\Delta u v \,\mathrm{d}x &= \sum_{i=1}^{2} \int_{\Omega_i} -\Delta u v \,\mathrm{d}x \\ &= \sum_{i=1}^{2} \Big(\int_{\Omega_i} \nabla u \cdot \nabla v \,\mathrm{d}x - \int_{\partial\Omega_i} (\partial_n u)_i v_i \,\mathrm{d}s \Big).\end{aligned} \tag{2.37}$$

We consider the boundary terms separately. They are

$$\begin{aligned}\sum_{i=1}^{2} \int_{\partial\Omega_i} (\partial_n u)_i v_i \,\mathrm{d}s &= \underbrace{\sum_{i=1}^{2} \int_{\partial\Omega_i \cap \partial\Omega_0} (\partial_n u)_i v_i \,\mathrm{d}s}_{=0, \text{ since } v|_{\partial\Omega_0}=0} + \sum_{i=1}^{2} \int_{\Gamma} (\partial_n u)_i v_i \,\mathrm{d}s \\ &= \int_{\Gamma} n_1 \cdot (\nabla u)_1 v_1 + n_2 \cdot (\nabla u)_2 v_2 \,\mathrm{d}s \\ &\overset{\text{3rd}}{=} \int_{\Gamma} n \cdot (\nabla u)_1 v_1 - n \cdot (\nabla u)_2 v_2 \,\mathrm{d}s = \int_{\Gamma} [\partial_n u v] \,\mathrm{d}s \\ &\overset{\text{5th}}{=} \int_{\Gamma} \underbrace{[\partial_n u]}_{=0} \langle v \rangle + \langle \partial_n u \rangle [v] + (\omega_2 - \omega_1) \underbrace{[\partial_n u]}_{=0} [v] \,\mathrm{d}s \\ &= \int_{\Gamma} \langle \partial_n u \rangle [v] \,\mathrm{d}s,\end{aligned} \tag{2.38}$$

where we have taken $n = n_1$, and thus get $n_2 = -n$, in the third step. In the fifth step we have used a jump identity, which is easily shown to hold by evaluating both sides. Combining (2.37) and (2.38), turns (2.36) into

$$\sum_{i=1}^{2} (\nabla u, \nabla v)_{\Omega_i} - (\langle \partial_n u \rangle, [v])_{\Gamma} = (f, v)_{\Omega_0}. \tag{2.39}$$

The second term on the left-hand side is the corresponding Nitsche *consistency* term. For the same reasons as before, we add corresponding *symmetry* and *coercivity* terms, and then formulate the continuous variational problem: Find $u \in H_0^1(\Omega_1, \Omega_2)$ such that

$$\begin{aligned}&\sum_{i=1}^{2} (\nabla u, \nabla v)_{\Omega_i} - (\langle \partial_n u \rangle, [v])_{\Gamma} - (\langle \partial_n v \rangle, [u])_{\Gamma} + \gamma([u], [v])_{\Gamma} \\ &= (f, v)_{\Omega_0}, \quad \forall v \in H_0^1(\Omega_1, \Omega_2).\end{aligned} \tag{2.40}$$

Discrete spaces for overlapping meshes

We continue by letting $\mathcal{T}_0$ and $\mathcal{T}_G$ be meshes of Ω_0 and G, respectively. We define the corresponding standard finite element spaces by

$$V_{h,0} := \{v \in C(\Omega_0) : v|_K \in \mathcal{P}^p(K), \forall K \in \mathcal{T}_0, v|_{\partial\Omega_0} = 0\}, \tag{2.41}$$

$$V_{h,G} := \{v \in C(G) : v|_K \in \mathcal{P}^p(K), \forall K \in \mathcal{T}_G\}. \tag{2.42}$$

Using these spaces we define the *broken* discrete space $V_h \subset H_0^1(\Omega_1, \Omega_2)$ by

$$\begin{aligned} V_h := \{v : v|_{\Omega_1} = v_0|_{\Omega_1}, \text{ for some } v_0 \in V_{h,0} \text{ and} \\ v|_{\Omega_2} = v_G|_{\Omega_2}, \text{ for some } v_G \in V_{h,G}\}. \end{aligned} \tag{2.43}$$

Functions in V_h are allowed to be discontinuous on the interface Γ between the two meshes; see Figure 4. On the overlapping mesh, there are two function parts: one from the overlapping mesh space $V_{h,G}$, and one from the background mesh space $V_{h,0}$.

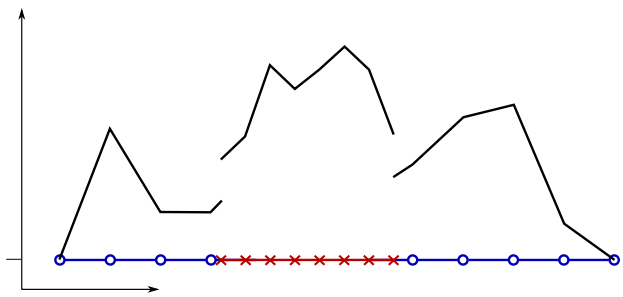

Figure 4: Example of a $v \in V_h$ for $p = 1$. The background mesh $\mathcal{T}_0$ is shown in blue and the overlapping mesh $\mathcal{T}_G$ in red.

Overlap stability term

Just as in the previously presented CutFEM on a fictitious domain, we include the mesh size parameter in the penalty parameter of the coercivity term. Here we take it to be h_{K_0}, where $K_0 \in \mathcal{T}_0$. But this is not enough for the corresponding discrete bilinear form to be coercive on V_h. There are various ways to ensure discrete coercivity. Here, we will add a stability term for the so called overlap domain. To define it, we will use the set of simplices

$$\mathcal{T}_{0,\Gamma} := \{K \in \mathcal{T}_0 : K \cap \Gamma \neq \emptyset\}, \tag{2.44}$$

which is all simplices in $\mathcal{T}_0$ that are cut by Γ. We define the overlap domain Ω_O by

$$\Omega_O := \bigcup_{K \in \mathcal{T}_{0,\Gamma}} K \cap \Omega_2. \tag{2.45}$$

We want the overlap stability term to connect the function parts from the two meshes and penalize the difference. It is thus natural to consider the jump on the overlap $[v] := v_1 - v_2$, where v_1 is the function part from the background mesh and v_2 the function part from the overlapping mesh. It is also natural to let a new stability term be symmetric, so that its inclusion does not ruin the symmetry of the bilinear form. Another aspect to consider is *discrete dimensional analysis*, i.e., to examine the power of the mesh size parameter h (length) in each term. With integrals over subsets of codimension k having discrete unit $[h^{d-k}]$, and derivatives of order l having discrete unit $[h^{-l}]$, performing a discrete dimensional analysis on one of the bulk terms on the left-hand side of (2.40) gives

$$(\nabla u, \nabla v)_{\Omega_i} = \int_{\Omega_i} \nabla u \cdot \nabla v \, \mathrm{d}x \sim [h^d] \cdot [h^{-1}] \cdot [h^{-1}] = [h^{d-2}]. \tag{2.46}$$

Performing this analysis on every term on the left-hand side of (2.40) gives the same answer: $[h^{d-2}]$. Typically we want all the terms in the bilinear form to have the same unit. Since $\int_{\Omega_O} \mathrm{d}x \sim [h^d]$ and by taking the other points into consideration as well, two natural choices for the overlap stability term are:

$$([\nabla w], [\nabla v])_{\Omega_O} \quad \text{and} \quad (h_K^{-2}[w], [v])_{\Omega_O}. \tag{2.47}$$

The latter term has the advantage of not vanishing if both function parts are constant. Therefore it is often preferred in implementations. The former might be more natural from an analysis viewpoint. But that does not really matter since one may easily obtain the latter from the former via an inverse estimate on discrete spaces.

Finite element formulation

We define the mesh-dependent bilinear form A_h on $H^1(\Omega_1, \Omega_2)$ by

$$\begin{aligned} A_h(w, v) := &\sum_{i=1}^{2} (\nabla w, \nabla v)_{\Omega_i} - (\langle \partial_n w \rangle, [v])_\Gamma - (\langle \partial_n v \rangle, [w])_\Gamma \\ &+ (\gamma h_K^{-1}[w], [v])_\Gamma + ([\nabla w], [\nabla v])_{\Omega_O}. \end{aligned} \tag{2.48}$$

The right-hand side linear form is just the same as in the standard finite element formulation. Thus

$$l(v) := (f, v)_{\Omega_0}, \tag{2.49}$$

where $(\cdot, \cdot)_{\Omega_0}$ is the $L^2(\Omega_0)$-inner product. We may now formulate the overlapping mesh cut finite element method as: Find $u_h \in V_h$ such that

$$A_h(u_h, v) = l(v), \quad \forall v \in V_h. \tag{2.50}$$

3 Analysis

Analysis of finite element methods is needed to verify that they produce "good" approximate solutions. For books on the analysis of finite element methods, see for example [20] for an introductory book and [21] for a more advanced book. Existence and uniqueness of a finite element solution is usually established via the application of the Lax-Milgram theorem to the continuous variational problem. The main components of an analysis of a finite element method are usually *stability analysis* and *error analysis*, and that is what we will focus on in this section. For theory about existence and uniqueness, we refer to the aforementioned books for details

In the stability analysis, the finite element solution is estimated in terms of problem data. For elliptic problems this means the right-hand side source function, and for time-dependent problems this includes the initial value as well. This is to make sure that the finite element solution depends continuously on the problem data. A small change in the problem data should result in a small change of the solution. This is in accordance with Hadamard's definition of a well-posed problem: a solution should exist, it should be unique, and it should depend continuously on the problem data.

In the error analysis, the approximation error is estimated. The approximation error is the difference between the *analytic solution* to the original PDE-problem and the discrete *finite element solution*. It is reasonable to assume that if a finer discretization is used, then a good discrete solution should be a better approximation to the analytic solution. The approximation error may be estimated in various norms. Two of the most common norms to use are an energy norm, which is a norm related to the bilinear form, and the standard L^2-norm. Depending on what norm is used, we will refer to the analysis as either an *energy analysis* or an L^2*-analysis*. The approximation error is estimated either in terms of the analytic solution or the finite element solution. An estimate of the former type is called an *a priori* error estimate and one of the latter type an *a posteriori* error estimate. Here we will only consider a priori error estimates.

The intended purpose of this section is to build up to, motivate, and summarize the analyses performed on the time-dependent CutFEMs on overlapping meshes in Paper I and Paper II. In both papers we consider a time-dependent parabolic model problem on two overlapping meshes, where the overlapping mesh is allowed to move around. The difference between the two papers is how we choose to represent the mesh movement discretely. In Paper I, we let its spatial location be a *continuous* and *piecewise linear* function with respect to time. We thus call this a *cG(1) mesh movement*, where cG(1) naturally stands for continuous Galerkin of order 1. In Paper II, we let its spatial location be

a *discontinuous* and *piecewise constant* function with respect to time. We thus call this a *dG(0) mesh movement*, where dG(0) naturally stands for discontinuous Galerkin of order 0. These two different mesh movements result in two different space-time discretizations; see Figure 5.

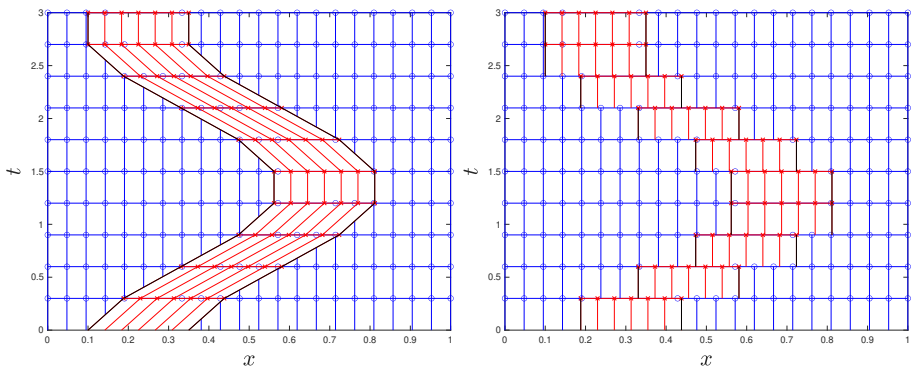

Figure 5: Space-time discretization for two overlapping meshes for $d = 1$ where the overlapping mesh moves around. *Left*: cG(1) mesh movement, studied in Paper I. *Right*: dG(0) mesh movement, studied in Paper II

In Paper I, we perform an energy analysis, and in Paper II, an L^2-analysis. We therefore go through the basic ideas for such analyses of parabolic problems for standard FEM in order to motivate and understand the analyses in Paper I and II. But we start even lighter by considering the analysis of an elliptic model problem. We do this because many core ideas are similar or even the same. To elucidate the similarities, each of the upcoming analysis sections contains the following parts:

1. *Problem formulation*: the finite element method is presented.

2. *Energy norm(s)*: energy norms are derived from the bilinear form together with corresponding coercivity results.

3. *Stability analysis*: stability estimates are presented.

4. *Energy error analysis*: energy error estimates are derived from Céa's lemma type arguments.

5. *L^2-error analysis*: L^2-error estimates are derived using a dual problem. For elliptic cases the corresponding energy error estimate is also used. For parabolic cases the stability estimates are used instead.

3.1 Elliptic problem: standard case

Problem formulation

Recall the elliptic model problem (2.1). The corresponding finite element formulation was: Find $u_h \in V_h$ such that

$$a(u_h, v) = l(v), \quad \forall v \in V_h, \tag{3.1}$$

where V_h was defined by (2.8), $a(w, v) = (\nabla w, \nabla v)_\Omega$, and $l(v) = (f, v)_\Omega$. We will perform the standard energy analysis and L^2-analysis on (3.1). For both elliptic and parabolic problems, the L^2-analysis is based on estimates from the energy analysis. We will therefore start by deriving energy estimates.

Energy norm

For an energy analysis, we need an energy norm, i.e., a norm that comes from the bilinear form. A good starting point to obtain an energy norm is to take the same function as both arguments of the bilinear form. Doing this with our a gives

$$a(v, v) = (\nabla v, \nabla v)_\Omega = \|\nabla v\|_\Omega^2, \tag{3.2}$$

where $\|\cdot\|_\Omega$ is the standard $L^2(\Omega)$-norm. In general $\|\nabla\{\cdot\}\|_\Omega$ is only a *semi*-norm, i.e., $\|\nabla v\|_\Omega = 0, \not\Rightarrow v = 0$, but on $H_0^1(\Omega)$ it is a norm that is equivalent to the full Sobolev 1-norm via the Poincaré inequality. Since $V_h \subset H_0^1(\Omega)$, $\|\nabla\{\cdot\}\|_\Omega$ is the natural energy norm for the case at hand. We henceforth denote it by $\|\cdot\|_a$. From (3.2) we have the trivial coercivity:

$$a(v, v) = \|v\|_a^2, \quad \forall v \in H_0^1(\Omega). \tag{3.3}$$

Stability analysis

As already mentioned, the objective of a stability analysis is to obtain estimates of the solution in terms of the data. This is done by testing with various functions in the finite element formulation. The basic stability estimate for both elliptic and parabolic problems is obtained by taking $v = u_h$ in the finite element formulation. Doing this in (3.1) gives

$$a(u_h, u_h) = l(u_h). \tag{3.4}$$

For the left-hand side we generally need coercivity, i.e., $\exists \alpha > 0 : a(v, v) \geq \alpha \|v\|_V^2$ for all v in some space V. In our case we have this trivially for the whole of $H_0^1(\Omega)$, since $a(v, v) = \|v\|_a^2$. Thus $a(u_h, u_h) = \|u_h\|_a^2$. The right-hand side is estimated by the Cauchy–Schwarz inequality and the Poincaré inequality as

$$l(u_h) = (f, u_h)_\Omega \leq \|f\|_\Omega \|u_h\|_\Omega \leq C\|f\|_\Omega \|\nabla u_h\|_\Omega = C\|f\|_\Omega \|u_h\|_a. \tag{3.5}$$

Using the coercivity and (3.5) in (3.4), we get

$$\begin{aligned} & \|u_h\|_a^2 = a(u_h, u_h) = l(u_h) \leq C\|f\|_\Omega \|u_h\|_a, \\ \implies \quad & \|u_h\|_a \leq C\|f\|_\Omega, \end{aligned} \tag{3.6}$$

which is the basic energy stability estimate. Using the Poincaré inequality and (3.6) we get

$$\begin{aligned} & \|u_h\|_\Omega \leq C\|u_h\|_a \leq C\|f\|_\Omega, \\ \implies \quad & \|u_h\|_\Omega \leq C\|f\|_\Omega, \end{aligned} \tag{3.7}$$

which is the basic L^2-stability estimate. We point out that the constant $C > 0$ changes in the above estimates, but since we do not really care about it more than that it is in fact some positive constant, we just keep it as C in accordance with standard practice. This concludes our stability analysis of (3.1).

Energy error analysis

We move on to the energy error analysis of (3.1) in which we will estimate the approximation error in the energy norm. We let u denote the analytic solution of (2.1). For the derivation of the energy error estimate, we follow the proof of Céa's lemma and use an interpolation error estimate. The steps are:

1. Coercivity of a: $\exists \alpha > 0 : a(v, v) \geq \alpha \|v\|_V^2, \forall v$ in some space V.
2. Galerkin orthogonality: $a(u - u_h, v) = 0, \forall v \in V_h$.
3. Continuity of a: $\exists C > 0 : a(w, v) \leq C\|w\|_V \|v\|_V, \forall w, v$ in some space V.
4. Optimal interpolation error estimate in the energy norm.

For our case, we have coercivity on the whole of $H_0^1(\Omega)$ from $\|v\|_a^2 = a(v, v)$. Galerkin orhogonality comes from consistency, i.e., the analytic solution also solves the finite element equation. To see this, we plug u into the left-hand side of (3.1) and integrate by parts. We get

$$a(u, v) = (\nabla u, \nabla v)_\Omega = (\underbrace{-\Delta u}_{=f}, v)_\Omega + \underbrace{(\partial_n u, v)_{\partial\Omega}}_{=0, v|_{\partial\Omega}=0} = (f, v)_\Omega = l(v), \tag{3.8}$$

which of course holds for all $v \in V_h$. We thus have

$$a(u_h, v) = l(v), \quad \forall v \in V_h, \tag{3.9}$$

$$a(u, v) = l(v), \quad \forall v \in V_h, \tag{3.10}$$

$$\implies \quad a(u - u_h, v) = 0, \quad \forall v \in V_h, \tag{3.11}$$

which is the Galerkin orthogonality. Continuity on $H_0^1(\Omega)$ also follows easily from the Cauchy–Schwarz inequality:

$$a(w,v) = (\nabla w, \nabla v)_\Omega \leq \|\nabla w\|_\Omega \|\nabla v\|_\Omega = \|w\|_a \|v\|_a. \tag{3.12}$$

Combining the coercivity, Galerkin orthogonality, and the continuity gives Céa's lemma. Letting $e = u - u_h$, we have

$$\begin{aligned} & \|e\|_a^2 \overset{1.}{=} a(e,e) = a(e, u - u_h) \overset{2.}{=} a(e, u - v) \overset{3.}{\leq} \|e\|_a \|u - v\|_a \\ \implies \quad & \|u - u_h\|_a \leq \|u - v\|_a, \quad \forall v \in V_h, \end{aligned} \tag{3.13}$$

where we have written out the steps taken in reference to the steps defined above. This means that u_h is the best approximation in V_h to u in the a-sense. Now consider an interpolation operator I_h that maps to V_h and that also gives optimal interpolation error estimates for sufficiently smooth functions, e.g., the Scott-Zhang interpolation operator. This means that

$$\|u - I_h u\|_a \leq C h^p \|D_x^{p+1} u\|_\Omega, \tag{3.14}$$

for some constant $C > 0$, where h is the largest diameter of a simplex in the underlying mesh to V_h, and p is the maximal polynomial degree in V_h. Since $I_h u \in V_h$, we may use it in (3.13) according to the fourth and final step. We thus obtain the optimal order a priori energy error estimate

$$\|u - u_h\|_a \leq C h^p \|D_x^{p+1} u\|_\Omega. \tag{3.15}$$

L^2-error analysis

For the derivation of the L^2-error estimate, we use the Aubin-Nitsche duality trick. The steps are:

1. Continuous dual problem: the error is the right-hand side data.
2. Integration by parts: go from L^2-inner product to bilinear form a.
3. Galerkin orthogonality: subtract an interpolant of ϕ.
4. Continuity of a: $a(w,v) \leq C\|w\|_a \|v\|_a$.
5. Optimal interpolation error estimate for ϕ-factor.
6. Elliptic regularity: get back e in L^2-norm for cancellation.
7. Optimal energy error estimate for e-factor.

We start by defining the continuous dual problem with the approximation error e as data: Find $\phi \in H^2(\Omega)$ such that

$$\begin{cases} -\Delta\phi = e & \text{in } \Omega, \\ \phi = 0 & \text{on } \partial\Omega. \end{cases} \tag{3.16}$$

Following the above steps, where we write out each step taken, we get

$$\begin{aligned} \|e\|_\Omega^2 &= (e,e)_\Omega \overset{1.}{=} (-\Delta\phi, e)_\Omega \overset{2.}{=} (\nabla\phi, \nabla e)_\Omega - (\partial_n\phi, e)_{\partial\Omega} \\ &= a(\phi, e) \overset{3.}{=} a(\phi - I_{h,p=1}\phi, e) \overset{4.}{\leq} \|\phi - I_{h,p=1}\phi\|_a \|e\|_a \\ &\overset{5.}{\leq} Ch\|D_x^2\phi\|_\Omega \|e\|_a \overset{6.}{\leq} Ch\|\Delta\phi\|_\Omega \|e\|_a = Ch\|e\|_\Omega \|e\|_a, \end{aligned} \tag{3.17}$$

where we have used that $e|_{\partial\Omega} = 0$. Cancelling a factor $\|e\|_\Omega$ on both sides, performing the seventh and final step, i.e., using the energy error estimate (3.15), gives us the optimal order a priori L^2-error estimate

$$\|u - u_h\|_\Omega \leq Ch^{p+1}\|D_x^{p+1}u\|_\Omega. \tag{3.18}$$

3.2 Elliptic problem: overlapping meshes

The analysis of CutFEM for elliptic problems on overlapping meshes is quite similar to the standard elliptic analysis. The greatest difference is that the bilinear form A_h does not induce an energy norm in the same way as a. This means that coercivity does not follow as easily as in the standard case presented in the previous section. Coercivity is used to obtain the energy estimates for both stability and error, which are subsequently used to obtain the corresponding L^2-estimates. Coercivity is thus the key to the whole analysis, but its importance may pass by rather unnoticed in the standard case since it appears so naturally. In this analysis however, coercivity will have a more prominent role.

Problem formulation

Recall the elliptic model problem (2.34). The corresponding finite element formulation was: Find $u_h \in V_h$ such that

$$A_h(u_h, v) = l(v), \quad \forall v \in V_h, \tag{3.19}$$

where V_h was defined by (2.43). The bilinear form A_h on $H^1(\Omega_1, \Omega_2)$, defined by (2.48), is

$$\begin{aligned} A_h(w,v) = &\sum_{i=1}^{2} (\nabla w, \nabla v)_{\Omega_i} - (\langle \partial_n w \rangle, [v])_\Gamma - (\langle \partial_n v \rangle, [w])_\Gamma \\ &+ (\gamma h_K^{-1}[w], [v])_\Gamma + ([\nabla w], [\nabla v])_{\Omega_O}. \end{aligned} \tag{3.20}$$

The linear form l is just as in the standard case, i.e., $l(v) = (f, v)_{\Omega_0}$.

Energy norm

Following the general analysis outline, we start with the energy analysis, for which we need an energy norm. Since we obtained such a norm in the standard case by taking the same function as both arguments of the bilinear form, we start by doing the same here. We get

$$A_h(v,v) = \sum_{i=1}^{2} \|\nabla v\|_{\Omega_i}^2 - 2(\langle \partial_n v \rangle, [v])_\Gamma + \gamma \|h_K^{-1/2}[v]\|_\Gamma^2 + \|[\nabla v]\|_{\Omega_O}^2, \quad (3.21)$$

where all terms, except the second, are good since they are non-negative whereas the second term may be negative. This is a problem since $A_h(v,v)$ may be negative for some functions v and thus cannot be a norm. To fix this, we note that the second factor in the problematic term is the same as we have in the third term except the h_K^{-1}-factor. The idea is thus to split the second term, using the Cauchy–Schwarz inequality and Young's inequality, and combine its second factor with the third term. To do this, we define the mesh-dependent norms

$$\|w\|_{1/2,h,\Gamma}^2 := \sum_{K \in \mathcal{T}_{0,\Gamma}} h_{K_0}^{-1} \|w\|_{\Gamma_K}^2, \quad (3.22)$$

$$\|w\|_{-1/2,h,\Gamma}^2 := \sum_{K \in \mathcal{T}_{0,\Gamma}} h_{K_0} \|w\|_{\Gamma_K}^2, \quad (3.23)$$

where $\Gamma_K = K \cap \Gamma$. The second term may now be split as

$$\begin{aligned} 2(\langle \partial_n v \rangle, [v])_\Gamma &\leq 2 \|\langle \partial_n v \rangle\|_{-1/2,h,\Gamma} \|[v]\|_{1/2,h,\Gamma} \\ &\leq \frac{1}{\varepsilon} \|\langle \partial_n v \rangle\|_{-1/2,h,\Gamma}^2 + \varepsilon \|[v]\|_{1/2,h,\Gamma}^2. \end{aligned} \quad (3.24)$$

We refer to the analytic preliminaries of either Paper I or Paper II for details on this and upcoming estimates. Using (3.24) in (3.21), we get

$$\begin{aligned} A_h(v,v) \geq \sum_{i=1}^{2} \|\nabla v\|_{\Omega_i}^2 - \frac{1}{\varepsilon} \|\langle \partial_n v \rangle\|_{-1/2,h,\Gamma}^2 \\ + (\gamma - \varepsilon) \|[v]\|_{1/2,h,\Gamma}^2 + \|[\nabla v]\|_{\Omega_O}^2, \end{aligned} \quad (3.25)$$

where the third term is good as long as $\gamma > \varepsilon$, but where something needs to be done about the non-positive second term. To treat it, we use the inverse inequality

$$\|\langle \partial_n v \rangle\|_{-1/2,h,\Gamma}^2 \leq C \Big(\sum_{i=1}^{2} \|\nabla v\|_{\Omega_i}^2 + \|[\nabla v]\|_{\Omega_O}^2 \Big), \quad \forall v \in V_h. \quad (3.26)$$

It is important to note that this inequality only holds for discrete functions. This is because an inverse estimate for polynomials is needed to show it. We refer to the appendix of either Paper I or Paper II for details. Using (3.26) in (3.25), we get for $v \in V_h$ that

$$\begin{aligned} A_h(v,v) &\geq \left(1 - \frac{2C}{\varepsilon}\right) \sum_{i=1}^{2} \|\nabla v\|_{\Omega_i}^2 + \frac{1}{\varepsilon} \|\langle \partial_n v \rangle\|_{-1/2,h,\Gamma}^2 \\ &\quad + (\gamma - \varepsilon) \|[v]\|_{1/2,h,\Gamma}^2 + \left(1 - \frac{2C}{\varepsilon}\right) \|[\nabla v]\|_{\Omega_O}^2 \\ &\geq \alpha \Bigg(\sum_{i=1}^{2} \|\nabla v\|_{\Omega_i}^2 + \|\langle \partial_n v \rangle\|_{-1/2,h,\Gamma}^2 \\ &\quad + \|[v]\|_{1/2,h,\Gamma}^2 + \|[\nabla v]\|_{\Omega_O}^2 \Bigg), \end{aligned} \tag{3.27}$$

where $\alpha > 0$ is obtained by taking ε sufficiently large, $\gamma > \varepsilon$, and the minimum over the constants. From this, we define the energy norm $|||\cdot|||_{A_h}$ on $H_0^1(\Omega_1, \Omega_2)$ by

$$|||w|||_{A_h}^2 := \sum_{i=1}^{2} \|\nabla w\|_{\Omega_i}^2 + \|\langle \partial_n w \rangle\|_{-1/2,h,\Gamma}^2 + \|[w]\|_{1/2,h,\Gamma}^2 + \|[\nabla w]\|_{\Omega_O}^2. \tag{3.28}$$

From (3.27) we directly obtain the *discrete* coercivity: $\exists \alpha > 0$ such that

$$A_h(v,v) \geq \alpha \, |||v|||_{A_h}^2, \quad \forall v \in V_h. \tag{3.29}$$

In the standard case we had coercivity on the whole space of the energy norm, i.e., $H_0^1(\Omega)$. Here we only get coercivity on the discrete space $V_h \subset H_0^1(\Omega_1, \Omega_2)$, since we have to use an inverse inequality that only holds on V_h to handle the consistency and symmetry terms in A_h.

Stability analysis

The stability analysis is analogous to the standard case with only two natural modifications to account for the CutFEM setting: the discrete coercivity (3.29), and the energy Poincaré inequality on $H_0^1(\Omega_1, \Omega_2)$

$$\|v\|_{\Omega_0} \leq C \, |||v|||_{A_h}. \tag{3.30}$$

We refer to the appendix of Paper I for details. Following the standard case, we take $v = u_h$ in the finite element equation, and thus get

$$A_h(u_h, u_h) = l(u_h). \tag{3.31}$$

The left-hand side is $A(u_h, u_h) \geq \alpha \left|\!\left|\!\left| u_h \right|\!\right|\!\right|_{A_h}^2$ from the discrete coercivity. The right-hand side is estimated by the Cauchy–Schwarz inequality and the energy Poincaré inequality as

$$l(u_h) = (f, u_h)_{\Omega_0} \leq \|f\|_{\Omega_0} \|u_h\|_{\Omega_0} \leq C \|f\|_{\Omega_0} \left|\!\left|\!\left| u_h \right|\!\right|\!\right|_{A_h}. \tag{3.32}$$

Combining this with the left-hand side, just as in the standard case, we get the basic energy stability estimate

$$\left|\!\left|\!\left| u_h \right|\!\right|\!\right|_{A_h} \leq C \|f\|_{\Omega_0}. \tag{3.33}$$

Using the energy Poincaré inequality and (3.33), just as in the standard case, we get the basic L^2-stability estimate

$$\|u_h\|_{\Omega_0} \leq C \|f\|_{\Omega_0}. \tag{3.34}$$

This concludes our stability analysis of (3.19).

Energy error analysis

The effect of only having discrete coercivity is mostly prominent in the energy error analysis. We let u denote the analytic solution of (2.34). We start by checking the points needed for the energy analysis of the standard case. Coercivity is already established, although only discrete. Galerkin orthogonality follows from consistency just as in the standard case. Plugging u into the bilinear form A_h gives

$$A_h(u, v) = (-\Delta u, v)_{\Omega_0} = (f, v)_{\Omega_0} = l(v), \quad \forall v \in V_h, \tag{3.35}$$

which is consistency. We refer to the appendix of Paper II for details on the integration by parts. We thus obtain the Galerkin orthogonality

$$A_h(u - u_h, v) = 0, \quad \forall v \in V_h. \tag{3.36}$$

Continuity of A_h on $\left(H_0^1(\Omega_1, \Omega_2), \left|\!\left|\!\left| \cdot \right|\!\right|\!\right|_{A_h}\right)$ is straighforward so we omit the proof. Writing the approximation error $e = u - u_h$, we would like to follow the proof of Céa's lemma by considering $\left|\!\left|\!\left| e \right|\!\right|\!\right|_{A_h}^2$ and using coercivity to go to A_h. But since $e \notin V_h$, from $u \notin V_h$, we cannot use our discrete coercivity. The remedy is an error split. We again consider an interpolation operator I_h that maps to V_h, and that also gives optimal interpolation error estimates for sufficiently smooth functions, e.g., a composite Scott-Zhang interpolation operator; see the appendix of Paper II for details. This means that

$$\left|\!\left|\!\left| u - I_h u \right|\!\right|\!\right|_{A_h} \leq C h^p \|D_x^{p+1} u\|_{\Omega_0}. \tag{3.37}$$

We use the interpolant $I_h u$ to split the approximation error e into an interpolation error ρ and a discrete error θ by

$$e = u - u_h = (u - I_h u) + (I_h u - u_h) = \rho + \theta. \tag{3.38}$$

The energy norm of the error is thus

$$|||e|||_{A_h} \leq |||\rho|||_{A_h} + |||\theta|||_{A_h}, \tag{3.39}$$

where we focus on the θ-part since the ρ-part is handled by the interpolation estimate (3.37). Since $\theta \in V_h$, we may use the discrete coercivity, but first we note that from the Galerkin orthogonality, we have

$$\begin{aligned} & 0 = A_h(e, \theta) = A_h(\rho + \theta, \theta), \\ \implies \quad & A_h(\theta, \theta) = A_h(-\rho, \theta). \end{aligned} \tag{3.40}$$

Following the proof of Céa's lemma using our tools, i.e., the discrete coercivity, the Galerkin orthogonality result (3.40), and the continuity of A_h, we get

$$\begin{aligned} & |||\theta|||_{A_h}^2 \leq \alpha^{-1} A_h(\theta, \theta) = \alpha^{-1} A_h(-\rho, \theta) \leq C\, |||\rho|||_{A_h}\, |||\theta|||_{A_h}, \\ \implies \quad & |||\theta|||_{A_h} \leq C\, |||\rho|||_{A_h}. \end{aligned} \tag{3.41}$$

Using this estimate in (3.39) gives

$$|||e|||_{A_h} \leq |||\rho|||_{A_h} + |||\theta|||_{A_h} \leq C\, |||\rho|||_{A_h}. \tag{3.42}$$

From this, by using the interpolation estimate (3.37), we obtain the optimal order a priori energy error estimate

$$|||u - u_h|||_{A_h} \leq C h^p \|D_x^{p+1} u\|_{\Omega_0}. \tag{3.43}$$

L^2-error analysis

The L^2-error estimate is derived as in the standard case by using the Aubin-Nitsche duality trick with some natural modifications to account for the CutFEM setting. We consider the continuous dual problem with the approximation error e as data: Find $\phi \in H^2(\Omega_0)$ such that

$$\begin{cases} -\Delta \phi = e & \text{in } \Omega_0, \\ \phi = 0 & \text{on } \partial\Omega_0, \\ [\partial_n \phi] = 0 & \text{on } \Gamma. \end{cases} \tag{3.44}$$

Following the same steps as in (3.17) with natural modifications for our case (see the proof of the estimates of the Ritz projection error in Paper II), and using the energy error estimate (3.43) gives us the optimal order a priori L^2-error estimate

$$\|u - u_h\|_{\Omega_0} \leq C h^{p+1} \|D_x^{p+1} u\|_{\Omega_0}. \tag{3.45}$$

This concludes our error analysis of (3.19).

3.3 Parabolic problem: standard case

Parabolic model problem

As our parabolic model problem, we extend the Poisson problem (2.1) to time. Let $T > 0$ be a given final time. We consider the heat equation in the space-time prism $\Omega \times (0, T]$ with given source function $f \in L^2((0, T], \Omega)$, homogeneous Dirichlet boundary conditions and initial data $u_0 \in H^2(\Omega)$. The problem is: Find $u \in H^1((0, T], L^2(\Omega)) \cap L^2((0, T], H^2(\Omega))$ such that

$$\begin{cases} \dot{u} - \Delta u = f & \text{in } \Omega \times (0, T], \\ u = 0 & \text{on } \partial\Omega \times (0, T], \\ u = u_0 & \text{in } \Omega \times \{0\}. \end{cases} \tag{3.46}$$

To obtain a corresponding finite element formulation, there are a number of different ways to include time. The most common is probably to consider a spatially semidiscrete problem with FEM in space, and then use a *finite difference* (FD) scheme to discretize time, e.g., the backward Euler method. Another way is to create a $d+1$-dimensional finite element mesh of the whole prism $\Omega \times (0, T]$, and use a *space-time* finite element formulation. We will consider neither but use ideas from both. In a typical application of overlapping meshes, the meshes are purely *spatial* discretizations, meaning that the second option is out. The time interval would be discretized separately into subintervals, just as in the first option. But depending on how the meshes move around, the standard discrete space-time product structure might be ruined. This structure is needed for a finite difference scheme, meaning that the first option is also out. The most natural choice for overlapping meshes is instead a mix between the two: using prismatic space-time cells as in the first option, but a space-time formulation as in the second. Also due to the mesh movement, the natural choice for time is to use a *discontinuous Galerkin* (dG) method. Space is handled as in the elliptic case which is with a *continuous Galerkin* (cG) method. The resulting method is therefore sometimes referred to as dG(q)cG(p), where q and p are the degrees of the piecewise polynomials used in time and space, respectively. We point out that the dG(q)cG(p)-method and the first option can coincide, e.g., dG(0) is the same scheme as backward Euler, but that the space-time formulation of dG(q)cG(p) makes it the preferred method for overlapping meshes.

Problem formulation

To formulate a corresponding dG(q)cG(p)-method for our parabolic model problem (3.46), we need a discrete space, and for that discretizations of time and space. We partition the time interval $(0, T]$ into N subintervals $I_n = (t_{n-1}, t_n]$ of length $k_n = t_n - t_{n-1}$, where $0 = t_0 < t_1 < \ \ldots \ < t_N = T$ and $n = 1, \ldots, N$. Let $\mathcal{T}$ be a finite element mesh of Ω. We denote by $V_h(t)$ the corresponding

finite element space defined by (2.8), i.e., the discrete space for the standard elliptic case. We define the dG(q)cG(p) space V_h by

$$V_h := \{v : v(\cdot, t) \in V_h(t), \forall t \in [0, T], \text{ and } v(x, \cdot)|_{I_n} \in \mathcal{P}^q(I_n), \forall x \in \Omega\}. \quad (3.47)$$

The dG(q)cG(p) space-time finite element formulation corresponding to (3.46) is: Find $u_h \in V_h$ such that

$$\begin{aligned} &\sum_{n=1}^{N} \left(\int_{I_n} (\dot{u}_h, v)_\Omega + a(u_h, v)\, \mathrm{d}t + ([u_h]_{n-1}, v_{n-1}^+)_\Omega \right) \\ &= \int_0^T (f, v)_\Omega \, \mathrm{d}t, \quad \forall v \in V_h. \end{aligned} \quad (3.48)$$

Here $a(w, v) = (\nabla w, \nabla v)_\Omega$ and the time-jump $[v]_n = v^+ - v^-$, i.e., the difference between the limits from above and below at time t_n. We take $u_{h,0}^- = P_h u_0$, where P_h is the standard $L^2(\Omega)$-projection operator and u_0 is the initial data in (3.46). We define the non-symmetric bilinear form B_h by

$$\begin{aligned} B_h(w, v) := &\sum_{n=1}^{N} \left(\int_{I_n} (\dot{w}, v)_\Omega + a(w, v)\, \mathrm{d}t \right) \\ &+ \sum_{n=1}^{N-1} ([w]_n, v_n^+)_\Omega + (w_0^+, v_0^+)_\Omega. \end{aligned} \quad (3.49)$$

With B_h we may write (3.48) more concisely as: Find $u_h \in V_h$ such that

$$B_h(u_h, v) = \int_0^T (f, v)_\Omega \, \mathrm{d}t + (u_0, v_0^+)_\Omega, \quad \forall v \in V_h. \quad (3.50)$$

Energy norms

Because of the non-symmetric temporal contribution in B_h, we will need more than one energy norm. We start with the usual trick of taking the same function as both arguments.

$$B_h(v, v) = \sum_{n=1}^{N} \left(\int_{I_n} (\dot{v}, v)_\Omega + \|v\|_a^2 \, \mathrm{d}t \right) + \sum_{n=1}^{N-1} ([v]_n, v_n^+)_\Omega + \|v_0^+\|_\Omega^2, \quad (3.51)$$

where the time-derivative terms and the time-jump terms need treatment since they can be negative. By integrating the time-derivative term, we get

$$\int_{I_n} (\dot{v}, v)_\Omega \, \mathrm{d}t = \int_{I_n} \frac{1}{2} \partial_t \|v\|_\Omega^2 \, \mathrm{d}t = \frac{1}{2} \|v_n^-\|_\Omega^2 - \frac{1}{2} \|v_{n-1}^+\|_\Omega^2. \quad (3.52)$$

From $(a-b)^2 = a^2 - 2ab + b^2$, the time-jump terms are

$$([v]_n, v_n^+)_\Omega = \frac{1}{2}\|[v]_n\|_\Omega^2 - \frac{1}{2}\|v_n^-\|_\Omega^2 + \frac{1}{2}\|v_n^+\|_\Omega^2. \tag{3.53}$$

Using (3.52) and (3.53) in (3.51) gives a telescoping sum, which leaves us with

$$B_h(v,v) = \sum_{n=1}^{N}\int_{I_n}\|v\|_a^2\,\mathrm{d}t + \frac{1}{2}\sum_{n=1}^{N-1}\|[v]_n\|_\Omega^2 + \frac{1}{2}\|v_N^-\|_\Omega^2 + \frac{1}{2}\|v_0^+\|_\Omega^2. \tag{3.54}$$

We thus define the space-time energy norm $|||\cdot|||_{B_h}$ on $L^2([0,T],H_0^1(\Omega))$ by

$$|||v|||_{B_h}^2 := \sum_{n=1}^{N}\int_{I_n}\|v\|_a^2\,\mathrm{d}t + \sum_{n=1}^{N-1}\|[v]_n\|_\Omega^2 + \|v_N^-\|_\Omega^2 + \|v_0^+\|_\Omega^2. \tag{3.55}$$

From the preceding treatment of $B_h(v,v)$, we immediately obtain the coercivity: $\exists\beta > 0$ such that

$$B_h(v,v) \geq \beta\,|||v|||_{B_h}^2\,, \quad \forall v \in H^1(\cup_n I_n, L^2(\Omega)) \cap L^2([0,T],H_0^1(\Omega)). \tag{3.56}$$

Note that this coercivity does not hold on the whole space of the energy norm $|||\cdot|||_{B_h}$, but that it definitely holds on the discrete space V_h. This is so since we need extra temporal regularity for the left-hand side to be well-defined. However, this coercivity is not enough for an energy analysis.

With the proof of Céa's lemma in mind, we need something more than the $|||\cdot|||_{B_h}$-coercivity, since $|||\cdot|||_{B_h}$ does not include the time derivative. We therefore need to consider other space-time energy norms. A common way to include the time derivative in an energy norm is to use the H^{-1}-norm; see, e.g., [22] for a good summary of energy methods for parabolic problems. The H^{-1}-norm is the norm of the dual space to H_0^1. For situations such as the current, this norm is often defined with the help of the solution operator T (not to be confused with final time T) to the corresponding elliptic problem; see, e.g., [23]. We thus define $T : L^2(\Omega) \to H^2(\Omega) \cap H_0^1(\Omega)$ by

$$u = Tf. \tag{3.57}$$

This means that $T = (-\Delta)^{-1}$. For $w \in L^2(\Omega)$ and $v \in H_0^1(\Omega)$, we have

$$(w,v)_\Omega = (\underbrace{-\Delta T}_{=\mathbb{1}}w, v)_\Omega = (\nabla Tw, \nabla v)_\Omega - \underbrace{(\partial_n Tw, v)_{\partial\Omega}}_{=0, v|_{\partial\Omega}=0} = a(Tw,v). \tag{3.58}$$

We define the $H^{-1}(\Omega)$-norm for a function $v \in L^2(\Omega)$ by

$$\|v\|_{-1}^2 := (v, Tv)_\Omega. \tag{3.59}$$

This norm figures in the following useful isometry and inequality:

$$\|w\|_{-1} = \|Tw\|_a, \quad \forall w \in L^2(\Omega), \tag{3.60}$$

$$(w, v)_\Omega \leq \|w\|_{-1}\|v\|_a, \quad \forall w \in L^2(\Omega) \text{ and } \forall v \in H_0^1(\Omega). \tag{3.61}$$

The isometry (3.60) is shown by using the norm definition and (3.58). The inequality (3.61) is shown by using (3.58) and the isometry.

To put the spotlight on the approach of using the H^{-1}-norm of the time derivative, we will use a *reconstruction*. We let $\cdot^\circ$ denote the interior of a set, thus $I_n^\circ = (t_{n-1}, t_n)$. For simplicity, we define the reconstruction $\widetilde{w} \in C([0,T]) \cap \mathcal{P}^{q+1}(\cup_n I_n^\circ)$ of a function w by $\widetilde{w}(0) = 0$ and

$$\sum_{n=1}^{N} \int_{I_n} \dot{\widetilde{w}} v \,\mathrm{d}t = \sum_{n=1}^{N} \int_{I_n} \dot{w} v \,\mathrm{d}t + \sum_{n=1}^{N-1} [w]_n v_n^+ + w_0^+ v_0^+, \quad \forall v \in \mathcal{P}^q(\cup_n I_n^\circ). \tag{3.62}$$

Meaning that $\widetilde{w}$ is a *continuous* piecewise polynomial of degree $q+1$, defined by testing against *discontinuous* piecewise polynomials of degree q. The extra degree of freedom in polynomial degree $(q+1)$ is payed for by giving up the degree of freedom in discontinuity at t_{n-1}.

With the reconstruction and the H^{-1}-norm, we define two space-time energy norms by

$$|||v|||_Y^2 := \sum_{n=1}^{N} \int_{I_n} \|v\|_a^2 \,\mathrm{d}t, \tag{3.63}$$

$$|||v|||_X^2 := \sum_{n=1}^{N} \int_{I_n} \|\dot{\widetilde{v}}\|_{-1}^2 \,\mathrm{d}t + |||v|||_Y^2 \,. \tag{3.64}$$

Note that $|||v|||_Y \leq |||v|||_X$ and $|||v|||_Y \leq |||v|||_{B_h}$. The X-norm is the main space-time energy norm and the Y-norm is an auxiliary norm.

These energy norms may be used to obtain an inf-sup condition that can be used in place of a coercivity in a Céa's lemma type energy argument. To derive this inf-sup condition, two estimates are needed. The first one is a perturbed coercivity on V_h and the second is a boundedness result. Letting P_h be the $L^2(\Omega)$-projection to $V_h(t)$, we note that $P_h T\dot{\widetilde{v}} \in V_h$ for $v \in V_h$. The discrete perturbed coercivity is: $\exists \delta, \xi > 0$ such that

$$B_h(v, v + \delta P_h T\dot{\widetilde{v}}) \geq \xi \,|||v|||_X^2 \,, \quad \forall v \in V_h. \tag{3.65}$$

The second estimate is: $\exists C > 0$ such that

$$\left|\left|\left| P_h T\dot{\widetilde{v}} \right|\right|\right|_Y \leq C \,|||v|||_X \,, \quad \forall v \in V_h. \tag{3.66}$$

We omit the proofs of these estimates since they are a bit too technical for the purpose of this section. For details, we instead refer to literature and the proof of the discrete perturbed coercivity in Paper I. One thing we point out though is that H^1-stability of P_h is essential to obtain these estimates, i.e., boundedness of P_h in the spatial energy norm $\|\cdot\|_a$. Using (3.65) and (3.66), we have for $v \in V_h \setminus \{0\}$ that

$$\begin{aligned} \xi \left|\!\left|\!\left| v \right|\!\right|\!\right|_X^2 \leq B_h(v, v + \delta P_h T\dot{\tilde{v}}) &= \frac{B_h(v, v + \delta P_h T\dot{\tilde{v}})}{\left|\!\left|\!\left| v + \delta P_h T\dot{\tilde{v}} \right|\!\right|\!\right|_Y} \left|\!\left|\!\left| v + \delta P_h T\dot{\tilde{v}} \right|\!\right|\!\right|_Y \\ &\leq C \frac{B_h(v, v + \delta P_h T\dot{\tilde{v}})}{\left|\!\left|\!\left| v + \delta P_h T\dot{\tilde{v}} \right|\!\right|\!\right|_Y} \left|\!\left|\!\left| v \right|\!\right|\!\right|_X . \end{aligned} \tag{3.67}$$

From this we may obtain the discrete inf-sup condition

$$\left|\!\left|\!\left| w \right|\!\right|\!\right|_X \leq C \sup_{v \in V_h \setminus \{0\}} \frac{B_h(w, v)}{\left|\!\left|\!\left| v \right|\!\right|\!\right|_Y}, \quad \forall w \in V_h. \tag{3.68}$$

Stability analysis

For elliptic problems, the stability and error analyses were not really connected. One could do the one without the other. This is also the case for the energy analysis of parabolic problems. However, for the L^2-analysis of parabolic problems, the stability analysis is crucial. This is because stability estimates of a dual problem are needed; see, e.g., [24, 25] or the stability analysis in Paper II. The right-hand side in this dual problem is zero. Therefore, we will only consider stability estimates with $f = 0$ here. We present three stability estimates.

Just as in the previous cases, we obtain the basic stability estimate by taking $v = u_h$ in the finite element formulation. Doing this in (3.50) with $f = 0$, using the $\left|\!\left|\!\left|\cdot\right|\!\right|\!\right|_{B_h}$-coercivity, and the definition of $\left|\!\left|\!\left|\cdot\right|\!\right|\!\right|_{B_h}$ (3.55), we get

$$\beta \left|\!\left|\!\left| u_h \right|\!\right|\!\right|_{B_h}^2 \leq B_h(u_h, u_h) = (u_0, u_{h,0}^+)_\Omega \leq \|u_0\|_\Omega \left|\!\left|\!\left| u_h \right|\!\right|\!\right|_{B_h} . \tag{3.69}$$

From this we obtain the corresponding *basic* stability estimate

$$\sum_{n=1}^{N} \int_{I_n} \|u_h\|_a^2 \,\mathrm{d}t + \sum_{n=1}^{N-1} \|[u_h]_n\|_\Omega^2 + \|u_{h,N}^-\|_\Omega^2 + \|u_{h,0}^+\|_\Omega^2 \leq C\|u_0\|_\Omega^2. \tag{3.70}$$

One may also use the perturbed coercivity (3.65) in the same way to obtain a stability estimate for u_h in the X-norm; see the stability analysis of Paper I for details.

We move on to the remaining two stability estimates. We omit the proofs of these estimates since they are a bit too technical for the purpose of this

section. For details, we instead refer to [24, 25] or the stability analysis in Paper II. Recall that we denote the spatial discrete space by $V_h(t)$, i.e., for $v \in V_h$, $v(\cdot, t) \in V_h(t)$ for all $t \in [0, T]$. We consider the discrete Laplacian $\Delta_h : H_0^1(\Omega) \to V_h(t)$, defined by

$$(-\Delta_h w, v)_\Omega = a(w, v), \quad \forall v \in V_h(t). \tag{3.71}$$

By testing with $-\Delta_h u_h$, then $(t - t_{n-1})\dot{u}_h$, and finally $[u_h]_{n-1}$, we obtain three intermediate estimates. Using these three estimates and the basic stability estimate (3.70), we may obtain the *strong* stability estimate

$$\sum_{n=1}^{N} t_n \int_{I_n} \|\dot{u}_h\|_\Omega^2 + \|\Delta_h u_h\|_\Omega^2 \,\mathrm{d}t + \sum_{n=2}^{N} \frac{t_n}{k_n} \|[u_h]_{n-1}\|_\Omega^2 \leq C \|u_0\|_\Omega^2. \tag{3.72}$$

The strong stability estimate is used to obtain the *main* stability estimate

$$\|u_{h,N}^-\|_\Omega + \sum_{n=1}^{N} \int_{I_n} \|\dot{u}_h\|_\Omega + \|\Delta_h u_h\|_\Omega \,\mathrm{d}t + \sum_{n=1}^{N} \|[u_h]_{n-1}\|_\Omega \leq C_1 \|u_0\|_\Omega, \tag{3.73}$$

where $C_1 = C(\log(t_N/k_1) + 1)^{1/2}$ and $C > 0$. It is the corresponding main stability estimate for the discrete dual problem that is crucial to the L^2-error analysis.

Energy error analysis

Here we derive an energy error estimate in our main space-time energy norm $|||\cdot|||_X$. Again recall the three ingredients for the proof of Céa's lemma: coercivity, Galerkin orthogonality, and continuity.

Due to the non-symmetric temporal terms in the bilinear form, we have instead of coercivity, the discrete inf-sup condition (3.68). In the elliptic CutFEM case, we saw that when the coercivity was only discrete, an error split was needed. This is needed here as well, since the inf-sup condition is discrete. Letting $\bar{\cdot}$ denote something related to space-time, we introduce a space-time interpolation operator $\bar{I}_h$ that maps to V_h and that has optimal interpolation properties. For a function v with sufficient regularity we thus assume that we have the interpolation error estimate

$$\left|\left|\left| v - \bar{I}_h v \right|\right|\right|_X \leq C k^{q+1/2} F_k(v) + C h^p F_h(v), \tag{3.74}$$

where k and h are the largest time step and simplex diameter, respectively. $F_k(v)$ and $F_h(v)$ are expressions involving derivatives of the function v with respect to space and time; see, e.g., the appendix of Paper I for details on space-time interpolation. We use the interpolant $\bar{I}_h u$ to split the approximation error e into an interpolation error ρ and a discrete error θ by

$$e = u - u_h = (u - \bar{I}_h u) + (\bar{I}_h u - u_h) = \rho + \theta. \tag{3.75}$$

For the Galerkin orthogonality, we plug the analytic solution u into B_h. Using that $[u]_n = 0$ from its regularity, and integrating by parts in space we obtain consistency just as in the previous cases. This gives us the Galerkin orthogonality

$$B_h(u - u_h, v) = 0, \quad \forall v \in V_h. \tag{3.76}$$

Recall from the elliptic CutFEM case that a twist on the Galerkin orthogonality was needed when working with the discrete error. Since we do that here as well, from using the discrete inf-sup condition, we do the following: For $v \in V_h$, we have

$$\begin{aligned} & 0 = B_h(e, v) = B_h(\rho + \theta, v), \\ \implies \quad & B_h(\theta, v) = B_h(-\rho, v). \end{aligned} \tag{3.77}$$

Continuity follows naturally with the X- and Y-norms. By using the reconstruction, the H^{-1}-norm, and standard estimates, we get: $\exists C > 0$ such that for any functions w and v with sufficient regularity we have that

$$B_h(w, v) \leq C \left|\!\left|\!\left| w \right|\!\right|\!\right|_X \left|\!\left|\!\left| v \right|\!\right|\!\right|_Y . \tag{3.78}$$

We thus have the three required ingredients for a Céa's lemma type energy argument. The energy norm of the error is

$$\left|\!\left|\!\left| e \right|\!\right|\!\right|_X \leq \left|\!\left|\!\left| \rho \right|\!\right|\!\right|_X + \left|\!\left|\!\left| \theta \right|\!\right|\!\right|_X , \tag{3.79}$$

where we focus on the θ-part since the ρ-part is handled by the interpolation estimate (3.74). Since $\theta \in V_h$, we follow the proof of Céa's lemma using our tools, i.e., the discrete inf-sup condition (3.68), the Galerkin orthogonality result (3.77), and the continuity of B_h. We thus get

$$\begin{aligned} \left|\!\left|\!\left| \theta \right|\!\right|\!\right|_X &\leq C \sup_{v \in V_h \setminus \{0\}} \frac{B_h(\theta, v)}{\left|\!\left|\!\left| v \right|\!\right|\!\right|_Y} = C \sup_{v \in V_h \setminus \{0\}} \frac{B_h(-\rho, v)}{\left|\!\left|\!\left| v \right|\!\right|\!\right|_Y} \\ &\leq C \sup_{v \in V_h \setminus \{0\}} \frac{\left|\!\left|\!\left| \rho \right|\!\right|\!\right|_X \left|\!\left|\!\left| v \right|\!\right|\!\right|_Y}{\left|\!\left|\!\left| v \right|\!\right|\!\right|_Y} = C \left|\!\left|\!\left| \rho \right|\!\right|\!\right|_X . \end{aligned} \tag{3.80}$$

Using this estimate in (3.79) gives

$$\left|\!\left|\!\left| e \right|\!\right|\!\right|_X \leq \left|\!\left|\!\left| \rho \right|\!\right|\!\right|_X + \left|\!\left|\!\left| \theta \right|\!\right|\!\right|_X \leq C \left|\!\left|\!\left| \rho \right|\!\right|\!\right|_X . \tag{3.81}$$

From which, by using the interpolation estimate (3.74), we obtain the optimal order a priori energy error estimate

$$\left|\!\left|\!\left| u - u_h \right|\!\right|\!\right|_X \leq C k^{q+1/2} F_k(u) + C h^p F_h(u). \tag{3.82}$$

L^2-error analysis

The L^2-error analysis for parabolic problems is heavily dependent on the stability analysis. Just as for the elliptic case, one uses a dual problem, albeit a discrete one instead of a continuous. The greatest differences are probably that in the parabolic case, stability of the dual problem is used, and corresponding energy error estimates are not used. The required stability of the dual problem is stronger than what the basic stability estimate provides. This makes the L^2-analysis more demanding than the energy analysis. The derivation of the error estimate below follows [24,25] and we refer to those works for details. The general steps are:

1. Error split: use an interpolant and focus on the discrete error θ.
2. Discrete dual problem: discrete error is initial data, may thus go to B_h.
3. Galerkin orthogonality: use twist to switch to the interpolation error ρ
4. Hölder inequalities in space and time: $\|wv\|_{L^1} \leq C\|w\|_{L^p}\|v\|_{L^q}$.
5. Stability estimate: get back data in L^2-norm for cancellation.
6. Optimal interpolation error estimates for ρ-factors.

By comparing these steps to the steps of the L^2-error analysis for the elliptic case, we note the following similarites:

- Both use a dual problem to go from the L^2-inner product to the bilinear form. Step 2 in parabolic case. Steps 1 and 2 in elliptic case.
- Both use Galerkin orthogonality to get an interpolation error. Step 3 in parabolic case. Step 3 in elliptic case.
- Both use inequalities to go from product terms to norms. Step 4 in parabolic case. Step 4 in elliptic case.
- Both use optimal order estimates for the interpolation error. Step 6 in parabolic case. Step 5 in elliptic case.

We also point out the following difference. In the elliptic case, the interpolation error estimate is used to get back the dual data (right-hand side data) for cancellation, and together with the energy error estimate give the optimal order. In the parabolic case, a stability estimate is used to get back the dual data (initial data) for cancellation.

We start by integrating by parts in B_h with respect to time, which gives the alternative form

$$\begin{aligned} B_h(w,v) = &\sum_{n=1}^{N} \int_{I_n} (w, -\dot{v})_\Omega \,\mathrm{d}t + \sum_{n=1}^{N} \int_{I_n} a(w,v)\,\mathrm{d}t \\ &+ \sum_{n=1}^{N-1} (w_n^-, -[v]_n)_\Omega + (w_N^-, v_N^-)_\Omega. \end{aligned} \tag{3.83}$$

See the section about B_h in either Paper I or Paper II for details. Using this alternative form of B_h, we consider a *discrete* dual problem to (3.50) that goes backwards in time, has $f = 0$, and initial data $z_{h,N}^+$. The problem is: Find $z_h \in V_h$ such that

$$B_h(v, z_h) = (v_N^-, z_{h,N}^+)_\Omega, \quad \forall v \in V_h. \tag{3.84}$$

Since the dual problem is discrete, it will only work with a discrete error, we therefore need to perform an error split. For that we need an interpolant. Recall that we denote the spatial discrete space by $V_h(t)$, i.e., for $v \in V_h$, $v(\cdot, t) \in V_h(t)$ for all $t \in [0, T]$. We consider the Ritz projection operator $R_h : H_0^1(\Omega) \to V_h(t)$, defined by

$$a(R_h w, v) = a(w, v), \quad \forall v \in V_h(t). \tag{3.85}$$

For $q \in \mathbb{N}$ and $n = 1, \ldots, N$, we also define the temporal interpolation operator $\tilde{I}^n : C(I_n) \to \mathcal{P}^q(I_n)$ by

$$(\tilde{I}^n v)_n^- = v_n^-, \tag{3.86a}$$

and with the additional condition for $q \geq 1$,

$$\int_{I_n} \tilde{I}^n v w \,\mathrm{d}t = \int_{I_n} v w \,\mathrm{d}t, \quad \forall w \in \mathcal{P}^{q-1}(I_n). \tag{3.86b}$$

For a sufficiently smooth function v, we have the following error estimates for R_h and $\tilde{I}^n$:

$$\|v - R_h v\|_\Omega \leq C h^{p+1} \|D_x^{p+1} v\|_\Omega, \tag{3.87}$$

$$\|v - \tilde{I}^n v\|_{\Omega, I_n} \leq C k_n^{q+1} \|\dot{v}^{(q+1)}\|_{\Omega, I_n}, \tag{3.88}$$

where $\|v\|_{\Omega, I_n} = \max_{t \in I_n} \|v\|_\Omega$. We use the interpolant $\tilde{u} = \tilde{I}^n R_h u \in V_h$ to split the approximation error e into an interpolation error ρ and a discrete error θ by

$$e = u - u_h = (u - \tilde{u}) + (\tilde{u} - u_h) = \rho + \theta. \tag{3.89}$$

The Galerkin orthogonality is used here as well, and since we will work with a discrete error, the twist (3.77) is needed. We now have all the tools needed

to start deriving the L^2-error estimate. We consider the $L^2(\Omega)$-norm of the approximation error at the final time and perform the first step. The error is

$$\|u(t_N) - u_{h,N}^-\|_\Omega = \|e_N^-\|_\Omega \leq \|\rho_N^-\|_\Omega + \|\theta_N^-\|_\Omega. \tag{3.90}$$

The ρ-part is estimated by using the definition of $\tilde{I}^n$ (3.86a) and the estimate for the Ritz projection error (3.87). We thus have

$$\|\rho_N^-\|_\Omega = \|u(t_N) - (\tilde{I}^n R_h u)_N^-\|_\Omega \leq Ch^{p+1}\|D_x^{p+1}u(t_N)\|_\Omega. \tag{3.91}$$

As in the previous error analyses with error splits, the θ-part needs the most treatment. We take the initial data of the discrete dual problem to be $z_{h,N}^+ = \theta_N^-$. Performing the second and third steps, i.e., using the discrete dual problem and the Galerkin orthogonality twist, we obtain a representation of the discrete error by

$$\begin{aligned}
\|\theta_N^-\|_\Omega^2 &= (\theta_N^-, \theta_N^-)_\Omega = (\theta_N^-, z_{h,N}^+)_\Omega \overset{2.}{=} B_h(\theta, z_h) \overset{3.}{=} B_h(-\rho, z_h) \\
&= \sum_{n=1}^N \int_{I_n} (\rho, \dot{z}_h)_\Omega \,\mathrm{d}t - \sum_{n=1}^N \int_{I_n} a(\rho, z_h)\,\mathrm{d}t \\
&\quad + \sum_{n=1}^{N-1} (\rho_n^-, [z_h]_n)_\Omega - (\rho_N^-, z_{h,N}^-)_\Omega.
\end{aligned} \tag{3.92}$$

The terms are treated separately, but we omit a rigorous treatment here since it is a bit too technical for the purpose of this section. We refer to the literature and the error analysis in Paper II for details. One thing we point out though is that giving the second term a special treatment for $q = 1$ is the key to obtain superconvergence. Recall the fourth step. For a generic product term in (3.92), the general approach is to use the Hölder inequality $(\rho, z_h) \leq \|\rho\|\|z_h\|$ for space, i.e., the Cauchy–Schwarz inequality, and then the Hölder inequality $\int_{I_n} \|\rho\|\|z_h\|\,\mathrm{d}t \leq \|\rho\|_{I_n} \int_{I_n} \|z_h\|\,\mathrm{d}t$ for time. Recall the fifth step. The z_h-factors are estimated by the corresponding *main* stability estimate for z_h. The main stability estimate for u_h (3.73) is a temporal L^1-estimate of various terms in terms of the initial data. The initial data for z_h is $z_{h,N}^+ = \theta_N^-$. Thus, collecting the ρ-factors in $F(u)$, we get from (3.92) that

$$\|\theta_N^-\|_\Omega^2 \leq C_N F_q(u)\|\theta_N^-\|_\Omega, \quad \Longrightarrow \quad \|\theta_N^-\|_\Omega \leq C_N F_q(u), \tag{3.93}$$

where $C_N = C(\log(t_N/k_N)+1)^{1/2}$. Combining the estimates for the ρ-part and the θ-part, and doing the sixth and final step, i.e., using standard estimates for the ρ-terms in $F_q(u)$, we may obtain the optimal order a priori L^2-error

estimate for $q = 0, 1$,

$$\begin{aligned} &\|u(t_N) - u_{h,N}^-\|_\Omega \\ &\leq C_N \max_{1\leq n\leq N} \left\{ k_n^{2q+1} \|\dot{u}^{(2q+1)}\|_{\Omega,I_n} + h^{p+1} \|D_x^{p+1} u\|_{\Omega,I_n} \right\}. \end{aligned} \tag{3.94}$$

We note the so called *superconvergence* with respect to the time step, which is of the *third* order for dG(1).

3.4 Parabolic problem: overlapping meshes

Here we summarize the analyses presented in Paper I and Paper II. To motivate the analysis approaches taken in these papers, we conclude some main points of consideration for finite element analysis of parabolic problems based on the contents of the previous sections:

- L^2-analysis: Requires more stability, thus more demanding than energy analysis.
- Energy analysis: As in the elliptic cases, a Céa's lemma type argument together with interpolation is used.
- Energy analysis: The time derivative needs to be handled. Standard way is to use the H^{-1}-norm.
- Energy analysis: The non-symmetric temporal contribution to the bilinear form leads to the need for an inf-sup condition derived from a perturbed coercivity. The inf-sup condition is used in the Céa's lemma type argument instead of a standard coercivity as in the elliptic case.

In both papers we consider the same parabolic model problem on two meshes: one background mesh $\mathcal{T}_0$ and one overlapping mesh $\mathcal{T}_G$ that is allowed to move around in the background mesh. The main difference is the discrete representation of the mesh movement which leads to quite different space-time discretizations. This in turn leads to different finite element formulations, both concerning the discrete spaces and the variational equations, which allow for different analysis approaches. In general the mesh movement may either be continuous or discontinuous. We have considered the simplest case of both of these two types, which we refer to as cG(1) and dG(0) mesh movement, where cG(r) and dG(r) stand for continuous and discontinuous Galerkin of order r, respectively. The mesh movements are named after what type of function the location of the overlapping mesh is when considered as a function of time.

In a very first study, we considered cG(1) mesh movement and attempted to follow the L^2-analysis methodology presented by Eriksson and Johnson in [24,

25]. However, due to the space-time discretization resulting from the cG(1) mesh movement, the L^2-analysis failed. The study, containing partial results of the incomplete L^2-analysis, was presented in the MSc-thesis [26]. With that very first study as a starting point, we retreated in two directions by considering a less demanding energy analysis and the simpler dG(0) mesh movement, meaning less complicated space-time discretization. This has resulted in two new studies with complete analyses that are presented in Paper I and Paper II. Paper I presents an energy analysis for cG(1) mesh movement and Paper II presents an L^2-analysis for dG(0) mesh movement. Table 1 gives an overview of the various studies of CutFEM for the heat equation on two overlapping meshes performed so far.

	dG(0) mesh movement	cG(1) mesh movement
Energy analysis	-	Paper I ✓
L^2-analysis	Paper II ✓	MSc-thesis [26] ✗

Table 1: Overview of studies of CutFEM for the heat equation on two overlapping meshes based on analysis and mesh movement type. The checkmark indicates a complete analysis and the x-mark one that is currently incomplete.

To go through the studies in Paper I and Paper II, we start by presenting the parabolic model problem that both have in common.

Parabolic model problem
For $d = 1, 2$, or 3, let $\Omega_0 \subset \mathbb{R}^d$ be a bounded convex domain with polygonal boundary $\partial\Omega_0$. Let $T > 0$ be a given final time. Let $G \subset \Omega_0 \subset \mathbb{R}^d$ be another bounded domain with polygonal boundary ∂G. We let the *location* of G be time-dependent by prescribing for G a velocity $\mu : [0, T] \to \mathbb{R}^d$. This means that G and ∂G are functions of time, i.e., $G = G(t)$ and $\partial G = \partial G(t)$ for $t \in [0, T]$. We point out that the shape of G remains the same for all times. From Ω_0 and $G(t)$, we define the following two domains:

$$\Omega_1(t) := \Omega_0 \setminus (G(t) \cup \partial G(t)), \quad \Omega_2(t) := \Omega_0 \cap G(t), \tag{3.95}$$

with boundaries $\partial\Omega_1(t)$ and $\partial\Omega_2(t)$, respectively. Let the common boundary between $\Omega_1(t)$ and $\Omega_2(t)$ be

$$\Gamma(t) := \partial\Omega_1(t) \cap \partial\Omega_2(t). \tag{3.96}$$

Note that for any $t \in [0, T]$, we have the partition

$$\Omega_0 = \Omega_1(t) \cup \Gamma(t) \cup \Omega_2(t). \tag{3.97}$$

We consider the heat equation in $\Omega_0 \times (0,T]$ with given source function $f \in L^2((0,T],\Omega_0)$, homogeneous Dirichlet boundary conditions, and initial data $u_0 \in H^2(\Omega_0) \cap H_0^1(\Omega_0)$. The problem is: Find $u \in H^1((0,T],L^2(\Omega_0)) \cap L^2((0,T],H^2(\Omega_0) \cap H_0^1(\Omega_0))$ such that

$$\begin{cases} \dot{u} - \Delta u = f & \text{in } \Omega_0 \times (0,T], \\ u = 0 & \text{on } \partial\Omega_0 \times (0,T], \\ u = u_0 & \text{in } \Omega_0 \times \{0\}. \end{cases} \tag{3.98}$$

3.4.1 cG(1) mesh movement (Paper I)

Problem formulation

As a discrete counterpart to the movement of the domain G, we prescribe a cG(1) movement for the overlapping mesh $\mathcal{T}_G$. By this we mean that the location of the overlapping mesh $\mathcal{T}_G$ is a cG(1) function with respect to time, i.e., *continuous* on $[0,T]$ and *linear* on each I_n; see Figure 6.

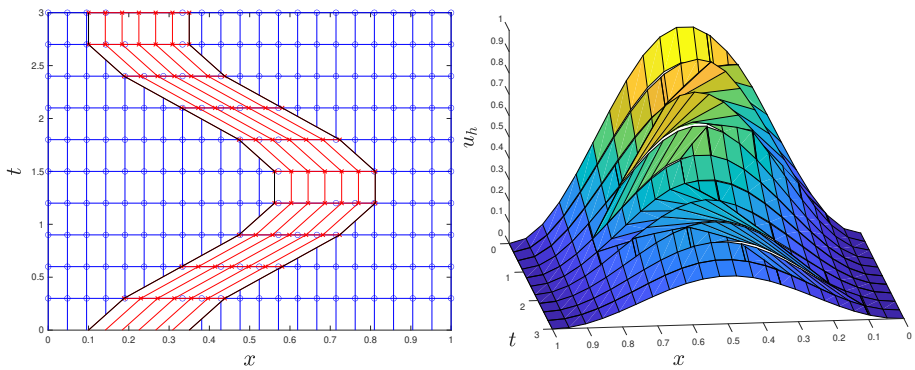

Figure 6: Overlapping meshes and solution with cG(1) mesh movement. *Left*: Space-time discretization for $d = 1$. *Right*: A corresponding dG(1)cG(1) cut finite element solution of the heat equation.

Just as in the elliptic CutFEM case, we use two spatial standard finite element spaces corresponding to the meshes $\mathcal{T}_0$ and $\mathcal{T}_G$, to define a *broken spatial* finite element space. Since the location of $\mathcal{T}_G$ varies in time, we get a different broken spatial finite element space $V_h(t)$ for every $t \in [0,T]$. Following the discretization steps of the parabolic standard case, we define two dG(q)cG(p) spaces, one for each mesh. Then, as in the elliptic CutFEM case, we use restrictions of functions from these spaces to define our *main* broken space-time discrete space V_h. We refer to Paper I for details on the construction of these spaces.

Since the location of $\Gamma(t)$ may be different for every $t \in [0,T]$, we use a time-dependent spatial bilinear form $A_{h,t}$ corresponding to the bilinear form A_h in the elliptic CutFEM case. The dG(q)cG(p) space-time finite element formulation corresponding to (3.98) with cG(1) mesh movement is: Find $u_h \in V_h$ such that

$$\begin{aligned}
&\sum_{i=1}^{2}\sum_{n=1}^{N}\int_{I_n}(\dot{u}_h, v)_{\Omega_i(t)}\,\mathrm{d}t + \sum_{n=1}^{N}\int_{I_n} A_{h,t}(u_h, v)\,\mathrm{d}t \\
&+\sum_{i=1}^{2}\sum_{n=1}^{N}([u_h]_{n-1}, v_{n-1}^{+})_{\Omega_{i,n-1}} + \sum_{n=1}^{N}\int_{\bar{\Gamma}_n} -\bar{n}^t[u_h]v_\sigma\,\mathrm{d}\bar{s} \qquad (3.99)\\
&= \sum_{i=1}^{2}\sum_{n=1}^{N}\int_{I_n}(f, v)_{\Omega_i(t)}\,\mathrm{d}t, \quad \forall v \in V_h.
\end{aligned}$$

Note the special last term on the left-hand side. It is present due to the continuous mesh movement. This term mimics the standard time-jump terms in the dG-method but over the space-time interface $\bar{\Gamma}_n$. The term contains the temporal component $\bar{n}^t$ of the space-time normal vector to $\bar{\Gamma}_n$, which makes the term proportional to $|\mu|$. Thus this term is typically quite small but it is essential for obtaining the basic stability estimate. We refer to Paper I for details on the finite element formulation.

We define the non-symmetric bilinear form B_h by

$$\begin{aligned}
B_h(w,v) := &\sum_{i=1}^{2}\sum_{n=1}^{N}\int_{I_n}(\dot{w}, v)_{\Omega_i(t)}\,\mathrm{d}t + \sum_{n=1}^{N}\int_{I_n} A_{h,t}(w, v)\,\mathrm{d}t \\
&+\sum_{i=1}^{2}\sum_{n=1}^{N-1}([w]_n, v_n^{+})_{\Omega_{i,n}} + \sum_{i=1}^{2}(w_0^{+}, v_0^{+})_{\Omega_{i,0}} \qquad (3.100)\\
&+\sum_{n=1}^{N}\int_{\bar{\Gamma}_n} -\bar{n}^t[w]v_\sigma\,\mathrm{d}\bar{s}.
\end{aligned}$$

We may then write (3.99) in compact form as: Find $u_h \in V_h$ such that

$$B_h(u_h, v) = \sum_{i=1}^{2}\int_0^T (f, v)_{\Omega_i(t)}\,\mathrm{d}t + \sum_{i=1}^{2}(u_0, v_0^{+})_{\Omega_{i,0}}, \quad \forall v \in V_h. \qquad (3.101)$$

A comment on standard analysis approaches

We start by considering a potential L^2-analysis of (3.101). The L^2-analysis of a parabolic problem requires the *main* stability estimate which in turn requires the *strong* stability estimate. Recall these estimates for the parabolic standard case given by (3.73) and (3.72), respectively. To start deriving the strong stability

estimate we recall that one tests with the discrete Laplacian of the finite element solution. The discrete Laplacian is defined using the spatial finite element space and the spatial bilinear form, which in our case are $V_h(t)$ and $A_{h,t}$, respectively. Since both of them are time-dependent due to the cG(1) mesh movement, the discrete Laplacian for our case is also time-dependent. For our case, the discrete Laplacian $\Delta_{h,t} : H^1(\Omega_1(t), \Omega_2(t)) \to V_h(t)$ is defined by

$$(-\Delta_{h,t} w, v)_{\Omega_0} = A_{h,t}(w, v), \quad \forall v \in V_h(t). \tag{3.102}$$

This definition gives our discrete Laplacian an intrinsic time-dependence. The time-dependence of $\Delta_{h,t} u_h$ is therefore a combination of the one from $\Delta_{h,t}$ and the one from $u_h \in V_h$, which means that $-\Delta_{h,t} u_h \notin V_h$. Thus we cannot test with $-\Delta_{h,t} u_h$ which means that we cannot obtain the strong stability estimate. This means that we cannot perform an L^2-analysis of (3.101) using the standard tools.

We add that in [26], an attempt at the L^2-analysis for cG(1) mesh movement was made, but did not succeed because of the mentioned issues. However, partial results of this incomplete analysis together with numerical results, suggested that the L^2-error's dependence on the time step k and the mesh size h for $q = 0, 1$ was

$$error \sim k^{2q+1} + h^{p+1} + |\mu| k^{q+1} \Longrightarrow \begin{cases} k^1 + h^{p+1}, & \text{for } q = 0, \\ (k^1 + |\mu|)k^2 + h^{p+1}, & \text{for } q = 1. \end{cases} \tag{3.103}$$

The convergence orders for $q = 0$ thus seem to be the same as in the standard case. For $q = 1$ however, the superconvergence with respect to the time step seems to be lost with cG(1) mesh movement. Since when the overlapping mesh $\mathcal{T}_G$ moves with speed $|\mu| > 0$, the k-convergence order seems to be of the *second* order instead of the *third* as in the standard case. Note that in the region $k \gg |\mu|$ the k-convergence seems to be of the expected third order though. In numerical simulations, the problem dimensions typically do not allow $|\mu|$ to become arbitrarily large, meaning that $k \ll |\mu|$ could be needed to verify the drop in convergence order. This can make a numerical convergence study quite demanding, not only because the k's might need to be small but also because h might be required to be small too so as not to interfere with the numerical k-convergence order. It is therefore not surprising if one appears to have numerical superconvergence for $|\mu| > 0$ since the k's needed to verify second order convergence might be required to be very small.

We go back to considering potential analysis approaches for cG(1) mesh movement. Since we are not able to establish the strong stability estimate for the case with cG(1) mesh movement, we retreat to a less demanding energy analysis. For a general energy analysis the first main concern is how to treat

the time derivative. The standard way is to use the H^{-1}-norm defined by (3.59), where the operator T is defined by (3.57). For the derivation of the discrete inf-sup condition (3.68), the $L^2(\Omega_0)$-projection to $V_h(t)$ is used. For our case, the $L^2(\Omega_0)$-projection $P_{h,t} : L^2(\Omega_0) \to V_h(t)$ is defined by

$$(P_{h,t}w, v)_{\Omega_0} = (w, v)_{\Omega_0}, \quad \forall v \in V_h(t). \tag{3.104}$$

For the derivation of the inf-sup condition to work, we need the $L^2(\Omega_0)$-projection of $T\dot{\tilde{v}}$ to lie in V_h for $v \in V_h$. However, since $V_h(t)$ is time-dependent in our setting, $P_{h,t}$ is also time-dependent. So just as the discrete Laplacian, $P_{h,t}$ will have an intrinsic time-dependence which results in $P_{h,t}T\dot{\tilde{v}} \notin V_h$ for $v \in V_h$.

An alternative is to consider a space-time L^2-projection $\bar{P}_h$ to the main discrete space V_h. Then $\bar{P}_h T\dot{\tilde{v}} \in V_h$ holds trivially. However, for the derivation of the inf-sup condition, the two estimates (3.65) and (3.66) are needed. To show those estimates, H^1-stability of the L^2-projection is needed, which in this case translates to showing that $\bar{P}_h$ is stable in either $|||\cdot|||_{A_{h,t}}$ or $\int_{I_n} |||\cdot|||^2_{A_{h,t}} \, \mathrm{d}t$. In the first case, one runs into problems since $\bar{P}_h$ is a space-time operator and $|||\cdot|||_{A_{h,t}}$ is a spatial norm, and we want the stability to hold for every $t \in [0, T]$. The second case also has its issues since obtaining that stability requires the assumption $k \leq Ch^2$, which is a severe restriction when using implicit dG-methods in time.

A third alternative could be to consider the corresponding *discrete* operator to T; see [23]. For our case, the discrete operator $T_{h,t} : L^2(\Omega_0) \to V_h(t)$ is defined by

$$(w, v)_{\Omega_0} = A_{h,t}(T_{h,t}w, v), \quad \forall v \in V_h(t). \tag{3.105}$$

But once again, using $V_h(t)$ and $A_{h,t}$ to define $T_{h,t}$ gives it an intrinsic time-dependence, thus $T_{h,t}\dot{\tilde{v}} \notin V_h$ for $v \in V_h$.

We conclude from the above discussion that standard spatial operators that map to $V_h(t)$ get an intrinsic time-dependence due to the continuous movement of the overlapping mesh $\mathcal{T}_G$. The intrinsic time-dependence of these operators ruins the ideas of standard analysis methodology that rely on a product structure between space and time. So far we have not encountered any existing analysis approach that is suitable for the method (3.101). We therefore propose a new energy analysis methodology that, to the best of our knowledge, has not been presented before.

New energy analysis approach

The natural first step is to decide on how to handle the time derivative in B_h. We saw that the standard way of using the H^{-1}-norm did not work well in our case. We do the following instead. Recall the discrete dimensional analysis performed in (2.46). Let h and k denote spatial and temporal discretization

units, respectively. Performing a discrete dimensional analysis on the bulk terms in the standard space-time bilinear form B_h gives

$$\int_{I_n} (\nabla w, \nabla v)_\Omega \,\mathrm{d}t \sim [k^1] \cdot [h^d] \cdot [h^{-1}] \cdot [h^{-1}] = [h^{d-2}k], \tag{3.106}$$

$$\int_{I_n} (\dot{w}, v)_\Omega \,\mathrm{d}t \sim [k^1] \cdot [h^d] \cdot [k^{-1}] \cdot [h^0] = [h^d]. \tag{3.107}$$

Since the terms should have the same unit, we deduce the unit correspondence $[k] \sim [h^2]$. This is a discrete equivalent of $\partial_t \sim \Delta$ for the heat equation. The symmetric first term naturally gives a useful energy norm term with the same unit, i.e., $[h^{d-2}k]$. It is thus reasonable to consider an energy norm term corresponding to the time-derivative term that has the same unit, i.e., $[h^d]$. Mimicking the symmetric spatial derivative term, we get

$$\int_{I_n} \|\dot{w}\|_\Omega^2 \,\mathrm{d}t \sim [k^1] \cdot [h^d] \cdot [k^{-2}] = [h^d k^{-1}]. \tag{3.108}$$

This term does not have the desired unit due to the factor $[k^{-1}]$. We remedy this by simply including the time step in the integral. We thus propose the following treatment of the time-derivative term:

$$\int_{I_n} k_n \|\dot{w}\|_\Omega^2 \,\mathrm{d}t \sim [k^1] \cdot [k^1] \cdot [h^d] \cdot [k^{-2}] = [h^d]. \tag{3.109}$$

We compare this to the standard treatment that uses the H^{-1}-norm. With the H^{-1}-norm, one has $\|\dot{w}\|_{-1}^2 = (\dot{w}, T\dot{w})_\Omega$. The operator $T = (-\Delta)^{-1}$ corresponds to the unit $[(h^{-2})^{-1}] = [h^2] \sim [k]$. Using the H^{-1}-norm via the operator T is thus in some sense equivalent to multiplying the L^2-norm by the temporal discretization unit. This is what we do explicitly in our approach to avoid any involvement of spatial operators with an intrinsic time-dependence.

Using (3.109) to include the time derivative in an energy norm sets the course for the rest of the energy analysis. The general is the same as in the standard case, i.e., using an inf-sup condition in place of a coercivity in a Céa's lemma type argument followed by interpolation estimates. However there are some key differences that we point out:

- Again, the time derivative is not handled with the H^{-1}-norm. Instead it is included in an energy norm by

$$\int_{I_n} k_n \|\dot{w}\|_\Omega^2 \,\mathrm{d}t. \tag{3.110}$$

- The corresponding X- and Y-norms are different due to the new time-derivative approach. This naturally results in different estimates, e.g., a different perturbed coercivity.

- The discrete inf-sup condition is different due to the new norms and estimates. Instead of having an auxiliary Y-norm in the denominator as in the standard case, the main X-norm figures there as well, i.e.,

$$|||w|||_X \leq C \sup_{v \in V_h \setminus \{0\}} \frac{B_h(w,v)}{|||v|||_X}, \quad \forall w \in V_h. \tag{3.111}$$

- A slightly different continuity result is used in the Céa's lemma type argument due to having the X-norm in the denominator of the inf-sup condition. This continuity result is derived from using the alternative form of B_h that comes from a temporal partial integration. The time derivative is thus moved from the first to the second argument in B_h which allows for cancellation of X-norm factors.

In the following, we give a concise presentation of the energy analysis, where we refer to Paper I for details.

Energy norms
Taking the same function as both arguments of B_h, and integrating the time-derivative term, etc., motivates us to define the following space-time energy norm:

$$\begin{aligned} |||v|||_{B_h}^2 := & \sum_{n=1}^{N} \left(\int_{I_n} |||v|||_{A_{h,t}}^2 \, \mathrm{d}t + |||\bar{n}^t|^{1/2}[v]\|_{\Gamma_n}^2 \right) \\ & + \sum_{i=1}^{2} \sum_{n=1}^{N-1} \|[v]_n\|_{\Omega_{i,n}}^2 + \sum_{i=1}^{2} \left(\|v_N^-\|_{\Omega_{i,N}}^2 + \|v_0^+\|_{\Omega_{i,0}}^2 \right). \end{aligned} \tag{3.112}$$

We may obtain the discrete coercivity: $\exists \beta > 0$ such that

$$B_h(v,v) \geq \beta \, |||v|||_{B_h}^2, \quad \forall v \in V_h. \tag{3.113}$$

As in the standard case, the B_h-norm is not enough for a complete energy analysis since it does not include the time derivative. For our case, instead of the partial time-derivative operator ∂_t, we consider a material-derivative operator D_t along the space-time trajectories of the underlying spatial domain. We define

three other space-time energy norms by

$$|||v|||_X^2 := \sum_{i=1}^{2}\sum_{n=1}^{N}\int_{I_n} k_n \|D_t v\|_{\Omega_i(t)}^2 \,\mathrm{d}t + |||v|||_{B_h}^2 \,, \tag{3.114}$$

$$|||v|||_{Y_+}^2 := \sum_{n=1}^{N}\Big(\int_{I_n}\frac{1}{k_n}\|v\|_{\Omega_0}^2 \,\mathrm{d}t + \int_{I_n}|||v|||_{A_{h,t}}^2 \,\mathrm{d}t + \|v_{n-1}^+\|_{\Omega_0}^2\Big), \tag{3.115}$$

$$|||v|||_{Y_-}^2 := \sum_{n=1}^{N}\Big(\int_{I_n}\frac{1}{k_n}\|v\|_{\Omega_0}^2 \,\mathrm{d}t + \int_{I_n}|||v|||_{A_{h,t}}^2 \,\mathrm{d}t + \|v_n^-\|_{\Omega_0}^2\Big). \tag{3.116}$$

The X-norm is the main norm in this energy analysis, and the Y-norms are auxiliary norms. By using the standard form of B_h and the alternative form that is obtained from an integration by parts with respect to time, we may obtain the two continuity results: $\exists C_+, C_- > 0$ such that for any functions w and v with sufficient regularity we have that

$$B_h(w,v) \le C_+ \,|||w|||_X \,|||v|||_{Y_+} \,, \tag{3.117}$$

$$B_h(w,v) \le C_- \,|||w|||_{Y_-} \,|||v|||_X \,. \tag{3.118}$$

We may obtain the discrete perturbed coercivity: For $q = 0, 1$, $\exists \delta, \xi > 0$ such that

$$B_h(v, v + \delta k_n D_t v) \ge \xi \,|||v|||_X^2 \,, \quad \forall v \in V_h. \tag{3.119}$$

We may also obtain the inverse estimate: For $q = 0, 1$, $\exists C > 0$ such that

$$|||k_n D_t v|||_X \le C \,|||v|||_X \,, \quad \forall v \in V_h. \tag{3.120}$$

The perturbed coercivity and the inverse estimate correspond to (3.65) and (3.66), respectively. The main tools used in obtaining (3.119) and (3.120) are two estimates that generalize corresponding standard estimates to the space-time CutFEM setting under consideration. The two estimates are: $\exists C_1, C_2 > 0$ such that for any $v \in V_h$ we have that

$$\int_{I_n}|||k_n D_t v|||_{A_{h,t}}^2 \,\mathrm{d}t \le C_1 \int_{I_n}|||v|||_{A_{h,t}}^2 \,\mathrm{d}t, \tag{3.121}$$

and, for $q = 0, 1$, that

$$\begin{aligned} &\sum_{i=1}^{2} k_n^2 \|(D_t v)_{n-1}^+\|_{\Omega_{i,n-1}}^2 \\ &\le C_2\Big(\sum_{i=1}^{2}\int_{I_n} k_n \|D_t v\|_{\Omega_i(t)}^2 \,\mathrm{d}t + |\mu|_{I_n}\int_{I_n}|||v|||_{A_{h,t}}^2 \,\mathrm{d}t\Big). \end{aligned} \tag{3.122}$$

By using (3.119) and (3.120), we may obtain the discrete inf-sup condition: For $q = 0, 1$, $\exists C > 0$ such that

$$|||w|||_X \leq C \sup_{v \in V_h \setminus \{0\}} \frac{B_h(w, v)}{|||v|||_X}, \quad \forall w \in V_h. \tag{3.123}$$

Note that we have the X-norm in the denominator here as well. This is in contrast to the inf-sup condition for the standard case (3.68) where we had the auxiliary Y-norm in the denominator.

Stability analysis
We use the discrete perturbed coercivity (3.119) to obtain the stability estimate for $q = 0, 1$,

$$\begin{aligned}
|||u_h|||_X^2 = & \sum_{n=1}^{N} \sum_{i=1}^{2} \int_{I_n} k_n \|D_t u_h\|_{\Omega_i(t)}^2 \, \mathrm{d}t \\
& + \sum_{n=1}^{N} \left(\int_{I_n} |||u_h|||_{A_{h,t}}^2 \, \mathrm{d}t + \||\bar{n}^t|^{1/2}[u_h]\|_{\bar{\Gamma}_n}^2 \right) \\
& + \sum_{i=1}^{2} \sum_{n=1}^{N-1} \|[u_h]_n\|_{\Omega_{i,n}}^2 + \sum_{i=1}^{2} \left(\|u_{h,N}^-\|_{\Omega_{i,N}}^2 + \|u_{h,0}^+\|_{\Omega_{i,0}}^2 \right) \\
\leq & C \left(\|u_0\|_{\Omega_0} + \|f\|_{L^2((0,T];L^2(\Omega_0))} \right)^2.
\end{aligned} \tag{3.124}$$

We note that this stability estimate is stronger than the corresponding basic one which is given in the B_h-norm.

Energy error analysis
We follow the standard energy analysis steps, i.e., Céa's lemma type argument and interpolation estimates. From having a discrete inf-sup condition we need an error split. We define a space-time interpolation operator $\bar{I}_h$ that maps to V_h and that has optimal interpolation properties. For a function v with sufficient regularity we have the interpolation error estimates

$$\left|\left|\left| v - \bar{I}_h v \right|\right|\right|_X^2 \leq C k^{2q+1} F_k^2(v) + C h^{2p} F_h^2(v), \tag{3.125}$$

$$\left|\left|\left| v - \bar{I}_h v \right|\right|\right|_{Y_-}^2 \leq C k^{2q+1} F_k^2(v) + C h^{2p} F_h^2(v), \tag{3.126}$$

where k and h are the largest time step and simplex diameter, respectively. $F_k(v)$ and $F_h(v)$ are expressions involving derivatives of the function v with respect to space and time. Let $e = u - u_h$ denote the approximation error.

By using the interpolant $\bar{I}_h u \in V_h$, we split the approximation error e into an interpolation error ρ and a discrete error θ by

$$e = u - u_h = (u - \bar{I}_h u) + (\bar{I}_h u - u_h) = \rho + \theta. \tag{3.127}$$

We then consider

$$|||e|||_X = |||\rho + \theta|||_X \leq |||\rho|||_X + |||\theta|||_X \,, \tag{3.128}$$

where we focus on the θ-part. The method (3.101) is consistent from which Galerkin orthogonality follows trivially. Thus for any $v \in V_h$

$$\begin{aligned} & 0 = B_h(e, v) = B_h(\rho + \theta, v) = B_h(\rho, v) + B_h(\theta, v), \\ \Longrightarrow \quad & B_h(\theta, v) = -B_h(\rho, v). \end{aligned} \tag{3.129}$$

The Céa's lemma type argument consists of the following ingredients: the discrete inf-sup condition (3.123), the Galerkin orthogonality twist (3.129), and the continuity result (3.118) where the alternative form of B_h obtained from integration by parts with respect to time has been used. We thus estimate the θ-part by

$$\begin{aligned} |||\theta|||_X &\leq C \sup_{v \in V_h \setminus \{0\}} \frac{B_h(\theta, v)}{|||v|||_X} = C \sup_{v \in V_h \setminus \{0\}} \frac{-B_h(\rho, v)}{|||v|||_X} \\ &\leq C \sup_{v \in V_h \setminus \{0\}} \frac{|||\rho|||_{Y_-} \, |||v|||_X}{|||v|||_X} = C \, |||\rho|||_{Y_-} \,. \end{aligned} \tag{3.130}$$

Using this estimate in (3.128) gives

$$|||e|||_X \leq |||\rho|||_X + |||\theta|||_X \leq |||\rho|||_X + C \, |||\rho|||_{Y_-} \,, \tag{3.131}$$

from which, by using the interpolation estimates (3.125) and (3.126), we obtain the optimal order a priori energy error estimate for $q = 0, 1$,

$$|||u - u_h|||_X^2 \leq C k^{2q+1} F_k^2(u) + C h^{2p} F_h^2(u). \tag{3.132}$$

3.4.2 dG(0) mesh movement (Paper II)

Problem formulation

As a discrete counterpart to the movement of the domain G, we prescribe a dG(0) movement for the overlapping mesh $\mathcal{T}_G$. By this we mean that the location of the overlapping mesh $\mathcal{T}_G$ is a dG(0) function with respect to time, i.e., *discontinuous* on $[0, T]$ and *constant* on each I_n; see Figure 7.

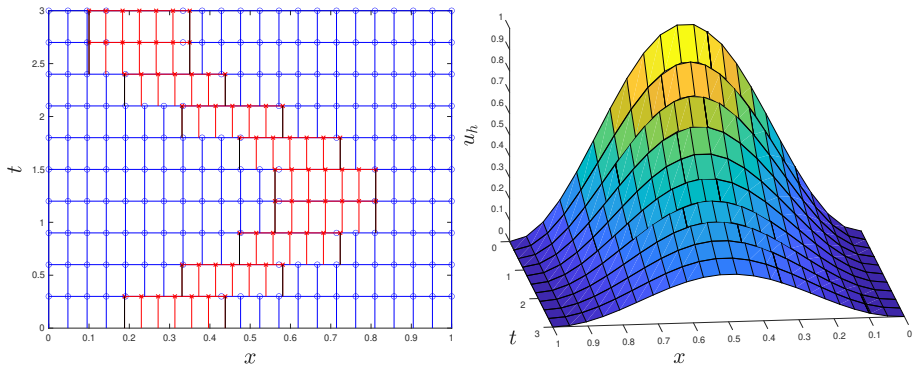

Figure 7: Overlapping meshes and solution with dG(0) mesh movement. *Left*: Space-time discretization for $d = 1$. *Right*: A corresponding dG(1)cG(1) cut finite element solution of the heat equation.

Just as in the elliptic CutFEM case, we use two spatial standard finite element spaces corresponding to the meshes $\mathcal{T}_0$ and $\mathcal{T}_G$, to define a *broken spatial* finite element space. Since the location of $\mathcal{T}_G$ is fixed for each I_n, we get the same broken spatial finite element space $V_{h,n}$ for every $t \in I_n$. Following the discretization steps of the parabolic standard case, we define two dG(q)cG(p) spaces, one for each mesh. Then, as in the elliptic CutFEM case, we use restrictions of functions from these spaces to define our *main* broken space-time discrete space V_h. We refer to Paper II for details on the construction of these spaces.

Since the location of $\Gamma(t)$ is the same for all $t \in I_n$, i.e., Γ_n, we use a slabwise spatial bilinear form $A_n = A_{h,n}$ corresponding to the bilinear form A_h in the elliptic CutFEM case. The dG(q)cG(p) space-time finite element formulation corresponding to (3.98) with dG(0) mesh movement is: Find $u_h \in V_h$ such that

$$\begin{aligned} &\sum_{n=1}^{N} \left(\int_{I_n} (\dot{u}_h, v)_{\Omega_0} + A_n(u_h, v)\,\mathrm{d}t + ([u_h]_{n-1}, v_{n-1}^+)_{\Omega_0} \right) \\ &= \int_0^T (f, v)_{\Omega_0}\,\mathrm{d}t, \quad \forall v \in V_h. \end{aligned} \tag{3.133}$$

Note the similarity to the standard formulation (3.48). We refer to Paper II for details on the finite element formulation. We define the non-symmetric bilinear

form B_h by

$$B_h(w,v) := \sum_{n=1}^{N} \int_{I_n} (\dot{w}, v)_{\Omega_0} \,\mathrm{d}t + \sum_{n=1}^{N} \int_{I_n} A_n(w,v) \,\mathrm{d}t + \sum_{n=1}^{N-1} ([w]_n, v_n^+)_{\Omega_0} + (w_0^+, v_0^+)_{\Omega_0}. \tag{3.134}$$

We may then write (3.133) in compact form as: Find $u_h \in V_h$ such that

$$B_h(u_h, v) = \int_0^T (f, v)_{\Omega_0} \,\mathrm{d}t + (u_0, v_0^+)_{\Omega_0}, \quad \forall v \in V_h. \tag{3.135}$$

A comment on standard analysis approaches
We start by considering a potential L^2-analysis of (3.135). The L^2-analysis of a parabolic problem requires the *main* stability estimate which in turn requires the *strong* stability estimate. Recall these estimates for the parabolic standard case given by (3.73) and (3.72), respectively. To start deriving the strong stability estimate we recall that one tests with the discrete Laplacian of the finite element solution. The discrete Laplacian is defined using the spatial finite element space and the spatial bilinear form, which in our case are $V_{h,n}$ and A_n, respectively. Since both of them are slabwise constant due to the dG(0) mesh movement, the discrete Laplacian will also be slabwise constant. For our case, the discrete Laplacian $\Delta_n : H^1(\Omega_{1,n}, \Omega_{2,n}) \to V_{h,n}$ is defined by

$$(-\Delta_n w, v)_{\Omega_0} = A_n(w,v), \quad \forall v \in V_{h,n}. \tag{3.136}$$

This definition does *not* give the discrete Laplacian an intrinsic time-dependence on each I_n. The time-dependence of $\Delta_n u_h$ is therefore the same as for $u_h \in V_h$, which means that $-\Delta_n u_h \in V_h$. Thus we may test with $-\Delta_n u_h$ which means that we can start deriving the strong stability estimate. This motivates us to attempt the more demanding L^2-analysis of (3.135).

The L^2-analysis closesly follows [24, 25]. Most modifications are natural extensions from the standard case to spatial CutFEM, since we have a nice separating product structure between space and time in every slab. The main difference is the introduction of a shift operator that generalizes the Ritz projection operator. The shift operator is used in the derivation of the strong stability estimate to handle the discontinuous shift of the discontinuity on Γ between slabs. In the following, we give a concise presentation of the L^2-analysis, where we refer to Paper II for details.

Stability analysis

Just as in the standard case, we need a stability estimate for a discrete dual problem with zero right-hand side. We therefore focus the stability analysis on deriving estimates with $f = 0$. We present three stability estimates.

By testing with $v = u_h$ in the finite element formulation (3.133) and integrating the time-derivative term, etc., just as in the standard case, we obtain the *basic* stability estimate

$$\|u_{h,N}^-\|_{\Omega_0}^2 + \sum_{n=1}^N \int_{I_n} |||u_h|||_{A_n}^2 \,\mathrm{d}t + \sum_{n=1}^N \|[u_h]_{n-1}\|_{\Omega_0}^2 \leq C\|u_0\|_{\Omega_0}^2. \tag{3.137}$$

We move on to the remaining two stability estimates. We start by testing with $-\Delta_n u_h$. In the resulting treatment we use the shift operator to handle the shifting discontinuity between slabs. Secondly we test with $(t - t_{n-1})\dot{u}_h$ which works just as in the standard case. Finally, we test with $P_n[u_h]_{n-1}$, where $P_n = P_{h,n}$ is the $L^2(\Omega_0)$-projection to $V_{h,n}$. We have to use P_n since the shifting discontinuity makes $[u_h]_{n-1} \notin V_h$. This is also done in [25]. Using the resulting estimates from testing with the aforementioned three functions, and also the basic stability estimate (3.137), we may obtain the *strong* stability estimate

$$\sum_{n=1}^N t_n \int_{I_n} \|\dot{u}_h\|_{\Omega_0}^2 + \|\Delta_n u_h\|_{\Omega_0}^2 \,\mathrm{d}t + \sum_{n=2}^N \frac{t_n}{k_n}\|[u_h]_{n-1}\|_{\Omega_0}^2 \leq C\|u_0\|_{\Omega_0}^2. \tag{3.138}$$

The strong stability estimate is used to obtain the *main* stability estimate

$$\begin{aligned} &\|u_{h,N}^-\|_{\Omega_0} + \sum_{n=1}^N \int_{I_n} \|\dot{u}_h\|_{\Omega_0} + \|\Delta_n u_h\|_{\Omega_0} \,\mathrm{d}t + \sum_{n=1}^N \|[u_h]_{n-1}\|_{\Omega_0} \\ &\leq C_1\|u_0\|_{\Omega_0}, \end{aligned} \tag{3.139}$$

where $C_1 = C(\log(t_N/k_1) + 1)^{1/2}$ and $C > 0$. It is the corresponding main stability estimate for the discrete dual problem that is crucial to and used in the L^2-error analysis.

L^2-error analysis

The dG(0) mesh movement results in a nice product structure between space and time in every slab. The L^2-error analysis of (3.133) therefore follows the standard case with only natural modifications to account for the CutFEM setting. We thus refer to the L^2-analysis of the parabolic standard case for a brief overview and to Paper II for a detailed presentation of the error analysis. From

the L^2-error analysis of (3.133), we obtain the optimal order a priori L^2-error estimate for $q = 0, 1$,

$$\begin{aligned} &\|u(t_N) - u^-_{h,N}\|_{\Omega_0} \\ &\leq C_N \max_{1\leq n\leq N} \left\{ k_n^{2q+1} \|\dot{u}^{(2q+1)}\|_{\Omega_0,I_n} + h^{p+1} \|D_x^{p+1} u\|_{\Omega_0,I_n} \right\}, \end{aligned} \tag{3.140}$$

where $C_N = C(\log(t_N/k_N)+1)^{1/2}$, and $C > 0$. Note that the superconvergence with respect to the time step, which is of the *third* order for dG(1), is preserved for the case with dG(0) mesh movement.

Extension to an L^2-analysis for cG(1) mesh movement?
We conclude this section about the L^2-analysis for dG(0) mesh movement by mentioning how it could be used to come closer to an understanding of the L^2-analysis for cG(1) mesh movement.

Recall that we could not perform a complete L^2-analysis for cG(1) mesh movement since we could not obtain the strong stability needed. This is however not all that is missing. Another tool in the L^2-analysis is the temporal interpolation operator that is used to construct an interpolant for the error split. The operator is defined partly via an orthogonality condition with lower degree polynomials. This orthogonality is an essential component in obtaining the superconvergence with respect to the time step. However, due to the cG(1) mesh movement, using the orthogonality condition to define the temporal interpolation operator gives it an intrinsic space-dependence, just as the discrete Laplacian receives an intrinsic time-dependence. This means that the temporal interpolation operator does not give the interpolant the required time-dependence for it to belong to the discrete subspace of the method. Hence, the discrete error in an error split is in fact not discrete. Instead using a temporal operator with the intended interpolation property, but without the orthogonality, results in an error estimate without the superconvergence.

To derive an a priori error estimate that does not include the superconvergence for $q = 1$ is a bit less demanding, and is definitely an option for the analysis of cG(1) mesh movement. Recall that in [26], partial results from an incomplete L^2-analysis together with numerical results suggested that the L^2-error for cG(1) mesh movement had the following k-dependence for $q = 1$:

$$error \sim k^3 + |\mu|k^2. \tag{3.141}$$

For $|\mu| > 0$, the superconvergence seems to be lost. It might thus suffice with an L^2-error estimate of the following type:

$$error \sim k^2, \tag{3.142}$$

since the k-convergence seems to be of the second order anyways. However, this estimate is not completely satisfactory since it does not reflect the L^2-error's suggested dependence on $|\mu|$. Recall that in a numerical convergence study it would not be surprising to get third order k-convergence for $|\mu| > 0$ since very small k's ($k \ll |\mu|$) might be needed for the second order. The apparent $|\mu|$-dependent switch in k-convergence order is thus something that could be of interest to understand from an analytic standpoint. This is where the L^2-analysis of dG(0) mesh movement could be of use.

Let u be the analytic solution of a parabolic model problem. Let u_h^0 and u_h^1 denote the corresponding finite element solutions of the problem with dG(0) and cG(1) mesh movements, respectively. From the L^2-analysis of dG(0) mesh movement we have the following k-dependence of the L^2-error for $q = 1$:

$$\|u - u_h^0\| \sim k^3. \tag{3.143}$$

We note that if $|\mu| \to 0$, then the method with cG(1) mesh movement approaches the one with dG(0) mesh movement, leading to the two finite element solutions converging to the same function. We thus have the following crude μ-dependence of the difference between the two discrete solutions:

$$\|u_h^0 - u_h^1\| \sim C(\mu). \tag{3.144}$$

We may use (3.143) and (3.144), to obtain the following k-dependence of the L^2-error for G(1) mesh movement for $q = 1$:

$$\|u - u_h^1\| \leq \|u - u_h^0\| + \|u_h^0 - u_h^1\| \sim k^3 + C(\mu). \tag{3.145}$$

Assume now that we could also prove a slightly easier a priori error estimate, not including the superconvergence, for cG(1) mesh movement. We would then have the following k-dependence of the L^2-error for $q = 1$:

$$\|u - u_h^1\| \sim k^2. \tag{3.146}$$

Combining (3.145) and (3.146), gives us the following k-$|\mu|$-dependence of the L^2-error for G(1) mesh movement for $q = 1$:

$$\|u - u_h^1\| \sim \min\{k^3 + C(\mu), k^2\} = \begin{cases} k^3, & \text{if } |\mu| \ll k < 1, \\ k^2, & \text{if } k \ll |\mu|. \end{cases} \tag{3.147}$$

This estimate reflects the apparent $|\mu|$-dependent switch in k-convergence order for $q = 1$ as suggested by the incomplete analysis and numerical results in [26].

4 Applications

The finite element method has been applied in many areas of engineering and science ever since development started around the 1950s with the classical civil engineering application in structural analysis. It has during the decades since then become a valuable tool for engineers and researchers; see, e.g., [27, 28].

An essential component of the finite element method is the computational mesh: the result of discretizing the solution domain of a PDE-problem. In standard finite element methods the mesh is usually created based on the domain geometry. Domain-fitted meshes are typically unstructured in real-world applications since such applications often concern problems with complicated geometry. Unstructured meshes are more memory demanding and lead to an increased computational cost compared to structured meshes, where the underlying structure may be used to implicitly store and use the mesh in computations. Another matter for consideration is when the problem geometry changes. Using a domain-fitted mesh means that the mesh changes and deforms when the geometry changes. This leads to a need for remeshing it the geometry changes too much. The remeshing might concern the whole mesh or just be local. Either way it comes with a computational cost.

Cut finite element methods do not rely on having a domain-fitted mesh. The mesh and the problem geometry may be completely unrelated. CutFEM thus allows for the usage of simple and structured meshes for problems with demanding geometry where standard methods would require unstructured discretizations and/or remeshing. This means that there is a computational benefit in using CutFEM instead of standard methods in applications concerning such situations. In the following, we consider the following application aspects of CutFEM on overlapping meshes:

- Application example: time-dependent geometry.
- Application example: evaluation of different geometries.
- Other application examples.
- Software.
- Computational cost and gain.

We discuss these cases separately in the following subsections.

4.1 Time-dependent geometry

With time-depended geometry we mean that the solution domain evolves during the solution process of the PDE. For example, consider a situation with a moving

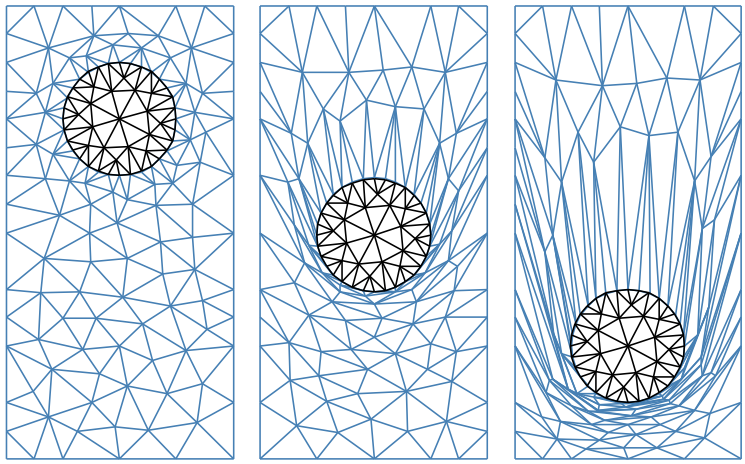

Figure 8: Problem with a moving object that deforms the mesh.

object in the solution domain. With a standard finite element method the movement of the object would deform the mesh. If this deformation is too severe it leads to degenerate mesh cells, meaning that the mesh is no longer suitable for computation and a new mesh needs to be generated; see Figure 8. By using CutFEM on overlapping meshes for such a situation, the moving object is instead encapsulated in an overlapping mesh. A background mesh is generated in the empty solution domain which means that it may be nice and structured. The body-fitted overlapping mesh is then placed on top of the background mesh where it is free to move around; see Figure 9 and Figure 10. This approach thus avoids remeshing altogether. The issue is instead to make sure that the finite element solution is well-behaved on the joint boundary between the meshes. For a problem with a moving object in the solution domain, interface CutFEM could also be used. The interface would then represent the boundary of the object. However, if the object is complicated, this could lead to more difficult cut situations between the interface and the mesh cells. An advantage of overlapping meshes is that a complicated object may be embedded in a body-fitted mesh that has a much simpler boundary, thus leading to simpler cut situations; see Figure 10 for an example.

In Paper I and Paper II, we consider two alternatives for representing the movement of the overlapping mesh: *continuous* and *discontinuous* mesh movement, respectively; see Figure 5. For problems with evolving geometry, such as a moving object or a bubble containing one liquid floating around in another, both continuous and discontinuous mesh movement could be used.

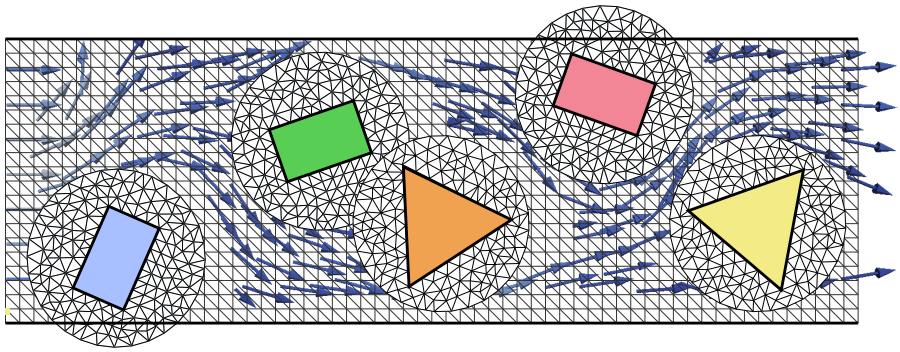

Figure 9: Application of CutFEM on overlapping meshes to a problem concerning the flow around several objects in a channel.

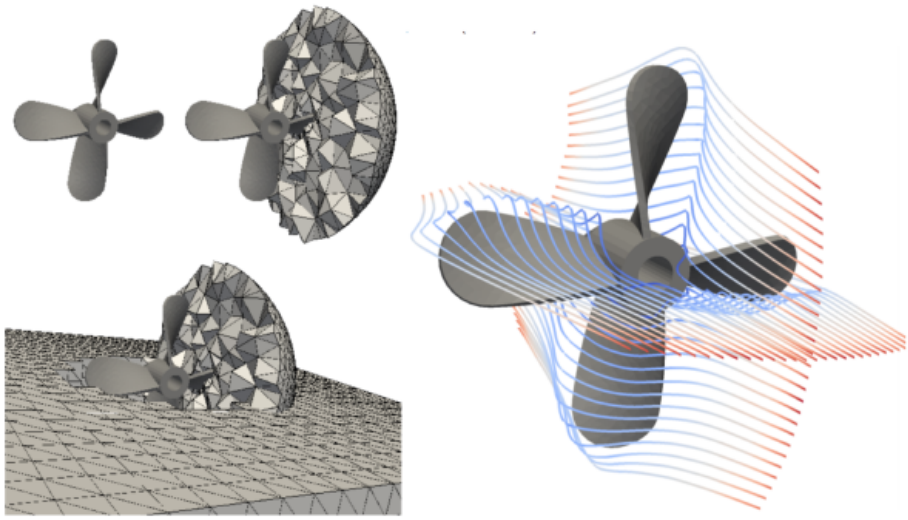

Figure 10: Application of CutFEM on overlapping meshes to a problem concerning the flow around a propeller.

The *continuous* mesh movement might be the preferred one when considering physical accuracy. This is so since the space-time cells of the overlapping mesh in one slab are connected exactly to their older versions in the previous slab. This means that the solution inside the prismatic space-time domain of the overlapping mesh in one slab naturally may have a strong influence on the one in the next. Compare this to the discontinuous mesh movement where the overlapping mesh may teleport to another region of the solution domain from one time to the next. The previous solution may of course be carried over directly to the next slab but this then reduces the degrees of freedom in the space-time domain of the overlapping mesh. The solution on the part of the background mesh where the overlapping mesh was located also needs to be treated so as not to cause leakage of information from the overlapping mesh.

A benefit with *discontinuous* mesh movement is that the space-time prisms of the background mesh may be cut in less complicated ways. This means that less demanding geometry computations are needed in an implementation. A way to also make discontinuous mesh movement more physically accurate is to consider something we refer to as "snapping". Consider a moving object encapsulated in an overlapping mesh. Instead of moving the entire object-mesh package during a discrete time-interval, the boundary of the overlapping mesh is held at a fixed position while the object moves inside the overlapping mesh. This naturally leads to deformed mesh cells, but instead of remeshing to avoid degenerate cells, one could move (or snap) the overlapping mesh around the object back into its initial position when the overlapping mesh cells have become too deformed. Snapping thus reduces potential physical inaccuracy for discontinuous mesh movement.

4.2 Evaluation of different geometries

There are also problems that concern the evaluation of different geometries, but where the geometry is stationary during the solution process of the PDE. This is the case when one is interested in studying the influence of the geometry on the solution of a PDE-problem. Typical applications concern problems involving both PDE-modeling and optimization. The objective in such situations is then often to optimize the geometry based on the solution of the PDE. One starts with an initial geometry, solves the corresponding PDE-problem, obtains a solution, and uses it to evaluate the geometry. The geometry is then changed and the procedure repeated. Such an optimization process can consist of several thousand evaluations, meaning that the same number of different geometries is needed. With standard finite element methods, each new geometry then requires some degree of remeshing. With CutFEM, a single initial mesh generation suffices since the same mesh can be used for an arbitrary number of

different geometries. Some examples of using CutFEM on overlapping meshes to avoid remeshing in an optimization processes are presented in [14] and Paper III.

In Paper III, we consider multi-objective optimization of configurations of buildings based on minimizing the surroundning wind and maximizing the view from the buildings. For the wind model, we use a CutFEM on overlapping meshes for the Stokes equations. The buildings are encapsulated in overlapping meshes, and are freely placed in a background mesh of the surroundings, thus enabling any number of configurations to be evaluated without remeshing; see Figure 11.

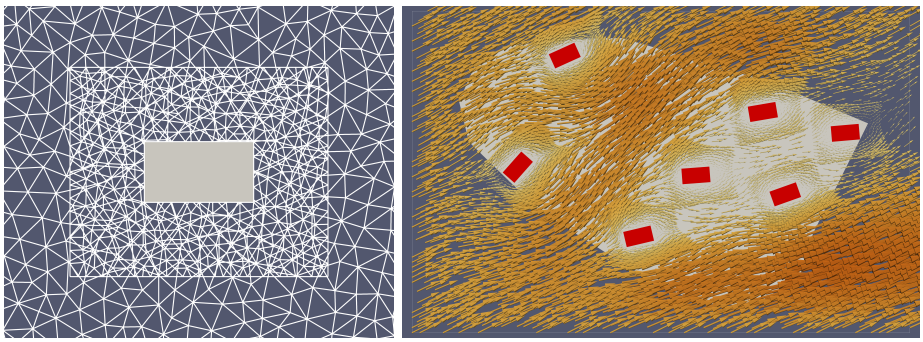

Figure 11: Application of CutFEM on overlapping meshes to optimize the configuration of buildings based on minimizing the surrounding wind. *Left*: Building-fitted overlapping mesh on top of a background mesh. *Right*: The computed flow around several buildings embedded in overlapping meshes.

In Paper IV and V, we present a *potential* application for CutFEM on overlapping meshes related to the evaluation of different geometries. The papers concern a system for immersive and interactive physics simulations, implemented as an app on MicroSoft's augmented reality glasses HoloLens. The app is called HoloFEM and allows a user to define and solve physical problems governed by PDEs in augmented reality; see Figure 12. The PDEs are solved with FEM by using a volume mesh computed from the real-world surroundings. The mesh is however a quite crude representation of the real-world geometry due to the hardware limitations of the HoloLens. Smaller interior geometries such as furniture are completely ignored, meaning that the mesh does not include any objects. CutFEM on overlapping meshes could be used to introduce virtual objects into the augmented reality environment of HoloFEM, thereby enabling more interesting simulations and the evaluation of different geometry configurations.

Figure 12: Augmented reality environment representing PDE-problems and their solutions in the author's office.

4.3 Other application examples

CutFEM on overlapping meshes may be used as an alternative to adaptive mesh refinement. Instead of local mesh refinements, one may simply patch over the area with a pre-generated sufficiently fine overlapping mesh. Consider for example a problem with a moving heat source, e.g., a welding application. The solution typically exhibits a peak around the location of the heat source, which might lead to the need for local mesh refinement in order to approximate the solution properly. By using overlapping meshes, one might instead make a fine overlapping mesh that follows the heat source around, thus enabling the computation of a nice approximate solution without any mesh refinement.

Again consider the *continuous* and *discontinuous* mesh movements studied in Paper I and Paper II, respectively. As an alternative to adaptive mesh refinement, the overlapping mesh is considered to be a fully *computational* feature. This means that we do not want the overlapping mesh to obstruct or influence the flow of information in the space-time discretization. It should be physically invisible. The dG(0) mesh movement studied in Paper II naturally allows information to flow through the overlapping mesh. With a continuous mesh movement however, the solution in the space-time domain of the overlapping mesh is smeared out in each slab; see Figure 6. We thus conclude that *discontinuous piecewise constant* mesh movement might be to prefer when the overlapping mesh is used as a fully computational feature.

Another example is domain construction. Several simply shaped overlapping meshes may be patched together to define a more complicated solution domain.

4.4 Software

Application and implementation of multi-mesh finite element methods rely on efficient and robust computation of mesh-mesh intersections. This puts high demands on the implementation and requires much more sophisticated algorithms from computational geometry than what is normally the case for finite element problems.

Support for the formulation of cut finite element methods on overlapping meshes for 2D problems exists in older versions of the popular open-source finite element package FEniCS [29]. In FEniCS, as well as in other works, CutFEM on overlapping meshes is referred to as multi-mesh finite element methods. This highlights the concept of having multiple arbitrarily overlapping meshes. Using the multi-mesh functionality of FEniCS, one may formulate and automatically discretize basic multi-mesh finite element formulations of systems of PDEs such as the Stokes problem. The implementation in FEniCS relies on an implementation of mesh-mesh intersections, based on generation and traversal of axis-aligned bounding box trees (AABB trees), low-level operations for computing and representing the intersections of simplices, and generation of quadrature points on cut cells. The implementation is integrated with the automatic code generation of FEniCS which allows multi-mesh discretizations to be formulated in (close to) natural mathematical language. In some of the referenced CutFEM literature, the numerical results have been obtained with FEniCS; see, e.g., [15] for a recent example.

In Paper III, the multi-mesh functionality of FEniCS is used for the implementation of a flow problem modeled by the Stokes equations. The multi-mesh finite element formulation of the Stokes equations that is used is analyzed by [13] as an extension to higher-order function spaces of the discretization previously analyzed by [12].

4.5 Computational cost and gain

With CutFEM on overlapping meshes, the computational issues with a changing problem geometry is shifted from mending deformed meshes to making sure that the approximate solution is well-behaved on the interface between meshes. The way to obtain a well-behaved and stable CutFEM solution is to add the appropriate Nitsche terms to the variational forms. This of course leads to a more complicated variational equation and thus a more involved and expensive assembly process for the corresponding system of algebraic equations. So by using CutFEM, there is a computational gain from avoiding remeshing, but also an increased computational cost in the assembly process.

In a numerical study performed with the multi-mesh functionality of FEn-

iCS, the computational gains and costs of using CutFEM on overlapping meshes have been measured. The study considers a multi-mesh finite element formulation for a *stationary* heat equation problem related to an application where the evaluation of different geometries is of interest; see Section 5.2 in [14]. The problem geometry is thus stationary during each solution process, but changes between solution processes. Three mesh compositions are used: coarse (20,216 cells), middle (175,196 cells), and fine (481,232 cells). For each mesh composition, three alternatives are considered: remeshing, deformation, and multi-mesh. For each of the resulting nine cases, the time for assembling the linear system, solving the linear system, and updating the meshes is measured. For multi-mesh, updating the meshes means computing the cut cells of the new mesh hierarchy. The results of the study are presented in Figure 13–15. These results demonstrate an overall computational gain when using CutFEM on overlapping meshes as opposed to the two standard methods for problems with changing geometry. For all three mesh compositions, the assembly time for multi-mesh is greater than for the two alternatives. This difference becomes less prominent when the number of mesh cells increases though. For all three mesh compositions, the greatest computational cost is however the mesh update. This cost is smaller for multi-mesh than for both of the two alternatives, resulting in the overall computational gain when using multi-mesh.

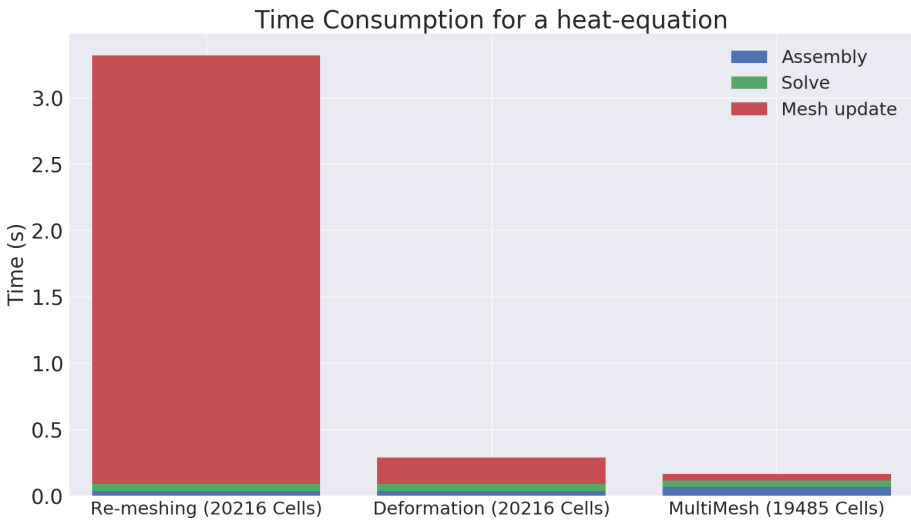

Figure 13: Times for assembly, solve, and mesh update for a stationary heat equation problem with three mesh changing alternatives using the *coarse* mesh composition. (Image courtesy of J.Dokken.)

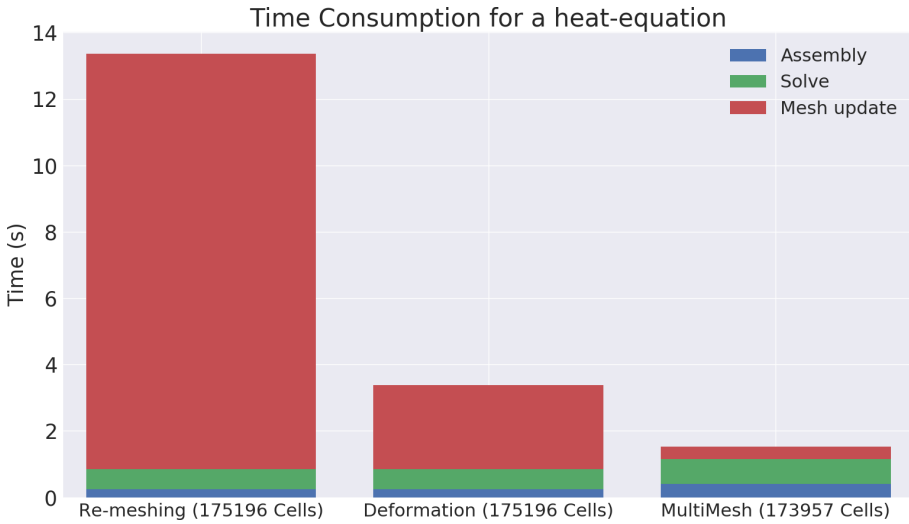

Figure 14: Times for assembly, solve, and mesh update for a stationary heat equation problem with three mesh changing alternatives using the *middle* mesh composition. (Image courtesy of J.Dokken.)

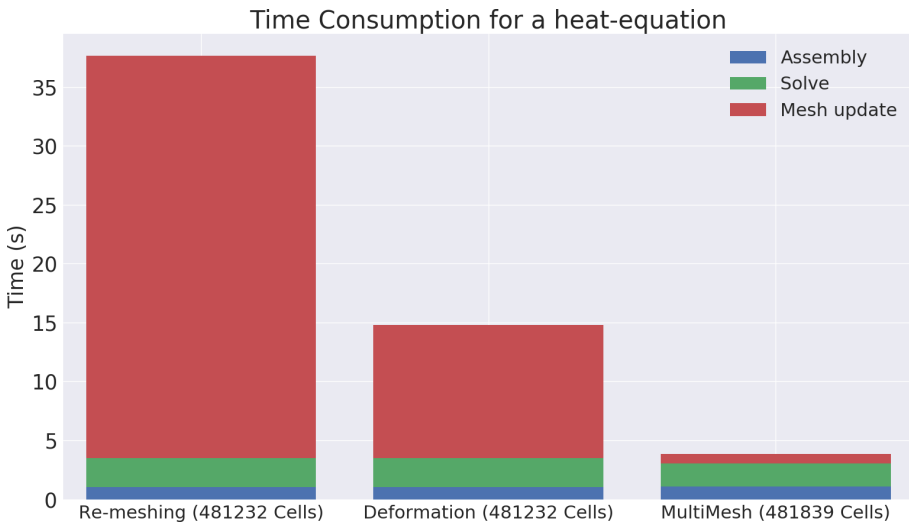

Figure 15: Times for assembly, solve, and mesh update for a stationary heat equation problem with three mesh changing alternatives using the *fine* mesh composition. (Image courtesy of J.Dokken.)

5 Summary of appended papers

This thesis deals with both analysis and applications of CutFEM on overlapping meshes. The thesis may therefore be partitioned into an analysis part and an applications part.

5.1 Analysis part

The analysis part concerns *time-dependent* CutFEM on overlapping meshes. We consider two alternatives of a space-time CutFEM for the heat equation on two overlapping meshes. This should be viewed as an initial step for developing time-dependent CutFEMs on overlapping meshes, since the heat equation is one of the simplest time-dependent PDEs and two meshes is the minimum number of meshes. The analysis part consists of two papers.

Paper I

In Paper I: *A cut finite element method for the heat equation on overlapping meshes: Energy analysis for cG(1) mesh movement*, the overlapping mesh is prescribed a cG(1) movement. This means that its location as a function of time is *continuous* and *piecewise linear*. Characteristics of the cG(1) mesh movement are skewed space-time nodal trajectories and cut prismatic space-time cells. This results in a discretization that lacks a nice product structure between space and time in every slab. Standard analysis methodology that heavily relies on such a product structure therefore fails in the current setting. We therefore propose a new analysis methodology that is general enough to be applicable to our situation. The analysis is of an energy type, where we use space-time energy norms to derive and obtain stability and error estimates. The error analysis produces an optimal order a priori estimate. We also include numerical results for a model problem in one spatial dimension that support the analytic convergence orders of the approximation error.

Paper II

In Paper II: *A cut finite element method for the heat equation on overlapping meshes: L^2-analysis for dG(0) mesh movement*, the overlapping mesh is prescribed a dG(0) movement. This means that its location as a function of time is *discontinuous* and *piecewise constant*. This results in a discretization that has a product structure between space and time in each slab. Standard analysis methodology therefore work with some modifications to handle the shift in the overlapping mesh's location at discrete times. The general analysis consists of stability and error estimates. The error analysis produces an optimal order a priori estimate of the L^2-norm of the approximation error at the final time. This estimate shows that the method preserves the so called superconvergence

of the error with respect to the time step. We also include numerical results for a model problem in one spatial dimension that support the analytic convergence orders of the approximation error.

5.2 Applications part

In the applications part, we consider two potential applications for CutFEM on overlapping meshes. Both applications present examples of when the evaluation of different geometry configurations are of interest. More specifically, how different configurations of a stationary solution domain influence the solution of a PDE-problem.

Paper III

The first application, presented in Paper III: *Multi-mesh multi-objective optimization with application to a model problem in urban design*, presents methodology for evaluating configurations of buildings. Two factors are taken into consideration when evaluating a configuration: the wind around the buildings, and the view from the buildings. The wind model is based on a CutFEM on overlapping meshes for Stokes equations. It is applied by encapsulating each building in its own mesh and also having a background mesh of the domain without any buildings. This set-up only requires initial mesh generations and allows the same meshes to be used for evaluation of a very large number of building configurations, thus avoiding costly remeshing for new configurations. The view models, a simple one for 2D settings, and a more elaborate one for 3D settings, relate to established concepts such as isovists and using different view weights depending on what type of object is seen. The models are used to define measures for wind and view that can be used to evaluate a configuration of buildings. These measures are then used to formulate a multi-objective optimization problem that is implemented and solved for 2D settings.

Paper IV and V

The second application, presented in Paper IV and Paper V, concerns a software application (app) for Microsoft's augmented reality glasses HoloLens. This app has yet to incorporate CutFEM, and thus only provides a potential application example. The app lets a user define and solve physical problems governed by PDEs in an immersive and interactive augmented reality environment. FEM is used to solve the PDE-problems and the real-world geometry is used as input to compute a computational mesh. All computations are performed on the HoloLens which due to hardware limitations put restrictions on the computational accuracy. This also affects the meshing. In the current state, smaller geometries such as furniture are ignored when generating a mesh. The gener-

ated mesh is therefore a discretization of the empty solution domain, which is exactly what the background mesh in a CutFEM multi-mesh hierarchy is. The idea would then be the same as in the previous application. Namely to generate meshes around the ignored smaller geometries so that CutFEM on overlapping meshes may be used to investigate how different configurations of the geometry affect the physical system. For example, how the location of furniture affect the airflow from a ventilation system.

Paper IV: *Solving Poisson's equation on the Microsoft HoloLens*, published in Proceedings of VRST'17, gives a brief presentation of the first version of the app. This version of the app can only handle stationary problems.

Paper V: *Finite element simulation of physical systems in augmented reality*, published in Advances in Engineering Software, contains a far more elaborate presentation and description of the app. Here, the app has also been improved and extended to handle time-dependent advection–diffusion problems.

References

[1] R. Glowinski, T. W. Pan, T. I. Hesla, D. D. Joseph, and J. Periaux, "A fictitious domain approach to the direct numerical simulation of incompressible viscous flow past moving rigid bodies: application to particulate flow," *J. Comput. Phys.*, vol. 169, no. 2, pp. 363–426, 2001.

[2] N. Moes, J. Dolbow, and T. Belytschko, "A finite element method for crack growth without remeshing," *Int. J. Numer. Meth. Engng*, vol. 46, pp. 131–150, 1999. [Online]. Available: http://venus.usc.edu/PAPERS/MultiScaleMechanics/XFEM.pdf

[3] J. Nitsche, "Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind," in *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, vol. 36. Springer, 1971, pp. 9–15.

[4] A. Hansbo and P. Hansbo, "An unfitted finite element method, based on Nitsche's method, for elliptic interface problems," *Comp. Methods Appl. Mech. Engrg.*, vol. 191, no. 47, pp. 5537–5552, 2002.

[5] A. Hansbo, P. Hansbo, and M. G. Larson, "A finite element method on composite grids based on Nitsche's method," *ESAIM, Math. Model. Numer. Anal.*, vol. 37, no. 03, pp. 495–514, 2003.

[6] E. Burman and M. A. Fernández, "Stabilized explicit coupling for fluid-structure interaction using Nitsche's method," *C. R. Math. Acad. Sci. Paris*, vol. 345, no. 8, pp. 467–472, 2007.

[7] E. Burman and P. Hansbo, "A unified stabilized method for Stokes' and Darcy's equations," *J. Comput. Appl. Math.*, vol. 198, no. 1, pp. 35–51, 2007.

[8] R. Becker, E. Burman, and P. Hansbo, "A Nitsche extended finite element method for incompressible elasticity with discontinuous modulus of elasticity," *Comp. Methods Appl. Mech. Engrg.*, vol. 198, no. 41, pp. 3352–3360, 2009.

[9] A. Massing, M. G. Larson, A. Logg, and M. E. Rognes, "A Stabilized Nitsche Fictitious Domain Method for the Stokes Problem," *Journal of Scientific Computing*, vol. 61, no. 3, pp. 604–628, 2014.

[10] C. Lehrenfeld and A. Reusken, "High Order Unfitted Finite Element Methods for Interface Problems and PDEs on Surfaces," pp. 33–63, 2017.

[11] Lehrenfeld, Christoph and Olshanskii, Maxim, "An eulerian finite element method for pdes in time-dependent domains," *ESAIM: M2AN*, vol. 53, no. 2, pp. 585–614, 2019. [Online]. Available: https://doi.org/10.1051/m2an/2018068

[12] A. Massing, M. G. Larson, A. Logg, and M. E. Rognes, "A stabilized Nitsche overlapping mesh method for the Stokes problem," *Numerische Mathematik*, vol. 128, no. 1, pp. 73–101, 2014.

[13] A. Johansson, M. G. Larson, and A. Logg, "High order cut finite element methods for the Stokes problem," *Advanced Modeling and Simulation in Engineering Sciences*, vol. 2, no. 1, pp. 1–23, 2015. [Online]. Available: http://dx.doi.org/10.1186/s40323-015-0043-7

[14] J. S. Dokken, S. W. Funke, A. Johansson, and S. Schmidt, "Shape optimization using the finite element method on multiple meshes with nitsche coupling," *SIAM Journal on Scientific Computing*, vol. 41, no. 3, pp. A1923–A1948, 2019.

[15] A. Johansson, B. Kehlet, M. G. Larson, and A. Logg, "Multimesh finite element methods: Solving PDEs on multiple intersecting meshes," *Computer Methods in Applied Mechanics and Engineering*, 2019.

[16] K. Eriksson, D. Estep, P. Hansbo, and C. Johnson, *Computational Differential Equations*. Lund: Studentlitteratur, 1996.

[17] M. G. Larson and F. Bengzon, *The Finite Element Method: Theory, Implementation, and Applications*. Springer Berlin Heidelberg, 2013.

[18] M. Asadzadeh, *An Introduction to the Finite Element Method for Differential Equations*. Wiley, 2020.

[19] T. Boiveau, E. Burman, S. Claus, and M. Larson, "Fictitious domain method with boundary value correction using penalty-free Nitsche method," *Journal of Numerical Mathematics*, vol. 26, no. 2, pp. 77–95. [Online]. Available: https://www.degruyter.com/view/journals/jnma/26/2/article-p77.xml

[20] S. Larsson and V. Thomée, *Partial Differential Equations with Numerical Methods*. Springer Berlin Heidelberg, 2003.

[21] S. Brenner and R. Scott, *The Mathematical Theory of Finite Element Methods*, 3rd ed. Springer-Verlag New York, 2008.

[22] F. Tantardini, "Quasi-optimality in the backward Euler-Galerkin method for linear parabolic problems," PhD-thesis, Universita degli Studi di Milano, 2013.

[23] V. Thomée, *Galerkin Finite Element Methods for Parabolic Problems*, 2nd ed. Springer Berlin Heidelberg, 2006.

[24] K. Eriksson and C. Johnson, "ADAPTIVE FINITE ELEMENT METHODS FOR PARABOLIC PROBLEMS I: A LINEAR MODEL PROBLEM," *SIAM Journal on Numerical Analysis*, vol. 28, no. 1, pp. 43–77, 1991.

[25] ——, "Adaptive Finite Element Methods for Parabolic Problems II: Optimal Error Estimates in $L_\infty L_2$ and $L_\infty L_\infty$," *SIAM Journal on Numerical Analysis*, vol. 32, no. 3, pp. 706–740, 1995.

[26] C. Lundholm, "A space-time cut finite element method for a time-dependent parabolic model problem," MSc thesis, Chalmers University of Technology and University of Gothenburg, 2015.

[27] K. K. GUPTA and J. L. MEEK, "A brief history of the beginning of the finite element method," *International Journal for Numerical Methods in Engineering*, vol. 39, pp. 3761–3774, 1996.

[28] L. T. Tenek and J. Argyris, *A brief history of FEM*. Dordrecht: Springer Netherlands, 1998, pp. 17–25. [Online]. Available: https://doi.org/10.1007/978-94-015-9044-0_2

[29] A. Logg, K. A. Mardal, and G. N. Wells, Eds., *Automated solution of differential equations by the finite element method*. Springer, 2012, vol. 84.