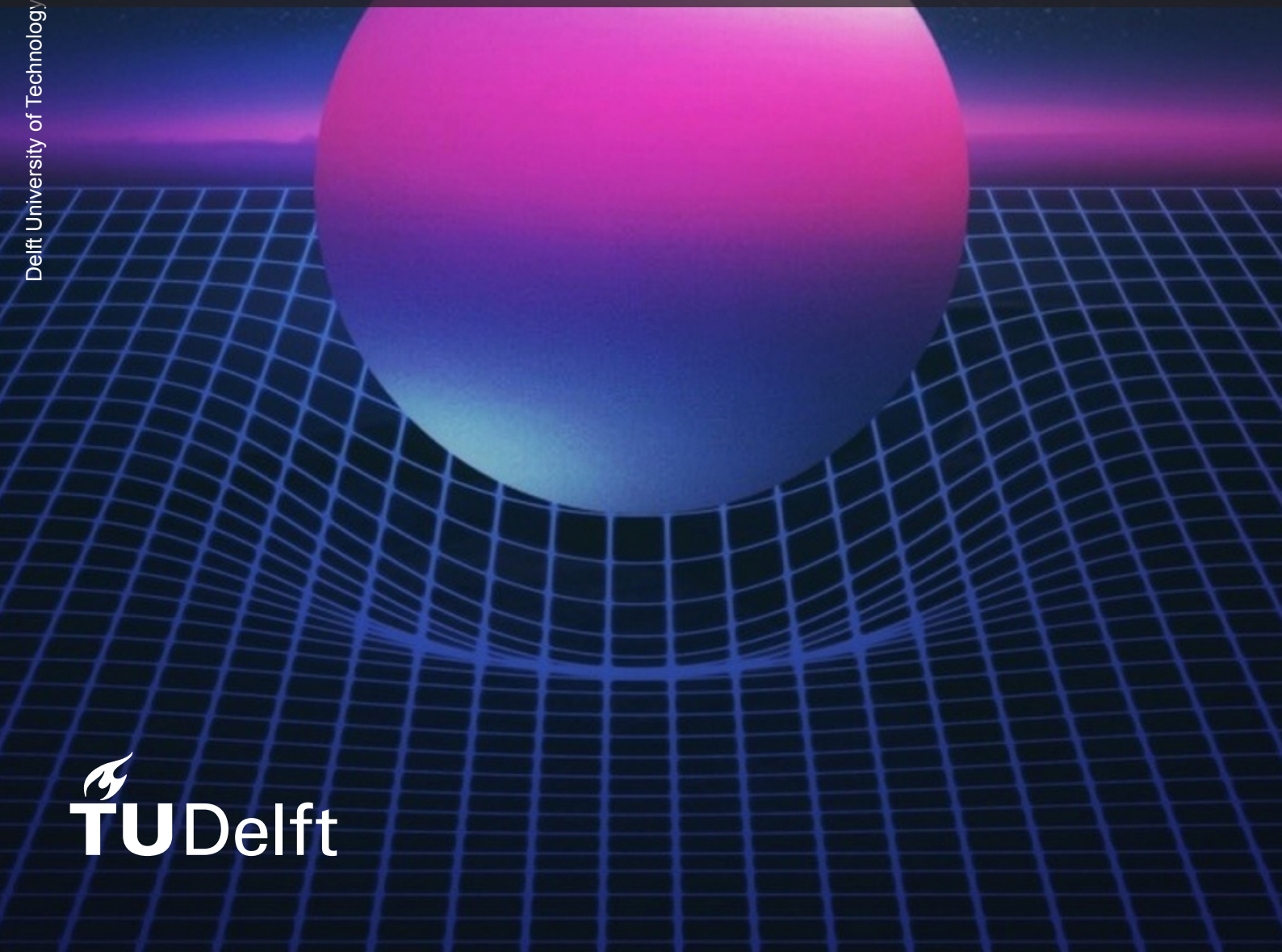
Space-Time Parallel Algorithms for Boundary Element Methods

An exploration of parallelization, preconditioning and implementation for the heat equation in a space-time setting

Thesis Msc. Applied Mathematics

Daniël Hoonhout



Delft University of Technology

TU Delft

Space-Time Parallel Algorithms for Boundary Element Methods

An exploration of parallelization, preconditioning and implementation for the heat equation in a space-time setting

by

Daniël Hoonhout

Student Name	Student Number
Daniël Hoonhout	4389565

Daily Supervisor:	dr. Carolina Urzúa-Torres
Responsible Professor:	Prof.dr.ir. Kees Vuik
Institution:	Delft University of Technology
Place:	Faculty of Applied Mathematics, Delft
Project Duration:	January, 2021 - September, 2021

Cover Image: Physical space-time representation, unrelated. Retreived at: https://wallpapercave.com/spacetime-wallpapers

TU Delft

Preface

In the past nine months, I worked on my master thesis, of which the result lies in front of you. In a way, this thesis will show not only what I have been working on in the past months. Instead, it provides an overview of the things I have learned in the two years of my masters degree Applied Mathematics at the University of Delft. One could even take it a step further and regard it as work resulting from being a student of the University of Delft for the past seven years.

I have performed research in parallelization of space-time boundary element methods, restricted to the heat equation. Research in this matter was proposed nine months ago by dr. Carolina Urzúa-Torres in collaboration with prof.dr.ir. Kees Vuik. This subject spoke to me for two reasons, which coincidentally corresponds to the two main subjects of this work: Space-time (boundary element) methods and parallel computing. I found the concept of combining space and time immediately fascinating, and was curious to the mathematics behind it. Conceptually, I had already heard of parallel programming, however, I had no prior encounters with the subject. I wished to gain more knowledge of a concept with increasing presence and effect in modern research, and, for that reason, was interested in this matter.

First of all, I want to thank dr. Carolina Urzúa-Torres, my daily supervisor, for managing to provide me with an immense amount of knowledge by introducing me to a whole new challenging world of mathematics. The accompanied enthusiasm to this introduction made me become infatuated with numerical analysis, to such a high degree even, that I will pursue researching in this area in the future.

Secondly, want to thank prof.dr.ir. Kees Vuik, for quickly bringing me up to speed on the theoretical and practical aspects of parallelization, and numerical methods in general. This passing of knowledge transcends the past nine months. Regarding both supervisors, I would like to emphasise that all this has been done without a single physical meeting; An unfortunate consequence of COVID-19, solved very efficiently.

Furthermore, I want to thank my parents for regularly asking about the project and showing support throughout, regardless of understanding everything I wrote.

Lastly, I would like to thank my roommates. With effort from your sides, we were able to create an environment at home where studying was always possible, even in times of quarantine.

Equipped with the knowledge and experiences I acquired in my years at the University of Delft. I am excited to apply this in the PhD track, which I plan to follow after graduation.

I hope you will enjoy reading this thesis as much as I enjoyed discovering this material myself for the first time.

Daniël Hoonhout
Delft, September 2021

Abstract

In this thesis we revisit theoretical background for space-time boundary element methods for the heat equation and its implementation. We restrict ourselves to solving the one-, and two dimensional Dirichlet heat equation. A new approach is proposed to approximate the Galerkin matrix entries in a semi-analytical fashion, requiring a reduced order of quadrature. This method can be applied to non-uniform meshes, but is restricted to right-triangular meshes.

Using this approach, a system of Galerkin equations is created and solved iteratively with the use of the Generalised Residual Method (GMRES). Operator preconditioners and preconditioners originating from domain decomposition methods are summarised and implemented for a two dimensional Dirichlet problem. In the case of operator preconditioning, a diagonal duality pairing proposed by Stevenson and van Venetië [23] is used in the implementations. The Restricted Additive Schwarz method is considered as both a preconditioner and a basic iterative method.

The Calderón preconditioner, an operator preconditioner, is used to test the efficiency of parallel preconditioned GMRES implementations, as this preconditioner provides a dense matrix. For different amount of processes, the parallel GMRES implementations are investigated. Using row-wise decomposition, parallel GMRES becomes increasingly time-efficient, as the level of refinement increases. However, the Induced Dimension Reduction method, a different non-symmetric solver, currently outperforms the parallel GMRES implementation.

Contents

List of Figures

List of Tables

1 Introduction

Boundary value problems for partial differential equations are omnipresent in mathematically oriented fields such as physics and a wide range of disciplines in engineering. A classical tool for the analysis and solution of boundary value problems are the *boundary integral equations* (BIE). Any method that provides an approximate numerical solution of these boundary integral equations, is categorised as a *boundary element method* (BEM) [2]. In terms of computational cost, these methods are especially useful when dealing with infinite domain problems or other problems where there is a large volume to surface ratio.

In general, the numerical solution of the evolution of partial differential equations relies on semi-discretizations, either in space or in time. In contrast, by treating the temporal dimension as another spatial dimension, a space-time discretization is obtained. This allows for adaptivity and parallelization in space and time simultaneously, without additional effort. Given today's computing capabilities, this feature is of great advantage, as it paves the way to develop algorithms that can use a large number of cores more efficiently. For this reason, we pursue space-time BEM in this thesis.

Recent developments in coupling *Finite Element Methods* (FEM) with BEM, and new space-time formulations for the wave equations, motivate us to conduct further research in space-time BEM parallelization. In contrast to the parallelization of space-time BEM studied by Dohr [3], these new formulations use a modified Hilbert transform, which causes the system-matrices to be also dense in time. Hence, we can no longer exploit the structure given by the fundamental solution in the choice of parallel algorithms. Moreover, in the spirit of true space-time BEM, this thesis focuses on space-time triangulations of the lateral boundary.

The space-time lateral boundary is discretized using triangular elements and the theory regarding the trial spaces given such a triangulation is examined. This theory is an extension of the already existing theory on error estimates given a tensor-product discretization [2, 15, 20, 3]. Furthermore, certain entries of the Galerkin matrix have singular integrands, one needs a strategy to increase the convergence rate of the quadrature in these cases.

Previous work mainly dealt with tensor-product meshes [15, 2, 3, 16], where *Boundary Integral Operators* (BIO) were integrated with a semi-analytical approach; or, more recently, with a general space-time triangulation [13]. This thesis offers, by restricting to right-triangles, a new semi-analytical approach for the integration of the BIOs. At the moment, this approach is only proven to be viable for space-time triangulations with an added constraint: the elements must be right-angled and two of the edges must be parallel to either the temporal or spatial axis. However, the method is not restricted to uniform-meshes.

For both the one-, and two dimensional case, the numerical approximation requires the solution of a linear system. These systems are solved using the iterative *Generalised Residual Method* (GMRES). In order to efficiently improve convergence rate of iterative solvers, one needs to use a preconditioner. Operator preconditioning is a commonly used preconditioning strategy when it comes to BEM, [8, 1, 21, 3, 23]. A new diagonal duality pairing for operator preconditioning has been proposed by Stevenson and van Venetië [23], which allows us to apply operator preconditioning more efficiently, given a general triangulation. For such general triangulation, the standard approach of operator preconditioning using a dual mesh has not yet been validated.

Apart from operator preconditioning, this thesis explores the possibility of using domain decomposition methods as a preconditioner and as a stand-alone iterative method. This will allow us to compare operator preconditioning, which is currently dominating in literature regarding space-time BEM, to other techniques. Alongside comparisons, introducing domain decomposition methods will also give us a broad impression of the effectiveness of inherently parallel preconditioners and iterative methods in a space-time setting. Though not required, this thesis only presents uniform decompositions of the space-time domain. In summary two forms of preconditioning are revisited: *Calderón* preconditioning and *Restricted Additive Schwarz* (RAS) preconditioning. The latter is also considered as a stand-alone iterative method.

As another comparison to GMRES, the *Induced Dimension Reduction* (IDR) method, a non-symmetric solver, is introduced. The efficiency of parallelization of this method on a basic level is explored.

The structure of this thesis is as follows: In Chapter 2, the theoretical framework regarding space-time BEM is revisited, mainly based on the one-dimensional Dirichlet heat equation. An implementation, using the tensor-product ansatz [15, 2, 3, 16] is presented.

A more general revision of the theoretical framework established in the second chapter is introduced in Chapter 3. A new semi-analytical approach is derived, to efficiently approximate the entries of the Galerkin matrix with lower order quadrature. Implementations of the two dimensional BEM, given different levels of uniform refinement, are considered.

The implementations presented in Chapter 3 are used to test the effectiveness of preconditioning and parallelization in Chapter 4. First, theory regarding different preconditioning strategies is summarised. The effect of the preconditioners is compared, using iteration count as a benchmark. Since the usage of Calderón preconditioning leads to a fully dense matrix, an implementation of this preconditioner is considered for parallelization. Basic parallelization schemes, such as row-wise parallel matrix vector products and parallel l^2-norms are presented, and used to construct a parallel GMRES method. From a computational-time perspective, the efficiency of parallel GMRES, given different degrees of parallelism is investigated.

As a comparison to GMRES, the IDR(s) method is also revisited. Using similar parallelization techniques as in the case of GMRES, the method is parallelized and, again tested for time-wise efficiency, given different degrees of parallelism.

2

1D problem

2.1. 1D Dirichlet problem

We start by considering the following relatively simple heat equation in one dimensional domain $\Omega = (0, 1)$ with boundary $\Gamma = \{0, 1\}$. This is

$$\partial_t u + \partial_{xx} u = 0, \tag{2.1}$$

with Dirichlet boundary conditions:

$$u|_\Gamma = g, \tag{2.2}$$

and initial condition:

$$u(x, 0) = f(x) = 0. \tag{2.3}$$

The goal is to solve these equations in a space-time setting up to time T, analogous to the multiple dimensional case described in Noon's thesis [15]. The space-time cylinder we are interested in is denoted as $Q := \Omega \times (0, T)$, and its lateral boundary as $\Sigma := \Gamma \times (0, T)$. The trace operators are defined as an extension of the following continuous linear operators

$$\begin{aligned} &\gamma_0 v := v|_\Sigma, \quad \forall v \in C(Q), \\ &\gamma_1 v := \partial_n v|_\Sigma = (\nabla v|_\Sigma) \cdot n, \quad \forall v \in C^1(Q), \end{aligned}$$

to the corresponding Sobolev spaces, which will be defined in the next section. With this, we can rewrite the Dirichlet problem as: Given $g : \Sigma \to \mathbb{R}$, find $u : Q \to \mathbb{R}$ such that

$$\begin{cases} \partial_t u + \partial_{xx} u = 0, & \text{in Q}, \\ u = 0, & \text{in } \Omega \times \{t = 0\}, \\ \gamma_0 u = g, & \text{in } \Sigma. \end{cases} \tag{2.4}$$

2.1.1. Formulation as boundary integral equations

In order to write the formulation of (2.4) in terms of boundary integral equations, first the required function spaces to solve this problem need to be established.

A d-tuple of non-negative integers $\alpha = (\alpha_1, \ldots, \alpha_d) \in \mathbb{N}^d$ is defined as a *multi-index* of dimension d. Its order is given by

$$|\alpha| := \alpha_1 + \cdots + \alpha_d,$$

and, for a C^∞-function f, we define

$$\partial^\alpha f := \partial^{\alpha_1} \ldots \partial_d^{\alpha_d} f.$$

This leads to the following definition:

Definition 2.1 ([14, Defn.6.6, Prop.6.7])**.** *The **Sobolev space** of functions with k-th weak derivative and L^2 integrability is denoted as*

$$H^k(\Omega) := \{f \in L^2(\Omega) : \partial^\alpha u \in L^2(\Omega)\ , \forall \alpha \text{ such that } |\alpha| < k\},$$

with norm

$$\|f\|_{H^k(\Omega)} := \sum_{|\alpha|\leq k} \|\partial^\alpha f\|_{L^2(\Omega)}.$$

Additionally, we define the *Bochner space*.

Definition 2.2 ([14, Ch.1])**.** *Let $(\Omega, \mathcal{F}, \mu)$ be a measure space and let X be a Banach space. For $1 \leq p \leq \infty$, we denote by the **Bochner space** $L^p(\Omega; X)$ the space of all equivalence classes of strongly measurable functions $f : \Omega \to X$, for which $\omega \mapsto \|f(\omega)\|$ belongs to $L^p(\Omega)$.*

Remark. *$L^p(\Omega; X)$ is a Banach space with respect to the norm*

$$\|f\|_p := \|\omega \mapsto \|f(\omega)\|\|_{L^p(\Omega)}\ .$$

In this thesis we are solely interested in the case $p = 2$.

Subsequently, the anisotropic Sobolev space on the space time cylinder for all $r, s > 0$ is defined as [15, 3, 16]:

$$H^{r,s}(\Omega) := L^2((0,T); H^r(\Omega)) \cap H^s((0,T); L^2(\Omega)),$$

where

$$H^s((0,T); L^2(\Omega)) := \{u \in L^2(Q) : |u|_{H^s((0,T;L^2(\Omega))} < \infty\},$$

with

$$|u|_{H^s((0,T);L^2(\Omega))} := \int_\Omega \|u(x,\cdot)\|^2_{H^s(0,T)}\ dx.$$

The norm corresponding to the anisotropic Sobolev space for $r, s \in (0,1)$ is given by

$$\|u\|^2_{H^{r,s}} = \|u\|^2_{L^2((0,T);H^r(\Omega))} + \|u\|^2_{H^s((0,T);L^2(\Omega))}.$$

We can further restrict ourselves to spaces with zero initial condition [15, 3, 16]:

$$\tilde{H}^{r,s}(Q) := \{u \in \tilde{H}^{r,s}(Q) : u(x,t) = 0,\ t < 0\}.$$

We will look for solutions of (2.4) in the space $\tilde{H}^{1,\frac{1}{2}}(Q)$. Given second order operator $L := \partial_t + \partial_{xx}$, let $\tilde{H}^{1,\frac{1}{2}}(Q; L) := \{u \in \tilde{H}^{1,\frac{1}{2}}(Q) : Lu \in L^2(Q)\}$. It is worth noting that the trace operators for this space are continuous linear mappings [1, Lem.2.4, Prop.2.18]

$$\gamma_0 : \tilde{H}^{1,\frac{1}{2}}(Q) \to H^{\frac{1}{2},\frac{1}{4}}(\Sigma),$$

and,

$$\gamma_1 : \tilde{H}^{1,\frac{1}{2}}(Q; L) \to H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma),$$

with $H^{\frac{1}{2},\frac{1}{4}}(\Sigma)$ defined analogously to $H^{r,s}$, and $H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma)$ the dual space of $H^{\frac{1}{2},\frac{1}{4}}(\Sigma)$. Then, by Green's second identity we have:

$$\begin{aligned}&\int_Q v(x,t-t_0)Lu(x,t) - u(x,t-t_0)Lv(x,t)\ dxdt \\ &= \int_\Sigma \gamma_0 u(x,t)\cdot\gamma_1 v(x,t-t_0)\ dxdt - \int_\Sigma \gamma_1 u(x,t)\cdot\gamma_0 v(x,t-t_0)\ dxdt,\end{aligned} \tag{2.5}$$

for $t_0 \in \mathbb{R}$ and $u, v \in \tilde{H}^{\frac{1}{2},\frac{1}{4}}(Q; L)$. The fundamental solution, $G(x,t)$, of the d-dimensional heat equation is given by

$$G(x,t) := \begin{cases} \frac{1}{(4\pi t)^{\frac{d}{2}}} e^{-|x|^2/4t} & t \geq 0 \\ 0 & t < 0 \end{cases}. \tag{2.6}$$

Substituting v with $G(x-y,t-s)$ in (2.5), yields:

$$u(x,t) = \int_\Sigma \left(G(x-y,t-s)\frac{\partial u}{\partial n_y}(y,s) - \frac{\partial}{\partial n_y} G(x-y,t-s)u(y,s) \right) \, dyds. \tag{2.7}$$

Remark. *The left hand-side of* (2.7) *follows from the fact that* (2.1) *implies that* $Lu = 0$*. And, as a property of the fundamental solution, we have that* $L(G(x,t)) * u(x,t) = u(x,t)$*, where* $*$ *denotes the convolution of the functions.*

Equation (2.7) is the *Representation formula* for the heat equation, given that there is no source function present. Using this representation, we can define the boundary layer potential operators. The *single layer potential operator* [2, 15, 3, 16]

$$\mathcal{K}_1(\phi)(x,t) := \int_\Sigma \phi(y,s) G(x-y,t-s) \, dyds, \quad (x,t) \in Q, \ \phi \in L^1(Q),$$

and the *double layer potential operator*

$$\mathcal{K}_2(\phi)(x,t) := \int_\Sigma \phi(y,s) \frac{\partial}{\partial n_y} G(x-y,t-s) \, dyds, \quad (x,t) \in Q, \ \phi \in L^1(Q).$$

By these definitions, we can rewrite (2.7) as

$$u(x,t) = \mathcal{K}_1(\gamma_1 u) - \mathcal{K}_2(\gamma_0 u), \quad (x,t) \in Q. \tag{2.8}$$

Now that the boundary layer potential operators are properly defined, we can define the boundary integral operators.

The single layer potential operator is continuous across the boundary [2]. Hence we can define *The single layer operator*

$$V\psi := \gamma_0 \mathcal{K}_1 \psi, \quad \psi \in H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma).$$

Given the domain Ω, let us denote the trace taken from inside Ω with $\gamma_i^{int}, i = \{0,1\}$. Similarly let us denote the trace taken from $\bar{\Omega}^c := \mathbb{R}^d \backslash \Omega$, by γ_i^{ext}. We define a "jump" across the boundary $[\cdot]_\Gamma$ by [2]

$$[\gamma_i v]_\Gamma := \gamma_i^{ext} v - \gamma_i^{int} v, \quad x \in \Gamma, \ i = 0,1. \tag{2.9}$$

Now, let us consider the following lemma:

Lemma 2.1 ([20, Ch.6], [3, Ch.5])**.** *Let* $v \in C(\bar{\Sigma})$ *and* $(x,t) \in \Sigma$*. The double layer potential* $\mathcal{K}_2 v$ *satisfies the following relations*

$$\gamma_0^{int}(\mathcal{K}_2 v)(x,t) = -\frac{1}{2} v(x,t) + \int_\Sigma \phi(y,s) \frac{\partial}{\partial n_y} G(x-y,t-s) \, dyds, \quad (x,t) \in \Sigma,$$

$$\gamma_0^{ext}(\mathcal{K}_2 v)(x,t) = \frac{1}{2} v(x,t) + \int_\Sigma \phi(y,s) \frac{\partial}{\partial n_y} G(x-y,t-s) \, dyds, \quad (x,t) \in \Sigma.$$

From Lemma 2.1 we obtain the following *jump relation* [20]

$$[\gamma_0 \mathcal{K}_2 v]_\Sigma = \gamma_0^{ext} \mathcal{K}_2 v(x) - \gamma_0^{int} \mathcal{K}_2 v(x) = v(x). \tag{2.10}$$

For the double layer potential and $\phi \in H^{\frac{1}{2},\frac{1}{4}}(\Sigma)$, there also holds:

$$\gamma_0^{int} \mathcal{K}_2 \phi(x,t) = -\frac{1}{2}\phi(x,t) + \int_\Sigma \phi(y,s) \frac{\partial}{\partial n_y} G(x-y,t-s) \, dyds, \quad (x,t) \in \Sigma. \tag{2.11}$$

From this we define the *double layer operator*

$$K\phi := \int_\Sigma \phi(y,s) \frac{\partial}{\partial n_y} G(x-y,t-s) \, dyds.$$

The conormal derivative of the double layer potential, defines the *hypersingular operator W*, i.e.

$$W\phi := -\gamma_1 \mathcal{K}_2 \phi, \tag{2.12}$$

which is a a bounded operator from $H^{\frac{1}{2},\frac{1}{4}}(\Sigma)$ to $H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma)$. From (2.8) and (2.11) it follows directly that

$$\gamma_0 u = \gamma_0 \mathcal{K}_1(\gamma_1 u) - \gamma_0 \mathcal{K}_2(\gamma_0 u) = V\gamma_1 u - K\gamma_0 u + \frac{1}{2}\gamma_0 u. \tag{2.13}$$

Next, another useful lemma is introduced:

Lemma 2.2 ([20, Ch.6],[3, Ch.5]). *Let $w \in C(\bar{\Sigma})$, and $(x,t) \in \Sigma$. The single layer potential satisfies the relation*

$$\gamma_1(\mathcal{K}_2 w)(x) = \frac{1}{2}\psi(x,t) + \int_\Sigma \psi(y,s)\frac{\partial}{\partial n_x} G(x-y,t-s)\, dyds.$$

From Lemma 2.2, we define the *adjoint double layer operator* K' as:

$$(K'\psi)(x,t) + \frac{1}{2}\psi(x,t) = \gamma_1(\mathcal{K}_1\psi)(x,t), \quad (x,t) \in \Sigma, \tag{2.14}$$

for $\psi \in H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma)$. The adjoint double layer operator $K' : H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma) \to H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma)$ is a linear and bounded operator [15, 3].

Using (2.8), (2.12) an (2.14), we have

$$\gamma_1 u = \gamma_1 \mathcal{K}_1(\gamma_1 u) - \gamma_1 \mathcal{K}_2(\gamma_0 u) = W\gamma_0 u + \frac{1}{2}\gamma_1 u + K'\gamma_1 u. \tag{2.15}$$

Equation (2.13) and (2.15) give us the *Calderón Projector* C, defined as

$$\vec{\gamma} u = C\vec{\gamma} u = \begin{pmatrix} \frac{1}{2} - K & V \\ W & \frac{1}{2} + K' \end{pmatrix} \begin{pmatrix} \gamma_0 u \\ \gamma_1 u \end{pmatrix}, \tag{2.16}$$

with

$$\vec{\gamma} := \begin{pmatrix} \gamma_0 \\ \gamma_1 \end{pmatrix}.$$

It is important to note that (2.16) does not hold in general, but only when no source function is present. With the use of the Caldéron projector, one can directly obtain the boundary integral equations corresponding to the Dirichlet problem (2.4).

Direct method (Dirichlet): Find $q \in H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma)$ such that either (i) or (ii) holds

$$(i) \quad Vq = (\frac{1}{2} + K)g, \quad \text{(first kind)}. \tag{2.17}$$

$$(ii) \quad (\frac{1}{2} - K')q = Wg, \quad \text{(second kind)}. \tag{2.18}$$

Then the unique solution $u \in \tilde{H}^{1,\frac{1}{2}}(Q)$ of (2.1) is given by

$$u = \mathcal{K}_1 q - \mathcal{K}_2 g. \tag{2.19}$$

The layer potentials satisfy the homogeneous equation [2, 3, 16], i.e.

$$(\partial_t + \partial_{xx})\mathcal{K}_i\phi_i = 0, \quad \text{for } i \in \{1,2\},\ \phi_i \in \begin{cases} H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma) & i = 1 \\ H^{\frac{1}{2},\frac{1}{4}}(\Sigma) & i = 2 \end{cases}.$$

This implies that we can solve immediately for the Dirichlet problem (2.4) by substituting $\mathcal{K}_i\phi$ for u, which gives

$$\gamma_0 \mathcal{K}_i \phi_i = g \quad \text{for } i \in \{1,2\},\ \phi_i \in \begin{cases} H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma) & i = 1 \\ H^{\frac{1}{2},\frac{1}{4}}(\Sigma) & i = 2 \end{cases}. \tag{2.20}$$

With the previously defined boundary integral operators and (2.20) another method for solving the Dirichlet problem (2.4) follows:

Indirect method (Dirichlet) Find $q \in H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma)$ such that either (i) or (ii) holds

$$(i) \quad Vq = g, \quad \text{(first kind).} \tag{2.21}$$

$$(ii) \quad (K - \frac{1}{2})q = g, \quad \text{(second kind).} \tag{2.22}$$

Then the unique solution $u \in \tilde{H}^{1,\frac{1}{2}}(Q)$ of (2.1) is given by

$$(i) \quad u = \mathcal{K}_1 q \quad \text{(first kind)} \tag{2.23}$$

$$(ii) \quad u = \mathcal{K}_2 q \quad \text{(second kind)} \tag{2.24}$$

2.1.2. Solution via boundary element method

We consider the boundary integral equation determined by the indirect method of the first kind (2.21). Hence, we need to take a closer look at the single layer integral operator. The solution to the boundary integral equation is approximated using the Galerkin approach.

Let X_h be a closed subspace of $H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma)$ and $q_h \in X_h$ the unique solution to the Galerkin equations

$$\langle p_h, Vq_h\rangle_Q = \langle p_h, g\rangle_Q \quad \forall p \in X_h, \tag{2.25}$$

where $\langle\cdot,\cdot\rangle_Q$ denotes the $L^2(Q)$ duality pairing. The construction of X_h is done using tensor-product ansatz. In other words we consider time basis functions $\chi_k(t)$, space basis functions $\nu_\alpha(x)$, such that the doubly indexed set $\{\nu_\alpha(x)\chi(t)\}$ forms a basis for X_h. In time we consider the basis functions

$$\chi_k(t) := \begin{cases} 1, & (k-1)h_t < t < kh_t \\ 0, & \text{otherwise} \end{cases}, \quad k = 1, \ldots, N_t, \tag{2.26}$$

where $h_t := \frac{T}{N_t}$ and N_t is the total amount of time steps.

In space, we consider the discretization Γ_n of the surface boundary Γ into quadrilateral elements τ_i^Γ, i.e.,

$$\Gamma_n := \cup_{i=0}^{N_x} \tau_i^\Gamma.$$

Remark. *In one dimension, the "quadrilateral" elements τ_i^Γ are just the boundary points.*

The linear basis functions ν_α are defined such that

$$\nu_\alpha(x) := \delta_{\tau_\alpha^\Gamma}(x), \tag{2.27}$$

where $\delta_{\tau_\alpha^\Gamma}(x)$ represents the Dirac delta distribution with support $\tau_\alpha^\Gamma(x)$.

As was the objective, the doubly indexed set $\{\nu_\alpha(x)\chi_k(t)\}$ forms a basis of X_h. Using this basis, we can rewrite the Galerkin equations

$$\langle \nu_\alpha\chi_n, Vq_h\rangle_Q = \langle \nu_\alpha\chi_n, g\rangle_Q, \quad \alpha = 1, 2;\ n = 1, \ldots, N_t. \tag{2.28}$$

Since q_h is an element of X_h, we can approximate q_h in terms of basis functions

$$q_h(x,t) \approx \sum_{k=1}^{N_t}\sum_{\beta=1}^{2} q_{\beta,k}\nu_\beta(x)\chi_k(t). \tag{2.29}$$

This will be an equality, by definition of the basis functions, as $N_t \to \infty$. The approximation (2.29) is substituted into the Galerkin equations (2.28) which gives

$$\sum_{k=1}^{N_t}\sum_{\beta=1}^{2}\langle \nu_\alpha\chi_n, V\nu_\beta\chi_k\rangle_Q\ q_{\beta,k} = \langle \nu_\alpha\chi_n, g\rangle_Q, \quad \alpha = 1, 2;\ n = 1, \ldots, N_t. \tag{2.30}$$

By definition of the (L^2)-inner product, (2.30) can be written out into

$$\sum_{k=1}^{N_t}\sum_{\beta=1}^{2} q_{\beta,k}\int_\Gamma\int_\Gamma \nu_\alpha(x)\nu_\beta(y)\mathcal{F}_{n,k}(x-y)dydx = \int_\Gamma \nu_\alpha(x)\int_{(n-1)h_t}^{nh_t} g(x,t)dtdx, \tag{2.31}$$

with

$$\mathcal{F}_{n,k}(x) := \int_{(n-1)h_t}^{nh_t}\int_{(k-1)h_t}^{kh_t} G(x-y,t-s)\; dsdt.$$

To keep things orderly, (2.31) is written in matrix form as the $2N_t \times 2N_t$ system

$$\begin{pmatrix} \mathcal{G}_{1,1} & \dots & \mathcal{G}_{1,N_t} \\ \vdots & \ddots & \vdots \\ \mathcal{G}_{Nt,1} & \dots & \mathcal{G}_{N_t,N_t} \end{pmatrix}\begin{pmatrix} \vec{q}_1 \\ \vdots \\ \vec{q}_{N_t} \end{pmatrix} = \begin{pmatrix} \vec{F}_1 \\ \vdots \\ \vec{F}_{N_t} \end{pmatrix}, \tag{2.32}$$

with

$$\vec{q}_n = \begin{pmatrix} q_{1,n} \\ q_{2,n} \end{pmatrix}, \quad n = 1,\dots,N_t,$$

$$\mathcal{G}_{n,k} := \begin{pmatrix} \int_\Gamma\int_\Gamma \nu_1(x)\nu_1(y)\mathcal{F}_{n,k}(x-y)dydx & \int_\Gamma\int_\Gamma \nu_1(x)\nu_2(y)\mathcal{F}_{n,k}(x-y)dydx \\ \int_\Gamma\int_\Gamma \nu_2(x)\nu_1(y)\mathcal{F}_{n,k}(x-y)dydx & \int_\Gamma\int_\Gamma \nu_2(x)\nu_2(y)\mathcal{F}_{n,k}(x-y)dydx \end{pmatrix} \in \mathbb{R}^{2\times 2},$$

and

$$\vec{F}_n := \begin{pmatrix} \int_\Gamma \nu_1(x)\int_{(n-1)h_t}^{nh_t} g(x,t)dtdx \\ \int_\Gamma \nu_2(x)\int_{(n-1)h_t}^{nh_t} g(x,t)dtdx \end{pmatrix}.$$

By definition of the fundamental solution G, $\mathcal{F}_{n,k}(x)$ can be expressed as

$$\mathcal{F}_{n,k}(x) = \int_{(n-1)h_t}^{nh_t}\int_{(k-1)h_t}^{kh_t} \frac{1}{\sqrt{4\pi(t-s)}} e^{-|x|^2/4(t-s)}\mathbb{1}_{(t\geq s)}\; dsdt.$$

If $k > n$, we have that $s > t$ and the integration bounds are disjoint, which implies:

$$\forall k > n: \quad \mathcal{F}_{n,k} = 0 \Rightarrow \mathcal{G}_{n,k} = 0.$$

The matrix from (2.32) is thus of lower-triangular form.

In order to solve the system of equations (2.32), we need to determine the matrices $\mathcal{G}_{n,k}$ (for $k \leq n$) and the right-handside vector. Let $l = n - k$ and consider $\mathcal{F}_{n,k}$:

$$\begin{aligned}
\mathcal{F}_{n,k}(x) &= \int_{(n-1)h_t}^{nh_t}\int_{(k-1)h_t}^{kh_t} G(x-y,t-s)\; dsdt \\
&= h_t^2\int_0^1\int_0^1 G(x-y,h_t(\sigma_t-\sigma_s+l))\; d\sigma_s d\sigma_t \quad \text{with: } \sigma_s := \frac{s}{h_t} - k + 1; \sigma_t = \frac{t}{h_t} - n + 1 \\
&= -h_t^2\int_0^1\int_{\sigma_t+l}^{\sigma_t+l-1} G(x-y,h_t\sigma_l)d\sigma_l d\sigma_t \quad \text{with: } \sigma_l = \sigma_t - \sigma_s + l \\
&= h_t^2\int_0^1\int_{\sigma_t+l-1}^{\sigma_t+l} G(x-y,h_t\sigma_l)d\sigma_l d\sigma_t \\
&= h_t^2\int_{l-1}^{l}\int_0^{\sigma_l-l+1} G(x-y,h_t\sigma_l)d\sigma_t d\sigma_l + h_t^2\int_l^{l+1}\int_{s-l}^1 G(x-y,h_t\sigma_l)d\sigma_t d\sigma_l \\
&= h_t^2\int_{l-1}^{l}(\sigma_l - l + 1)G(x-y,h_t\sigma_l)d\sigma_l + h_t^2\int_l^{l+1}(l+1-\sigma_l)G(x-y,h_t\sigma_l)d\sigma_l.
\end{aligned}$$

If $l = 0$, the first integral will be zero by definition of fundamental solution.

Next, we define the incomplete gamma functions:

$$\mathbb{G}(s,x) := \int_x^\infty t^{s-1}e^{-t}\,dt. \tag{2.33}$$

By substitution, we have for all $r > 0$

$$\int_0^a \frac{1}{\sqrt{t}}e^{-r^2/4t}dt = \frac{r}{2}\int_{\frac{r^2}{4a}}^\infty u^{-\frac{3}{2}}e^{-u}du = \frac{r}{2}\mathbb{G}\left(-\frac{1}{2},\frac{r^2}{4a}\right). \tag{2.34}$$

if $r = 0$,

$$\int_0^a \frac{1}{\sqrt{t}}dt = \lim_{b\to 0} 2\sqrt{t}|_b^a = 2\sqrt{a} - 0 = 2\sqrt{a}.$$

Similarly, by substitution we have for $r > 0$:

$$\int_0^a \sqrt{t}e^{\frac{r^2}{4t}}dt = \frac{r^3}{4^{3/2}}\int_{\frac{r^2}{4a}}^\infty u^{\frac{-5}{2}}e^{-u}du = \frac{r^3}{4^{3/2}}\mathbb{G}\left(-\frac{3}{2},\frac{r^2}{4a}\right),$$

and, for $r = 0$:

$$\int_0^a \sqrt{t}dt = \frac{2}{3}a^{3/2}.$$

For notational convenience, we denote

$$\kappa(x,a) := \frac{x^2}{4h_t a}. \tag{2.35}$$

We also define the following functions:

$$f_1(x,l) := \begin{cases} \left|\frac{h_t}{\sqrt{4\pi}}\frac{x}{2}\mathbb{G}(-1/2,\kappa(x,l+1))\right| & x \neq 0, l \geq 0 \\ \frac{2h_t^{3/2}}{\sqrt{4\pi}}\sqrt{l+1} & x = 0, l \geq 0 \\ 0 & l < 0 \end{cases}, \tag{2.36}$$

and

$$f_2(x,l) := \begin{cases} \left|\frac{1}{\sqrt{4\pi}}\frac{x^3}{4^{3/2}}\mathbb{G}(-3/2,\kappa(x,l+1))\right| & x \neq 0, l \geq 0 \\ \frac{2h_t^{3/2}}{3\sqrt{4\pi}}(l+1)^{3/2} & x = 0, l \geq 0 \\ 0 & l < 0 \end{cases}. \tag{2.37}$$

These functions can be combined to give the matrices $\mathcal{G}_{n,k}$ for each n, k. Because we are considering the one dimensional problem, we have

$$\begin{aligned}\int_\Gamma\int_\Gamma \nu_1(x)\nu_2(y)\mathcal{F}(x-y)dydx &= \nu_1(x)\nu_2(y)\mathcal{F}(x-y)|_{y=0}^{y=1}|_{x=0}^{x=1} \\ &= \nu_1(x)\nu_2(1)\mathcal{F}(x-1) - \nu_1(x)\nu_2(0)\mathcal{F}(x)|_{x=0}^{x=1} \\ &= \nu_1(x)\nu_2(1)\mathcal{F}(x-1)|_{x=0}^{x=1} \\ &= \nu_1(1)\nu_2(1)\mathcal{F}(0) - \nu_1(0)\nu_2(1)\mathcal{F}(-1) = -\mathcal{F}(-1).\end{aligned}$$

Since $\Gamma = \{0,1\}$, we can rewrite the general form of $\mathcal{G}_{n,k}$ to:

$$\mathcal{G}_{n,k} = \begin{pmatrix} \mathcal{F}_{n,k}(0) & -\mathcal{F}_{n,k}(-1) \\ -\mathcal{F}_{n,k}(1) & \mathcal{F}_{n,k}(0) \end{pmatrix}.$$

These matrices can be solved directly since by $l = k - n$ we have

$$\begin{aligned}\mathcal{F}_l(x) = (1-l)(f_1(x,l-1) - f_1(x,l-2)) + f_2(x,l-1) - f_2(x,l-2) + \\ + (l+1)(f_1(x,l) - f_1(x,l-1)) - f_2(x,l) + f_2(x,l-1).\end{aligned} \tag{2.38}$$

Now, (2.32) can be solved for $\vec{q}$. As the matrix is lower triangular, solving requires relatively little computational cost. As stated in the indirect method (2.23), by applying the first layer potential operator to $q_h(x,t)$, we will obtain the discretization of the solution on the entire domain. We define

$$f_3^k(z) := \int_{(k-1)h_t}^{kh_t} G(z, t-s)ds = \int_{t-kh_t}^{t-(k-1)h_t} G(z,\sigma)d\sigma. \tag{2.39}$$

From (2.34) and (2.39) we obtain

$$f_3^k(z) = \begin{cases} \left| \frac{x}{2\sqrt{4\pi}} \left(E(-\frac{1}{2}, \frac{z^2}{4(t-(k-1)h_t)}) - E(-\frac{1}{2}, \frac{z^2}{4(t-kh_t)}) \right) \right| & x \neq 0 \\ 2\sqrt{t-(k-1)h_t} - 2\sqrt{t-kh_t} & x = 0 \end{cases},$$

and

$$\begin{aligned} u_h(x,t) := \mathcal{K}_1 q_h(x,t) &= \sum_{k=1}^{N_t} \sum_{\beta=1}^{2} q_{\beta,k} \int_0^T \chi_k(s) \int_\Gamma \nu_\beta(y) G(x-y, t-s)dyds \\ &= \sum_{k=1}^{N_t} \sum_{\beta=1}^{2} q_{\beta,k} \int_\Gamma \nu_\beta(y) \int_0^T \chi_k(s) G(x-y,t-s)dsdy \quad \text{(Fubini)} \\ &= \sum_{k=1}^{N_t} \sum_{\beta=1}^{2} q_{\beta,k} \int_\Gamma \nu_\beta(y) \int_{(k-1)h_t}^{kh_t} G(x-y,t-s)dsdy \\ &= \sum_{k=1}^{N_t} \sum_{\beta=1}^{2} q_{\beta,k} \int_\Gamma \nu_\beta(y) f_3^k(x-y)dy \\ &= \sum_{k=1}^{N_t} q_{2,k} f_3^k(x-1) - q_{1,k} f_3^k(x). \end{aligned} \tag{2.40}$$

2.1.3. Numerical results

We proceed to test our implementation with an example for which we know the solution. The Dirichlet problem (2.4) is assumed to have time dependent boundary conditions, on interval (0,1), more specifically:

$$g(x,t) = \begin{cases} 1-e^{-5t} & x = 0 \\ 1-e^{-10t} & x = 1 \end{cases}, \quad \forall t \geq 0, \tag{2.41}$$

and zero initial condition. For convenience we define $Bc(x,t)$ as

$$Bc(x,t) := \int g(x,t)\, dt = \begin{cases} t + 5e^{-5t} & x = 0 \\ t + 10e^{-10t} & x = 1 \end{cases}.$$

The right hand-side vector can be written in the 1D case and time dependent boundary conditions as

$$\begin{aligned} \vec{F}_n &= \begin{pmatrix} \int_\Gamma \nu_1(x) Bc(x,(n+1)h_t) - Bc(x,nh_t)dx \\ \int_\Gamma \nu_2(x) Bc(x,(n+1)h_t) - Bc(x,nh_t)dx \end{pmatrix} = \begin{pmatrix} \nu_1(x) Bc(x,(n+1)h_t) - Bc(x,nh_t)|_{x=0}^{x=1} \\ \nu_2(x) Bc(x,(n+1)h_t) - Bc(x,nh_t)|_{x=0}^{x=1} \end{pmatrix} \\ &= \begin{pmatrix} -Bc(0,(n+1)h_t) + Bc(0,nh_t) \\ Bc(1,(n+1)h_t) - Bc(1,nh_t) \end{pmatrix}, \quad n = 1,\ldots,N_t. \end{aligned}$$

The numerical results of the previously stated problem were obtained using an implementation in Python and compared to the solution. In the case of the Dirichlet problem with boundary conditions g(x,t) as given in (2.41), the exact solution $u(x,t)$ can be found by decomposing the solution into two components:

$$u(x,t) = w(x,t) + v(x,t). \tag{2.42}$$

We define

$$w(x,t) := g(0,t) + x(g(1,t) - g(0,t)),$$

such that $w(x,t)$ satisfies the same boundary conditions as $u(x,t)$. Note that $w_{xx} = 0$, and by (2.1) we have

$$w_t + v_t = v_x x \Rightarrow v_t = v_x x - w_t, \tag{2.43}$$

with

$$v(0,t) = v(1,t) = 0,\ v(0,t) = 0. \tag{2.44}$$

Hence we have to solve for $v(x,t)$, a homogeneous boundary value problem with zero initial condition and source term. $v(x,t)$ can be done by eigenfunction expansion with associated eigenvalues $\lambda_n = n\pi$ and eigenfunctions $\phi_n(x) = \sin(n\pi x)$. We can therefore write

$$w_t(x,t) = \sum_{n=1}^{\infty} \hat{w}_n(t)\phi_n(x),$$
$$v(x,t) = \sum_{n=1}^{\infty} \hat{v}_n(t)\phi_n(x).$$

Using this expression in combination with (2.43) implies

$$\sum_{n=1}^{\infty} (\dot{\hat{v}}_n + \lambda_n^2 \hat{v}_n - \hat{w}_n)\phi_n(x) = 0. \tag{2.45}$$

By the linear independency of the eigenfunctions, (2.45) implies

$$\dot{\hat{v}}_n(t) + \lambda_n^2 \hat{v}_n(t) = \hat{w}_n(t), \tag{2.46}$$

which is a first order linear ordinary differential equation. This can be solved using an integration factor and yields

$$\hat{v}_n(t) = \int_0^t e^{\lambda_n^2(t-\tau)} \hat{w}_n(\tau) d\tau + C_1(t). \tag{2.47}$$

Since the initial condition for $v(x,t)$ equals zero, we have that $C_1(t) = 0$. Since w_t is a known expression, it can be written as a Fourier sine series with known expression

$$\hat{w}_n(t) = 2\int_0^1 w_t(x,t)\phi_n(x)\, dx. \tag{2.48}$$

Now, all the expressions are available and thus we have that the exact solution to 2.4 is given by

$$u(x,t) = w(x,t) + \sum_{n=1}^{\infty} \hat{v}_n(t)\phi_n(x). \tag{2.49}$$

Both the exact and the numerical solution will be evaluated at the time interval (0,1). The infinite sums present in the analytical expression of the solution are truncated at $n = 1000$. The time-step difference h_t is set to $\frac{1}{100}$ and the steps in spatial direction, Δx, are taken to be $\frac{1}{100}$ as well. The numerical result is visualised with the use of a two dimensional heat map given in Figure 2.1.

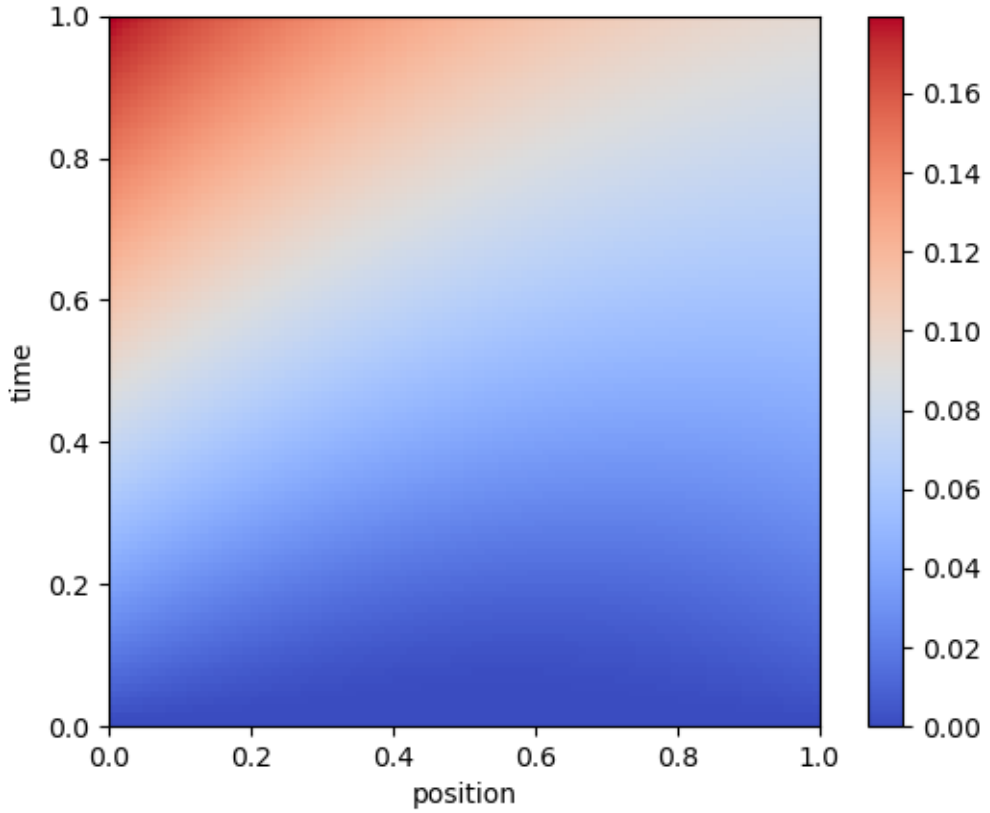


Figure 2.1: Heat map of numerical solution with h_t=0.01 and Δx=0.01

The numerical procedure described in Section 2.1.2, will leave us with the coefficient matrix u_h, constructed with the use of (2.40). With the use of interpolation functions, the solution can be expressed on the entire space-time domain $(0,1)^2$. In this case, the solution $\hat{u}(x,t)$ is defined as

$$\hat{u}(x,t) := \sum_{i=1}^{N_t} \sum_{j=1}^{N_x} u_h(i,j)\chi_i(t)\nu_j(x). \tag{2.50}$$

Typically, the numerical solution $\hat{u}$ will be compared to the exact solution with respect the L_2-norm. However, since we are considering a uniform mesh, the Euclidean norm can also be used to quantify the error. This is done as follows: we create a matrix whose entries consist of values of $\hat{u}(x,t)$, chosen such that every basis function χ_i and ν_j in $\hat{u}(x,t)$ contains a single value in its support. For example, the coefficient matrix u_h itself satisfies this constraint. Next, we compare the created matrix, or vectors of which the matrix is composed of, to the exact solution at the same points in time and space. This will yield a comparison of matrices or vectors, which can be described by the Euclidean norm. Due to the uniformity of the mesh, the norm will be equivalent to the L_2-norm up to a constant of $\Delta x \Delta t$.

The error over time is quantified with the use of the relative error

$$\epsilon = \frac{\|\hat{u} - u\|_2}{\|u\|_2}.$$

To get an idea of the relative error on the entire discretized domain and time interval, the Frobenius norm is used, i.e. given approximation matrix $\hat{A}$ and actual solution matrix A we have:

$$\frac{\|\hat{A} - A\|_F}{\|A\|_F},$$

with

$$\|A\|_F := \sqrt{\sum_{i,j} a_{i.j}^2}, \quad a_{i,j} \in A. \tag{2.51}$$

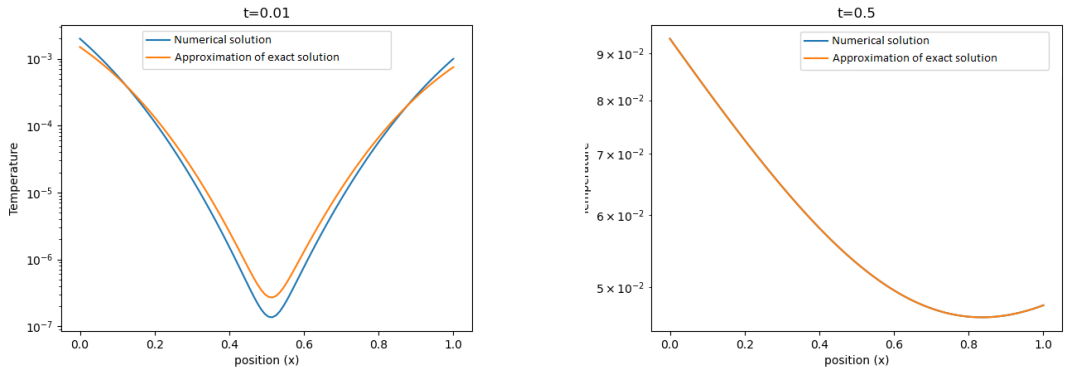
Given the discretisation we obtain a relative error with respect to the Frobenius norm of 1.3E-4. To see where the error takes most effect, we look at the relative error over time in certain parts of the domain, namely on $x = 0, x = 0.25$ and $x = 0.5$. The results are given in Table 2.1.

Table 2.1: Relative and absolute error on time interval (0,1), at different points in the domain with $\Delta t = \Delta x = 0.01$.

Position	Relative error	Absolute error
x=0.01	5.2E-4	5.4E-4
x=0.25	1.8E-5	1.5E-5
x=0.5	3.9E-6	2.6E-6
x=0.99	4.6E-4	2.5E-4

As expected, the error becomes larger when nearing the boundaries and becomes relatively small in the centre of the domain. This is a consequence of the Gaussian quadrature used to approximate $u_h(x,t)$ as defined in (2.40). Given $u_h(x,t)$, as x approaches the boundary, the regularity of the integrand in (2.40) reduces. The loss of the regularity yields a larger quadrature error, in the case of Gaussian quadrature.

Similarly, we can look at the error over the domain at certain points in time. As can be seen in Figure 2.2



(a) Temperature at time t=0.01

(b) Temperature at time t=0.5

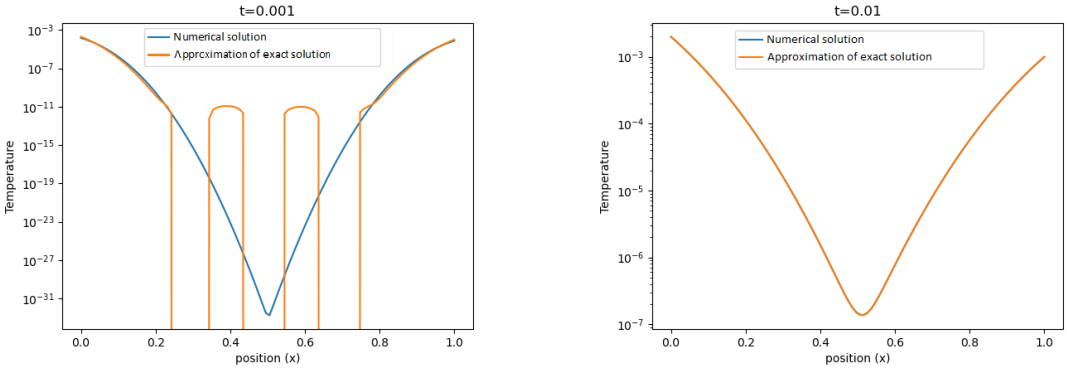
Figure 2.2: Temperatures at different points in time over the domain. Figure 2.2b seems to be a single line, this is because the solutions are nearly identical and the lines coincide.

The error hence diminishes when moving away from the initial condition and boundaries. To obtain a better approximation on the boundaries and initial condition, a smaller step-size in time is required. This step-size is used in both the construction of the Galerkin equations (2.32) as in the construction of the numerical solution u_h described in (2.40). Taking a smaller h_t improves the accuracy of the quadrature for the single layer potential in (2.40). In the one-dimensional case we are currently reviewing, there is no notion of h_x in (2.32), only of h_t. So, naturally, h_t will thus affect the error estimate in (2.32), and a reduction of h_t will yield a reduced error estimate. To validate this, the errors close to the spatial boundaries are given in Table 2.2, with a time step size of $\Delta t = 0.001$.

Table 2.2: Relative and absolute error on time interval (0,1), at different points close to the boundary, with $\Delta t = 0.001$ and $\Delta x = 0.01$.

Position	Relative error	Absolute error
x=0.01	1.2E-5	3.9E-5
x=0.99	1.1E-5	1.9E-5

Also, as expected, the error over the domain close to the initial condition will reduce both relatively (given a single timestep Δt) and in absolute sense ($\Delta t = 0.01$). This is visualised in Figure 2.3



(a) Temperature at time t=0.001

(b) Temperature at time t=0.01

Figure 2.3: Temperatures at different points in time over the domain. Note that the solution in Figure (a) is of such a small order, that the analytical approximation seems to falter due to machine precision. Figure 2.2b seems to be a single line, this is because the solutions are nearly identical and the lines coincide.

3

2D problem

Considering a two dimensional bounded Lipschitz domain Ω and boundary $\Gamma = \partial\Omega$, the Dirichlet problem can be formulated as

$$\begin{cases} \partial_t u - \Delta_x u = f, \\ u|_\Gamma = g, \\ u(\cdot, 0) = u_0. \end{cases} \tag{3.1}$$

3.1. Space-time formulation

To switch to a space time setting we define the space time cylinder $Q := \Omega \times (0, T)$, the lateral boundary $\Sigma := \Gamma \times (0, T)$, the initial region $\Omega_0 := \overline{\Omega} \times \{0\}$ and the final region $\Omega_T := \overline{\Omega} \times \{T\}$. Let $L := \partial_t - \Delta_x$ and consider the following proposition by van Kan et al. [11], which states

Proposition 3.1 ([11, p.141])**.** *If the linear operator L is of order $2s$, then the corresponding weak solution is of order s.*

The operator, L, is first order in time and second order in space. By Proposition 3.1, the weak solution of the heat equation is therefore in $H^{1,\frac{1}{2}}(Q)$.

For the first order (in time) operator$L$, $u \in C(\bar{Q})$ and $v \in C^1(\bar{Q})$, the trace operators are defined as

$$\gamma_0 u := \lim_{(\tilde{x},\tilde{t})\in Q \to (x,t)\in\Sigma} u(\tilde{x}, \tilde{t}), \tag{3.2}$$

and

$$\gamma_1 v := \lim_{(\tilde{x},\tilde{t})\in Q \to (x,t)\in\Sigma} \vec{n}_x \cdot \gamma_0(\nabla v(\tilde{x}, \tilde{t})). \tag{3.3}$$

Where $\vec{n}_x$ is the outward normal vector.

With the side-note that the Dirichlet problem in general can be rewritten into a Dirichlet problem with zero initial condition, u_0 is taken to be zero. Let us now introduce the following Sobolev spaces:

$$\begin{aligned} H^{r,s}(\Sigma)_{0;} &:= \{u \in H^{r,s}(\Sigma) : u|_\Gamma = 0\}, \\ H^{r,s}(\Sigma)_{;0,} &:= \{u \in H^{r,s}(\Sigma) : u(\cdot, t) = 0, t \le 0\}, \\ H^{r,s}(\Sigma)_{;,0} &:= \{u \in H^{r,s}(\Sigma) : u(\cdot, t) = 0, t \ge T\}, \\ H^{r.s}(Q, L) &:= \{u \in H^{r,s}(Q) : Lu \in L^2(Q)\}. \end{aligned}$$

Combination of indices for these Sobolev space will be used throughout this thesis as well, inheriting the properties of the previously defined spaces. For example:

$$H^{r,s}(\Sigma)_{0;,0} := \{u \in H^{r,s}(\Sigma) : u|_\Gamma = 0, u(\cdot, t) = 0, t \ge T\}.$$

3.1.1. Trace mapping properties

The trace operators can be extended to the anisotropic Sobolev-spaces. This extension of the Dirichlet trace operator can be proven with the use of the following theorems [12].

Theorem 3.2 (**Trace Theorem**, ([12, Thm.2.1])**.** *Let $u \in H(Q)^{r,s}(Q)$ with $r > \frac{1}{2}$ and $s \geq 0$. Then there exists a bounded linear operator $\gamma_0^{int} : H^{r,s}(Q) \to H^{\alpha,\beta}(\Sigma)$ with*

$$\|\gamma_0^{int} u\|_{H^{\alpha,\beta}(\Sigma)} \leq c_T \|u\|_{H^{r,s}(Q)},$$

for all $u \in H^{r.s}(Q)$. With $\alpha = r - \frac{1}{2}$ and $\beta = s - \frac{s}{2r}$. This bounded linear operator is an extension of the Dirichlet trace operator $\gamma_0^{int} u = u|_\Sigma$ for $C(\bar{Q})$.

Proof. For the proof, the reader is referred to the proof by Lions and Magenes ([12], Theorem 2.1).

Corollary 3.3. *The interior lateral trace operator*

$$\gamma_0^{int} : H^{1,\frac{1}{2}}(Q) \to H^{\frac{1}{2},\frac{1}{4}}(\Sigma),$$

is linear and bounded satisfying

$$\|\gamma_0^{int} v\|_{H^{\frac{1}{2},\frac{1}{4}}(\Sigma)} \leq c_T \|v\|_{H^{\frac{1,1}{2}}(Q)} \quad \forall v \in H^{1,\frac{1}{2}}(Q),$$

with c_T a constant depending only on s and Q.

Proof. This corollary is a direct consqeuence of Theorem 3.2, with $r = 1$ and $s = \frac{1}{2}$. □

Theorem 3.4 (**Inverse Trace Theorem**,[1, Ch.2], [3, Thm.2.4])**.** *The interior lateral trace operator*

$$\gamma_0^{int} : H^{1,\frac{1}{2}}(Q) \to H^{\frac{1}{2},\frac{1}{4}}(\Sigma)$$

has a continuous right inverse operator:

$$\mathcal{E} : H^{\frac{1}{2},\frac{1}{4}}(\Sigma) \to H^{1,\frac{1}{2}}(Q),$$

satisfying $\gamma_0^{int} \mathcal{E} w = w$ for all $w \in H^{\frac{1}{2},\frac{1}{4}}(\Sigma)$ as well as

$$\|\mathcal{E} w\|_{H^{1,\frac{1}{2}}(Q)} \leq c_{IT} \|w\|_{H^{\frac{1}{2},\frac{1}{4}}(\Sigma)}, \quad \forall w \in H^{\frac{1}{2},\frac{1}{4}}(\Sigma),$$

with c_{IT} a constant depending only on s and Ω.

3.1.2. Green's theorem

Using integration by parts, the weak formulation of the Dirichlet problem can be expressed as the domain-variational problem

Variational problem: Find $u \in H^{1,\frac{1}{2}}_{;0,}(Q)$ (as defined in Section 3.1) such that

$$\alpha(u, v) = l(v), \quad \forall v \in H^{1,\frac{1}{2}}_{0;,0}(Q), \tag{3.4}$$

with

$$\alpha(u, v) := \langle \partial_t u, v \rangle_Q + \langle \nabla_x u, \nabla_x v \rangle_Q,$$

and

$$l(v) := \langle f, v \rangle_Q.$$

The choice of these Sobolev spaces can be regarded as a result of reverse engineering, since it will give desired vanishing properties in Green's second identity (which will be discussed in this section). In order to arrive at the boundary integral equations and corresponding representation formula, we use the following lemmas [14, 1, 3]:

Lemma 3.5 ([14, Prop 4.7]). *Suppose X is a normed space and Y is a Banach space, and let X_0 be a dense subspace of X. If $T_0 : X_0 \to Y$ is a bounded operator, there exists a unique bounded operator $T : X \to Y$ extending T_0. The norm of this extension satisfies $\|T\| = \|T_0\|$.*

Lemma 3.6 ([1, Lem.2.22], [3, Lem.3.6]). *Let $C_0^\infty(\overline{\Omega} \times (0,T])$ denote the space of functions in $C_0^\infty(\mathbb{R} \times (0,\infty)$ restricted to $\overline{Q}$. Then $C_0^\infty(\overline{\Omega} \times (0,T])$ is dense in the anisotropic Sobolev spaces $H^{1,\frac{1}{2}}_{;0,}(Q)$ and $H^{1,\frac{1}{2}}_{;0,}(Q;L)$.*

Analogously, by defining $C_0^\infty(\overline{\Omega} \times [0,T))$ as the space of functions $C_0^\infty(\mathbb{R} \times (-\infty,T))$ restricted to $\overline{Q}$, we arrive a the Corollary 3.7 [3]

Corollary 3.7 ([3, Cor.3.7]). *The space $C_0^\infty(\overline{\Omega} \times [0,T))$ is dense in $H^{1,\frac{1}{2}}_{;,0}(Q)$ and $H^{1,\frac{1}{2}}_{;,0}(Q;L^*)$. Where L^* denotes the adjoint of L*

These results allow us to extend the classical Green's identities to the anisotropic Sobolev spaces. We obtain

Theorem 3.8 (**Green's first identity**, [16, Thm.2.2.1]). *For $u \in H^{1,\frac{1}{2}}_{;0,}(Q;L)$ and $v \in H^{1,\frac{1}{2}}_{;,0}(Q)$ there holds*

$$\alpha(u,v) = \langle \gamma_1^{int} u, \gamma_0^{int} v\rangle_{\partial Q} + \langle Lu, v\rangle_Q.$$

Proof. Though this theorem itself is not a new result, a new variation of the proof is added to this thesis and shown for sake of completeness. Let $u \in C_0^\infty(\overline{\Omega} \times (0,T])$ and $v \in C_0^\infty(\overline{\Omega} \times [0,T))$. Then the classical Green's first identity holds, i.e.

$$\alpha(u,v) = \langle \gamma_1^{int} u, \gamma_0^{int} v\rangle_{\partial Q} + \langle Lu, v\rangle_Q.$$

We will denote this bilinear mapping $u, v \mapsto \alpha(u,v)$ as the operator

$$T_0 : C_0^\infty(\overline{\Omega} \times (0,T]) \times C_0^\infty(\overline{\Omega} \times [0,T)) \to \mathbb{R}.$$

The operator T_0 is bounded in u and v in their respective Sobolev norms. Also, $\mathbb{R}$ clearly is a Banach space. By Lemma 3.6 and Corollary 3.7 it follows that the space $C_0^\infty(\overline{\Omega} \times (0,T]) \times C_0^\infty(\overline{\Omega} \times [0,T))$ is a dense subspace of $H^{1,\frac{1}{2}}_{;0,}(Q;L) \times H^{1,\frac{1}{2}}_{;,0}(Q)$. Hence, according to Lemma 3.5, there exists a norm-preserving extension of T_0 to the Sobolev spaces, which yields the desired result.

□

In a similar fashion we can extend Green's second identity to the desired Sobolev spaces.

Theorem 3.9 (**Green's second identity**, [16, Thm.2.2.1]). *For $u \in H^{1,\frac{1}{2}}_{;0,}(Q;L)$ and $v \in H^{1,\frac{1}{2}}_{;,0}(Q;L^*)$ there holds*

$$\langle Lu, v\rangle_Q - \langle u, L^* v\rangle_Q = \langle \gamma_0^{int} u, \gamma_1^{int} v\rangle_\Sigma - \langle \gamma_1^{int} u, \gamma_0^{int} v\rangle_\Sigma.$$

Proof. The extension follows again by a density argument analogous to the proof Theorem 3.8. However, note that the resulting duality pairing is no longer restricted to ∂Q but to Σ. Since $\partial Q = \Omega_0 \cup \Sigma \cup \Omega_T$, we can rewrite the result of Green's second identity to

$$\begin{aligned}\langle Lu, v\rangle_Q - \langle u, L^* v\rangle_Q = {} & \langle u(\cdot,0), \gamma_1 v\rangle_{\Omega_0} - \langle \gamma_1 u, v(\cdot,0)\rangle_{\Omega_0} \\ & + \langle \gamma_0 u, \gamma_1 v\rangle_\Sigma - \langle \gamma_1 u, \gamma_0 v\rangle_\Sigma \\ & + \langle u(\cdot,T), \gamma_1 v\rangle_{\Omega_T} - \langle \gamma_1, v(\cdot,T)\rangle_{\Omega_T}.\end{aligned}$$

By definition of the trace operator γ_1, this reduces to

$$\langle Lu, v\rangle_Q - \langle u, L^* v\rangle_Q = \langle u(\cdot,0), v(\cdot,0)\rangle_{\Omega_0} + \langle \gamma_0 u, \gamma_1 v\rangle_\Sigma - \langle \gamma_1 u, \gamma_0 v\rangle_\Sigma - \langle u(\cdot,T), v(\cdot,T)\rangle_{\Omega_T}.$$

Since we assumed $u \in H^{1,\frac{1}{2}}_{;0,}(Q;L)$ and $v \in H^{1,\frac{1}{2}}_{;,0}(Q;L^*)$, the initial and final time terms vanish and the statement is proven.

□

3.1.3. Representation formula

In Chapter 2, the representation formula has been given in (2.7). A slight generalisation is added in this section. Given the linear differential operator L, the representation formula follows by applying the two dimensional fundamental solution, $G(x-y,t-s)$ of L. For notational convenience we abbreviate $G(x-y,t-s)$ in the upcoming equations by G For $u \in H^{1,\frac{1}{2}}(Q;L)$ and $v \in H^{1,\frac{1}{2}}_{;,0}(Q;L^*)$ we obtain

$$\langle Lu, G\rangle_Q - \langle u, L^*G\rangle_Q = \langle u(\cdot,0), G\rangle_{\Omega_0} + \langle \gamma_0^{int}u, \gamma_1^{int}G\rangle_\Sigma - \langle \gamma_1^{int}u, \gamma_0^{int}G\rangle_\Sigma.$$

By definition of the fundamental solution, we have

$$\langle u, L^*G\rangle_Q = \int_Q u(y,s)\delta_Q(x-y,t-s)\,dQ = u(x,t).$$

By (3.1), we have $Lu = f$, and thus we obtain the general *representation formula*

$$u(x,t) = \langle u(\cdot,0), G\rangle_{\Omega_0} + \langle f, G\rangle_Q + \langle \gamma_1^{int}u, \gamma_0^{int}G\rangle_\Sigma - \langle \gamma_0^{int}u, \gamma_1^{int}G\rangle_\Sigma. \tag{3.5}$$

As shown in (3.5), the possibility of a source function and initial condition in (3.1) is now included into the representation formula.

3.1.4. Boundary integral equations

Given no source functions and a zero initial condition, the boundary integral equations are based on (2.16) introduced in Chapter 2. For sake of completeness the Calderon system is generalised to the case we have source function f and initial condition u_0, as stated in (3.1). Let us define the initial potential operators

$$M_i u_0(x,t) := \gamma_i^{int}\int_\Omega G(x-y,t)u_0(y)dy, \quad i=1,2,$$

and the Newton potential operators

$$N_i f(x,t) := \gamma_i^{int}\int_Q G(x-y,t-s)f(y,s)dyds, \quad i=1,2.$$

The by taking traces on (3.5) we obtain the general form

$$\vec{\gamma}u = C\vec{\gamma}u = \begin{pmatrix} \frac{1}{2}-K & V \\ W & \frac{1}{2}+K' \end{pmatrix}\begin{pmatrix}\gamma_0 u \\ \gamma_1 u\end{pmatrix} + \begin{pmatrix} M_0u_0 \\ M_1u_0\end{pmatrix} + \begin{pmatrix} N_0 f \\ N_1 f\end{pmatrix}, \tag{3.6}$$

with

$$\vec{\gamma} := \begin{pmatrix}\gamma_0 \\ \gamma_1\end{pmatrix}.$$

From (3.6) the construction of the generalised boundary integral equations is straightforward.

3.2. Space-time BEM

There are several ways to tackle the variational Dirichlet problem (3.4). Since the layer potentials solve the homogeneous problem Dirichlet problem, we can write:

$$u(x,t) = \mathcal{K}_1 q + \int_\Omega G(x-y,t)u_0(y)dydt + \int_Q G(x-y,t-s)f(y,s)dyds. \tag{3.7}$$

By taking the Dirichlet trace over (3.7) we obtain the indirect formulation of the first kind:
Find $q \in H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma)$ such that

$$Vq = g - M_0u_0 - N_0 f, \quad \text{on } \Sigma, \tag{3.8}$$

which will yield the unique solution $u \in H^{1,\frac{1}{2}}_{;0,}(Q)$ given by

$$u(x,t) = \mathcal{K}_1 q + M_0 u_0 + N_0 f.$$

This corresponds to the variational formula

$$\langle Vq, v\rangle_\Sigma = \langle g - M_0 u_0 - N_0 f, v\rangle_\Sigma, \quad \forall v \in H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma). \tag{3.9}$$

3.3. Triangulation

To approximate the solution to the variational formula, triangulation is performed on the space-time lateral boundary Σ. We follow the cue from [15, 3, 16]. We assume that the spatial boundary Γ is Lipschitz, and can be divided into M closed subsets Γ_m such that

$$\begin{aligned} &(i) \quad \Gamma = \bigcup_{m=1}^{M} \Gamma_m, \\ &(ii) \quad \mu(\Gamma_{m_1} \cap \Gamma_{m_2}) = 0, \ \forall m_1 \neq m_2, \\ &(iii) \quad \vec{n}|_{\Gamma_{m_1}} \cdot \vec{n}|_{\Gamma_{m_2}} \neq 1, \ \forall m_1 \neq m_2. \end{aligned} \tag{3.10}$$

where $\mu(\cdot)$ denotes the Lesbesgue measure. Now, we can divide the lateral boundary Σ into J closed subsets $\Sigma_j := \Gamma_j \times (0,T)$ with

$$\Sigma = \bigcup_{j=1}^{J} \Sigma_j. \tag{3.11}$$

Consider the family of admissible non-curved triangulations $\mathcal{T}$, of the lateral boundary. Then every triangulation $\tau \in \mathcal{T}$ can be decomposed into a finite union of boundary elements $e_l \in \Sigma$, i.e.

$$\Sigma_h \in \mathcal{T} \iff \Sigma_h = \bigcup_{l=1}^{N_h} e_l, \quad N_h < \infty.$$

We define the volume of the elements by

$$\Delta_l := \int_{e_l} ds,$$

its local mesh size by

$$h_l := \Delta_l^{\frac{1}{d-1}}, \tag{3.12}$$

and the global mesh size as

$$h := \max_{l=1,\dots,L} h_l, \tag{3.13}$$

with d the dimensions of the element. Let us consider such an admissible triangulation composed non-curved triangles, $\Sigma_h \in \mathcal{T}$.

Then, we can decompose Σ_h in such a way that for all elements e_l, there exists exactly one $j \in (1,\dots,J)$ such that $e_l \subset \Sigma_{h_j}$.

As stated by Reinarz [16], in two dimensions, the boundary Γ (of a simply connected domain) can be parameterized by a single periodic function:

$$\kappa : [0,1] \to \Gamma. \tag{3.14}$$

This function can be assumed to be analytic. Before the boundary elements in (x,y,t) can be transformed to a two dimensional reference element, the elements must be mapped to the two dimensional plane, in this case, $[0,1] \times (0,T)$ with the mapping $(x,y,t) \mapsto (\kappa^{-1}(x,y),t)$. For convenience, we will

denote $\kappa^{-1}(x, y)$ as $z(x, y)$. Now, the elements can be mapped to the reference element in the (ξ, η)-plane. When using piecewise linear basis functions (ϕ_i) must satisfy $\phi_i(\xi_j, \eta_j) = \delta_{ij}$ on the reference triangle with vertices $(0, 0)$, $(0, 1)$, $(1, 0)$. This yields:

$$\begin{aligned}\phi_1(\xi, \eta) &= 1 - \xi - \eta,\\ \phi_2(\xi, \eta) &= \xi,\\ \phi_3(\xi, \eta) &= \eta.\end{aligned}$$

Let the coordinates of the vertices of a (triangular) *spatially parameterized* element e_l be denoted by $\mathbf{x_1}, \mathbf{x_2}, \mathbf{x_3} \in \mathbb{R}^2$.

Remark. *The original elements will have vertices in $\mathbb{R}^3$. Consider, the example element with vertices (x_1, y_1, t_1), (x_2, y_2, t_2) and (x_3, y_3, t_3). After parameterization we can denote its vertices as $(z(x_1, y_1), t_1)$, $(z(x_2, y_2), t_2)$ and $(z(x_3, y_3), t_3) \in \mathbb{R}^2$. For notational convenience, we will denote $z(x_i, y_i)$ by z_i, $i \in \mathbb{N}$.*

Then the linear transformation from the element to the reference triangle is given by

$$\mathbf{x_1} \mapsto (0, 0), \quad \mathbf{x_2} \mapsto (1, 0), \quad \mathbf{x_3} \mapsto (0, 1),$$

and hence

$$z = z_1 + (z_2 - z_1)\xi + (z_3 - z_1)\eta, \tag{3.15}$$
$$t = t_1 + (t_2 - t_1)\xi + (t_3 - t_1)\eta. \tag{3.16}$$

3.4. Trial spaces

For a general domain X, we define the trial spaces of local polynomials. We will denote the trial space of piecewise constant functions as $S_h^0(X)$, and the trial space of piecewise linear continuous functions as $S_h^1(X)$. First, the approximation properties for $S_h^0(\Sigma_h)$ are presented following the use of Steinbach's and Dohr's work [20, 3]. Then, in a similar fashion, the properties of $S_h^1(\Sigma_h)$ summarised.

3.4.1. Piecewise constant basis functions

Tensor-product discretization

By viewing the the space-time lateral boundary Σ as a tensor product of time and space, we are able to derive the approximation properties of $S_h^0(\Sigma_h)$ by first treating time and space separately. These approximation properties can in turn be extended to the desired space $S_h^0(\Sigma_h)$.

The spatial boundary Γ, as stated in section 3.3, is assumed to be Lipschitz, piecewise smooth and (3.10) holds. By letting $\{\Gamma_{N_x}\}_{N_x} \in \mathbb{R}$ denote the the family of admissible triangulations on Γ, we have

$$\Gamma_{N_x} = \bigcup_{k=1}^{N_x} \overline{e_k^{(x)}}, \tag{3.17}$$

with $e_k^{(x)}$ the elements discretizing Γ. Similarly, the boundary with respect to the time, $I := (0, T)$ can be decomposed into line segments by considering the family of admissible decompositions $\{I_{N_t}\}_{N_t \in \mathbb{R}}$ satisfying

$$I_{N_t} = \bigcup_{k=1}^{N_t} \overline{e_k^{(t)}}. \tag{3.18}$$

given $e_k^{(t)}$ the elements discretizing I. The trial space of piecewise constant functions for the spatial and temporal triangulations respectively can now formally be defined as

$$S_{h_x}^0(\Gamma_{N_x}) := \mathsf{span}\{\phi_k^{\Gamma,0}\}_{k=1}^{N_x},$$

and

$$S^0_{h_t}(I_{N_t}) := \text{span}\{\psi^{I,0}_k\}_{k=1}^{N_t}.$$

The piecewise constant functions $\phi^{\Gamma,0}$ and $\psi^{I,0}$ are given by

$$\phi^{\Gamma,0}_k(x) := \begin{cases} 1, & x \in e^{(x)}_k, \\ 0, & \text{otherwise}, \end{cases}, \text{for } k = 1, \dots, N_x$$

and

$$\psi^{I,0}_k(t) := \begin{cases} 1, & t \in e^{(t)}_k, \\ 0, & \text{otherwise}. \end{cases}, \text{for } k = 1, \dots, N_t.$$

Next, let us define the L^2 projection operator and a contraction.

Definition 3.1. *Let X be an arbitrary function space defined over (Lipschitz) domain Q. The L^2 projection $Pu \in V$ of a function $u \in L^2(Q)$ is defined by*

$$\langle u - Pu, v\rangle_Q = 0, \quad \forall v \in X.$$

Remark. *Clearly, the L^2 projection operator is an orthogonal projection.*

Consider the L^2 projections $P_x u \in S^0_{h_x}(\Gamma_{N_x})$ and $P_t u \in S^0_{h_t}(I_{N_t})$. Then the following theorem holds [20]:

Theorem 3.10 ([20, Thm.10.2])**.** *Let Y be either Γ_{N_x} or I_{N_t}, with respective L^2 projections P_y being either P_x or P_t. Let $u \in H^s(Y)$, for some $s \in [0, 1]$. Then, there holds the error estimates*

$$\|u - P_y u\|^2_{L^2(Y)} \le c \sum_{k=1}^{N_y} h_k^{2s} |u|^2_{H^s(e^{(y)}_l)}, \tag{3.19}$$

and

$$\|u - P_y u\|_{L^2(Y)} \le c h^s |u|_{H^s(Y)}. \tag{3.20}$$

Proof. For completeness, a part of the proof is added to the thesis. The proof is divided in three cases: $s = 0,\ s \in (0, 1),\ s = 1$. For the latter two cases, the reader is referred to to Theorem 10.2 by Steinbach [20]. The case $s = 0$ follows rather quickly and will now be proven. Since $u \in L^2(Y)$, we have $P_y u \in S^0_h(Y)$. As a consequence of this fact and Definition 3.1 we have

$$\langle u - P_y u, v - P_y v\rangle_{L^2(Y)} = \langle u - P_y u, v\rangle_{L^2(Y)} - \langle u - P_y u, P_y v\rangle_{L^2(Y)} = \langle u - P_y u, v\rangle_{L^2(Y)}. \tag{3.21}$$

Hence,

$$\|u - P_y u\|^2_{L^2(Y)} = \langle u - P_y u, u\rangle_{L_2(Y)} \le \|u - P_y y\|_{L_2(Y)} \|u\|_{L_2(Y)}.$$

Which gives the desired estimate for $s = 0$.

□

As a consequence of Theorem 3.10, we obtain Corollary 3.11 [20].

Corollary 3.11 ([20, Cor.10.3])**.** *Let Y and P_y be defined as in Theorem 3.10. Let $u \in H^s(Y)$ be given for some $s \in [0, 1]$. For $\sigma \in [-1, 0)$, the following error estimates hold*

$$\|u - P_y u\|^2_{H^\sigma(Y)} \le c h^{-2\sigma} \sum_{k=1}^{N_y} h_k^2 |u|^2_{H^s(e^{(y)}_l)},$$

and

$$\|u - P_y u\|_{H^\sigma(Y)} \le c h^{s-\sigma} |u|_{H^s(Y)}.$$

Proof. For the sake of completeness a summary of the proof by Steinbach is given. Some trivial parts of the proof are left out while other steps are given more attention. Since $H^{\sigma}(Y)$ is the dual space of $H^{-\sigma}$ we obtain the dual norm

$$\|u - P_y u\|^2_{H^{\sigma}(Y)} = \sup_{0 \neq v \in H^{-\sigma}(Y)} \frac{\langle u - P_y u, v \rangle_{L^2(Y)}}{\|v\|_{H^{-\sigma}(Y)}}. \tag{3.22}$$

By combining the fact that $v \in H^{-\sigma}(Y) \subset L^2(Y)$, implies $P_y v \in S^0_h(Y)$, and (3.21) we obtain

$$\begin{aligned}\|u - P_y u\|^2_{H^{\sigma}(Y)} &= \sup_{0 \neq v \in H^{-\sigma}(Y)} \frac{\langle u - P_y u, v - P_y v \rangle_{L^2(Y)}}{\|v\|_{H^{-\sigma}(Y)}} \\ &\leq \|u - P_y u\|_{L^2(Y)} \sup_{0 \neq v \in H^{-\sigma}(Y)} \frac{\|v - P_y\|_{L^2(Y)}}{\|v\|_{H^{-\sigma}(Y)}} \quad \text{(Cauchy-Schwarz).}\end{aligned}$$

By using the error estimates (3.19) and (3.20) for $\|u - P_y y\|_{L^2(Y)}$ and $\|v - P_y v\|_{L^2(Y)}$ respectively, the assertion follows.

□

Theorem 3.10 and Corollary 3.11 lead to the final approximation property of the trial space(s) $\Sigma^0_{h_y}(Y)$ [20].

Theorem 3.12 ([20, Thm.10.4])**.** *Let $\sigma \in [-1, 0]$. For $u \in H^s(Y)$ with some $s \in [\sigma, 1]$ there holds the approximation property of $S^0_{h_y}(Y)$*

$$\inf_{v_h \in S^0_{h_y}(Y)} \|u - v_h\|_{H^{\sigma}(Y)} \leq c h^{s-\sigma} |u|_{H^s(Y)}. \tag{3.23}$$

Remark. *For $\sigma \in [-1, 0)$ and $s \in (0, 1]$, Theorem 3.12 is not a direct consequence of Theorem 3.10 and Corollary 3.11. The approximation property still holds and is proven in Steinbach Theorem 10.4 [20].*

Let us now consider the space-time tensor product space of piecewise constant basis functions $S^{0,0}_{h_x,h_t}(\Sigma)$ defined as

$$S^{0,0}_{h_x,h_t}(\Sigma) := S^0_{h_x}(\Gamma_{N_x}) \otimes S^0_{h_t}(I_{N_t}). \tag{3.24}$$

The goal is to derive the approximation properties for the L^2 projection with respect to the tensor product space $S^{0,0}_{h_x,h_t}(\Sigma)$, which will be denoted as $P_{x,t}$. To find an expression for this L^2 projection $P_{x,t}$, we make use of a well known proposition about projection operators [14].

Proposition 3.13 ([14, Prop.8.8])**.** *For a projection P on Hilbert space H, the following assertions are equivalent*

(i) *P is orthogonal,*
(ii) *P is self-adjoint.*

Now, we define the following projections for $u \in L^2(\Sigma)$

$$\begin{aligned}P^{\Sigma}_x u(x,t) &:= (P_x u(\cdot, t))(x), \\ P^{\Sigma}_t u(x,t) &:= (P_t u(\cdot, t))(t).\end{aligned}$$

Our aim is to obtain error estimates in the anisotropic Sobolev spaces. We will slightly deviate from the original method introduced by Dohr [3] and will make use of the self-adjointness obtained by Proposition 3.13, to show how to obtain these estimates. Consider

$$\begin{aligned}\langle P^{\Sigma}_x P^{\Sigma}_t u - P_{x,t} u, v \rangle_{L^2(\Sigma)} &= \langle P^{\Sigma}_x P^{\Sigma}_t u, v \rangle_{L^2(\Sigma)} + \langle P_{x,t} u, v \rangle_{L^2(\Sigma)} = \langle P^{\Sigma}_x u, P^{\Sigma}_t v \rangle_{L^2(\Sigma)} + \langle P_{x,t} u, v \rangle_{L^2(\Sigma)} \\ &= \langle u - P_{t,x}, v \rangle_{L^2(\Sigma)} = 0 \quad \forall v \in S^{0,0}_{h_x,h_t}(\Sigma).\end{aligned}$$

By taking $v = P_x^\Sigma P_t^\Sigma u - P_{x,t} u \in S^{0,0}_{h_x,h_t}(\Sigma)$ we obtain

$$\|P_x^\Sigma P_t^\Sigma u - P_{x,t} u\|_{L^2(\Sigma)} = 0.$$

Hence, we define $P_{x,t}$ as $P_x^\Sigma P_t^\Sigma$, and use the previously obtained approximation results. As a consequence for $r, s \in [0,1]$ and $u \in H^{r,s}(\Sigma) \subset L^2(\Sigma)$, we can apply Theorem 3.10 and the triangle inequality to arrive at

$$\begin{aligned}
\|u - P_{x,t}u\|_{L^2(\Sigma)} &= \|u - P_x^\Sigma P_t^\Sigma u\|_{L^2(\Sigma)}\|_{L^2(\Sigma)} = \|u - P_x^\Sigma u - P_x^\Sigma(u - P_t^\Sigma u)\|_{L^2(\Sigma)} \\
&\leq \|u - P_x^\Sigma u\|_{L^2(\Sigma)} + \|P_x^\Sigma(u - P_t^\sigma u)\|_{L^2(\Sigma)} \\
&\leq \|u - P_x^\Sigma u\|_{L^2(\Sigma)} + \|P_x^\Sigma\|_{L^2(\Sigma)}\|u - P_t^\sigma u\|_{L^2(\Sigma)} \\
&\leq \|u - P_x^\Sigma u\|_{L^2(\Sigma)} + \|u - P_t^\sigma u\|_{L^2(\Sigma)} \leq c(h_x^r + h_t^s)|u|_{H^{r,s}(\Sigma)}.
\end{aligned} \tag{3.25}$$

In this derivation the contraction property of the projections is used i.e. $\|P_x^\Sigma\| \leq 1$. Also, the Slobodeckii norm on the anisotropic Sobolev space is defined as

$$|\cdot|^2_{H^{r,s}(\Sigma)} := |\cdot|^2_{L^2(I;H^r(\Gamma))} + |\cdot|^2_{H^s(I;L^2(\Gamma))}.$$

In a similar fashion, Corollary 3.11 can be extended to $S^{0,0}_{h_t,h_x}(\Sigma_h)$. Analogous to the proof of Corollary 3.11, if we let $u \in H^{r,s}$ with $r, s \in [0,1]$, we obtain the following error estimate for $\sigma, \mu \in (-1, 0]$

$$\|u - P_{x,t}u\|_{H^{\sigma,\mu}(\Sigma)} \leq c(h_x^r + h_t^s)(h_x^{-\sigma} + h_t^{-\mu})|u|_{H^{r,s}(\Sigma)}. \tag{3.26}$$

The approximation properties for on $S^{0,0}_{h_t,h_x}(\Sigma_h)$ are henceforth derived and given by (3.25) and (3.26).

Space-time triangulation
Let $S_h^0(\Sigma_h)$ be the space of piecewise constant functions on the triangulation Σ_h such that

$$S_h^0(\Sigma_h) := \mathsf{span}\{\phi_k^{\Sigma,0}\}_{k=1}^N, \tag{3.27}$$

with

$$\phi_k^{\Sigma,0}(z,t) := \begin{cases} 1 & (z,t) \in e_k \\ 0 & \text{otherwise} \end{cases}. \tag{3.28}$$

Remark. *z can either be a value received after parameterization of the boundary Γ, or a vector $(x, y) \in \mathbb{R}^2$.*

We now consider the approximation properties of using space-time boundary elements on Σ_h. We denote the L^2 projection on $S_h^0(\Sigma_h)$ with P_0. To find estimates on this operator, we make use of the following lemma [3]:

Lemma 3.14 ([3, Lem.2.1])**.** *For $r, s \in [0,1]$, the continuous embedding*

$$H^{\mathsf{max}(r,s)}(\Sigma) \hookrightarrow H^{r,s}(\Sigma) \hookrightarrow H^{\mathsf{min}(r,s)},$$

holds.

With the use of Theorem 3.10 and Lemma 3.14, the following theorem arises [3]:

Theorem 3.15 ([3, Thm.6.14])**.** *Let $u \in H^{r,s}$ for some $r, s \in [0,1]$. Then the error estimates:*

$$\|u - P_0 u\|_{L^2(\Sigma)} \leq \|u\|_{L^2(\Sigma)},$$

$$\|u - P_0 u\|_{L^2(\Sigma)} \leq c h^{\mathsf{min}(r,s)} \|u\|_{H^{r,s}(\Sigma)},$$

hold.

*Proof:*The proof, which is mainly based on the proof Dohr has proposed for discretization using the tensor product ansatz, is added to this thesis. At the moment of writing this thesis, a proof of Theorem 3.15 was not available in the consulted literature. The first inequality can be proven following the same steps as in the proof of Theorem 3.10. For the second inequality, let $m := \min(r, s)$. Using Theorem 3.10 we obtain the inequality

$$\|u - P_0 u\|_{L^2(\Sigma)} \leq c h^m \|u\|_{H^m(\Sigma)}.$$

As a result of Lemma 3.14, $\|u\|_{H^m(\Sigma)} \leq \|u\|_{H^{r,s}(\Sigma)}$, proving the second assertion.

□

3.5. Computing the Galerkin Matrix

Now, that the approximation properties are properly defined, we can apply the apply the Galerkin-Bubnov formulation on (3.9). As we are mainly concerned in the structure of the matrix in the left hand-side, the initial condition and source is set to zero. Consider $S^0(\Sigma_h) \subset H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma_h)$. The Galerkin-Bubnov variational formula reads: Find $q_h \in S^0(\Sigma_h)$ such that

$$\langle V q_h, v_h \rangle_\Sigma = \langle g, v_h \rangle_\Sigma, \quad \forall v_h \in S_h^0(\Sigma_h). \tag{3.29}$$

Since $q_h \in S_h^0(\Sigma_h)$, we have

$$q_h := \sum_{k=1}^{N} q_k \phi_k^{\Sigma,0}(z,t).$$

Therefore (3.29) can be rewritten as

$$\sum_{k=1}^{N} q_k \langle V \phi_k^{\Sigma,0}, \phi_l^{\Sigma,0} \rangle_\Sigma = \langle g, \phi_l \rangle_\Sigma, \quad l = 1, \ldots, N. \tag{3.30}$$

This system of linear equations can be written into matrix form as

$$\begin{pmatrix} \langle V\phi_1^{\Sigma,0}, \phi_1^{\Sigma,0}\rangle_\Sigma & \ldots & \langle V\phi_N^{\Sigma,0}, \phi_1^{\Sigma,0}\rangle_\Sigma \\ \vdots & \ddots & \vdots \\ \langle V\phi_1^{\Sigma,0}, \phi_N^{\Sigma,0}\rangle_\Sigma & \ldots & \langle V\phi_N^{\Sigma,0}, \phi_N^{\Sigma,0}\rangle_\Sigma \end{pmatrix} \begin{pmatrix} q_1 \\ \vdots \\ q_N \end{pmatrix} = \begin{pmatrix} \langle g, \phi_1^{\Sigma,0}\rangle_\Sigma \\ \vdots \\ \langle g, \phi_N^{\Sigma,0}\rangle_\Sigma \end{pmatrix}. \tag{3.31}$$

For $i, j \in [1, N]$, the elements of the matrix are given by

$$\langle V\phi_i^{\Sigma,0}, \phi_j^{\Sigma,0}\rangle_\Sigma = \int_\Sigma \left(\int_\Sigma \phi_i^{\Sigma,0}(y,s) G(z-y, t-s)\, dyds \right) \phi_j^{\Sigma,0}(x,t)\, dzdt.$$

Which, by definition of the piecewise constant functions $\phi_i^{\Sigma,0}$, the inner product can be reduced to

$$\langle V\phi_i^{\Sigma,0}, \phi_j^{\Sigma,0}\rangle_\Sigma = \int_{e_j} \int_{e_i} G(z-y, t-s)\, dy\, ds\, dz\, dt. \tag{3.32}$$

The reference triangle will from now on be denoted as K_{ref}, and is defined as a triangle in $\mathbb{R}^2$ with vertices $(0,0)$, $(0,1)$, $(1,0)$.

Given an element $e \in \Sigma_h$, with vertices $(x_i, y_i, t_i) \in \mathbb{R}^3$, $i = \{1,2,3\}$, we consider the parameterization of the vertices to the two dimensional plane. The vertices after parameterization will be denoted as $(z_i, t_i) \in \mathbb{R}^2$, $z_i = \kappa^{-1}(x_i, y_i)$, with κ the parameterization defined in (3.14). Now, we can define the Jacobian matrix obtained from the transformation of the element to the reference triangle described by (3.15),

$$J := \begin{pmatrix} \frac{\partial z}{\partial \xi} & \frac{\partial z}{\partial \eta} \\ \frac{\partial t}{\partial \xi} & \frac{\partial t}{\partial \eta} \end{pmatrix} = \begin{pmatrix} z_2 - z_1 & z_3 - z_1 \\ t_2 - t_1 & t_3 - t_1 \end{pmatrix}. \tag{3.33}$$

With the use of (3.33), the integral (3.32) can be rewritten into

$$\int_{e_j(z,t)} \int_{e_i(y,s)} G(z-y,t-s)\, dy\, ds\, dz\, dt =$$
$$= |\det J_i||\det J_j| \int_{K_{ref}} \int_{K_{ref}} G(z(\xi_1)-z(\xi_2), t(\eta_1)-t(\eta_2))\, d\xi_2\, d\eta_2\, d\xi_1\, d\eta_1, \tag{3.34}$$

with

$$K_{ref} \times K_{ref} :=$$
$$\{[z(\xi_1), t(\eta_1), z(\xi_2), t(\eta_2)]^T \in \mathbb{R}^4 : z(\xi_1), t(\eta_1), z(\xi_2), t(\eta_2) > 0; z(\xi_1) + t(\eta_1) < 1; z(\xi_2) + t(\eta_2) < 1\}.$$

When the original elements e_i, e_j have coinciding vertices or edges, singularities might occur. Therefore, we take a closer look at the behaviour of the function $G(z-y, t-s)$ at its limits. We define $r := z-y$ and $\tau := t-s$. Note that $G(r,\tau) = 0$ if $\tau < 0$. Thus, for a non-zero fundamental solution, we have

$$G(z-y, t-s) = G(r,\tau) = \frac{1}{4\pi|\tau|e^{r^2/(4|\tau|)}} = \frac{C}{F(r,\tau)},$$

with

$$F(r,\tau) := |\tau|e^{r^2/(4|\tau|)}.$$

and $C := \frac{1}{4\pi}$. We can rewrite this with the definition of an exponential expansion as

$$F(r,\tau) = |\tau| \sum_{n=0}^{\infty} \frac{1}{n!} \left(\frac{r^2}{4\tau}\right)^n = |\tau| \left(1 + \frac{r^2}{8\tau} + \frac{r^4}{96\tau^2} + \mathcal{O}\left(\frac{r^6}{\tau^3}\right)\right). \tag{3.35}$$

From this expansion, the following limits become clearly visible

- if $r \neq 0$, $\lim_{\tau\to 0} F(r,\tau) = \infty$.
- if $\tau \neq 0$, $\lim_{r\to 0} F(r,\tau) = \tau$.
- $\lim_{(r,\tau)\to(0,0)} F(r,\tau) = 0$.

As $G(r,\tau)$ is some constant divided by $F(r,\tau)$, we have a singularity at the origin. In the case of a $\mathcal{O}(\frac{1}{\tau})$ problem, i.e. the Taylor expansion of $F(r,\tau)$ is of order $\frac{1}{\tau}$, this singularity can be suppressed with the use of gamma functions. At the moment of writing this thesis, this procedure was not reported in the literature, which is the reason why we proceed to derive the required formulas. As shown in (3.35), the two dimensional problem is of $\mathcal{O}(\frac{1}{\tau})$.

For notational simplicity, consider, the following integral on the unit triangle K_{ref}

$$\int_{K_{ref}} G(x,t)dtdx = \int_0^1 \int_0^{1-x} G(x,t)dtdx.$$

Note that the inner integral can be rewritten into

$$\int_0^{1-x} G(x,t)dt = \frac{1}{4\pi} \int_{\frac{x^2}{4(1-x)}}^{\infty} \sigma^{-1}e^{-\sigma}d\sigma = \mathbb{G}\left(0, \frac{x^2}{4(1-x)}\right), \tag{3.36}$$

where $\mathbb{G}$ denotes the incomplete Gamma function, defined in Chapter 2. We are left with

$$\int_0^1 \mathbb{G}\left(0, \frac{x^2}{4(1-x)}\right) dx,$$

which can be approximated with the use of quadrature rules. In the case of the integration over the tensor product of the unit triangles, we make use of the following identity

$$\int \mathbb{G}\left(0, \frac{a}{4(b+x)}\right)\, dx = \left(\frac{a}{4} + b + x\right) \mathbb{G}\left(0, \frac{a}{4(b+x)}\right) - (b+x)e^{-\frac{a}{4(b+x)}} + C. \tag{3.37}$$

We can write the tensor product integral as

$$\int_{K_{ref}}\int_{K_{ref}} G(x-y,t-s)\,dydsdxdt = \int_0^1\int_0^1\int_0^{1-x}\int_0^{1-y} G(x-y,t-s)\,dsdtdydx. \tag{3.38}$$

Next, the integral (3.38) is evaluated step by step with the use of gamma functions, starting with the inner-most integral

$$\int_0^{1-y} G(x-y,t-s)\,ds = \frac{1}{4\pi}\int_0^{1-y}\frac{1}{t-s}e^{-\frac{(x-y)^2}{4(t-s)}}\,ds. \tag{3.39}$$

By letting $\sigma := \frac{(x-y)^2}{4(t-s)}$ we have

$$\frac{\partial\sigma}{\partial s} = \frac{(x-y)^2}{4(s-t)^2} = \frac{\sigma}{(t-s)}.$$

So by substitution of σ in (3.39), we obtain

$$\int_0^{1-y} G(x-y,t-s)\,ds = \frac{1}{4\pi}\int_{\alpha(x,y,t)}^{\beta(x,y,t)}\frac{1}{\sigma}e^{-\sigma}\,d\sigma = \frac{1}{4\pi}\left(\mathbb{G}(0,\alpha(x,y,t)) - \mathbb{G}(0,\beta(x,y,t))\right), \tag{3.40}$$

with

$$\alpha(x,y,t) := \frac{(x-y)^2}{4t}, \quad \beta(x,y,t) := \frac{(x-y)^2}{4(t+y-1)}.$$

The incomplete Gamma function $\mathbb{G}(0,x)$ is well defined for all $x\in\mathbb{R}^+$, with $\mathbb{G}(0,x)=0$ as $x\to\infty$ and $\mathbb{R}^+ := \{x\in\mathbb{R}, x>0\}$. When taking a closer look at the anti-derivative of the incomplete Gamma function (3.37), terms of the form $x\mathbb{G}(0,\frac{1}{x})$ arise. Terms of this form have the distinctive property that their right hand-side limit is well defined, more specifically

$$\lim_{x\to 0^+} x\mathbb{G}\left(0,\frac{1}{x}\right) = 0.$$

Hence, (3.40) is integrable with respect to t, for any domain in $\mathbb{R}^+$. Using (3.37) and the fact that (by definition of a unit triangle) $x-1\geq 0$, the result of (3.40) can be integrated and evaluated as well. We obtain

$$\begin{aligned}
&\frac{1}{4\pi}\int_0^{1-x}\mathbb{G}(0,\alpha(x,y,t)) - \mathbb{G}(0,\beta(x,y,t))\,dt = \\
&\frac{1}{4\pi}\left[\left(\frac{(x-y)^2}{4}+1-x\right)\mathbb{G}\left(0,\frac{(x-y)^2}{4(1-x)}\right) - (1-x)e^{-\frac{(x-y)^2}{4(1-x)}} - \int_0^{1-x}\mathbb{G}\left(0,\frac{(x-y)^2}{4(t+y-1)}\right)\,dt\right] = \\
&\frac{1}{4\pi}\left[\left(\frac{(x-y)^2}{4}+1-x\right)\mathbb{G}\left(0,\frac{(x-y)^2}{4(1-x)}\right) - (1-x)e^{-\frac{(x-y)^2}{4(1-x)}} + (y-x)e^{\frac{-(x-y)^2}{4(y-x)}} + \right. \\
&\left.\left(\frac{(x-y)^2}{4}+y-1\right)\mathbb{G}\left(0,\frac{(x-y)^2}{4(y-1)}\right) - (y-1)e^{\frac{(x-y)^2}{4(y-1)}} - \left(\frac{(x-y)^2}{4}+y-x\right)\mathbb{G}\left(0,\frac{(x-y)^2}{4(y-x)}\right)\right].
\end{aligned} \tag{3.41}$$

The expression (3.41) and all prior calculations can be summarised by

$$4\pi\int_0^{1-x}\int_0^{1-y} G(x-y,t-s)\,dsdt = \mathcal{I}_1(x,y) + \mathcal{I}_2(x,y) + \mathcal{I}_3(x,y), \tag{3.42}$$

with

$$\begin{aligned}
\mathcal{I}_1 &:= \left(\frac{(x-y)^2}{4}+1-x\right)\mathbb{G}\left(0,\frac{(x-y)^2}{4(1-x)}\right) - (1-x)e^{-\frac{(x-y)^2}{4(1-x)}},\\
\mathcal{I}_2 &:= (y-x)e^{\frac{-(x-y)^2}{4(y-x)}} - \left(\frac{(x-y)^2}{4}+y-x\right)\mathbb{G}\left(0,\frac{(x-y)^2}{4(y-x)}\right),\\
\mathcal{I}_3 &:= \left(\frac{(x-y)^2}{4}+y-1\right)\mathbb{G}\left(0,\frac{(x-y)^2}{4(y-1)}\right) - (y-1)e^{\frac{(x-y)^2}{4(y-1)}}.
\end{aligned}$$

Remark. *Since the fundamental solution is zero for $s > t$, the expression* (3.42) *holds only on the domain where $t > s$. Indirectly, the spatial variables x and y in* (3.42) *are dependent on t and s.*

As stated in the remark, the result (3.42) will not yield a complete description of an arbitrary triangulation of the space-time lateral trace, as we did not yet take in account the behaviour of the fundamental solution. To see why this matters, we will take a closer look at the integration bounds in (3.42). For the two given reference triangles, for example the point $x = 0.8$ and $y = 0.1$ is within the integration domain of (3.38). However, when looking at (3.42), or more specifically $\mathcal{I}_2$, this would yield a negative entry in the gamma function, which in turn will result in a complex value. Since the described integrals should not result in complex values, we need to add constraints to (3.42).

In order to obtain the complete description of (3.38), we reconsider the first two inner integrals:

$$\begin{aligned}\int_0^{1-x}\int_0^{1-y} G(x-y,t-s)dsdt &= \int_0^{1-x}\int_0^{1-y} G(x-y,t-s)\mathbb{1}_{\{0\le s<t<\infty\}}dsdt \\ &= \int_0^{1-y}\int_s^{1-x} G(x-y,t-s)\mathbb{1}_{\{0\le s<t<\infty\}}\ dtds.\end{aligned}$$

Similar to (3.39), with $\sigma = \frac{(x-y)^2}{4(t-s)}$ we obtain

$$\int_0^{1-y}\int_s^{1-x} G(x-y,t-s)\ dtds = \frac{1}{4\pi}\int_0^{1-y}\int_{\frac{(x-y)^2}{4(1-x-s)}}^{\infty}\frac{1}{\sigma}e^{-\sigma}\ d\sigma ds = \frac{1}{4\pi}\int_0^{1-y}\mathbb{G}\left(0,\frac{(x-y)^2}{4(1-x-s)}\right)\ ds. \tag{3.43}$$

Where we used the fact that $t > 0$ for a non zero integral so we take the limit from the right when calculating the bounds, i.e.

$$\lim_{t\to s^+}\frac{(x-y)^2}{4(t-s)} = \infty, \forall x \neq y.$$

If $x = y$ both the upper and lower bound of the integral will become zero, and therefore this occurence can be neglected. Using (3.37), we get

$$\begin{aligned}&\int_0^{1-y}\int_s^{1-x} G(x-y,t-s)\ dtds = \\ &\frac{1}{4\pi}\mathbb{1}_{\{0\le s<1-x<\infty\}}\left(s+x-1-\frac{(x-y)^2}{4}\right)\mathbb{G}\left(0,\frac{(x-y)^2}{4(1-x-s)}\right) - (s+x-1)e^{-\frac{(x-y)^2}{4(1-x-s}}\Bigg|_0^{1-y}.\end{aligned}$$

Which yields the complete result

$$4\pi\int_0^{1-x}\int_0^{1-y} G(x-y,t-s)dsdt = \mathcal{I}_1^g(x,y) + \mathcal{I}_2^g(x,y), \tag{3.44}$$

with

$$\mathcal{I}_1^g(x,y) := \mathbb{1}_{\{y>x\}}\left[\left(x-y-\frac{(x-y)^2}{4}\right)\mathbb{G}\left(0,\frac{(x-y)^2}{4(y-x)}\right) - (x-y)e^{-\frac{(x-y)^2}{4(y-x)}}\right]$$

and

$$\mathcal{I}_2^g(x,y) := -\left(x-1-\frac{(x-y)^2}{4}\right)\mathbb{G}\left(0,\frac{(x-y)^2}{4(1-x)}\right) + (x-1)e^{-\frac{(x-y)^2}{4(1-x)}}.$$

With the use of quadrature rules on (3.44), we are able to approximate the final integral

$$\frac{1}{4\pi}\int_0^1\int_0^1 \mathcal{I}_1^g + \mathcal{I}_2^g\ dxdy. \tag{3.45}$$

When taking a closer look at $\mathcal{I}_1^g$, ignoring the indicator function, it becomes clear by definition of the incomplete gamma functions, that

$$\mathfrak{Im}(\mathcal{I}_1^g) \neq 0 \iff x > y.$$

The relation $x > y$ would imply, however, that one of the lower bounds in (3.43) is strictly negative. This, in turn, implies that there exists a domain in which $\sigma < 0$, i.e. $\frac{(x-y)^2}{4(t-s)} < 0$. This can only occur when $s > t$, which corresponds to a zero fundamental solution. $\mathcal{I}_2^g$ will not yield imaginary parts, as $0 < x < 1$. Hence, the final solution will look like

$$\int_0^1 \int_0^{1-x} \int_0^1 \int_0^{1-y} G(x-y, t-s)\, dsdydtdx = \frac{1}{4\pi} \int_0^1 \int_y^1 \mathcal{I}_1^g\, dxdy + \frac{1}{4\pi} \int_0^1 \int_0^1 \mathcal{I}_2^g\, dxdy.$$

To complete the calculations, the required mappings from a parameterized domain in $\mathbb{R}^2$, to the unit triangles need to be included. Let us consider two arbitrary triangles e_1 and e_2 with vertices $\mathbf{x}_i = (x_i, t_i)$ and $\mathbf{y}_i = (y_i, s_i)$, i=1,2,3, respectively. Let us denote the linear transformation to the reference triangle by Φ_1 for e_1 and Φ_2 for e_2, i.e.:

$$\Phi_i : e_i \to K, \quad \text{for } i = 1, 2.$$

We define the pullback operator as the inverse of this linear transformation

$$\Phi_i^* := \Phi_i^{-1} : K \to e_i, \quad \text{for } i = 1, 2.$$

The Jacobian matrices J_i are defined in the same fashion as (3.33) for both elements e_i and their respective determinant by Δ_i. The operator Φ_i^* maps to two variables, we will make a distinction between the mapping to spatial coordinates

$$\Phi_i^{(1*)} : (\xi, \eta) \mapsto X_i, \quad X_i := \{x, y\},$$

and the temporal coordinates

$$\Phi_i^{(2*)} : (\xi, \eta) \mapsto T_i, \quad T_i := \{t, s\}.$$

To further simplify notation, we write

$$\begin{pmatrix} \hat{x} \\ \hat{t} \\ \hat{y} \\ \hat{s} \end{pmatrix} := \begin{pmatrix} \Phi_1^{(1*)}(\xi_1, \eta_1) \\ \Phi_1^{(2*)}(\xi_1, \eta_1) \\ \Phi_2^{(1*)}(\xi_2, \eta_2) \\ \Phi_2^{(2*)}(\xi_2, \eta_2) \end{pmatrix}.$$

The transformed integrals to calculate the stiffness matrix are then given by

$$|\Delta_i||\Delta_j| \int_0^1 \int_0^1 \int_0^{1-\xi_1} \int_0^{1-\xi_2} G(\hat{x} - \hat{y}, \hat{t} - \hat{s})\, d\eta_2 d\eta_1 d\xi_2 d\xi_1. \tag{3.46}$$

Again the focus lies on the two inner integrals. We write

$$\int_0^{1-\xi_1} \int_0^{1-\xi_2} G(\hat{x} - \hat{y}, \hat{t} - \hat{s})\, d\eta_2 d\eta_1 = \int_0^{1-\xi_2} \int_{\hat{s}}^{1-\xi_1} G(\hat{x} - \hat{y}, \hat{t} - \hat{s})\, d\eta_1 d\eta_2.$$

Interestingly, both arguments in the fundamental solution may be depending on the same variable(s). It might be possible that the inner integrals are not possible to evaluate analytically, and this is indeed the case as will be shown now. First, we define a special function, *the exponential integral* :

Definition 3.2 ([16, Defn.4.4.2])**.** *The generalised exponential integral is given by*

$$E_n(x) := \int_{-\infty}^{x} \frac{e^t}{t^n}\, dt.$$

In the case $n = 1$*, for positive real x we have*

$$-E_1(-x) = \mathbb{G}(0, x),$$

where $\mathbb{G}$ *denotes the incomplete gamma function.*

An important property of the exponential integral is the following.

Proposition 3.16 ([16, Lem.7.1.5],[6]). *Let $E_n(x)$ be the generalised exponential integral defined in Definition 3.2. The derivative of $E_n(x)$ with respect to x is given by*

$$E_n'(x) = -E_{n-1}(x).$$

This is an useful property due to the fact $E_0(x) := \frac{e^{-x}}{x}$. Now, let us define $\sigma(\xi_1, \eta_1, \xi_2, \eta_2) := \frac{(\hat{x}-\hat{y})^2}{4(\hat{t}-\hat{s})}$. We will take a look at the inner most integral

$$\int_0^{1-\xi_2} G(\hat{x}-\hat{y}, \hat{t}-\hat{s})\, d\eta_2 = \frac{1}{4\pi}\int_0^{1-\xi_2} \frac{1}{\hat{t}-\hat{s}} e^{-\sigma} d\eta_2 = \frac{1}{4\pi}\int_0^{1-\xi_2} \frac{1}{\hat{t}-\hat{s}} e^{-\sigma} d\eta_2 = \frac{1}{\pi}\int_0^{1-\xi_2} \frac{\sigma^2}{(\hat{x}-\hat{y})^2} E_0(\sigma) d\eta_2. \tag{3.47}$$

Next (3.47) is rewritten as

$$\frac{1}{4\pi}\int_0^{1-\xi_2} \frac{\sigma^2}{(\hat{x}-\hat{y})^2} E_0(\sigma) d\eta_2 = \frac{1}{\pi}\int_0^{1-\xi_2} \frac{\sigma^2}{(\hat{x}-\hat{y})^2} \frac{1}{\frac{\partial\sigma}{\partial v}} \frac{\partial}{\partial v}(-E_1(\sigma)) d\eta_2 = A(\eta_2, \cdot)\frac{\partial}{\partial v}(-E_1(\sigma)) d\eta_2,$$

with

$$A(\eta_2, \cdot) := \frac{\sigma^2}{(\hat{x}-\hat{y})^2} \frac{1}{\frac{\partial\sigma}{\partial v}}.$$

Using integration by parts, (3.47) thus equals

$$\int_0^{1-\xi_2} G(\hat{x}-\hat{y}, \hat{t}-\hat{s})\, d\eta_2 = -A(\eta_2, \cdot) E_1(\sigma(\eta_2, \cdot))\Big|_0^{1-\xi_2} + \int_0^{1-\xi_2} A(\eta_2, \cdot) E_1(\sigma)\, d\eta_2. \tag{3.48}$$

At the moment of writing this thesis, no analytical form of the integral on the right-hand side of (3.48), has been found in the literature (see for example the list of integrals in [6]). At this moment, the problem can thus not be solved (semi-) analytically given an arbitrary triangulation.

3.5.1. Right triangular grid

When looking at a more specific grid, consisting of right-angled triangles with one side parallel to the spatial direction and one side parallel to the temporal direction, a semi-analytical solution exists for each element. To be clear, this form triangulation does not require all the elements to be of the same size or same shape. Let e_i and e_j, be elements of such a triangulation.

Same elements

We proceed to derive the matrix entries from a space-time triangulation of Σ. First we will look at the case when $e_i = e_j$, i.e., $i = j$. We denote the length of the leg of the element in spatial direction by X. Similarly, in time the length of the leg is denoted by T. Mirroring the element along the hypotenuse will yield the same results when integrating, because the distances between the (same) triangles is preserved. We can therefore always find a transformation in time to the reference triangle of the form

$$\hat{t}(t) := a + bt, \quad b \in \mathbb{R}^+. \tag{3.49}$$

The positivity of b is of importance, as the fundamental solution is in general not symmetric in time. However due to the positivity of b, the integral (3.32) with same elements will be symmetric in time. Since the fundamental solution is symmetric in spatial direction, we can (if needed) mirror the triangle along the temporal axis to obtain an integral of the form

$$\int_0^X \int_0^X \int_0^{T-\frac{T}{X}y} \int_s^{T-\frac{T}{X}x} G(x-y, t-s)\, dtdsdydx. \tag{3.50}$$

Using previous calculations, (3.50) can be rewritten as:

$$\frac{1}{4\pi}\left(\int_0^X \int_x^X \hat{I}\left(x, y, T-\frac{T}{X}y\right)\, dydx - \int_0^X \int_0^X \hat{I}\,(x, y, 0)\; dydx\right), \tag{3.51}$$

with

$$\hat{I}(x,y,s) := \frac{\left(-x^2-y^2+2xy-4T+4s+4\frac{T}{X}x\right)\mathbb{G}\left(0,\frac{(x-y)^2}{4(T-\frac{T}{X}x-s)}\right)}{4} + e^{-\frac{(x-y)^2}{4(T-\frac{T}{X}x-4s)}}(T-\frac{T}{X}-s). \quad (3.52)$$

This expression can be approximated using quadrature rules, like Gauss-Legendre.

Spatially abutting elements
There are several cases in which elements can share a common edge. For each of these cases, the analytically solved part of the solution differs. As stated before, the fundamental solution symmetric in space, and therefore each case holds as well for the mirrored case in space.

The visual representation of each case is given in the Figures 3.1-3.4.

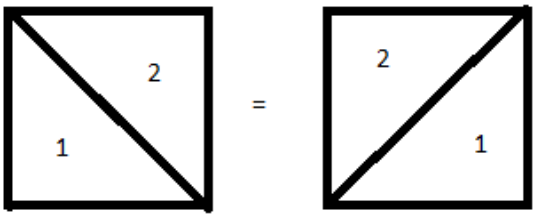


Figure 3.1: Type 1 abutting elements

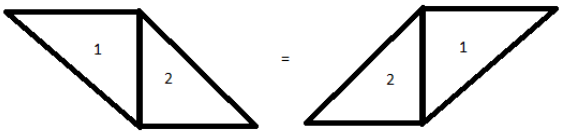


Figure 3.2: Type 2-a abutting elements, forming a parallelogram

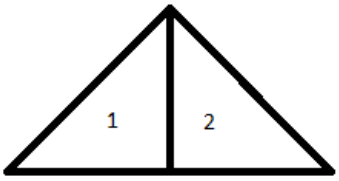


Figure 3.3: Type 2-b abutting elements, forming a triangle

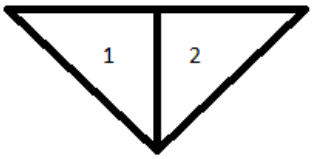


Figure 3.4: Type 2-c abutting elements, forming a triangle

Type 1:
Elements sharing a common hypotenuse will be regarded as the first type of abutting elements. Let e_1 and e_2 represent the elements denoted by "1" and "2" in Figure 3.1 respectively. Integrating over the tensor product space $e_1 \times e_2$ will in general not yield the same result as integrating over $e_2 \times e_1$. First, we will look at the $e_1 \times e_2$. Again, X denotes the *length* of the leg in spatial direction and T the length in temporal direction. As we are dealing with distances between elements, we can perform a volume preserving mapping to the origin, which will give the same result. This fact will be used throughout the rest of the derivations. The integral (3.32) can be written as

$$\int_0^X \int_0^X \int_{T-\frac{T}{X}y}^{T} \int_s^{T-\frac{T}{X}x} G(x-y,t-s)\, dtdsdydx. \tag{3.53}$$

This expression (3.53) can be rewritten, using the same substitution as in (3.36) into

$$\frac{1}{4\pi} \int_0^X \int_0^X \int_{T-\frac{T}{X}y}^{T} \mathbb{G}\left(0, \frac{(x-y)^2}{4(T-\frac{T}{X}x-s)}\right) dsdydx. \tag{3.54}$$

The indefinite form of the inner-most integral can be determined with the use of (3.37). By definition of the fundamental solution, $s < T - \frac{T}{X}x$. Also, with the integral tells us that $T - \frac{T}{X}y < s < T$. As $T - \frac{T}{X}x \leq T$, the inner-most integral in (3.54) will vanish when s approaches T. From the same inequalities it also follows that $y > x$. Combining these facts, we arrive at the final solution for this case:

$$-\frac{1}{4\pi} \int_0^X \int_x^X \hat{I}\left(x, y, T-\frac{T}{X}y\right) dydx. \tag{3.55}$$

Which again can be well approximated with the use of quadrature.

Let Q denote the area composed of e_1 and e_2. For the reverse case, $e_2 \times e_1$, we will use the fact that $e_2 \times e_1 = Q \times Q - e_1 \times e_1 - e_2 \times e_2 - e_1 \times e_2$. The integrals on tensor product spaces between elements are already derived in this section. All that is left is $Q \times Q$. The integral on this domain has already been derived in Section 2.1.2, but, for the sake of completeness will again be given:

$$\int_0^X \int_0^X \int_0^T \int_s^T G(x-y,t-s)\, dtdsdydx = \int_0^X \int_0^X \hat{I}_Q(x,y,0)\, dydx, \tag{3.56}$$

with

$$\hat{I}_Q(x,y,s) := \frac{(-x^2-y^2+2xy-4T+4s)\mathbb{G}\left(0, \frac{(x-y)^2}{4(T-s)}\right)}{4} + e^{-\frac{(x-y)^2}{4(T-s)}}(T-s).$$

Type 2-a:
Another type of abutting elements which can occur, is the type in which the elements share a common leg perpendicular to the spatial direction. Within the collection of elements of this type, there are two possible scenarios: elements which form together a parallelogram or elements which will result in another triangle when combined. The first case is given in Figure 3.2

We denote again e_1 and e_2 in a similar fashion as before. This time, we start with the case $e_2 \times e_1$. Let X_i denote the leg in spatial direction for e_i and T the shared leg length in temporal direction. The integral (3.32) is then given by:

$$\int_{X_1}^{X_1+X_2} \int_0^{X_1} \int_{T-\frac{T}{X_1}y}^{T} \int_s^{T\left(1+\frac{X_1}{X_2}\right)-\frac{T}{X_2}x} G(x-y,t-s)\, dtdsdydx. \tag{3.57}$$

This gives

$$\int_{X_1}^{X_1+X_2} \int_0^{X_1} \int_{T-\frac{T}{X_1}y}^{T} \mathbb{G}\left(0, \frac{(x-y)^2}{4(T\left(1+\frac{X_1}{X_2}\right)-\frac{T}{X_2}x-s)}\right) dsdydx. \tag{3.58}$$

This time, we have that $T - \frac{T}{X_1}y < s < T$ and $s < T(1 + \frac{X_1}{X_2}) - \frac{T}{X_2}x$. Given $X_1 < x < X_2$, this implies again that the inner integral vanishes as s goes to T. Also, the inequalities result in the requirement that $y > \frac{X_1}{X_2}x - \frac{X_1^2}{X_2}$. The final solution becomes

$$-\int_{X_1}^{X_1+X_2}\int_{\frac{X_1}{X_2}x-\frac{X_1^2}{X_2}}^{X_1} \hat{I}_2\left(x, y, T - \frac{T}{X1}y\right)\, dydx, \tag{3.59}$$

in which $\hat{I}_2(x, y, s)$ is a generalised version of $\hat{I}(x, y, s)$, given by

$$\hat{I}_2(x, y, s) := \int \mathbb{G}\left(0, \frac{(x-y)^2}{4(T\left(1 + \frac{X_1}{X_2}\right) - \frac{T}{X_2}x - s)}\right)\, ds.$$

For notational convenience, the expression $\hat{I}_2(x, y, s)$ is not written out in its entirety.

For the case $e_1 \times e_2$ is a bit more complicated to find an analytical expression of the solution. In this case (3.32) becomes:

$$\int_0^{X_1}\int_{X_1}^{X_1+X_2}\int_0^{T\left(1+\frac{X_1}{X_2}\right)-\frac{T}{X_2}y}\int_{T-\frac{T}{X_1}x}^{T} G(x-y, t-s)\, dtdsdydx. \tag{3.60}$$

The difficulty for solving this integral lies within the lower bound of the inner-most integral. In the previous cases, we were able to change this bound to s since $s \geq 0$. In this case, however, $T - \frac{T}{X_1}x < s$ does not hold for all x. Instead, the inner-most integral can be expressed as

$$\int_{T-\frac{T}{X_1}x}^{T} G(x-y, t-s)\, dt := \int_{\mathsf{max}(T-\frac{T}{X_1}x, s)}^{T} G(x-y, t-s)\, dt.$$

When using substitution similar to (3.36), the lower bound will not necessarily become infinitely large. Instead, we obtain the expression

$$\int_0^{X_1}\int_{X_1}^{X_1+X_2}\int_0^{T\left(1+\frac{X_1}{X_2}\right)-\frac{T}{X_2}y} \mathbb{G}\left(0, \frac{(x-y)^2}{4(T-s)}\right) - \mathbb{G}\left(0, \frac{(x-y)^2}{4(\mathsf{max}(T-\frac{T}{X_1}x, s)-s)}\right)\, dsdydx.$$

The solution is divided up into two parts. First consider the inner integral

$$\int_0^{T\left(1+\frac{X_1}{X_2}\right)-\frac{T}{X_2}y} \mathbb{G}\left(0, \frac{(x-y)^2}{4(T-s)}\right)\, ds.$$

In this case, there will be no extra constraints on x or y, so the solution will become

$$\int_0^{X_1}\int_{X_1}^{X_1+X_2} \hat{I}_Q\left(x, y, T(1 + \frac{X_1}{X_2}) - \frac{T}{X_2}y\right) - I_Q\left(x, y, 0\right)\, dydx. \tag{3.61}$$

The other part is a bit more tricky, consider

$$\int_0^{T\left(1+\frac{X_1}{X_2}\right)-\frac{T}{X_2}y} \mathbb{G}\left(0, \frac{(x-y)^2}{4(\mathsf{max}(T-\frac{T}{X_1}x, s)-s)}\right)\, ds. \tag{3.62}$$

The solution to this integral will only exist when $\mathsf{max}(T-\frac{T}{X1}x, s) > s$. Which is the case if $T-\frac{T}{X1}x > s$. This in turn implies $T - \frac{T}{X_1}x > T(1 + \frac{X_1}{X_2}) - \frac{T}{X_1}y \Rightarrow y > \frac{X_2}{X_1}x + X_1$. By setting X to X_1, we obtain the final solution

$$\int_0^{X_1}\int_{X_1+\frac{X_2}{X_1}x}^{X_1+X_2} \hat{I}\left(x, y, T(1 + \frac{X_1}{X_2}) - \frac{T}{X_2}y\right)\, dydx - \int_0^{X_1}\int_{X_1}^{X_1+X_2} \hat{I}_Q(x, y, 0)\, dydx. \tag{3.63}$$

Substracting (3.63) from (3.62) will yield the true final solution.

Type 2-b:
The last form of (horizontally) abutting panels can once again be divided into two possibilities. The first possible scenario is given in Figure 3.3.

The integral, over the domain $e_2 \times e_1$ reads

$$\int_{X_1}^{X_1+X_2} \int_0^{X_1} \int_0^{\frac{T}{X_1}y} \int_s^{T(1+\frac{X_1}{X_2}-\frac{x}{X_2})} G(x-y,t-s)\, dtdsdydx.$$

Using similar steps as the previous cases, one can quickly observe that the final solution consist of $\hat{I}_2\left(x,y,\frac{T}{X_2}y\right) - \hat{I}_2(x,y,0)$. In the case of $\hat{I}_2\left(x,y,\frac{T}{X_2}y\right)$, the following inequality should hold: $\frac{T}{X_1}y < T(1+\frac{X_1}{X_2}-\frac{x}{X_2}) \Rightarrow y < X_1(1+\frac{X_1}{X_2}-\frac{x}{X_2})$. This yields the final solution

$$\int_{X_1}^{X_1+X_2} \int_0^{X_1(1+\frac{X_1}{X_2}-\frac{x}{X_2})} \hat{I}_2(x,y,\frac{T}{X_1}y)\, dydx - \int_{X_1}^{X_1+X_2} \int_0^{X_1} \hat{I}_2(x,y,0)\, dydx. \tag{3.64}$$

Type 2-c:
The other possible scenario given a type 2-c abutting panel is represented in Figure 3.4.

The integral corresponding to abutting elements of this type, given $e_2 \times e_1$, can be represented in integral form as:

$$\int_{X_1}^{X_1+X_2} \int_0^{X_1} \int_{T-\frac{T}{X_1}y}^{T} \int_{\frac{T}{X_2}x-\frac{X_1T}{X_2}}^{T} G(x-y,t-s)\, dtdsdydx.$$

This gives

$$\int_{X_1}^{X_1+X_2} \int_0^{X_1} \int_{T-\frac{T}{X_1}y}^{T} \mathbb{G}\left(0,\frac{(x-y)^2}{4(T-s)}\right) - \mathbb{G}\left(0,\frac{(x-y)^2}{4(\mathsf{max}(\frac{T}{X_2}x-\frac{X_1T}{X_2},s)-s)}\right)\, dsdydx.$$

Both gamma functions will vanish at the upper bound T. For the gamma function $\mathbb{G}\left(0,\frac{(x-y)^2}{4(\mathsf{max}(\frac{T}{X_2}x-\frac{X_1T}{X_2},s)-s)}\right)$, the following inequality must hold: $T-\frac{T}{X_1}y < \frac{T}{X_2}x - \frac{X_1T}{X_2} \Rightarrow y > X_1 + \frac{X_1^2}{X_2} - \frac{X_1}{X_2}x$. This gives the final solution for this type of abutting elements:

$$\int_{X_1}^{X_1+X_2} \int_{X_1+\frac{X_1^2}{X_2}-\frac{X_1}{X_2}x}^{X_1} \hat{I}_3\left(x,y,T-\frac{T}{X_1}y\right)\, dydx - \int_{X_1}^{X_1+X_2} \int_0^{X_1} \hat{I}_Q\left(x,y,T-\frac{T}{X_1}y\right)\, dydx, \tag{3.65}$$

with

$$\hat{I}_3(x,y,s) := \int \mathbb{G}\left(0,\frac{(x-y)^2}{4(\frac{T}{X_2}x-\frac{X_1T}{X_2}-s)}\right).$$

Temporally abutting elements
When dealing with temporally abutting elements, basic quadrature rules will result in faster convergence than the horizontal case. This is a consequence of the fact that the abutting leg in this case needs to be approached from a single side only (from top to bottom). Which is a continuous process. It is still worth the effort to come up with semi analytical expressions of the solutions since it reduces the approximation from a fourth order quadrature to a second order quadrature. We make again a distinction between two types of elements which can occur. First, consider the case represented by Figure 3.5

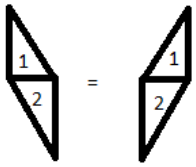


Figure 3.5: Type 3-a temporally abutting elements, forming a quadrilateral

Only $e_1 \times e_2$ type of domains are relevant, because otherwise the solution will be zero. Now the legs parallel to time are denoted as T_1 and T_2, we also have $X_1 = X_2$ and we obtain the integral

$$\int_0^{X_1}\int_0^{X_1}\int_{T_2-\frac{T_2}{X_1}y}^{T_2}\int_{T_2}^{T_1+T_2-\frac{T_1}{X_1}x} G(x-y,t-s)\,dtdsdydx.$$

We need not to worry about $s > t$, which becomes even more clear from the integration bounds. We thus obtain

$$\int_0^{X_1}\int_0^{X_1}\int_{T_2-\frac{T_2}{X_1}y}^{T_2} \mathbb{G}\left(0, \frac{(x-y)^2}{4(T_1+T_2-\frac{T_1}{X_1}x-s)}\right) - \mathbb{G}\left(0, \frac{(x-y)^2}{4(T_2-s)}\right)\,dsdydx.$$

The only possible restriction follows from $T_2 - \frac{T_2}{X_1}y < T_1 + T_2 - \frac{T_1}{X_1}x \Rightarrow y > \frac{T_1}{T_2}x - \frac{T_1X_1}{T_2}$, which is only present in the lower bound of $\mathbb{G}\left(0, \frac{(x-y)^2}{4(T_1+T_2-\frac{T_1}{X_1}x-s)}\right)$ and always true. The solution thus becomes:

$$\int_0^{X_1}\int_0^{X_1} \hat{I}_{h_1}(x,y,T_2) - \hat{I}_{h_1}\left(x,y,T_2-\frac{T_2}{X_1}y\right) + \hat{I}_{h_2}\left(x,y,T_2-\frac{T_2}{X_1}y\right)\,dydx,$$

with

$$\hat{I}_{h_1}(x,y,s) := \int \mathbb{G}\left(0, \frac{(x-y)^2}{4(T_1+T_2-\frac{T_1}{X_1}x-s)}\right)\,ds,$$

and

$$\hat{I}_{h_2}(x,y,s) := \int \mathbb{G}\left(0, \frac{(x-y)^2}{4(T_2-s)}\right)\,ds.$$

The final form of abutting elements in vertical direction is represented in Figure 3.6

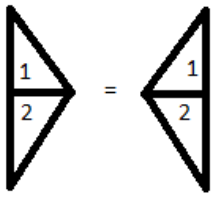


Figure 3.6: Type 3-b temporally abutting elements, forming a triangle

The corresponding integral is in this case

$$\int_0^{X_1}\int_0^{X_1}\int_{\frac{T_2}{X_1}y}^{T_2}\int_{T_2}^{T_1+T_2-\frac{T_1}{X_1}x} G(x-y,t-s)\,dtdsdydx.$$

One can observe the similarities to a type 3-a problem, and use the fact that $y < \frac{T_1X_1}{T_2} + X_1 - \frac{T_1}{T_2}x$ holds for all $0 < x, y < X_1$, to quickly obtain the result:

$$\int_0^{X_1}\int_0^{X_1} \hat{I}_{h_1}(x,y,T_2) - \hat{I}_{h_1}\left(x,y,\frac{T_2}{X_1}y\right) + \hat{I}_{h_2}\left(x,y,\frac{T_2}{X_1}y\right)\,dydx.$$

Non co-planar elements generalisation

So far we have analyzed situation in which elements are co-planar. We now would like to turn our attention to the cases where elements are abutting and have different unit normal vectors. When dealing with a two dimensional spatial domain, a slightly modified fundamental solution, compared to (2.6), is given by

$$G(\mathbf{x},t) := \frac{1}{4\pi t}e^{-\frac{\|\mathbf{x}\|^2}{4t}}.$$

where $\|\cdot\|$ denotes the Euclidean norm. As stated in (3.11), the space-time lateral boundary Σ, which is described *with coordinates* in $\mathbb{R}^3$, can be written as the union of closed subsets Σ_j. As each of these subsets is considered piecewise smooth, there exists a (continuous) bijective mapping of Σ_j in $\mathbb{R}^3$ to $\mathbb{R}^2$. This suggests there exists an invertible mapping from every element in the space time lateral boundary to a given reference element in $\mathbb{R}^2$. Hence, every element of Σ_h of which its vertices are described in $\mathbb{R}^3$, can be identified by a reference element in $\mathbb{R}^2$ with vertices: $\mathbf{x_1} = (0,0)$, $\mathbf{x_2} = (1,0)$ and $\mathbf{x_3} = (1,1)$. We will denote this reference element by $\hat{K}$. Let τ be an element of Σ with vertices $\boldsymbol{\tau_i} \in \mathbb{R}^3$, $i = 1,2,3$. Then the mapping $\Phi : \hat{K} \to \tau$ is given by

$$\Phi(\xi,\eta) := \boldsymbol{\tau_1} + \mathbf{M}\begin{pmatrix}\xi\\ \eta\end{pmatrix}, \tag{3.66}$$

with the 3×2 matrix $\mathbf{M}$

$$\mathbf{M} := \begin{pmatrix}\boldsymbol{\tau_2} - \boldsymbol{\tau_1} & \boldsymbol{\tau_3} - \boldsymbol{\tau_2}\end{pmatrix}. \tag{3.67}$$

When dealing with elements which do not intersect in any way, using quadrature to obtain the matrix entries will yield relatively fast convergence. However, a new type of abutting elements arise, namely elements from different Σ_j which share an edge. By way of example, we will first show why these types of elements still can be expressed semi-analytically. Next, we will generalise the weakly singular matrix entries given any two elements with the use of reference mapping.

Consider the abutting elements represented in Figure 3.7.

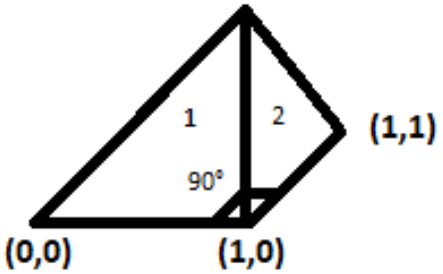


Figure 3.7: Example abutting elements from different Σ_j

Consider the matrix entry of the weakly singular matrix, given by the domain $e_1 \times e_2$, as shown in Figure 3.7, written into integral form:

$$\int_0^1\int_0^1\int_0^{1-y}\int_s^x G\left(\begin{pmatrix}x\\0\end{pmatrix} - \begin{pmatrix}1\\y\end{pmatrix}, t-s\right)\, dtdsdydx.$$

Using the properties of Gamma functions we get

$$\frac{1}{4\pi}\int_0^1\int_0^1\int_0^{1-y}\mathbb{G}\left(0, \frac{\|(x-1,-y)^T\|^2}{4(x-s)}\right)\, dsdydx.$$

Since the next task is to integrate over s, the expression inside the norm will not affect the outcome. We can therefore find an exact analytical term

$$I^{2D}(x,y,s) := \int \mathbb{G}\left(0, \frac{\|(x-1,-y)^T\|^2}{4(x-s)}\right)\, ds.$$

The only requirement is given by the inequality, $x > 1-y \Rightarrow y > 1-x$. We thus obtain the expression

$$\frac{1}{4\pi}\left[\int_0^1\int_{1-x}^1 I^{2D}(x,y,1-y)\, dydx - \int_0^1\int_0^1 I^{2D}(x,y,0)\, dydx\right] \approx 0.0367\ldots$$

The above example motivates us to find a general semi-analytical expression to (3.32). This generalisation is based on the following proposition:

Proposition 3.17. *For each triangular element τ in Σ_j satisfying the following conditions*

(i) one edge is perpendicular to spatial boundary Γ_j.
(ii) one edge is parallel to the spatial boundary Γ_j.

the following holds: There exists a mapping $\Phi : \hat{K} \to \tau$ given by (3.66)*, such that **M** is of the form*

$$\boldsymbol{M} := \begin{pmatrix} a & 0 \\ b & 0 \\ 0 & c \end{pmatrix}, \quad a, b \in \mathbb{R},\ \max(|a|, |b|) \neq 0,\ c \in \mathbb{R}\backslash\{0\}.$$

Proof. We denote the vertices of the element τ by $\boldsymbol{\tau}_i$, $i = 1, 2, 3$. Each vertex can be written into vector form $(x_i, y_i, t_i)^T$. Since one edge of τ is perpendicular to Γ_j, we have:

$$\exists!\ i_1, i_2 \in \{1, 2, 3\} : t_{i_1} - t_{i_2} = 0 \text{ and } \|(x_{i_1}, y_{i_1})^T - (x_{i_2}, y_{i_2})^T\| \neq 0. \tag{3.68}$$

Similarly, due to the parallel edge, we have

$$\exists!\ k_1, k_2 \in \{1, 2, 3\} : \|(x_{i_1}, y_{i_1})^T - (x_{k_2}, y_{k_2})^T\| = 0 \text{ and } t_{k_1} - t_{k_2} \neq 0. \tag{3.69}$$

Without loss of generality, we can denote $\boldsymbol{\tau}_1$ and $\boldsymbol{\tau}_2$ as the elements satisfying (3.68). By uniqueness of the pair (up to symmetry), $\boldsymbol{\tau}_3$ cannot satisfy (3.68). Also, $\boldsymbol{\tau}_1$ and $\boldsymbol{\tau}_2$ cannot satisfy (3.68) and (3.69) simultaneously, which implies that $\boldsymbol{\tau}_3$ must satisfy (3.69) in combination with either $\boldsymbol{\tau}_1$ or $\boldsymbol{\tau}_2$. Without loss of generality, we are now able to assume that $\boldsymbol{\tau}_2$ and $\boldsymbol{\tau}_3$ satisfy (3.69). Given this choice of vertices, the matrix **M** will become

$$\mathbf{M} = \begin{pmatrix} x_2 - x_1 & 0 \\ y_2 - y_1 & 0 \\ 0 & t_3 - t_2 \end{pmatrix} = \begin{pmatrix} a & 0 \\ b & 0 \\ 0 & c \end{pmatrix}.$$

We have by (3.68) that $\|(a, b)^T\| \neq 0 \Rightarrow \max(|a|, |b|) \neq 0$, and by (3.69) that $c \neq 0$, proving the proposition.

□

From now on for each element, we will assume that the reference element mapping is described as in Proposition 3.17. We denote the Gram matrix corresponding to a certain element,e, by $J_g(e)$, with J_{gram} defined as the 3×2 matrix

$$J_g = \begin{pmatrix} \frac{\partial \Phi(\xi,\eta)}{\partial \xi} & \frac{\Phi(\xi,\eta)}{\partial \eta} \end{pmatrix}.$$

Remark. *By our assumption on **M**, the Gram matrix will be equal to **M** given the corresponding mapping function of an element.*

Next, we denote:

$$|\Delta_{J_g}|(e) := \sqrt{J_g(e)^T \cdot J_g(e)}.$$

Now, consider the integral (3.32), given elements e_1, e_2 and corresponding mappings $\Phi_1(\xi_1, \eta_1)$ and $\Phi_2(\xi_2, \eta_2)$. For notational convenience, we denote:

$$\begin{pmatrix} \hat{x}(\xi_1) \\ \hat{y}(\xi_2) \\ \hat{t}(\eta_1) \\ \hat{s}(\eta_2) \end{pmatrix} := \begin{pmatrix} \Phi_1|_{\xi_1} \\ \Phi_2|_{\xi_2} \\ \Phi_1|_{\eta_1} \\ \Phi_2|_{\eta_2} \end{pmatrix}. \tag{3.70}$$

The integral hence becomes

$$|\Delta_{J_g}|(e_1)|\Delta_{J_g}|(e_2) \int_0^1 \int_0^1 \int_0^{\xi_2} \int_0^{\xi_1} G(\hat{x} - \hat{y}, \hat{t} - \hat{s})\ d\eta_1 d\eta_2 d\xi_2 d\xi_1. \tag{3.71}$$

Let $\hat{t}^{-1}$ denote the inverse of $\hat{t}$. If dealing with the same elements, by definition of the fundamental solution, we can rewrite (3.71) into

$$|\Delta_{J_g}|(e_1)|\Delta_{J_g}|(e_2) \int_0^1 \int_0^1 \int_0^{\xi_2} \int_{\hat{t}^{-1}(\hat{s}(\eta_2))}^{\xi_1} G(\hat{x} - \hat{y}, \hat{t} - \hat{s})\ d\eta_1 d\eta_2 d\xi_2 d\xi_1.$$

Changing first changing the variable η_1 by u by $\sigma := \frac{\|\hat{x}-\hat{y}\|^2}{4(\hat{t}-\hat{s})}$ yields a slightly different result as in the previous cases, as we have to account for $\frac{\hat{t}}{\eta_1}$ as well. We therefore get the following expression, in terms of gamma functions:

$$\frac{|\Delta_{J_g}|(e_1)|\Delta_{J_g}|(e_2)}{\frac{\partial \hat{t}}{\partial \eta_1}} \int_0^1 \int_0^1 \int_0^{\xi_2} \mathbb{G}\left(0, \frac{\|\hat{x}-\hat{y}\|^2}{4(\hat{t}(\xi_1)-\hat{s})}\right) d\eta_2 d\xi_2 d\xi_1.$$

This in turn can be expressed as

$$\frac{|\Delta_{J_g}|(e_1)|\Delta_{J_g}|(e_2)}{4\pi \frac{\partial \hat{t}}{\partial \eta_1}\frac{\partial \hat{s}}{\partial \eta_2}} \left(\int_0^1 \int_0^1 I_G(\xi_1,\xi_2,\xi_2) - I_G(\xi_1,\xi_2,0)\, d\xi_1 d\xi_2 \right), \tag{3.72}$$

with

$$I_G(\xi_1,\xi_2,\eta_2) := \begin{cases} 0 & \hat{t}(\xi_1) \leq \hat{s}(\eta_2) \\ \int \mathbb{G}\left(0, \frac{\|\hat{x}-\hat{y}\|^2}{4(\hat{t}(\xi_1)-\hat{s})}\right) d\eta_2 & \text{otherwise} \end{cases}.$$

If the elements are not equal, the corresponding integrals can also be generalised to a semi-analytical form. We define the following function:

$$I_{G_2}(x,y,t,s) := \begin{cases} 0 & t > s \\ \frac{1}{4\pi} \int \mathbb{G}\left(0, \frac{\|x-y\|^2}{4(t-s)}\right) ds & \text{otherwise} \end{cases}. \tag{3.73}$$

Following a similar procedure as in the case of abutting elements, we obtain the following semi analytical expression:

$$\begin{aligned} &\left| \int_0^1 \int_0^1 I_{G_2}(\hat{x}(x),\hat{y}(y),\hat{t}(0),\hat{s}(y)) - I_{G_2}(\hat{x}(x),\hat{y}(y),\hat{t}(0),\hat{s}(0))\, dy\, dx - \right. \\ &\left. \int_0^1 \int_0^1 I_{G_2}(\hat{x}(x),\hat{y}(y),\hat{t}(x),\hat{s}(y)) - I_{G_2}(\hat{x}(x),\hat{y}(y),\hat{t}(x),\hat{s}(0))\, dy\, dx \right|. \end{aligned} \tag{3.74}$$

Where $\hat{x}, \hat{y}, \hat{t}, \hat{s}$ depend on the elements of choice and are set up as in (3.70).

Remark. *As can be seen in* (3.74)*, the determinants of the gram matrices corresponding to two given elements is not present in the final expression. This is a consequence of the application of the chain rule, which cancels out these determinants (given our right-triangular grid). Taking the absolute value corrects the potential sign difference when using the chain rule.*

3.6. Numerical results

We consider a Dirichlet problem (3.1) with homogeneous boundary conditions, no source terms and a non-zero initial condition, on the domain $\Omega = (0,1)^2$. More specifically, we will base our numerical results on the following Dirichlet problem

$$\begin{cases} \partial_t u - \Delta_x u = 0, \\ u|_\Gamma = 0, \\ u(\cdot,0) = 1. \end{cases} \tag{3.75}$$

The corresponding variational formulation of this problem reads: find $q \in H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma)$ such that

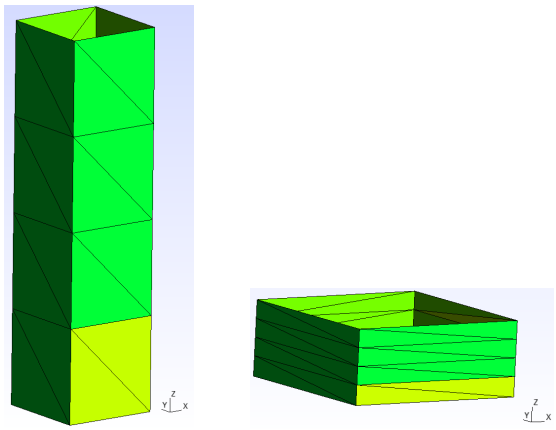
$$\langle Vq, v\rangle_\Sigma = -\langle M_0(1), v\rangle_\Sigma, \quad \forall v \in H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma). \tag{3.76}$$

Let us define $\vec{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$ and $\vec{y} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$. The initial potential operator M_0 can be expressed analytically:

$$M_0(1)(\vec{x},t) = \int_0^1 \int_0^1 G(\vec{x}-\vec{y},t)\, d\vec{y} = \frac{\left(\text{erf}\,\frac{x_2}{2\sqrt{t}} - \text{erf}\,\frac{x_2-1}{2\sqrt{t}}\right)\left(\text{erf}\,\frac{x_1}{2\sqrt{t}} - \text{erf}\,\frac{x_1-1}{2\sqrt{t}}\right)}{4}, \tag{3.77}$$

where erf denotes the (Gauss) error function.

Now, we consider two basis meshes, consisting of 32 right triangular elements of equal size. The first basis mesh corresponds to the Dirichlet problem on the time interval (0,4) in seconds, whilst the second mesh considers the smaller time interval (0, 0.41). Even though the step-size in temporal direction differs between the meshes, the spatial is taken to be equal for both meshes. The basis meshes are given in Figure 3.8.



(a) Basis mesh on time interval (0,4)

(b) Basis mesh on time interval (0,0.41)

Figure 3.8: Basis meshes used for Dirichlet problem

From these basis meshes, two levels of refinement are created: one refinement consisting of 128 elements and one consisting of 512 elements.

The Dirichlet problem is solved on these different meshes with the use the Galerkin-Bubnov variational formula with the space of piecewise constants as a test and trial space, similar to (3.29)-(3.31). The left hand-side of the variational formula is approximated using the procedure described in Section 3.5. The right hand-side is approximated directly by the use of (Gaussian) quadrature. Solving the Dirichlet system as described in (3.31), will yield a vector $\vec{q}$, which does not equal the numerical approximation of the solution $u_h(x,y,t)$. To obtain this approximation we make use of the general representation formula (3.5) to obtain $u_h(x,t)$:

$$u_h(\vec{x},t) := \mathcal{K}_1 q(\vec{x},t) + M_0(1)(\vec{x},t) = \sum_{k=1}^{N} q_k \mathcal{K}_1 \phi_k^{\Sigma,0}(\vec{x},t) + M_0(1)(\mathbf{x},t), \quad (\vec{x},t) \in Q, \tag{3.78}$$

with $\phi_k^{\Sigma,0}(x,y,t)$ given in (3.28), N the total amount of elements in the mesh and $\mathcal{K}_1$ the weakly singular operator.

3.6.1. Analytical expression of solution

The solution of Dirichlet problem described by (3.75) can be expressed analytically. With the use of the "separation of variables" procedure, we obtain the general solution:

$$u(x,y,t) = (C_1 \sin(\lambda x) + C_2 \cos(\lambda x))\,(D_1 \sin(\mu y) + D_2 \cos(\mu y))\, e^{-(\lambda^2+\mu^2)t}, \tag{3.79}$$

with the coefficients C_1, C_2, D_1, D_2 and λ, μ unknown (for now). By applying the boundary conditions (3.75), we have that $C_2 = D_2 = 0$. Thus we obtain

$$u(x,y,t) = C_1 D_1 \sin(\lambda x) \sin(\mu y) e^{(\lambda^2+\mu^2)t}. \tag{3.80}$$

Again, looking at the boundary conditions and assuming the solution is non-trivial, we see that $\lambda = n\pi$ and $\mu = m\pi$, $n, m \in \mathbb{N}$. By the superposition principle, we have

$$u(x, y, t) = \sum_{n=1}^{\infty} \sum_{m=1}^{\infty} A_{nm} \sin(\lambda_n x) \sin(\mu_m y) e^{-(\lambda_n^2 + \mu_m^2)t}, \tag{3.81}$$

with A_{nm} unknown. The initial condition (3.75) and the use of Fourier series theory implies:

$$1 = \sum_{n=1}^{\infty} \sum_{m=1}^{\infty} A_{nm} \sin(\lambda_n x) \sin(\mu_m y) \Rightarrow A_{nm} = 4 \int_0^1 \int_0^1 \sin(\lambda_n x) \sin(\mu_m y)\, dxdy. \tag{3.82}$$

The integral in (3.82) can be expressed analytically and hence we have

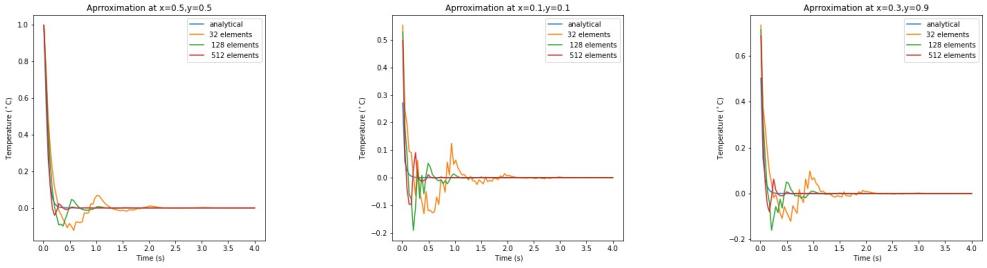
$$A_{nm} = \frac{4(\cos(\mu_m) - 1)(\cos(\lambda_n - 1)}{nm\pi^2},$$

which yields the solution when combined with (3.81).

3.6.2. Discrete solution and error analysis

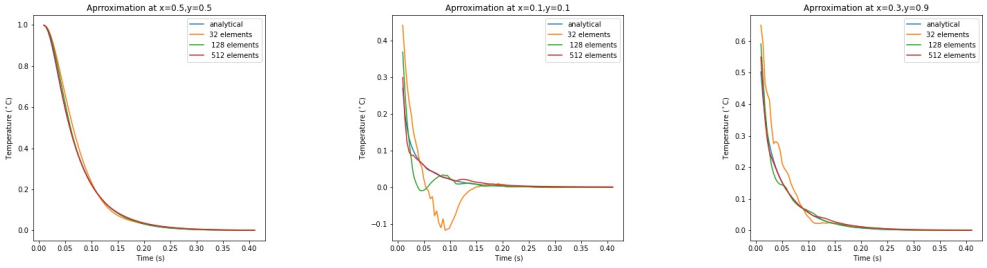
The discrete approximation is compared to the analytical expression of the solution in a similar fashion to the one dimensional case. The space-time domain is discretized into quadrilaterals and the discrete approximation $u_h(\vec{x}, t)$ is evaluated at the centre of each of these quadrilaterals. In a similar fashion, the exact solution is evaluated at the same points. Since the different meshes are all uniform, the relative Euclidean norm of the error is equivalent to the L^2-norm up to a constant h, (the global mesh size defined in (3.13)). Due to the equivalence and convenience, the (relative) Euclidean norm is used in the error analysis.

The (double) infinite summation present in (3.81) is truncated at $N = 100$. The visual representation for both the mesh corresponding to $Q = (0, 1)^2 \times (0, 4)$ and the mesh corresponding to $Q = (0, 1)^2 \times (0, 0.41)$ and their refinements are given in Figure 3.9 and Figure 3.10 respectively.



(a) (b) (c)

Figure 3.9: Approximation of the solution at different positions in the domain $(0, 1)^2$, given the mesh in Figure 3.8 (a). The time interval (0,4) is evaluated at 100 different equidistant points.



(a) (b) (c)

Figure 3.10: Approximation of the solution at different positions in the domain $(0, 1)^2$, given the mesh in Figure 3.8 (b). The time interval (0,0.41) is evaluated at 100 different equidistant points.

In all cases, higher refinement seems to increase the accuracy of the approximation, as expected. Both models perform better when the distance between the point of evaluation and the closest boundaries becomes larger. Analogous to the one dimensional problem, this is a consequence of the Gaussian quadrature error becoming larger as the regularity of the integrand reduces.

Decreasing the temporal step-size seems to improve the overall accuracy of the approximation. Similar to the one dimensional case, the reduction of the temporal step-size h_t improves the accuracy of the quadrature for the single layer potential in (3.78). Given the approximation properties derived by [2, p. 20], which are also summarised in (3.25) and (3.26), one can see that a reduction of h_t will lead to a reduction of the error estimate in (3.31). As h_t (and h_x) are taken to be the same for both (3.31) and (3.78), the error estimate hence reduces.

The actual errors, given hundred equidistant evaluation points are given in the tables below.

Table 3.1: The relative (absolute) error for different levels of refinement given the mesh in Figure (3.8) (a). The error is taken as with respect to the Euclidean norm.

	Relative Error	**(Absolute Error)**	
Evaluation point	**x=0.5, y =0.5**	**x=0.1, y =0.1**	**x=0.3, y =0.9**
32 elements	0.420 (0.505)	1.980 (0.552)	1.006 (0.534)
128 elements	0.266 (0.320)	1.456 (0.406)	0.721 (0.383)
512 elements	0.125 (0.150)	1.074 (0.299)	0.468 (0.249)

As shown in 3.1, the approximations is relatively inaccurate. When looking at the solutions, this makes sense as the solution becomes zero in approximately half a second whilst the temporal step-size of the space time boundary is one second. Since the approximation in each element is given by a piecewise constant, it will not be possible to properly approximate the solution in that case. As shown in Table 3.2, the error becomes significantly smaller when decreasing the temporal step-size.

Table 3.2: The relative (absolute) error for different levels of refinement Figure (3.8) (b). The error is taken as with respect to the Euclidean norm.

	Relative Error	**(Absolute Error)**	
Evaluation point	**x=0.5, y =0.5**	**x=0.1, y =0.1**	**x=0.3, y =0.9**
32 elements	0.072 (0.224)	1.122 (0.535)	0.377 (0.392)
128 elements	0.029 (0.091)	0.495 (0.236)	0.136 (0.142)
512 elements	0.011 (0.035)	0.158 (0.075)	0.067 (0.070)

Comparing Table 3.1 and Table 3.2, one can observe that the error reduces more promptly when dealing with larger meshes. This is consistent with the estimate (3.25), as the temporal step-size has an $\mathcal{O}(h_t^{\frac{1}{2}})$ effect on the error, whilst the spatial step-size affect the magnitude of the error by $\mathcal{O}(h_x)$. For small mesh sizes, and therefore relatively large h_x, the error is dominated by h_x. For increasing level of refinement however, the influence of h_t becomes more visible. This is especially true at $(x, y) = (0.5, 0.5)$, where the accuracy of the numerical solution is not affected by proximity to the boundaries.

4

Parallelization

In this chapter, we will consider parallelization techniques to improve the efficiency of solving a system of linear equations, based on the Galerkin-Bubnov variational formula. First, we consider the weakly singular matrix (given N elements)

$$\mathbf{V}_h := \begin{pmatrix} \langle V\phi_1^{\Sigma,0}, \phi_1^{\Sigma,0}\rangle_\Sigma & \dots & \langle V\phi_N^{\Sigma,0}, \phi_1^{\Sigma,0}\rangle_\Sigma \\ \vdots & \ddots & \vdots \\ \langle V\phi_1^{\Sigma,0}, \phi_N^{\Sigma,0}\rangle_\Sigma & \dots & \langle V\phi_N^{\Sigma,0}, \phi_N^{\Sigma,0}\rangle_\Sigma \end{pmatrix},$$

with V the single layer operator and $\phi_i^{\Sigma,0}$ the basis function given a triangulation Σ. If the triangulation consists of right angled triangles with one edge parallel to the time domain and one parallel to the boundary, we can evaluate the entries of $\mathbf{V}_h$ with the expression (3.72).The right hand side vector **F** is defined as

$$\mathbf{F} := \begin{pmatrix} \langle g, \phi_1^{\Sigma,0}\rangle_\Sigma \\ \vdots \\ \langle g, \phi_N^{\Sigma,0}\rangle_\Sigma \end{pmatrix},$$

with g, the spatial boundary condition. This gives the following system of linear equations with unknown vector **q** of length N

$$\mathbf{V}_h\mathbf{q} = \mathbf{F}. \tag{4.1}$$

4.1. Weakly singular matrix structure

The structure of the matrix may affect the parallelization algorithms, and therefore worth looking into. We consider a right-angled triangulation of the space-time boundary with a rectangular spatial domain. An example of such a triangulation is given in Figure 4.1.

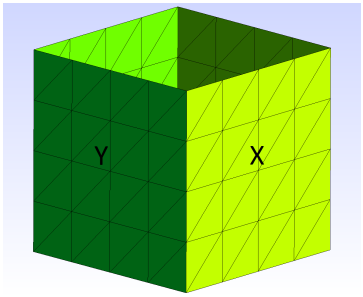


Figure 4.1: Example of space time triangulation, with faces X and Y

When we do not specifically look into the way the elements are numbered, this structure of the weakly singular matrix could, for example, look as denoted in Figure 4.2.

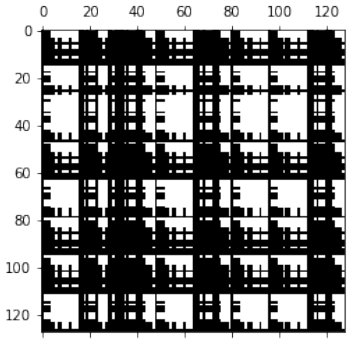
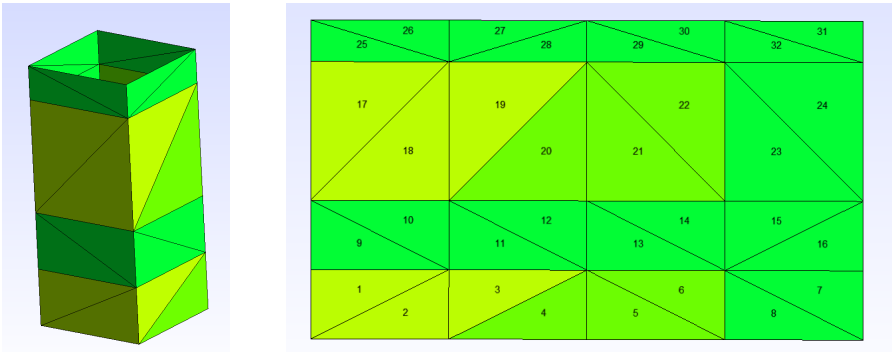


Figure 4.2: Example of weakly singular matrix structure (128 elements), where the white spaces denote zero entries

The structure of the weakly singular matrix changes when the numbering changes. We can, for example number the elements along each ”layer” in time, as presented in Figure 4.3.



(a) 3D representation **(b)** ”Folded out” representation

Figure 4.3: Example triangulation with structured numbering by time-layer

This results in the following weakly singular matrix structure:

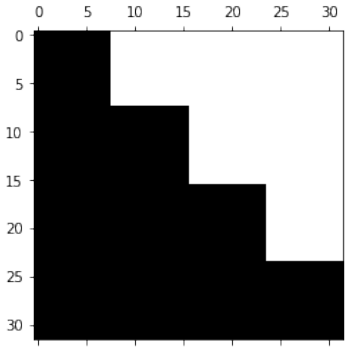


Figure 4.4: Example of weakly singular matrix structure (32 elements) when numbering by time-layer, where the white spaces denote zero entries

The matrix has a lower triangular block form. We can also choose to number by face instead of time layer. This results in the following ”folded out” representation.

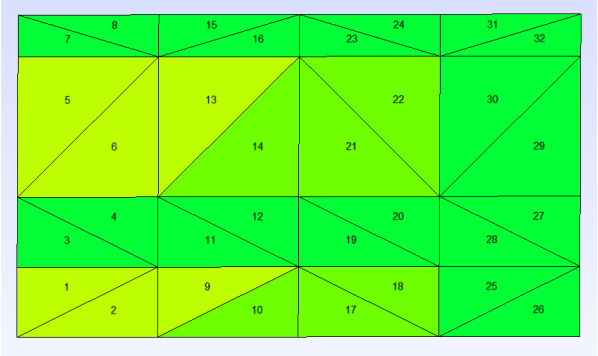


Figure 4.5: ”Folded out” representation (32 elements) when numbering by face

.

This gives the following structure on $\mathbf{V}_h$

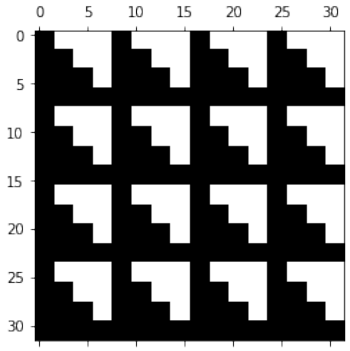


Figure 4.6: Example of weakly singular matrix structure (32 elements) when numbering by face, where the white spaces denote zero entries.

As can be seen in Figure 4.6, the matrix consists of blocks of lower triangular block matrices.

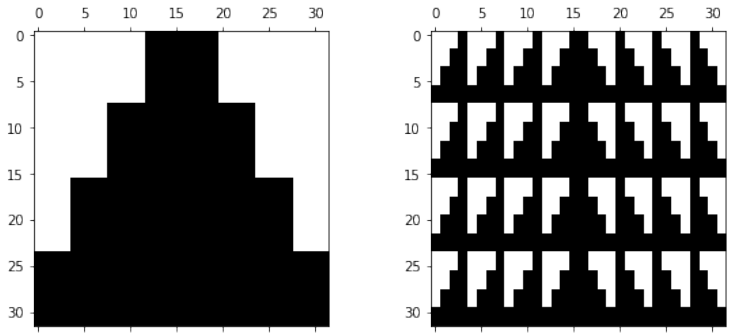
Another way to change the matrix structure, is by changing the way the matrix is assembled, instead of changing the way the elements are numbered. For example, we could create a sequence k defined as

$$k := \{N-1, N-3, \ldots, 3, 1, 0, 2, \ldots, N-2, N\}.$$

This time, we define the entries of the matrix as

$$\mathbf{V}_h[i,j] := \langle V\phi_i^{\Sigma,0}, \phi_{k_j}^{\Sigma,0}\rangle_\Sigma.$$

Depending on the way of (structured) numbering the elements, this can give different results, as shown in Figure 4.7.



(a) Example matrix structure with numbering by time layer

(b) Example matrix structure with numbering by faces

Figure 4.7: Matrix structures given a different assembly method

These figures are examples of the broad scala of matrix structures which can be obtained by either changing the numbering of the elements or the way the matrix is assembled.

4.2. Generalised residual method

The system of linear equations arising from the weakly singular matrix and corresponding right-hand side can be solved directly of with the use of an iterative method. In this thesis the (iterative) Krylov subspace method ”Generalised residual method” (GMRES) will be used to solve the linear system. To properly grasp the possibilities of parallelization, we need to take a closer look at the operations performed by this algorithm.

As already mentioned, GMRES is a Krylov subspace method, i.e. $\mathbf{q}_k$, the approximation of the solution $\mathbf{q}$ in (4.1), at the k-th iteration, is an element of k-dimensional Krylov subspace $K^k(\mathbf{V}_h; \mathbf{r}^0)$ [25], with

$$K^k(\mathbf{V}_h; \mathbf{r}^0) := \mathbf{q}_0 + \text{span}\{\mathbf{r}^0, \mathbf{V}_h\mathbf{r}^0, \ldots, \mathbf{V}_h^{k-1}\mathbf{r}^0\},$$

and

$$\mathbf{r}^k := \mathbf{F} - \mathbf{V}_h\mathbf{q}_k.$$

Within this increasingly expanding subspace, we search for an approximate solution of the form

$$\mathbf{q}_k = \mathbf{q}_0 + \mathbf{z}_k, \tag{4.2}$$

with $\mathbf{z}_k \in K^k(\mathbf{V}_h; \mathbf{r}^0)$, such that

$$\|\mathbf{r}^k\|_2 = \min_{\mathbf{z} \in K^k(\mathbf{V}_h; \mathbf{r}^0)} \|\mathbf{F} - \mathbf{V}_h\mathbf{z}\|_2. \tag{4.3}$$

The GMRES algorithm is based on solving the minimisation problem (4.3) efficiently with the use of an orthonormal basis of the corresponding Krylov subspace. Various methods for finding the orthonormal basis of the Krylov subspace, such as Arnoldi’s method and Modified Gram-Schmidt method [25], are available and will not be discussed in detail in this thesis.

By left-preconditioning, we transform the system of linear equations (4.1) into the following system:

$$\mathbf{M}^{-1}\mathbf{V}_h\mathbf{q} = \mathbf{M}^{-1}\mathbf{F}, \tag{4.4}$$

where $\mathbf{M}$, a non-singular square matrix of equal dimensions as $\mathbf{V}_h$, represents an arbitrary *preconditioner*. Given such a preconditioner, the GMRES algorithm for k iterations can be described as follows [17]:

Algorithm 1: Left-preconditioned GMRES

Result: $\mathbf{q}_k$, approximation of the solution $\mathbf{M}^{-1}\mathbf{V}_h\mathbf{q} = \mathbf{M}^{-1}\mathbf{F}$
Define initial guess $\mathbf{q}_0$;
Compute $\mathbf{r}^0 = \mathbf{M}^{-1}(\mathbf{F} - \mathbf{V}_h\mathbf{q}_0)$, $\beta = \|\mathbf{r}^0\|_2$ and $v_1 = \frac{\mathbf{r}_0}{\beta}$;
for *j=1,...,k* **do**
  Compute $w := \mathbf{M}^{-1}\mathbf{V}_h v_j$;
  for *i = 1,...,j* **do**
    $h_{i,j} := (w, v_i)$;
    $w := w - h_{i.j}v_i$;
  end
  Compute $h_{j+1,j} = \|w\|_2$ and $v_{j+1} = \frac{w}{h_{j+1,j}}$;
end
Define $\mathbf{V}_k := [v_1, \ldots, v_k]$, $\bar{\mathbf{H}}_k = \{h_{i,j}\}_{1\leq i\leq j; 1\leq j\leq k}$;
Compute $\mathbf{y}_k$ = argmin$_{\mathbf{y}}\|\beta\mathbf{e}_1 - \bar{\mathbf{H}}_k\mathbf{y}\|_2$, and $\mathbf{q}_k = \mathbf{q}_0 + \mathbf{V}_k\mathbf{y}_k$;
if *Satisfied* **then**
  Stop
else
  Set $\mathbf{q}_0 := \mathbf{q}_k$;
  Repeat algorithm;
end

Remark. *The vector* $\mathbf{e}_1$ *is the canonical vector* $(1, 0, \ldots, 0)^T$.

The main operations in Algorithm 1 can be distinguished by 5 different types of operations [17]:

1. Preconditioner setup.
2. Matrix vector multiplications.
3. Vector updates.
4. Dot products.
5. Preconditioning operations.

For parallel machines, the *potential* bottlenecks in the above list are (1) and (5). Matrix vector multiplications and dot products are relatively easy to implement on high performance computers. Also, vector updates will not cause significant difficulties in general on parallel machines [17]. We can henceforth conclude that the choice of preconditioner can significantly affect the efficiency of the parallelization.

4.3. Calderón preconditioning

The reduction of the condition number of $\mathbf{V}_h$, by applying a preconditioner $\mathbf{M}^{-1}\mathbf{V}_h$, *might* result in desired convergence properties. It should be noted, however, that a small condition number is by no means a necessary and sufficient condition for GMRES. Given a diagonalizable matrix A, roughly speaking, the GMRES algorithm converges quickly if it is 'close to' normal with respect to the condition number of the spectral decomposition of A and if properly normalised degree n polynomials their size on the spectrum of the matrix rapidly decreases with n ([24], Thm. 35.2). Although not sufficient from a theoretical perspective, Calderón preconditioning, which is a particular form of operator preconditioning, usually provides a good preconditioner when used with GMRES.

Operator preconditioning makes use of the following idea. If we are given a continuous bijective linear operator $A : V \to W$ and an isomorphism $B : W \to V$, on function spaces V and W, then AB will provide an endomorphism on V. Discretisation of such an endomorphism often gives rise to well-conditioned matrices [8].

In order to quantify how operator preconditioning affects the condition number, we will make use of the following theory by Hiptmair [8]. On two reflexive Banach spaces A,B we consider two continuous sesquilinear forms $a \in \mathcal{L}(A \times A, \mathbb{C})$ and $b \in \mathcal{L}(B \times B.\mathbb{C})$, where $\mathcal{L}(X, Y)$ denotes the the set of bounded

linear operators from space X into Y. Let $A_h := span\{\phi_k\}_1^N \subset A$ and $B_h := span\{\psi_j\}_1^N \subset B$ be finite dimensional subspaces on which the following conditions hold:

$$\sup_{0\neq v_h\in A_h} \frac{|a(u_h, v_h)|}{\|v_h\|_A} \geq c_1\|u_h\|_A \quad \forall u_h \in A, \tag{4.5}$$

$$\sup_{0\neq w_h\in B_h} \frac{|b(z_h, w_h)|}{\|w_h\|_B} \geq c_2\|z_h\|_B \quad \forall z_h \in B. \tag{4.6}$$

Further we assume there exists a sesquilinear form $d \in \mathcal{L}(A \times B, \mathbb{R})$ satisfying

$$\sup_{0\neq w_h\in B_h} \frac{|d(v_h, w_h)|}{\|w_h\|_B} \geq c_3\|v_h\|_A \quad \forall v_h \in A. \tag{4.7}$$

This results in the following Galerkin matrices

$$\mathcal{A}[i,j] := a(\phi_i, \phi_j), \quad \mathcal{B}[i,j] := b(\psi_i, \psi_j), \quad D[i,j] := d(\phi_i, \psi_j), \quad i,j = 1,\dots N.$$

The following result is obtained:

Theorem 4.1 ([8, Thm. 2.1])**.** *Assume* (4.5) - (4.7) *hold and* dim $A_h =$ dim B_h*, then*

$$\kappa(D^{-1}\mathcal{B}D^{-T}\mathcal{A}) \leq \frac{\|a\|\|b\|\|d\|^2}{c_1c_2c_3^2},$$

where $\kappa(\cdot)$ *denotes the spectral condition number.*

Proof. The proof is added for the sake of completeness. It goes along the lines oft he proof introduced by Hiptmair [8] with some added intermediate steps for elaboration.

From now on, we denote the dual of a Banach space X by X^*. Let $J_A : A_h \to A_h^*$, $J_B : B_h \to B_h^*$ and $J_D : A_h \to B_h^*$ be bounded linear operators associated with the sesquilinear forms a,b and d respectively. We can define an element $\tau_h \in A_h^*$ by the function $\tau : A_h \to \mathbb{K}$, such that for any $t \in A_h$ we have $\tau_h : t \mapsto a(t,h)$, for a given $h \in A_h$. This can be done in a similar fashion in the other dual spaces. Note that this implies

$$\|J_A\| = \sup_{\|x\|_{A_h}=1} \|J_Ax\|_{A_h^*} = \sup_{\|x\|_{A_h}=1} \sup_{\|y\|_{A_h}=1}, |a(v_h, x)| = \|a\|. \tag{4.8}$$

Taking a closer look at (4.5) will show us that, $\forall y \in A_h : \ \|y\|_{A_h^*} = 1 \Rightarrow \|y\|_{A_h} \leq \frac{1}{c_1}$, we also have

$$\|J_A^{-1}\| = \sup_{\|y\|_{A_h^*}=1} \|J_A^{-1}y\|_{A_h} \leq \frac{1}{c_1}.$$

Next, we denote the adjoint operator of J_D by J_D^*. Using similar steps as in the case of J_A, and recalling that $J_D^* := D^T$ we can conclude

$$\|J_D^{-1}J_BJ_D^{-*}J_A\| \leq \frac{1}{c3^2}\|a\|\|b\|, \quad \|J_A^{-1}J_D^*J_B^{-1}J_D\| \leq \|d\|^2\frac{1}{c_1c_2}.$$

Equipping $\mathbb{C}^n$ with the same norm as the Banach space A, we obtain the final inequalities

$$|\lambda_{\max}(D^{-1}\mathcal{B}D^{-T}\mathcal{A})| \leq \|D^{-1}\mathcal{B}D^{-T}\mathcal{A}\| \leq \frac{1}{c_3^2}\|a\|\|b\|, \tag{4.9}$$

$$|\lambda_{\max}(D^{-1}\mathcal{B}D^{-T}\mathcal{A})^{-1}| \leq \|\mathcal{A}^{-1}D^T\mathcal{B}^{-1}D\| \leq \frac{1}{c_1c_2}\|d\|^2, \tag{4.10}$$

where $\lambda_{max}(X) := \{\max x \in \sigma(X)\}$, $\lambda_{min}(X) := \{\min x \in \sigma(X)\}$ and $\sigma(X)$ defines the spectrum of X. The product of the norms in (4.9) and (4.10) equals the spectral condition number, and hence the statement is proven.

□

If we use a left preconditioner

$$C_v^{-1} := D^{-1}\mathcal{B}D^{-T},$$

applied by $C_v^{-1}\mathcal{A}$, we obtain a preconditioned form of matrix $\mathcal{A}$ with bounded condition number, which is the desired result.

4.3.1. Dirichlet Boundary value problem

Consider the Dirichlet domain variational problem (3.4) which leads to the indirect formulation of the first kind given by (3.8). As stated by Noon, the single layer integral operator $V : H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma) \to H^{\frac{1}{2},\frac{1}{4}}(\Sigma)$ and the hypersingular integral operator $W : H^{\frac{1}{2},\frac{1}{4}}(\Sigma) \to H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma)$ are both elliptic and their composition $WV : H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma) \to H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma)$ defines an operator of order zero [15]. The stability assumptions (4.5) and (4.6) are therefore satisfied, which in turn implies that we can find a suitable preconditioner of $\mathbf{V}_h$ in (4.1) by using the Galerkin matrix corresponding to the hypersingular operator. The single later operator is discretized with respect to either $X_h := S_h^0(\Sigma_h)$ or $X_h := S_{h_x,h_t}^{0,0}(\Sigma_h)$, which are defined by (3.27) and (3.24) respectively. We can hence write $X_h := \text{span}\{\phi_i^0\}_1^N$. For the hypersingular operator however, we need to use a trial space $Y_h := \text{span}\{\psi_i\}_1^N \subset H^{\frac{1}{2},\frac{1}{4}}(\Sigma)$. As a corollary of Theorem 4.1 we have [3]:

Corollary 4.2 ([8, Thm.2.1],[3, Thm.8.2])**.** *Assume the stability condition*

$$\sup_{0\neq v_h\in Y_h} \frac{\langle u_h, v_h\rangle_\Sigma}{\|v_h\|_{H^{\frac{1}{2},\frac{1}{4}}(\Sigma)}} \geq C_M \|u_h\|_{H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma)}, \quad \forall u_h \in X_h, \tag{4.11}$$

holds. Then there exists a constant c_κ such that

$$\kappa(D_h^{-1} W_h D_h^{-T} \mathbf{V}_h) \leq c_\kappa,$$

with

$$\mathbf{V}_h[i,j] = \langle V\phi_j^0, \phi_i^0\rangle_\Sigma, \quad W_h[i.j] = \langle W\psi_j, \psi_i\rangle_\Sigma, \quad D_h[i,j] = \langle \phi_j^0, \psi_i\rangle_\Sigma, \quad i,j = 1\ldots, N.$$

Proof. The proof is a direct application of Theorem 4.1, with

$$a(u,v) = \langle Vu, w\rangle_\Sigma, \quad b(q,z) = \langle Wq, z\rangle_\Sigma, \quad d(v,z) = \langle v, z\rangle_\Sigma, \quad u,v \in H^{\frac{1}{2},\frac{1}{4}}(\Sigma) \text{ and } q,z \in H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma).$$

□

If we can find a boundary element space Y_h, such that the stability condition (4.11) is satisfied, then $C_v^{-1} := D_h^{-1} W_h D_h^{-T}$ can be used as a preconditioner of $\mathbf{V}_h$.

Dual mesh approach

One of the challenges of operator preconditioning is finding subspace Y_h such that Theorem 4.1 can be applied. When $X_h := S_{h_x,h_t}^{0,0}(\Sigma_h)$, Steinbach and Dohr [21, 3] have shown that there are several viable candidates for Y_h satisfying the stability condition (4.11), by using the notion of *dual meshes*.

Definition 4.1. *If the vertex-edge and edge-cell incidence matrices of one mesh (the primal mesh), agree with the cell-edge and edge-vertex of another mesh (the dual mesh), the meshes are called the* ***dual*** *of each other [8].*

Examples of finding a dual to a mesh are described in the work of Steinbach [21]. The possible candidates for Y_h are given in the following proposition.

Proposition 4.3 ([3, Ch.8])**.** *Given $X_h := S_{h_x,h_t}^{0,0}(\Sigma_h)$, the following candidates for Y_h satisfy the stability condition* (4.11)*.*

- Y_h=$S_{h_x}^1(\tilde{\Gamma}_{N_x}) \otimes S^0(I_{N_t})$
- Y_h=$S_{h_x}^1(\tilde{\Gamma}_{N_x}) \otimes S^1(\tilde{I}_{N_t})$

Where $\tilde{\Gamma}_{N_x}$ is the dual mesh of Γ_{N_x}, and $\tilde{I}N_t$ the dual mesh of I_{N_t}

In the following, we analyse the situation for a general space-time discretization. In other words, we consider a triangulation Σ_h of Σ and denote its dual mesh by $\tilde{\Sigma}_h$. When dealing with the more general form $X_h := S_h^0(\Sigma_h)$, we make use of the embedding property of Sobolev spaces in Lemma 3.14. In the two dimensional case, we have that $S_h^0(\Sigma_h)$ and $S_h^1(\tilde{\Sigma}_h)$ are proper subspaces of $H^{\frac{1}{4}}(\Sigma)$ and

$H^{\frac{1}{2}}(\tilde{\Sigma})$ respectively. Next, we define the generalised L^2-projection operators $P_\Sigma^0 : L^2(\Sigma) \to S^0(\Sigma_h)$ and $P_\Sigma^1 : L^2(\Sigma) \to S^1(\tilde{\Sigma}_h)$, i.e., they are the solution to the respective variational problem

$$\langle P_\Sigma^0 u, \tau\rangle_\Sigma = \langle u, \tau\rangle_\Sigma, \quad \forall \tau \in S_h^0(\Sigma_h), \tag{4.12}$$

and

$$\langle P_\Sigma^1 u, \tau\rangle_\Sigma = \langle u, \tau\rangle_\Sigma, \quad \forall \tau \in S_h^0(\Sigma_h). \tag{4.13}$$

We will use the following theorem by Dohr [3]:

Theorem 4.4 ([3, Thm.8.3])**.** *Let X_h and Y_h be given boundary element spaces satisfying*

$$\sup_{0\neq\tau\in X_h} \frac{\langle v, \tau\rangle_\Sigma}{\|\tau\|_\Sigma} \geq c\|v\|_\Sigma, \quad \forall v \in Y_h, \tag{4.14}$$

where the constant c is independent of h. Moreover let the L^2 projection operator P_Σ defined by either (4.12) *or* (4.13) *be $H^{\frac{1}{2},\frac{1}{4}}$-stable, i.e., there exists a constant $c > 0$, such that*

$$\|P_\Sigma u\|_{H^{\frac{1}{2},\frac{1}{4}}(\Sigma)} \leq c\|u\|_{H^{\frac{1}{2},\frac{1}{4}}(\Sigma)}, \quad \forall u \in H^{\frac{1}{2},\frac{1}{4}}(\Sigma). \tag{4.15}$$

Then the stability condition (4.11) *holds.*

Remark. *For a globally quasi-uniform mesh,* (4.15) *holds due to appropriate error estimates and the use of the inverse inequality [22, Ch.2].*

Since (4.12) and (4.13) are a Galerkin-Bubnov and Galerkin-Petrov variational formulation respectively, the formulations are uniquely solvable and their trial and test spaces satisfy (4.14) when dealing with a locally quasi-uniform mesh ([21], Lemma 3.1). However, if this is not the case, the stability assumption (4.14) needs to be verified when creating the mesh.

To verify the $H^{\frac{1}{2},\frac{1}{4}}$-stability of the projections, consider an element τ_l of the (locally quasi-uniform) mesh. Let $N_l := \dim X_h(\tau_l) = \dim X_h|_{\tau_h}$. As mentioned before, we can define the spaces test and trial spaces by their respective bases, i.e., $X_h := \text{span}\{\phi_i\}_1^N$ and $Y_h := \text{span}\{\psi_j\}_1^N$. For each k-th basis function we denote its support by ω_k. Let N_{vertex} and $N_{elements}$ denote the total amount of nodes and elements of mesh Σ_h, respectively. Next, we define two index sets as follows [22]:

$$I(k) := \{l \in \mathbb{N} : \tau_l \cap \omega_k \neq \emptyset\}, \quad k = 1, \ldots, N_{vertex},$$
$$J(l) := \{k \in \mathbb{N} : \tau_l \cap \omega_k \neq \emptyset\}, \quad l = 1, \ldots, N_{elements}.$$

Now, we define the local Gram matrices

$$G_l[j, i] := \langle \phi_{J(l)_i}, \psi_{J(l)_j}\rangle_{L^2(\tau_l)}, \quad i, j = 1, \ldots, N_l.$$

We define the nodal mesh size $\hat{h}$ as

$$\hat{h}_k := \frac{1}{|I(k)|} \sum_{l\in I(k)} h_l, \quad k = 1, \ldots, N,$$

where $|I(k)|$ denotes the cardinality of $I(k)$, and h_l is defined as in (3.12). Consider

$$H_l := \text{diag}(\hat{h}_{J(l)_i})_{i=1}^{N_l},$$

and

$$D_l := \text{diag}G_l.$$

To satisfy the stability, we will assume all elements τ_l satisfy the following condition [22, 21, 3]:

$$\exists C > 0 : \ (H_l G_l H_l^{-1} x_l, x_l) \geq C(D_l x_l, x_l), \quad \forall x_l \in \mathbb{R}^{N_l}. \tag{4.16}$$

To link this assumption to the required stability, we make use of the following theorem by Steinbach [21]:

Theorem 4.5 ([21, Thm.4.2]). *Assume* (4.16) *holds. Then for all* $s \in (0, 1]$

$$\|P_\Sigma u\|_{H^s(\Sigma)} \leq c_s \|u\|_{H^s(\Sigma)}, \tag{4.17}$$

holds.

Proof. The proof is given in Theorem 4.2 in [21].

Assuming (4.16) for the given meshes, hence results in the bound given by (4.17). By Lemma 3.14, with using $\frac{1}{4} := \min(\frac{1}{2}, \frac{1}{4})$ we have

$$\|P_\Sigma u\|_{H^{\frac{1}{4}}(\Sigma)} \leq c_1 \|u\|_{H^{\frac{1}{2},\frac{1}{4}}(\Sigma)}, \quad \forall u \in H^{\frac{1}{2},\frac{1}{4}}(\Sigma). \tag{4.18}$$

The inequality (4.18) does not coincide with the stability condition (4.11). Next, we will use the *inverse inequality*, given in the following theorem [18].

Theorem 4.6 ([18, Thm.4.4.2]). *Consider the surface of a polyhedron* Γ. *Assume the triangulation* $\mathcal{T}$ *on* Ω, *consists of plane panels with straight edges with mesh width* $h > 0$. *We have for* $0 \leq m \leq l$, *all* $\tau \in \mathcal{T}$ *and all* $v \in \mathbb{P}_k^\tau$, *(the space of polynomials of degree* k *and support* τ*):*

$$\|v\|_{H^l(\tau)} \leq C h_\tau^{m-l} \|v\|_{H^m(\tau)}. \tag{4.19}$$

The constant C only depends on h_{max}, l, k *and the shape regularity of the mesh.*

As stated in Section 3.3, we assume a triangulation with straight edges, as required in Theorem 4.6. So with the use Theorem 4.6 and setting $m = \frac{1}{4}$, $l = \frac{1}{2}$, we have for all $\tau \in \mathcal{T}$

$$\|P_\Sigma u\|_{H^{\frac{1}{2}}(\tau)} \leq C h_\tau^{-\frac{1}{4}} \|P_\Sigma u\|_{H^{\frac{1}{4}}(\tau)}. \tag{4.20}$$

For notational convenience $Ch_\tau^{-\frac{1}{4}}$ will be denoted as $C(h)$. By using the embedding property Lemma 3.14 again, with $\frac{1}{2} = \max(\frac{1}{2}, \frac{1}{4})$, we obtain the the bound

$$\|P_\Sigma u\|_{H^{\frac{1}{2},\frac{1}{4}}(\Sigma)} \leq c_2 \|P_\Sigma u\|_{H^{\frac{1}{2}}(\Sigma)}, \quad \forall u \in H^{\frac{1}{2},\frac{1}{4}}(\Sigma). \tag{4.21}$$

Combining (4.18) with (4.20) and (4.21) implies the inequality (4.15) holds:

$$\|P_\Sigma u\|_{H^{\frac{1}{2},\frac{1}{4}}(\Sigma)} \leq c_2 c_1 C(h) \|u\|_{H^{\frac{1}{2},\frac{1}{4}}(\Sigma)}, \quad \forall u \in H^{\frac{1}{2},\frac{1}{4}}(\Sigma). \tag{4.22}$$

Since the bound obtained in (4.22) depends on h, we are currently not able to show Y_h is a viable candidate such that the preconditioner C_v^{-1} can be used. Hence, at the moment of writing this thesis, this remains an open problem. Note that, if we were able to proof the missing case, to obtain a Galerkin matrix for the hypersingular operator W that leads to a suitable preconditioner in 2D, we can only choose $S_h^1(\tilde{\Sigma}_h)$.

Diagonal duality pairing

In 2018, Stevenson and Venetië proposed another approach to the duality pairing $d(\cdot, \cdot)$ for operator preconditioning [23]. The appealing part of their proposal is the fact that the discrete form of the duality pairing operators are diagonal matrices. The way this is achieved will be described in this section. For the mathematical theory behind this method, however, the reader is referred to the paper by Stevenson and Venetië [23].

The weakly singular operator is assumed to be discretized by piecewise constants throughout this section. Recall that our preconditioner system will read as $\mathbf{M}_h^{-1}\mathbf{W}_h\mathbf{M}_h^T\mathbf{V}_h$. We denote the basis of $\mathbf{V}_h$, originating from the space of piecewise constant functions, as Ξ. Similarly, we denote the basis of the hypersingular matrix W_h, consisting of piecewise linear functions, as Φ. It is clear that in this case, the bases are not *bi-orthogonal*, i.e. the Matrix $D_h := \langle \Xi, \Phi \rangle_\Sigma$ need not be diagonal. In order to achieve bi-orthogonality we thus need to find another discrete version of the hypersingular operator with basis Ψ, which still satisfies the stability condition (4.11), and has diagonal duality pairing $D_h := \langle \Xi, \Psi \rangle_\Sigma$.

Consider the triangulation Σ_h, of the lateral space-time boundary Σ. Given $\Sigma := \Gamma \times (0, T)$, the boundaries of Σ are $\Gamma \times \{0\}$ and $\Gamma \times \{T\}$ and are denoted by $\partial\Sigma$. For a triangulation Σ_h, let $\mathcal{V}(\Sigma_h)$ denote the set of vertices of Σ_h and define

$$N^0_{\Sigma_h} := \{v \in \mathcal{V}(\Sigma_h) : v \notin \partial\Sigma\}.$$

Now, we can define the *valence* of each vertex as

$$d_v := |\{\tau \in \Sigma_h : v \in \bar{\tau}\}|,$$

where $|\cdot|$ denotes the cardinality of the set. For the basis of the preconditioner we will make use of the space of piecewise linear functions which vanish on $\partial\Sigma$, i.e.

$$S^1_{\partial\Sigma}(\Sigma_h) := \{\psi_v(v') \in S^1_h(\Sigma) : \psi_v(v') = 0,\ \forall v, v' \in \mathcal{V}(\Sigma_h) \setminus N^0_{\Sigma_h}\}.$$

Similar as proposed by Stevenson, the construction of the basis Ψ builds on two collections, Θ and Υ, of locally supported functions in $H^{\frac{1}{2},\frac{1}{4}}(\Sigma)$ whose cardinalities equal Ξ, one of which is bi-orthogonal to Ξ and one is inside $S^1_{\partial\Sigma}(\Sigma)$ and satisfies the *approximation property*.

Definition 4.2 (**Approximation property**,[10, Ch.3])**.** *Given a Banach space X, if every compact operator is a limit of finite rank operators, X is said to be satisfying the approximation property.*

Consider $\Theta := \{\theta_\tau : \tau \in \Sigma_h\}$, such that $\theta_\tau \geq 0$, supp $\theta_\tau \subset \bar{\tau}$ and, for convenience,

$$\langle \theta_\tau, \xi_\tau \rangle_\Sigma = |\tau|,\ \xi_\tau \in \Xi,\ \forall \tau \in \Sigma_h.$$

One possible construction Θ is with the use of 'bubble functions', however the emphasis lies not on the actual construction of Θ, but on its existence.

The other collection of locally supported functions Υ is defined as $\Upsilon := \{\upsilon_\tau : \tau \in \Sigma_h\} \subset S^1_{\partial\Sigma}(\Sigma_h)$ with

$$\upsilon_\tau := \sum_{v \in N^0_{\Sigma_h} \cap \mathcal{V}(\Sigma_h)} d_v^{-1} \psi_v.$$

Finally, we can define $\Psi := \{\psi_\tau : \tau \in \Sigma_h\} \subset S^1_{\partial\Sigma}(\Sigma_h) \oplus \text{span}\ \Theta$, with

$$\psi_\tau := \upsilon_\tau + \frac{\langle \mathbb{1} - \sigma_\tau, \xi_\tau \rangle_\Sigma}{\langle \theta_\tau, \xi_\tau \rangle_\Sigma} \theta_\tau - \sum_{\tau' \in \Sigma_h \setminus \tau} \frac{\langle \sigma_\tau, \xi_{\tau'} \rangle_\Sigma}{\langle \theta_{\tau'}, \xi_{\tau'} \rangle_\Sigma} \theta_{\tau'}. \tag{4.23}$$

It can be verified that $\langle \xi_\tau, \psi_\tau \rangle_\Sigma = |\tau|,\ \forall \tau \in \Sigma_h$. As a candidate duality pairing we will therefore define

$$D_h := \langle \Xi, \Psi \rangle_\Sigma, \tag{4.24}$$

with Ξ and Ψ described above. Another interesting property of the duality pairing is given in the following theorem [23]

Theorem 4.7 ([23, Cor.3.3])**.** *Let $\mathbb{T}$ denote the collection of triangulations on Σ. Let $D_h : S^0_h(\Sigma_h) \to (\text{span}\ \Psi)'$. It holds that D_h is an invertible operator whose inverse is in $\mathcal{L}((\text{span}\ \Psi)', S^0_h(\Sigma_h))$, with*

$$\|D_h\|_{\mathcal{L}(S^0_h(\Sigma_h), (\text{span}\ \Psi)')} \leq 1,$$

and

$$\sup_{\mathcal{T} \in \mathbb{T}} \|D_h^{-1}\|_{\mathcal{L}((\text{span}\ .\Psi)', S^0_h(\Sigma_h))} < \infty$$

Due to the uniform boundedness implied by Theorem 4.7, the results are thus valid without any additional assumptions on the mesh grading.

Having established an uniformly bounded duality pairing, all that remains is to find an discrete version of the operator $W : H^{\frac{1}{2},\frac{1}{4}}(\Sigma) \to H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma)$, denoted as W_h. To obtain a uniform preconditioner, W_h and its inverse needs to be uniformly bounded with respect to the triangulations. The span of the collection Θ will be referred to as $\mathcal{B} := \text{span}\ \Theta$. Given the direct sum decomposition $S^1_{\partial\Sigma}(\Sigma) \oplus \mathcal{B}$, we consider to projections P_S, $P_\mathcal{B}$ on this domain whose ranges are defined by $S^1_{\partial\Sigma}(\Sigma_h)$ and $\mathcal{B}$, respectively. Next, we define

$$B^{S\oplus\mathcal{B}}v(w) := B^S(P_S v)(P_S w) + B^{\mathcal{B}}(P_{\mathcal{B}} v)(P_{\mathcal{B}} w),$$

with $B^S \in \mathcal{L}(S^1_{\partial\Sigma}(\Sigma_h), (S^1_{\partial\Sigma}(\Sigma_h))')$ and $B^{\mathcal{B}} \in \mathcal{L}(\mathcal{B}, \mathcal{B}')$. As a choice of B^S, the hypersingular operator can be used. Using the same construction proposed by Stevenson, we can make sure $B^{\mathcal{B}}$ is uniformly bounded (and its inverse as well).

Remark. *Two assumptions on the space of bubble functions $\mathcal{B}$ are actually required to complete the construction of $B^{\mathcal{B}}$. These assumptions are stated in (3.24) and (3.25) in the work of Stevenson [23].*

Setting

$$(W_h v)(w) := B^{S\oplus\mathcal{B}}v(w), \tag{4.25}$$

will now give rise to a well-defined uniformly bounded discretized version of W. In turn, this leads to the resulting preconditioner:

$$C_v := D_h^{-1} W_h D_h^{-T}, \tag{4.26}$$

with D_h and W_h defined by (4.24) and (4.25) respectively. More specifically we obtain:

$$\mathbf{W}_h := \mathbf{p}^T \mathbf{B}^S \mathbf{p} + \mathbf{q}^T \mathbf{B}^{\mathcal{B}} \mathbf{q}, \tag{4.27}$$

with

$$\mathbf{B}^{\mathcal{B}} := \beta D_h^{1-\frac{2s}{d}}, \tag{4.28}$$

$$\mathbf{p}_{vT} := \begin{cases} d_v^{-1}, & v \in N^0 \cap \mathcal{V}(T), \\ 0, & \text{otherwise.} \end{cases} \tag{4.29}$$

$$\mathbf{q}_{T'T} := \delta_{T'T} - \frac{1}{d+1} \sum_{v \in N^0 \cap \mathcal{V}(T)\ cap \mathcal{V}(T')} d_v^{-1}. \tag{4.30}$$

In the above definitions, β is a real scalar valued constant, d denotes the dimension of the domain and s is defined as a scalar depending on the Sobolev space H^s such that $W : H^s \to (H^s)'$. Since in a space-time setting, we are dealing with anisotropic Sobolev spaces, s is taken to be $\min(\frac{1}{2}, \frac{1}{4}) = \frac{1}{4}$. $\mathbf{B}^S$ is taken to be the discretisation of the hypersingular operator with respect to piecewise linear functions in $S^0_{\partial\Sigma}(\Sigma)$.

4.3.2. Construction of the hypersingular matrix

Both the dual mesh approach and the approach proposed by Stevenson, requires the construction of the hypersingular matrix. Recall from Chapter 2 that the hypersingular operator $W : H^{\frac{1}{2},\frac{1}{4}} \to H^{-\frac{1}{2},-\frac{1}{4}}$ is defined as

$$W\phi := -\gamma_1 \mathcal{K}_2 \phi.$$

The bilinear form induced by the hypersingular operator is given by [2]:

$$\langle Wu, v\rangle_\Sigma = \int_\Sigma v(x,t)\partial_{n_x} \int_\Sigma \partial_{n_y} G(x-y, t-s) u(y,s)\, d\sigma_y ds dt d\sigma_x. \tag{4.31}$$

With the use of integration by parts this can be written into [1]:

$$\langle Wu, v\rangle_\Sigma = \langle V \,\mathsf{curl}\, u, \mathsf{curl}\, v\rangle_\Sigma - \langle \frac{\partial}{\partial t} V(u\vec{n}^\Sigma), v\vec{n}^\Sigma\rangle_\Sigma, \quad \forall u, v \in H^{\frac{1}{2},\frac{1}{4}}, \tag{4.32}$$

where

$$\mathsf{curl}\, v(x,t) := n_1^\Sigma(x) \frac{\partial}{\partial x_2} v(x,t) - n_2^\Sigma(x) \frac{\partial}{\partial x_1} v(x,t),$$

in two dimensions and

$$\text{curl}\, v(x,t) := \vec{n}^{\Sigma} \times \nabla_x v(x,t), \quad x,t \in \Sigma,$$

in three dimensions. In these expressions $\vec{n}^{\Sigma}$ denotes the outward normal vector with respect to the space-time boundary. Note that the hypersingular integrals can hence be reduced to weakly singular ones. The curl operators can be defined with the use of piecewise constant basis functions. Therefore, the Galerkin-Bubnov variational formula corresponding to the bilinear form $\langle V\text{curl}\ u, \text{curl}\ v\rangle_{\Sigma}$ can be solved using the methods already described in Chapter 3 as will be shown more explicitly. The other inner product $\langle \frac{\partial}{\partial t} V(u\vec{n}), v\vec{n}\rangle_{\Sigma}$ however, requires some further consideration and can not be solved with the previous findings.

Let us subdivide (4.32) into two parts, the first part being $\langle V\text{curl}\ u, \text{curl}\ v\rangle_{\Sigma}$ and the second $\langle \frac{\partial}{\partial t} V(u\vec{n}), v\vec{n}\rangle_{\Sigma}$. For the first part we will define an element matrix, given two elements e_1 and e_2, originating from a triangulation satisfying the right-angled triangulation specified in Section 3.5.1. The vertices of the element e_i are represented by v_1^i, v_2^i, v_3^i, and corresponding *barycentric (linear) basis functions* $\lambda_1^i,\ \lambda_2^i,\ \lambda_3^i$, satisfying $\lambda_j^i(v_k) = \delta_{j,k}$ and $\lambda_j^i(x) \geq 0,\ \forall j,\ \forall x \in e_i$. Let E_k be the edge of the element such that $v_k \notin E_k$. Let n_k denote the outward normal vector at edge E_k. Then, for a general mesh, the barycentric basis functions are given in (4.33)-(4.35).

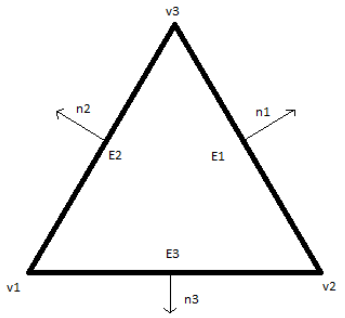


Figure 4.8: Example of an element and corresponding notation.

Every element lies in a two dimensional plane in the space-time lateral boundary. Given a rectangular domain we can assume without loss of generality that the barycentric basis functions are depending on only two variables: one spatial and one temporal direction. Every vertex can in that case thus be written as $v_j^i := (v_j^i(x), v_j^i(t))^T$, $j = 1,2,3$. The resulting basis functions are presented in the second part of (4.33)-(4.35).

$$\lambda_1^i(\mathbf{x}) := -\frac{|E_1^i|}{2|e_i|}(\mathbf{x} - v_2^i)n_1^i \overset{2D}{=} \frac{1}{2|e_i|}(\mathbf{x} - v_2^i) \cdot \begin{pmatrix} v_2^i(t) - v_3^i(t) \\ v_3^i(x) - v_2^i(x) \end{pmatrix} \mathbb{1}_{e_i}, \tag{4.33}$$

$$\lambda_2^i(\mathbf{x}) := -\frac{|E_2^i|}{2|e_i|}(\mathbf{x} - v_3^i)n_2^i \overset{2D}{=} \frac{1}{2|e_i|}(\mathbf{x} - v_3^i) \cdot \begin{pmatrix} v_3^i(t) - v_2^i(t) \\ v_1^i(x) - v_3^i(x) \end{pmatrix} \mathbb{1}_{e_i}, \tag{4.34}$$

$$\lambda_3^i(\mathbf{x}) := -\frac{|E_3^i|}{2|e_i|}(\mathbf{x} - v_1^i)n_3^i \overset{2D}{=} \frac{1}{2|e_i|}(\mathbf{x} - v_1^i) \cdot \begin{pmatrix} v_1^i(t) - v_2^i(t) \\ v_2^i(x) - v_1^i(x) \end{pmatrix} \mathbb{1}_{e_i}. \tag{4.35}$$

The gradient of each of these basis functions can be directly deduced from (4.33)-(4.35) and are given by:

$$\nabla\lambda_1^i = -\frac{|E_1^i|}{2|e_i|}n_1^i \overset{2D}{=} \frac{1}{2|e_i|} \begin{pmatrix} v_2^i(t) - v_3^i(t) \\ v_3^i(x) - v_2^i(x) \end{pmatrix} \mathbb{1}_{e_i}, \tag{4.36}$$

$$\nabla\lambda_2^i = -\frac{|E_2^i|}{2|e_i|}n_2^i \overset{2D}{=} \frac{1}{2|e_i|} \begin{pmatrix} v_3^i(t) - v_2^i(t) \\ v_1^i(x) - v_3^i(x) \end{pmatrix} \mathbb{1}_{e_i}, \tag{4.37}$$

$$\nabla\lambda_3^i = -\frac{|E_3^i|}{2|e_i|}n_3^i \overset{2D}{=} \frac{1}{2|e_i|} \begin{pmatrix} v_1^i(t) - v_2^i(t) \\ v_2^i(x) - v_1^i(x) \end{pmatrix} \mathbb{1}_{e_i}. \tag{4.38}$$

Now, as we use $S_h^1(\Sigma_h)$ as the test and trial space of the Galerkin-Bubnov approximation of the variational problem, the barycentric basis functions will be elements of the test and trial space. When dealing with a rectangular domain Ω, the outward normal vector in the curl operator will be zero for one of the variables. In general we can write:

$$\text{curl } \lambda_j^i = \begin{pmatrix} n_1^\Sigma \begin{pmatrix} 1 & 0 & 0 \end{pmatrix} \cdot \nabla\lambda_j^i - n_2^\Sigma \begin{pmatrix} 0 & 1 & 0 \end{pmatrix} \cdot \nabla\lambda_j^i \end{pmatrix} \mathbb{1}_{e_i} =: \alpha_j^i \mathbb{1}_{e_i}, \tag{4.39}$$

which equals a piecewise constant function, as normal vectors in scalars in the two dimensional plane. We therefore obtain the local 3×3 element matrix $R_{n,m}$ for e_n, e_m:

$$R_{n,m}[i,j] = \alpha_i^n \alpha_j^m \langle \mathbf{V}_h \mathbb{1}_{e_n}, \mathbb{1}_{e_m} \rangle_\Sigma. \tag{4.40}$$

The inner product in (4.40) can be approximated by following the procedure described in Section 3.5.

For the second part of (4.32), we make use of the distributional identity [1]:

$$\langle \frac{\partial}{\partial t} V(u\vec{n}^\Sigma), v\vec{n}^\Sigma \rangle_\Sigma = \langle V(u\vec{n}), \frac{\partial}{\partial t} v\vec{n} \rangle_\Sigma. \tag{4.41}$$

Given the previously defined basis functions, we can directly obtain an expression for the derivative of λ with respect to time

$$\frac{\partial \lambda_j^i}{\partial t} = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix} \nabla\lambda_j^i.$$

Again, the outward normal vectors for each element are constant within the element, we can therefore define the constant

$$\beta_j^i := \vec{n}_i^\Sigma \begin{pmatrix} 0 & 0 & 1 \end{pmatrix} \nabla\lambda_j^i,$$

where $\vec{n}_i^\Sigma$ denotes the outward normal vector in element e_i. The local 3 element matrix of the second part, $R2_{nm}$, given elements $e_n,\ e_m$, now reads

$$R2_{nm}[i,j] := \vec{n}_n^\Sigma \cdot \beta^m \langle V\lambda_i^n, \mathbb{1}_{e_m} \rangle_\Sigma. \tag{4.42}$$

The inner product in (4.42) is approximated with the use of Gaussian quadrature.

Regularisation

Consider the bilinear form of the hypersingular operator as stated in (4.31). Let f, g be some non-zero constants with supp $f =$ supp $g = \Sigma$. We have that $f, g \in H^{\frac{1}{2},\frac{1}{4}}(\Sigma)$. Inserting these functions into the bilinear form (4.31) yields

$$\langle Wf, g \rangle_\Sigma = fg \int_\Sigma \partial_{n_x} \int_\Sigma \partial_{n_y} G(x-y, t-s)\ d\sigma_y ds dt d\sigma_x.$$

We consider the following theoretical result regarding coercivity of the hypersingular operator [1, 16].

Theorem 4.8 ([1, Thm.4.16], [16, Thm.2.2.8]). *The hypersingular operator* $W : H^{\frac{1}{2},\frac{1}{4}}(\Sigma) \to H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma)$ *defined in* (2.12) *is an isomorphism and*

$$\exists c > 0,\ \langle W\phi, \phi \rangle_\Sigma \geq c \|\phi\|^2_{H^{\frac{1}{2},\frac{1}{4}}(\Sigma)},\ \forall \phi \in H^{\frac{1}{2},\frac{1}{4}}(\Sigma).$$

From Theorem 4.8, we directly see that $\langle Wf, f \rangle_\Sigma > 0$, since $f \neq 0$ almost everywhere (a.e). This is expected behaviour as, it is intuitively clear that some examples exist of the heat kernel with respect to the normal derivative which will not be zero almost a.e. nor additively invertible along the temporal axis.

If we were, however, to insert these functions into the right hand-side of the integration by parts formula (4.32), we obtain the following

$$\langle V\text{curl } f, \text{curl } f \rangle_\Sigma - \langle \frac{\partial}{\partial t} V(f\vec{n}^\Sigma), f\vec{n}^\Sigma \rangle_\Sigma \overset{(4.41)}{=} \langle V\text{curl } f, \text{curl } f \rangle_\Sigma - \langle V(f\vec{n}^\Sigma), \frac{\partial}{\partial t} f\vec{n}^\Sigma \rangle_\Sigma = 0. \tag{4.43}$$

So the bilinear form of the hypersingular operator, described by the integration by parts formula has a one dimensional kernel due to the presence of constant functions in $H^{\frac{1}{2},\frac{1}{4}}(\Sigma)$. Regularisation is thus required. We denote bilinear forms $b(u,v) := \langle Wu, v\rangle_\Sigma$ and $\tilde{b}(u,v) := \langle V\mathsf{curl}\ u, \mathsf{curl}\ v\rangle_\Sigma - \langle V(u\vec{n}^\Sigma), \frac{\partial}{\partial t} v\vec{n}^\Sigma\rangle_\Sigma$. The "analytical" regularisation is then given by

$$b(u,v)_{req} := \tilde{b}(u,v) + b(\mathbb{1}_\Sigma, \mathbb{1}_\Sigma). \tag{4.44}$$

In practice however, for the Calderon preconditioner, the following heuristic regularisation suffices [9]:

$$b(u,v)_{req_h} := \tilde{b}(u,v) + C\langle u, \mathbb{1}_\Sigma\rangle_\Sigma \langle v, \mathbb{1}_\Sigma\rangle_\Sigma, \tag{4.45}$$

with $C \in \mathbb{R}$. The regularising term in the heuristic regularisation can be rewritten as $Cr \cdot r^T$ with $r := M_{mass} \cdot \mathbb{1}$ and M_{mass} the mass matrix corresponding to a piecewise linear test-, and trial space.

4.4. Domain Decomposition

We now turn our attention to another mathematical tool that will be relevant for the parallelization of the system of linear equations (4.1) arising from the Dirichlet problem given by (3.1) on domain $Q := \Omega \times [0,T]$. By decomposing this domain into N_d sub-domains $Q_i := \Omega_i \times [t_{i_1}, t_{i_2}]$, such that

$$\bigcup_{i=1}^{N_d} Q_i = Q,$$

and Ω_i is a Lipschitz domain for all $i = [1, \dots N_d]$, we obtain N_d boundary value sub-problems which we strive to solve in parallel.

Remark. *Note that it is not a necessary condition for the decomposition to be disjoint in its sub-domains, i.e, to be a partition of the domain Σ. The necessity of this assumption varies between choice of solver.*

Schwarz came up with an iterative method to solve the Dirichlet problem in the union of its sub-domains, called *Schwarz alternating method* [17]. In the space time setting, the algorithm is described with the use of the a simple example decomposition of the two dimensional space-time domain $Q := (x_1, x_2) \times (t_0, t_1)$. The decomposition is represented in Figure 4.9.

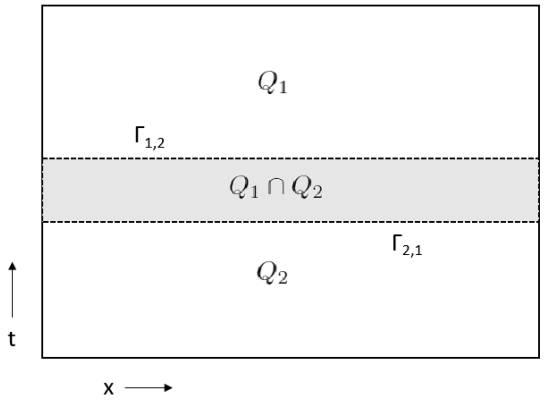


Figure 4.9: Example of a 1D space-time decomposition

As is clear from the Figure, the domain is decomposed in overlapping regions. The alternating Schwarz algorithm given this decomposition now reads:

Algorithm 2: Schwarz alternating method

Result: $\mathbf{u}_k$, approximation of the solution u, of the Dirichlet problem (3.1)
Define initial guess $\mathbf{u}_0$;
while *No convergence* **do**
  for *i=1,2* **do**
    Solve $\frac{\partial u}{\partial t} + \frac{\partial^2 u}{\partial x^2} = f$ on Q_i, with $u = u_{i,j}$ in $\Gamma_{i,j}$;
    Update u values on $\Gamma_{j,i} \forall, j$
  end
end

Note that the decomposition $Q = \cup_{i=1}^{N_D} Q_i$ induces a decomposition of the lateral boundary $\Sigma = \cup_i^{N_d} \Sigma_i$, with $\Sigma_i = \partial Q_i \setminus \Gamma_{i,j}$, $j = 1, \ldots, N_d$. The algorithm is not stated in discrete form, which is actually required because the original algorithm was developed for elliptic boundary value problems. When transferring to space-time boundary element setting, the weakly singular operator, for the heat equation, is elliptic [15, 3].

4.4.1. Multiplicative Schwarz

Consider the weakly singular matrices $\mathbf{V}_h^{(1)}$ and $\mathbf{V}_h^{(2)}$ arising from the Dirichlet problem (3.1), restricted to Σ_1 and Σ_2 respectively. Depending on how the space-time domain is decomposed, either an unknown boundary condition is added, or a unknown initial condition. Figure 4.9 represents a decomposition in which a unknown initial condition arises. By switching for each sub-problem to the indirect formulation of the first kind (3.8), and by assuming no forcing is present, we obtain

$$V q_i = g_i - M_0 u_0^i, \quad q_i \in H^{-\frac{1}{2},-\frac{1}{4}}(\Sigma_i), \tag{4.46}$$

which will yield the system of linear equations

$$\mathbf{V}_h^{(i)} \mathbf{q}_i = \mathbf{F}^i - \mathbf{M}^i, \quad i = 1, \ldots, N. \tag{4.47}$$

In (4.47), $\mathbf{M}^i$ is the vector arising from $\langle M_0 u_0^1, \phi_i^{\Sigma,0} \rangle_\Sigma$. Note that, if the decompositions follows along the lines of Figure 4.9, $M_0 u_0^1$ is known, the influence of the unknown boundary $\Gamma_{1,2}$ vanishes on this domain by choosing the test and trial function such that this is true. $M_0 u_0^2$, however, is not known, and depends on the value $u_0^2 = u|_{\Gamma_{2,1}}$. If we were to make an initial guess of q on the entire domain Σ, which we denote as q^0, we immediately obtain a guess on the unknown boundary as well by the identity

$$u = \mathcal{K}_1 q^0.$$

Let us denote $\mathbf{V}_h$ as the weakly singular Galerkin matrix on the lateral boundary Σ. The Multiplicative Schwarz procedure in space time setting is now given by Algorithm 3.

Algorithm 3: Schwarz Multiplicative method in space time setting

Result: $\mathbf{u}_k$, approximation of the solution $\mathbf{q}$, of the system of linear equations (4.1).
Define initial guess $\mathbf{q}^0$;
while *No convergence* **do**
  $\mathbf{q}^{k+\frac{1}{2}} = \mathbf{q}^k + \begin{pmatrix} (\mathbf{V}_h^{(1)})^{-1} & 0 \\ 0 & 0 \end{pmatrix} (\mathbf{F} - \mathbf{V}_h \mathbf{q}^k)$;
  $\mathbf{q}^{k+1} = \mathbf{q}^{k+\frac{1}{2}} + \begin{pmatrix} 0 & 0 \\ 0 & (\mathbf{V}_h^{(2)})^{-1} \end{pmatrix} \left(\mathbf{F} - \mathbf{V}_h q^{k+\frac{1}{2}}\right)$;
end

Remark. *This algorithm can be extended to a decomposition of N_d multiple sub-domains, where the vector* $\mathbf{q}$ *will be updated N_d times each loop.*

It is clear that the steps in the loop of the algorithm are serial and therefore, the multiplicative Schwarz method has a low degree of parallelism. For that reason, we move to another algorithm called the *Restricted Additive Schwarz method* (RAS).

4.4.2. Restricted Additive Schwarz

To improve parallelism we strive to solve for the different sub-domains simultaneously. The Restricted Additive Schwarz procedure is an convergence-wise optimised version of the original Additive Schwarz procedure. The original Schwarz method makes use of the so-called *Restriction operators* $\mathbf{R}_i$, which restrict a matrix to its sub-domain Σ_i by

$$\mathbf{V}_h^{(i)} = \mathbf{R}_i \mathbf{V}_h \mathbf{R}_i^T.$$

We thus have $(\mathbf{V}_h^i)^{-1} = (R_i \mathbf{V}_h R_i^T)^{-1}$. This results in the following algorithm [17]:

Algorithm 4: Additive Schwarz procedure in space time setting iterations

Result: $\mathbf{u}_k$, approximation of the solution $\mathbf{q}$, of the system of linear equations (4.1).
Define initial guess $\mathbf{q}^0$;
for *Amount of iterations* **do**
  for *i=1,…*N_d **do**
    $\delta_i = \mathbf{R}_i^T (\mathbf{V}_h^{(i)})^{-1} \mathbf{R}_i (\mathbf{F} - \mathbf{V}_h \mathbf{q}^k)$;
  end
  $\mathbf{q}^{k+1} = \mathbf{q}^k + \sum_{i=1}^N \delta_i$;
end

Now, the procedure in the for loop can be performed in parallel. However, this procedure does not need to converge in general, and is therefore mainly used as a preconditioner [17]. By considering the approximation of $\mathbf{q}$ in each subdomain to be of a different entity, we finally arrive at the Restricted Additive Schwarz method. Let us denote the non-overlapping part of a sub-domain Σ_i, by $\tilde{\Sigma}_i$ and their corresponding matrices by $\tilde{\mathbf{V}}_h^i$. Next we can define the non overlapping restriction operators $\tilde{R}_i$ by

$$\tilde{\mathbf{V}}_h^{(i)} = \tilde{\mathbf{R}}_i \mathbf{V}_h \tilde{\mathbf{R}}_i^T.$$

Now we define the Restricted Additive Schwarz algorithm by [5]:

Algorithm 5: Restricted Additive Schwarz procedure in space time setting iterations

Result: $\mathbf{u}_k$, approximation of the solution $\mathbf{q}$, of the system of linear equations (4.1).
Define initial guess $\mathbf{q}^0$;
for *Amount of iterations* **do**
  for *i=1,…N* **do**
    $\delta_i = \tilde{\mathbf{R}}_i^T (\tilde{\mathbf{V}}_h^{(i)})^{-1} \mathbf{R}_i (\mathbf{F} - \mathbf{V}_h \mathbf{q}^k)$;
  end
  $\mathbf{q}^{k+1} = \mathbf{q}^k + \sum_{i=1}^N \delta_i$;
end

This procedure does converge as an iterative method [5] and is therefore an interesting candidate to use for parallel implementations.

4.4.3. Schwarz procedures as preconditioners

Apart from being a stand-alone basic iterative method, RAS can also be applied as a preconditioner for GMRES. Since the preconditioner is not symmetric, it is sub-optimal for CG. Given N_d subdomains, the preconditioner is defined as [4]

$$\mathbf{M}_{ras}^{-1} := \sum_{i=1}^{N_d} \tilde{\mathbf{R}}_i^T (\tilde{\mathbf{V}}_h^{(i)})^{-1} \mathbf{R}_i.$$

The efficiency of the preconditioner depends on the choice of sub-domains. The preconditioner is tested for a few example domain decompositions in the next section.

4.5. Preconditioning results

The Dirichlet problem (3.1) is solved using the meshes defined in Section 3.6. Note that these meshes are uniform and results might differ for quasi-uniform meshes. Two preconditioning techniques for GMRES are compared, Calderon preconditioning and Restrictive Additive Schwarz (RAS) preconditioning. Also, RAS is considered as a Basic Iterative Method (BIM). Each iteration of RAS as BIM is highly parallelizable and, for that reason, comparisons of iteration counts between (the Krylov subspace method) GMRES and RAS should be dealt with cautiously.

In the case of the Calderon preconditioner, the approach proposed by Stevenson and van Venetië [23] is used. In this approach, choice of constant β is required, as stated in (4.28). They use $\beta = 1.25$ as a (near) optimal choice, which consequently will be the choice in this thesis as well.

The hypersingular matrix is obtained following the procedure described in Section 4.3.2. The hypersingular matrix is regularised heuristically, which also requires the choice of a scalar C as can be seen in (4.45). In this case, C is taken to be 0.1. In the numerical experiments, this value provided a reduction of iteration count, but need not be the optimal choice of C.

In the case of RAS, both as preconditioner and BIM, the domain is decomposed in (near) equal parts. When using overlap, the overlap between two domains is given as a percentage of area of the overlapping part compared to the total area of the sub-domains.

The right hand-side of 4.1 is taken at random. The numerical tests are done in Python with a seed of 10. The initial guess of the solution u^0 is taken to be the zero vector. The tolerance is set to 1E-5. The results are presented in Table 4.1.

Table 4.1: Table of iteration count given a preconditioner $\mathbf{M}^{-1}$. $\mathbf{M}^{-1} = I$ is equivalent to GMRES without preconditioning. The meshes are ordered on their level of refinement. The 's' behind the refinement level refers to the "scaled" mesh as presented in Figure 3.8(b). RAS[n,m%] denotes the RAS preconditioner given a domain decomposed in n subdomains and with m% overlap. No 'm' implies no overlap.

Mesh\Iteration count	Preconditioner M^-1						
	I	Cv	RAS[2]	RAS[2,5%]	RAS[2,10%]	RAS[4]	RAS[8]
32	13	11	2	2	2	4	12
32s	14	12	2	2	2	4	10
128	20	18	16	16	16	19	17
128s	19	15	10	10	10	13	12
512	38	20	18	18	17	20	19
512s	29	16	12	12	13	13	14

The results of the RAS method as BIM are given in Table 4.2.

Table 4.2: Iteration count given a RAS[n,m] basic iterative method. RAS[n,m] represents a decomposition in n equal-size sub-domains with m% overlap. The meshes are ordered on their level of refinement. The 's' behind the refinement level refers to the "scaled" mesh as presented in Figure 3.8(b).

Mesh\Iteration count	Decomposition				
	RAS[2]	RAS[2,5%]	RAS[2,10%]	RAS[4]	RAS[8]
32	2	2	2	4	25
32s	2	2	2	4	12
128	25	22	22	81	92
128s	13	13	13	17	20
512	30	29	28	126	139
512s	18	18	18	22	25

Both Table 4.1 and 4.2 seem to indicate that the contribution of implementing overlap is negligible. As overlap increases matrix size, it seems to be not an effective method. To get a better insight at the behaviour of the Calderon preconditioner, we take a look at the spectra of several matrices related to the preconditioning process. First, we take a look at the absolute values of the eigenvalues of the matrices, given different levels of refinement. This is visually presented in Figure 4.10.

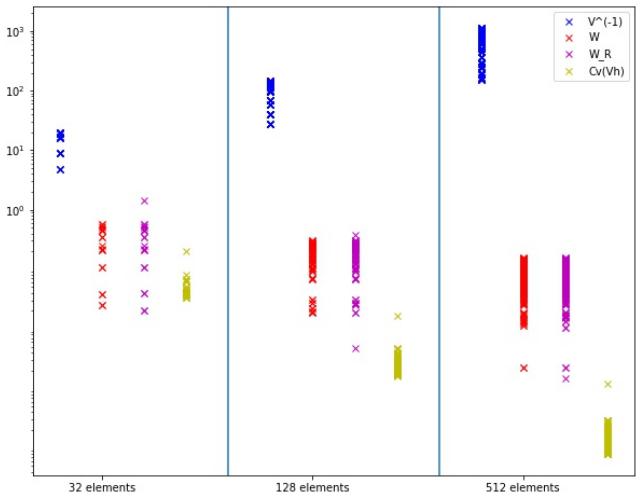
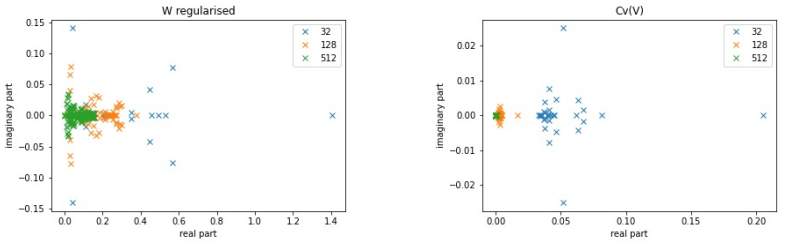


Figure 4.10: Absolute values of the eigenvalues of the following matrices: The inverse of the weakly singular matrix (V^-1), The hypersingular matrix without the zero eigenvalue (W), the regularized hypersingular matrix (W_R) and the matrix-matrix product of the Calderón preconditioner and the weakly singular matrix $Cv(Vh)$.

As expected, the clusters with regard to the weakly singular shift and grows with the refinement level. The behaviour of the spectrum of the hypersingular and preconditioned weakly singular is a bit more puzzling. Therefore, we take a closer look at the eigenvalues of these matrices as shown in Figure 4.11.

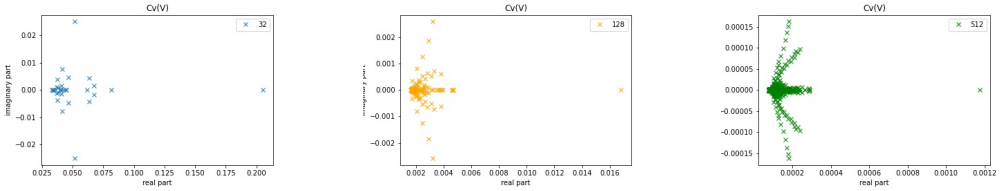


(a) Spectrum of Regularised hypersingular matrix

(b) Spectrum of preconditioned weakly singular

Figure 4.11: Spectra of hyper singular and preconditioned weakly singular matrices, given different levels of refinement.

The spectra seem to converge to some point near zero. As this is still unexpected behaviour we take another closer look at the preconditioned weakly singular.



(a) 32 elements

(b) 128 elements

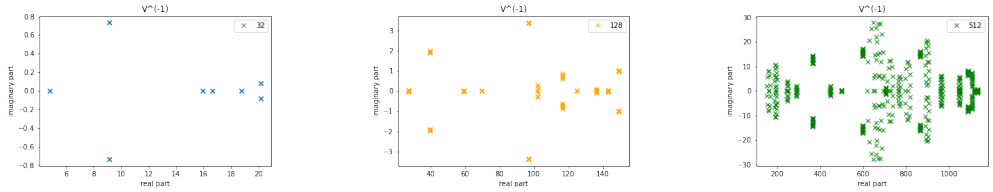
(c) 512 elements

Figure 4.12: Spectra of the preconditioned weakly singular matrices, given different levels of refinement. Note that the axis differ between graphs.

Figure 4.11 reveals that the clustering of the spectra is as expected. This explains why the iteration

count is low, in spite of the fact that the eigenvalues of $C_v(\mathbf{V}_h)$ shift. In order to assess if we are in a pre-asymptotic region, we take a look at the spectral condition number of each refinement.

For reference, we present the spectra of the inverse weakly singular matrices as well in Figure 4.13.



(a) 32 elements **(b)** 128 elements **(c)** 512 elements

Figure 4.13: Spectra of the inverted weakly singular matrices, given different levels of refinement. Note that the axis differ between graphs.

Table 4.3: Spectral condition number for each level of refinement, for different operators. V^-1 denotes the inverse of the weakly singular matrix, W_R the regularised hypersingular matrix and Cv(V) the preconditioned weakly singular.

Operator\Mesh size	Spectral condition number		
	32	128	512
V^-1	107	335	924
W_R	139.000	84.188	187.072
Cv(V)	58	224	283

As can be seen in the last row of Table 4.3, the rate of growth of the condition number of preconditioned weakly singular seems to decrease, which suggests asymptotic behaviour.

4.6. Parallel GMRES implementation

After obtaining several preconditioning strategies, the next step is to solve the (preconditioned) system of linear equations (4.4) in parallel. The matrix structure and density might change when applying preconditioning compared to the structures obtained in Section 4.1. In the case of Calderón preconditioning, the matrix becomes fully dense. When applying Restricted Additive Schwarz preconditioning, the matrix-matrix product structure more or less remains the same compared to the original matrix. Especially in the case of dense matrices, parallelization can cause significant reduction of computational time [17]. For that reason, we will use Calderon preconditioning and its structure as a basis of our parallelization strategy. Message Passing Interface (MPI) software is used as a means to parallelize the GMRES method. Python is used as a wrapper for MPI using the package ”mpi4py”.

4.6.1. Least squares problem

One of the practical implementation issues of Algorithm 1 lies in the fact it does not provide an explicit approximate solution of $\mathbf{q}_k$ at each step. As a result the algorithm will not stop when the approximation $\mathbf{q}_m$ satisfies a sufficiently small relative error, with $m < k$. We will define the relative error as

$$\mathbf{r}_k = \frac{\|\mathbf{V}_h\mathbf{q}_k - \mathbf{f}\|_2}{\|\mathbf{f}\|_2}. \tag{4.48}$$

There is an elegant way proposed by Saad ([17], Prop. 6.9) to solve this practical issue, which is related to the way the least squares problem $\min_y \|\beta\mathbf{e}_1 - \bar{\mathbf{H}}_k y\|_2$ is solved. For sake of completeness, the method proposed by Saad is now presented.

Let us consider the Hessenberg matrix $\bar{\mathbf{H}}_m$ obtained at step $m \leq k$ of the GMRES method. The matrix will have the following structure:

$$\bar{\mathbf{H}}_m = \begin{pmatrix} h_{1,1} & h_{1,2} & \dots & h_{1,m} \\ h_{2,1} & h_{2,2} & \dots & h_{2,m} \\ 0 & h_{3,2} & \dots & h_{3,m} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & h_{m+1,m} \end{pmatrix}. \tag{4.49}$$

We will transform this matrix into upper triangular form by premultiplying the matrix with *rotation matrices* of size $(m+1) \times (m+1)$, $\mathcal{R}_i$, defined as [17]:

$$\mathcal{R}_i = \begin{pmatrix} 1 & & & & & & & \\ & \ddots & & & & & & \\ & & 1 & & & & & \\ & & & c_i & s_i & & & \\ & & & -s_i & c_i & & & \\ & & & & & 1 & & \\ & & & & & & \ddots & \\ & & & & & & & 1 \end{pmatrix},$$

where $\mathcal{R}_i[i,i] = \mathcal{R}_i[i+1,i+1] = c_i$ and $\mathcal{R}_i[i,i+1] = -\mathcal{R}_i[i+1,i] = s_i$. Before we give the expression of c_i and s_i, we define

$$\mathcal{Q}_j := \mathcal{R}_j \mathcal{R}_{j-1} \dots \mathcal{R}_1, \quad 1 \le j \le m,$$

$$\bar{\mathbf{H}}_m^j := \mathcal{Q}_j \bar{\mathbf{H}}_m.$$

and

$$\bar{\mathbf{g}}_m := \mathcal{Q}_m(\beta \mathbf{e}_1) = (g^{(1)}, g^{(2)}, \dots, g^{(m+1)})^T.$$

Let us denote the element of matrix $\bar{\mathbf{H}}_m^j$ at row r and column s by $h_{r,s}^{(j)}$. Now we can define s_i and c_i as

$$s_i := \frac{h_{i+1,i}}{\sqrt{(h_{i,i}^{(i-1)})^2 + h_{i+1,i}^2}},$$

and

$$c_i := \frac{h_{i,i}^{(i-1)}}{\sqrt{(h_{i,i}^{(i-1)})^2 + h_{i+1,i}^2}}.$$

Using these definitions we have that $c_i^2 + s_i^2 = 1$ and $\bar{H}_m^m$ becomes an upper triangular matrix. Since Q_m is unitary [17], we have

$$\min_{\mathbf{y}} \|\beta \mathbf{e}_1 - \bar{\mathbf{H}} y\|_2 = \min_{\mathbf{y}} \|\bar{\mathbf{g}}_m - \bar{\mathbf{H}}_m^m \mathbf{y}\|_2. \tag{4.50}$$

This leads to the following useful proposition [17]:

Proposition 4.9 ([17, Prop.6.9])**.** *Let H_m^m be the $m \times m$ matrix obtained by removing the last row of $\bar{H}_m^m$. Similarly let g_m be the m-dimensional vector obtained by deleting the last element of $\bar{g}_m$. Then, the following holds*

(i) The vector $\mathbf{y}_m$ which minimises $\|\beta \mathbf{e}_1 - \bar{\mathbf{H}}_m \mathbf{y}\|_2$ is given by

$$\mathbf{y}_m = (\mathbf{H}_m^m)^{-1} \mathbf{g}_m.$$

(ii) The residual vector at step m satisfies

$$\mathbf{M}^{-1}(\mathbf{F} - \mathbf{V}_h\mathbf{q}_m) = \mathbf{V}_{m+1}(\beta(\mathbf{e}_1) - \bar{\mathbf{H}}_m\mathbf{y} = \mathbf{V}_{m+1}\mathcal{Q}_m^T(\mathbf{g}^{(m+1)}\mathbf{e}_{m+1}),$$

and as a result,

$$\|\mathbf{M}^{-1}(\mathbf{F} - \mathbf{V}_h\mathbf{q}_m)\|_2 = |\mathbf{g}^{(m+1)}|.$$

Proof: The proof will follows for a large part the proof given by Saad in proposition 6.9 [17]. Some parts are deliberately left out as they do not contribute much to this thesis and other parts are elaborated.

To prove (i), we make use of the fact that $\mathcal{Q}_m$ is a unitary operator and therefore isometric. We hence have

$$\begin{aligned}\|\beta\mathbf{e}_1 - \bar{\mathbf{H}}_m\mathbf{y}\|_2^2 = \|\mathcal{Q}_m(\beta\mathbf{e}_1 - \bar{\mathbf{H}}_m y)\|_2^2 = \|\bar{\mathbf{g}}_m - \bar{\mathbf{H}}_m^m y\|_2^2 \\ = |g^{(m+1)}|^2 + \|\mathbf{g}_m - \mathbf{H}_m^m y\|_2^2.\end{aligned} \tag{4.51}$$

This norm (4.51) is minimised if the second term in the last equality vanishes. This is achieved if $\mathbf{H}_m^m y = g_m \Rightarrow \mathbf{y} = (\mathbf{H}_m^m)^{-1}\mathbf{g}_m$, proving the first statement.

As for the second statement (ii), we take a closer look at the inner-loop of the GMRES algorithm (Algorithm 1). It can be quickly seen that we have

$$\mathbf{M}^{-1}\mathbf{V}_h v_j - \sum_{i=1}^{j} h_{i,j}v_i = w = v_{j+1}h_{j+1,j}. \tag{4.52}$$

From (4.52) we can deduce that

$$\mathbf{M}^{-1}\mathbf{V}_h v_j = \sum_{i=1}^{j+1} h_{i,j}v_i \Rightarrow \mathbf{M}^{-1}\mathbf{V}_h\mathbf{V}_m = \mathbf{V}_{m+1}\bar{\mathbf{H}}_m, \tag{4.53}$$

with $\mathbf{V}_m$ defined as in Algorithm 1. Since the approximate solution $\mathbf{q}_m$ is defined as $\mathbf{q}_m = \mathbf{q}_0 + \mathbf{V}_m\mathbf{y}$, we have

$$\begin{aligned}\mathbf{M}^{-1}\mathbf{F} - \mathbf{M}^{-1}\mathbf{V}_h\mathbf{q}_m = \mathbf{M}^{-1}\mathbf{F} - \mathbf{M}^{-1}\mathbf{V}_h(\mathbf{q}_0 + \mathbf{V}_m\mathbf{y}) = \mathbf{r}^0 - \mathbf{V}_{m+1}\bar{\mathbf{H}}_m\mathbf{y} \\ = \beta v_1 - \mathbf{V}_{m+1}\bar{\mathbf{H}}_m y = \mathbf{V}_{m+1}(\beta(\mathbf{e}_1) - \bar{\mathbf{H}}_m\mathbf{y}),\end{aligned} \tag{4.54}$$

proving the first equality in (ii). The second equality quickly follows from the definition of $\bar{\mathbf{g}}_m$ and the fact that $\mathcal{Q}_m$ is unitary:

$$\mathbf{V}_{m+1}(\beta(\mathbf{e}_1) - \bar{\mathbf{H}}_m\mathbf{y}) = \mathbf{V}_{m+1}\mathcal{Q}_m^T\mathcal{Q}_m(\beta(\mathbf{e}_1) - \bar{\mathbf{H}}_m\mathbf{y}) = \mathbf{V}_{m+1}\mathcal{Q}_m^T(\bar{\mathbf{g}}_m - \bar{\mathbf{H}}_m^m\mathbf{y}).$$

As seen in (4.51), $\bar{g}_m - \bar{H}_m^m y$ is minimised with respect to the 2-norm if y annihilates all elements of $\bar{g}_m$ except for the last term. Hence

$$\mathbf{V}_{m+1}\mathcal{Q}_m^T(\bar{\mathbf{g}}_m - \bar{\mathbf{H}}_m^m\mathbf{y}) = \mathbf{V}_{m+1}\mathcal{Q}_m^T(g^{(m+1)}\mathbf{e}_{m+1}), \tag{4.55}$$

proving the second equality. As the matrix product of two orthonormal matrices remains orthonormal, we have

$$\|\mathbf{V}_{m+1}\mathcal{Q}_m^T(\bar{\mathbf{g}}_m - \bar{\mathbf{H}}_m^m\mathbf{y})\|_2 = \|\mathbf{V}_{m+1}\mathcal{Q}_m^T(g^{(m+1)}\mathbf{e}_{m+1})\|_2 = \|g^{(m+1)}\mathbf{e}_{m+1}\| = |g^{(m+1)}|,$$

which yields the desired result.

□

The results described in Proposition 4.9 show us that with using this routine, we can come up with a norm of the residual at each step of the GMRES algorithm and find a minimizer when a satisying

norm of residual is reached. The routine described in this section is therefore added to the GMRES algorithm. This results in the following algorithm

Algorithm 6: Left-preconditioned GMRES with stopping criterion

Result: $\mathbf{q}_k$, approximation of the solution $\mathbf{M}^{-1}\mathbf{V}_h\mathbf{q} = \mathbf{M}^{-1}\mathbf{F}$
Define initial guess $\mathbf{q}_0$;
Compute $\mathbf{r}^0 = \mathbf{M}^{-1}(\mathbf{F} - \mathbf{V}_h\mathbf{q}_0)$, $\beta = \|\mathbf{r}^0\|_2$ and $v_1 = \frac{\mathbf{r}_0}{\beta}$;
for *k=1,…restart* **do**
  for *j=1,…,k* **do**
    Compute $w := \mathbf{M}^{-1}\mathbf{V}_h v_j$ ← **Inner-loop 1**;
    for *i = 1,…,j* **do**
      $h_{i,j} := (w, v_i)$;
      $w := w - h_{i.j}v_i$;
    end
    Compute $h_{j+1,j} = \|w\|_2$ and $v_{j+1} = \frac{w}{h_{j+1,j}}$;
  end
  Define $\mathbf{V}_k := [v_1, \ldots, v_k]$, $\bar{\mathbf{H}}_k = \bar{\mathbf{H}}_k^0 = \{h_{i,j}\}_{1\leq i\leq j; 1\leq j\leq k}$;
  for *m=1,…,k* **do**
    define $\mathcal{R}_m$ ← **Inner-loop 2**;
    $\bar{\mathbf{H}}_k^m = \mathcal{R}_m\bar{\mathbf{H}}_k^{m-1}$;
  end
  Define $\mathcal{Q}_m := \mathcal{R}_m \ldots \mathcal{R}_i$, $\bar{\mathbf{g}}_m = \mathcal{Q}_m(\beta\mathbf{e}_1)$
  if $|\bar{\mathbf{g}}_m\mathbf{e}_{m+1}| < tol$ **then**
    $(\mathbf{H}_k^k)^T := (\bar{\mathbf{H}}_k^k)^T(\mathbf{e}_1, \ldots, \mathbf{e}_k)$;
    $\mathbf{g}_k := (\bar{\mathbf{g}}_k)^T(\mathbf{I}, \mathbf{0})^T$;
    $\mathbf{y}_k := (\mathbf{H}_k^k)^-1\mathbf{g}_k$;
    $\mathbf{q}_k := \mathbf{q}_0 + \mathbf{V}_k\mathbf{y}_k$;
  else
    Set $\mathbf{q}_0 := \mathbf{q}_{restart}$;
    Repeat algorithm;
  end
end

By storing the matrices $\mathbf{V}_m$, $\bar{\mathbf{H}}_m^m$ and $\mathcal{Q}_m$, the inner loops denoted by 1 and 2, will not be necessary. Instead only a single update of each matrix is required in that case. This speeds up the process significantly at the cost of needing extra memory. For the parallel implementations described in the upcoming sections, the matrices are assumed to be stored and overwritten at each step of Algorithm 6. This results in the final algorithm, Algorithm 7.

Algorithm 7: Left-preconditioned GMRES with stopping criterion with saved matrices

Result: $\mathbf{q}_k$, approximation of the solution $\mathbf{M}^{-1}\mathbf{V}_h\mathbf{q} = \mathbf{M}^{-1}\mathbf{F}$
Define initial guess $\mathbf{q}_0$;
Compute $\mathbf{r}^0 = \mathbf{M}^{-1}(\mathbf{F} - \mathbf{V}_h\mathbf{q}_0)$, $\beta = \|\mathbf{r}^0\|_2$ and $v_1 = \frac{\mathbf{r}_0}{\beta}$;
for *k=1,...restart* **do**
  Compute $w := \mathbf{M}^{-1}\mathbf{V}_h v_{k-1}$;
  for *i = 1,...,k* **do**
    $h_{i,j} := (w, v_i)$;
    $w := w - h_{i.j}v_i$;
  end
  Compute $h_{j+1,j} = \|w\|_2$ and $v_{j+1} = \frac{w}{h_{j+1,j}}$;
  Define $\mathbf{V}_k := [v_1, \ldots, v_k]$, $\bar{H}_k = \bar{\mathbf{H}}_k^0 = \{h_{i,j}\}_{1\leq i\leq j; 1\leq j\leq k}$;
  Define $\mathcal{R}_k$;
  if *k=1* **then**
    $\mathcal{Q}$=$\mathcal{R}_k$;
  else
    $\mathcal{Q}$= $\mathcal{R}_k\mathcal{Q}$;
  end
  $\bar{\mathbf{H}}_k^k = \mathcal{Q}_k\bar{\mathbf{H}}_k$;
  $\bar{\mathbf{g}}_m = \mathcal{Q}_k(\beta\mathbf{e}_1)$;
  if $|\bar{\mathbf{g}}_m\mathbf{e}_{m+1}| < tol$ **then**
    $(\mathbf{H}_k^k)^T := (\bar{\mathbf{H}}_k^k)^T(\mathbf{e}_1, \ldots, \mathbf{e}_k)$;
    $\mathbf{g}_k := (\bar{\mathbf{g}}_k)^T(\mathbf{I}, \mathbf{0})^T$;
    $\mathbf{y}_k := (\mathbf{H}_k^k)^-1\mathbf{g}_k$;
    $\mathbf{q}_k := \mathbf{q}_0 + \mathbf{V}_k\mathbf{y}_k$;
  else
    Set $\mathbf{q}_0 := \mathbf{q}_{restart}$;
    Repeat algorithm;
  end
end

4.6.2. Matrix vector multiplications

As stated in Section 4.2, one of the main operations in Algorithm 1, 6 and 7 are the matrix vector multiplications. At the start of each iteration step a vector w is defined as a matrix vector product. Also, the calculation of the residual and approximate solution, using rotation matrices, requires solving both matrix vector-, and matrix matrix multiplications. The latter multiplications will not be solved in parallel, as the dimensions of these matrices are relatively small, which in turn reduces the efficiency of parallelization and can even increase computation time.

Since the matrices are assumed to be dense, balanced load distribution among processes is relatively straightforward. Two popular parallel (Dense) matrix vector product procedures are introduced and tested: Row-wise data decomposition and checkerboard partitioning [7, 17].

Given p processes, a $n \times n$ matrix A and $n \times 1$ vector x, in the case of row-wise data decomposition, A is decomposed in p stripes of dimensions $n/p \times n$. Each process is assigned such a stripe and row-corresponding vector, of dimension $n \times 1$. An example of such a decomposition with $p = 4$ is visualised in Figure 4.14

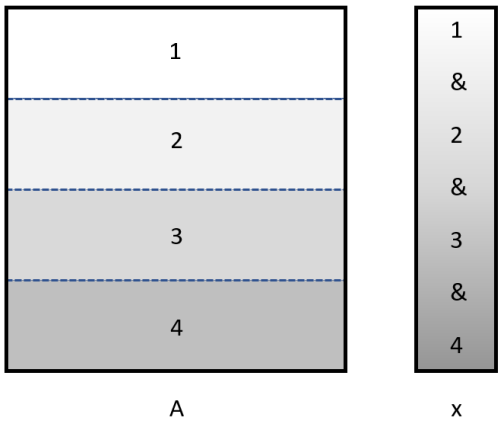


Figure 4.14: Example of row-wise decomposition of matrix A and vector x, given four processes.

Given such a decomposition, each process simultaneously calculates the reduced matrix vector product on its given domain. When the calculations are done, the processes either broadcast their result among the other processes, or send their result to an assigned root-process. When vertically concatenated, the gathered information equals the result of the matrix vector product. The algorithm looks as follows:

Algorithm 8: Parallel matrix vector using row-wise decomposition, given p processes

Result: Solution **y** for $\mathbf{y} = \mathbf{Ax}$
Partition matrix **A** and **x** row-wise in stripes $\mathbf{A}_i$ and **x** of equal length among each process;
for *i=1:p* **do**
  $\mathbf{y}_i$:= $\mathbf{A_i x}$
end
Gather $\mathbf{y}_i$

Another approach to load balancing is with use of so-called checkerboard partitioning. Given p processes, the $n \times n$ matrix is this time decomposed into blocks of size $n/\sqrt{p} \times n/\sqrt{p}$. The corresponding vector is partitioned in vectors of size $n/\sqrt{p} \times 1$. Each process is assigned to the vector with matching matrix columns to vector rows. For $p = 4$, an example decomposition is shown in Figure 4.15

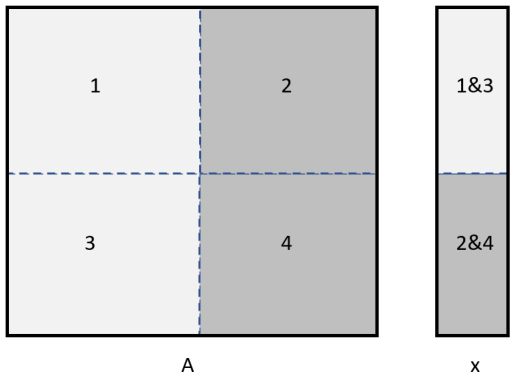


Figure 4.15: Example of checkerboard decomposition of matrix A and vector x, given four processes.

The checkerboard decomposition requires more communication between processes than in the case of row-wise partitioning. Every process computes the matrix-vector product on its given domain. Next, the processes which are on the same row, are added to each other. This requires communication between processes. As an optional step, the summed vectors can be vertically concatenated to obtain the original solution. Again, this requires communication between processes. When dealing with two

processes, the checkerboard decomposition coincided with either the row-wise decomposition or the column-wise decomposition. In these cases, column-wise decomposition is used, i.e. the matrix is divided into sections of size $n \times n/p$ with corresponding partioned vector of size n/p. The above partitioning (for $n \neq 2$), leads to the following algorithm:

Algorithm 9: Parallel matrix vector using checkerboard decomposition, given p processes

Result: Solution **y** for $\mathbf{y} = \mathbf{Ax}$
Partition matrix **A** and **x** in blocks $\mathbf{A}_i$ and $\mathbf{x}_i$ of size $n/\sqrt{p} \times n/\sqrt{p}$ and $n/\sqrt{p} \times 1$ respectively;
for *i=1:p* **do**
  $\mathbf{c}_i$:= $\mathbf{A}_i\mathbf{x}_i$;
  k= $\left\lfloor \frac{1}{\sqrt{p}} \right\rfloor$ mod $\sqrt{p}$;
  $\mathbf{y}_k$+= $\mathbf{c}_i$
end
Gather y_k

Note that, as was also not the case for $p = 2$, it is assumed that $\sqrt{p} \in \mathbb{N}$.

4.6.3. Dot products and vector norms

Parallelization of dot products and vector l^2-norms is relatively straightforward. Their algorithms are given below, given p processes.

Algorithm 10: Parallel dot product, given p processes

Result: Solution **y** for $\mathbf{y} = (w, v)$
Partition vector w and v in vectors w_i and v_i of size $n/p \times 1$;
for *i=1:p* **do**
  $\mathbf{y}_i$:= (w_i, v_i);
end
y = $\sum_{i=1}^{p} y_i$

Algorithm 11: Parallel l^2-norm, given p processes

Result: Norm $\|v\|_2$
Partition vector and v in vectors v_i of size $n/p \times 1$;
for *i=1:p* **do**
  y_i:= $\sum_{j=1}^{n/p} v_i(j)^2$;
end
$\|v\|_2$ = $\sqrt{\sum_{i=1}^{p} y_i}$

When considering algorithm 7, with using row-wise parallel matrix vector products for computing w, the inner product can be done directly in parallel as well, as the same partitioning can be used. This way, less communication between processes is required, which in turn implies a lower computation time. When applying a checkerboard decomposition for the construction of w, computing y_k in Algorithm 9, will suffice. Communication between processed will thus still remain necessary in this case. In the case of $p = 2$, a column-wise decomposition is used, in which case y_k equals the final solution vector.

4.6.4. Numerical results

The solvers coded using MPI software with a Python wrapper. The code is run using an Intel®Core™i7-7700HQ CPU @ 2.80GHz (4 core). The solution process is timed at the startup of the GMRES procedure, i.e. setup of preconditioners and the initial residual are not taken into account. The right-handside vector is created by pseudo-random generating an array in Python with seed set to 10. The initial guess for all solutions is set to zero. Every solution procedure is run 100 times for each different case, and the 100 different run times are gathered. To get a better understanding of the effect of the parallelization, another mesh refinement, consisting of 2048 elements, is added. Due to the increasing computational time of the weakly and hyper singular matrices for finer meshes, the order of quadrature for the 2048-

element mesh is lowered during assembly. The upcoming results for this mesh however, will remain relevant, as the overall structure of the matrices is preserved, and, use of (Calderon) preconditioning still decreases the iteration count significantly. The relative error is set to 1E-5. Given this error, the 2048-mesh converges in 42 and 23 iterations given no preconditioning and Calderon preconditioning respectively.

The obtained run-times can slightly vary when repeating the solution process given the same parameters. The mean run-time for each case is given in Table 4.4. The parallelization is performed with the use of row-wise decomposition.

Table 4.4: Mean run time in seconds **until convergence** of row-wise parallelized GMRES method with stopping criterion (Algorithm 7). The emphasised run times are the fastest times for a given mesh. The relative error tolerance is set to 1E-5.

Preconditioner	**No preconditioner**			**Calderon preconditioner**		
Mesh size/ no. of processes	**1**	**2**	**4**	**1**	**2**	**4**
32	0.0074	0.0101	0.0118	**0.0056**	0.0077	0.0095
128	0.0239	0.0229	0.031	**0.015**	**0.015**	0.0186
512	0.1139	0.0818	0.0948	0.0579	**0.0415**	0.0499
2048	0.9596	0.7549	0.5689	0.3266	0.2587	**0.2168**

From Table 4.4, we observe an overall better performance when using preconditioning. As the level of refinement (and the dimensions of the weakly-, and hyper singular matrices) increases, the efficiency of using multiple processes increases.

To get a better insight in the margin of errors in the obtained run times, for each level of refinement the preconditioned are visualised using a boxplot.

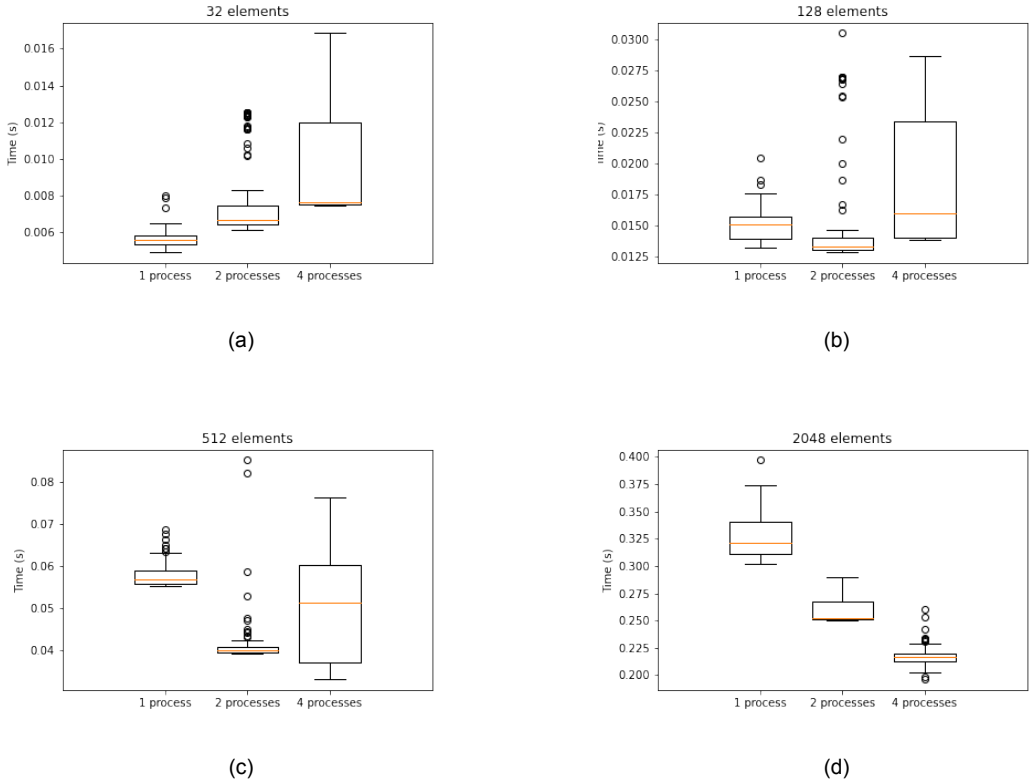


Figure 4.16: Boxplot of run times, for different levels of refinement. The orange line represents the median time. The lower and upper part of the "box" represent the 25th (Q_1) and 75th (Q_3) percentile respectively . The "whiskers" are obtained,with the use of interquartile range (I_{QR}), as follows: $Q1 - 1.5I_{QR}$ and $Q3 + 1.5I_{QR}$. Any measurement outside this range is considered an outlier and represented by a circle.

As can be seen in Figure 4.16, given a mesh of 2048 elements, near all registered run times with four parallel processes are lower than the run times obtained with less than four processes. This suggests parallelization becomes an increasingly important tool when the size of the matrices increases.

In Table 4.4, the run time until converges is denoted, given an predefined error tolerance. In order to see the effect of each iteration on the run time, we take a look at the parallel version of Algorithm 1, i.e. without stopping criterion. With the use of row wise decomposition, the run time of each mesh refinement is measured. The mean results are summarised in Table 4.5 - 4.8. The results are only obtained for Calderon preconditioned problems, as they are more relevant run time-wise than the non-preconditioned cases.

Table 4.5: Mean run times per iteration count vs total processes for GMRES without stopping criterion. Given a mesh of 32 elements and simulating the same experiment 100 times. The speedup column shows the ratio between times of one process and four processes

iterations/ no. of processes	1	2	4	Speedup
10	0.0040	0.0058	0.0072	0.56
20	0.0117	0.0169	0.0231	0.51
30	0.0252	0.0339	0.0510	0.49
40	-	-	-	-

Table 4.6: Mean run times per iteration count vs total processes for GMRES without stopping criterion. Given a mesh of 128 elements and simulating the same experiment 100 times. The speedup column shows the ratio between times of one process and four processes

iterations/ no. of processes	1	2	4	Speedup
10	0.0065	0.0069	0.0089	0.73
20	0.0215	0.0209	0.0292	0.74
30	0.0443	0.0440	0.0592	0.75
40	0.0738	0.0730	0.1093	0.67

Table 4.7: Mean run times per iteration count vs total processes for GMRES without stopping criterion. Given a mesh of 512 elements and simulating the same experiment 100 times.The speedup column shows the ratio between times of one process and four processes

iterations/ no. of processes	1	2	4	Speedup
10	0.0193	0.0144	0.0150	1.29
20	0.0575	0.0415	0.0430	1.34
30	0.1122	0.0796	0.0919	1.22
40	0.1926	0.1331	0.1520	1.26

Table 4.8: Mean run times per iteration count vs total processes for GMRES without stopping criterion. Given a mesh of 2048 elements and simulating the same experiment 100 times. The speedup column shows the ratio between times of one process and four processes

iterations/ no. of processes	1	2	4	Speedup
10	0.0856	0.0523	0.0457	1.87
20	0.2327	0.1411	0.1185	1.96
30	0.4440	0.2809	0.2246	1.98
40	0.7312	0.4337	0.3742	1.95

As stated in the description of the aforementioned tables, the denoted run times are the mean of repeating the same experiment a hundred times. With the use of (Monte-Carlo) bootstrapping ($N = 10.000$) a 95% confidence interval is created around the mean run times. In Figure 4.17, the iteration count versus time, given 95% confidence, are visualised.

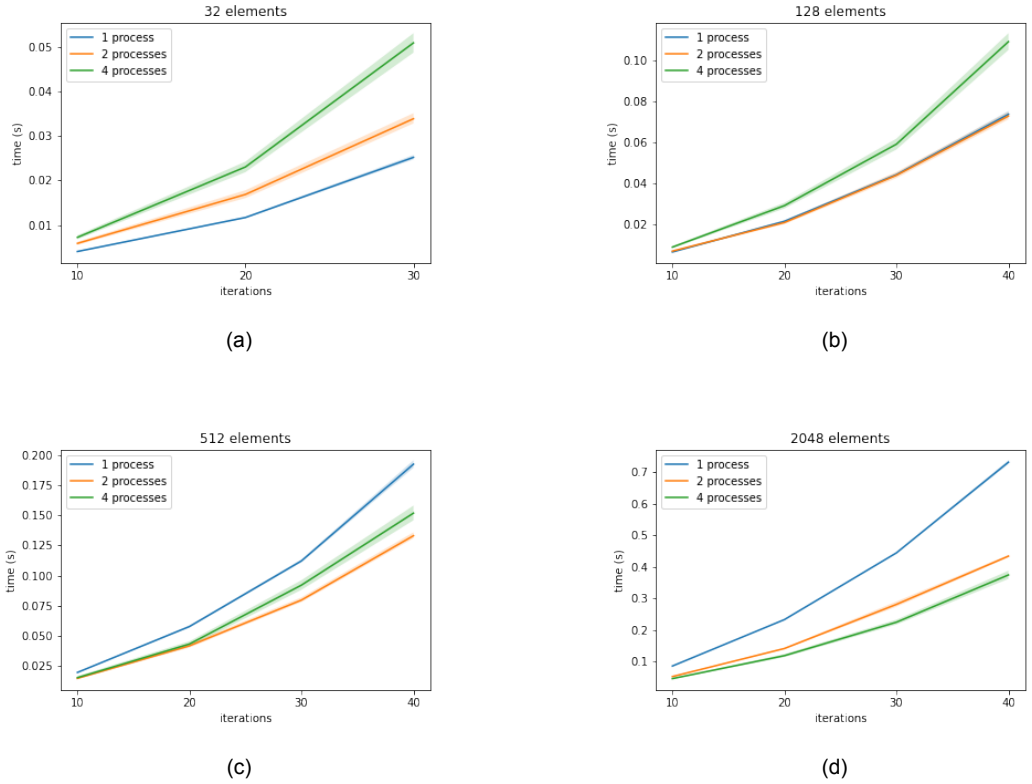


Figure 4.17: Run time per iteration count, for various processes and a 95% bootstrapped confidence interval ($N = 10.000$). Note that the mesh refinement consisting of 32 elements is not able to perform 40 iterations. Figure (b) seems to contain two lines, this is because the line of 1 process and 2 processes (nearly) coincides.

Not only does Figure 4.17 properly visualises the effectiveness of parallelization, it also seems to show non-linear behaviour between computation time and iteration count. This behaviour is typical for the GMRES method, since as the iteration count increases, the orthogonalisation procedure (Arnoldi or modified Gram-Schmidt) will become increasingly dominant. In contrast to the (linear) matrix vector products present in the algorithm, the orthogonalisation procedure behaves quadratic.

In the previous results matrix vector parallelization was performed with the use of row wise decompositions. Now, we will compare this parallelization procedure with checkerboard partitioning. We maintain an error tolerance of 1E-5, and consider the mean run times until convergence using checkerboard partitioning in Table 4.9. Given a single process, the two algorithms coincide and for that reason, columns with one process are purposely left out of the table.

Table 4.9: Mean run time in seconds **until convergence** of checkerboard/column-wise parallelized GMRES method with stopping criterion (Algorithm 7). The emphasised run times are faster than the serial algorithm. The relative error tolerance is set to 1E-5.

Preconditioner	**No preconditioner**		**Calderon preconditioner**	
Mesh size/ no. of processes	**2**	**4**	**2**	**4**
32	0.0109	0.017	0.0083	0.0122
128	0.0294	0.0433	0.0186	0.0262
512	0.1100	0.1276	0.0634	0.0664
2048	0.8804	0.7650	0.3368	**0.3063**

As is clear from Table 4.9, parallelization using this method will only become efficient for large size meshes. This is also visualised in Figure 4.18. This was expected for the columnwise (2 processes parallelization, as the work distribution among processes is the same as with row wise decomposition, whilst the amount of communication between processes increases.

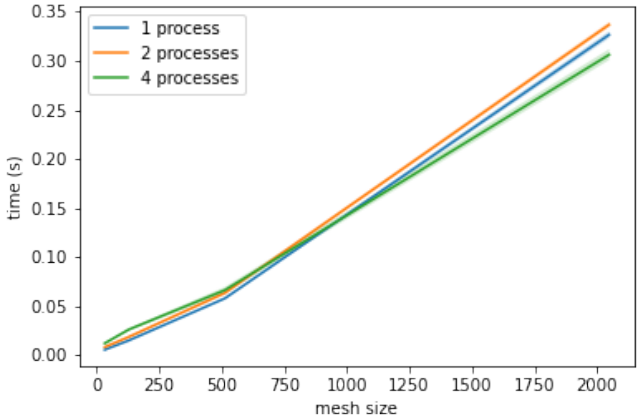


Figure 4.18: Mean convergence time versus mesh size, using checkerboard parallelization of matrix vector products. The confidence interval of the mean is calculated with bootstrap method $(N = 10.000)$.

It could be the case that the efficiency of checkerboard parallelisation versus mesh size increases faster than row wise parallelisation. However, as can be seen in Figure 4.19, for the case of 4 processes, this is not the case.

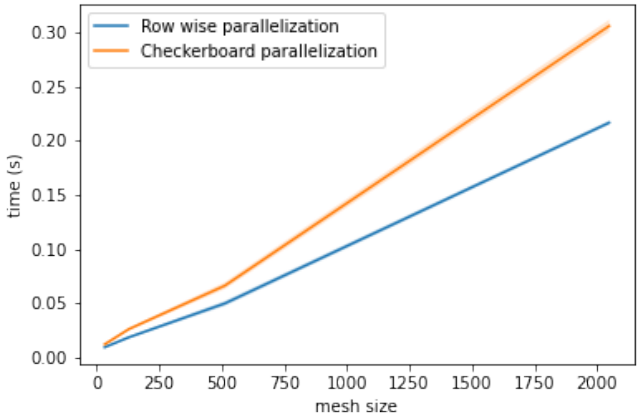


Figure 4.19: Mean convergence time versus mesh size, given different types of parallelization of the matrix vector products and four processes. The confidence interval of the mean is calculated with bootstrap method $(N = 10.000)$.

4.6.5. IDR(s)

A non-symmetric solver introduced by Sonneveld in 1997 has been reconsidered and generalized recently by Sonneveld and van Gijzen: the Induced Dimension Reduction method (IDR). The algorithm for the solver, given a choice of $s \in \mathbb{N}$, is given in Algorithm 12 [19]. This is a mathematical algorithm and its main purpose is to show in a broad sense with which types of potentially parallelizable operations we are dealing. The theory behind this algorithm will not be discusses as the main goal of this section is to provide a comparison between the parallel GMRES method and other non symmetric solvers.

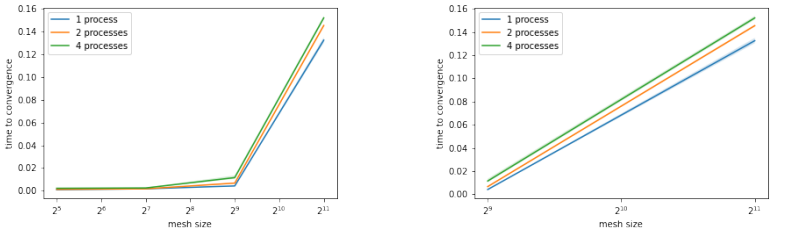
Increasing s will lead to faster convergence, however also to a higher computational cost per iteration. An optimal choice for s, obtained from literature, is $s = 4$. As stated by Sonneveld and van Gijzen, the operation count for the main operations to perform a full cycle of $s + 1$ $IDR(s)$ steps yields: $(s + 1)$ matrix-vector products, $s^2 + s + 2$ inner products and $2s^2 + \frac{7}{2}s + \frac{5}{2}$ vector updates [19]. All of these operations can potentially be parallelized.

Algorithm 12: Mathematical IDR(s) algorithm

Result: $\mathbf{q}_k$, approximation of the solution $\mathbf{M}^{-1}\mathbf{V}_h\mathbf{q} = \mathbf{M}^{-1}\mathbf{F}$
Define initial guess $\mathbf{q}_0$;
Compute $\mathbf{r}^0 = \mathbf{M}^{-1}(\mathbf{F} - \mathbf{V}_h\mathbf{q}_0)$ Let $P \in \mathbb{R}^{N\times s}$ Compute $w := \mathbf{M}^{-1}\mathbf{V}_h v_{k-1}$;
for *n = 0,…,s-1* **do**
  $v = \mathbf{M}^{-1}\mathbf{V}_h\mathbf{r}^n$, $\omega = (v^T\mathbf{r}^n)/(v^T v)$;
  $d\mathbf{q}_n = \omega\mathbf{r}^n$, $d\mathbf{r}^n = -\omega v$;
  $\mathbf{r}^{n+1} = \mathbf{r}^n + d\mathbf{r}^n$, $\mathbf{q}_{n+1} = \mathbf{q}_n + d\mathbf{q}_n$;
end
$dR_{n+1} = (d\mathbf{r}^n \dots d\mathbf{r}^0)$, $dX_{n+1} = (d\mathbf{x}_n \dots d\mathbf{x}_0)$;
while $\|\mathbf{r}_n\|_2 > TOL$ **do**
  for $k = 0, \dots s$ **do**
    solve c from $P^T dR_n c = P^T\mathbf{r}_n$;
    $v = \mathbf{r}_n - dR_n c$;
    if $k = 0$ **then**
      $t = \mathbf{M}^{-1}\mathbf{V}_h v$;
      $\omega = (t^T v)/(t^T t)$;
      $d\mathbf{r}_n = -dR_n c - \omega t$;
      $d\mathbf{x}_n = -dX_n c + \omega t v$;
    else
      $d\mathbf{x}_n = -dX_n c + \omega v$;
      $d\mathbf{r}_n = -\mathbf{M}^{-1}\mathbf{V}_h d\mathbf{x}_n$
    end
    $\mathbf{r}_{n+1} = \mathbf{r}_n + d\mathbf{r}_n$;
    $\mathbf{x}_{n+1} = \mathbf{x}_n + d\mathbf{x}_n$;
    $n = n + 1$;
    $dR_n = (d\mathbf{r}_{n-1} \dots d\mathbf{r}_{n-s})$;
    $dX_n = (d\mathbf{x}_{n-1} \dots d\mathbf{x}_{n-s})$;
  end
end

Using row-wise partitioning, the matrix-vector products are parallelized. At first hand, also the vector updates and inner products are parallelized, however, it turns out to be less efficient (computational time-wise) compared to the parallel algorithm where these operations are done in serial. One possible explanation is that the increase of communication time dominates the time ”won” by decreasing the computational cost per process. Another closely related reason could of course be sub-optimal programming.

An open source python implementation of IDR(s) is used as a starting point for parallelization. Given several meshes, the result of parallelization is represented visually in Figure 4.20



(a) With mesh size ranging from 32 to 2048 elements.

(b) With mesh size ranging from 512 to 2048 elements.

Figure 4.20: Computation time until convergence, with relative error of 1E-5, given different mesh sizes and amount of processes.

For any mesh size IDR(s) performs better time-wise compared to (parallel) GMRES. The parallelisation seems, however to be inefficient, and not becoming more efficient as mesh size increases. As

to why the IDR(s) method seems to perform better time-wise, a possible explanation could be that the implementation is coded more efficiently compared to the GMRES implementation, for which reaching code-wise efficiency was not the main goal. Another possibility is that the GMRES requires a higher amount of floating point operations until convergence. When using stored matrices, only one matrix vector product per iteration is required (except when convergence is achieved), as can be seen in Algorithm 7. However, the amount of flops per iteration for the orthogonalisation procedure at step m is of order $\mathcal{O}(m^2)$. It could thus be the case that GMRES has a higher computational cost in general until convergence. The inefficiency of the parallelization can be explained again due to sub-optimal programming and/or dominating communication-time between processes. Further research in these matters is required and encouraged.

5

Conclusion and Discussion

In this work, the theoretical background for space-time boundary element methods for the heat equation and its implementation were revisited. The discussion was restricted to solving the one and two dimensional Dirichlet heat equation.

Numerical experiments indicated that both the introduced one-, and two dimensional implementations converged to the exact solution of their respective problems. Regarding the numerical experiments of the two dimensional problem, a more general overview of the Dirichlet problem was considered, as the initial condition is taken to be non-zero. The reported new semi-analytical approach for approximating the Galerkin matrix entries, restricted to a triangulation consisting of right-triangular elements, was only considered for the two dimensional case. A similar approach for Dirichlet problems in different dimensions was not investigated in this thesis. The proposed approach was used for the Galerkin discretization of the related boundary integral equations, on which the presented preconditioning and parallelization techniques are based.

Two types of preconditioning techniques were investigated: Calderón preconditioning and Restricted Additive Schwarz (RAS) preconditioning. For the two dimensional problem, these preconditioners were implemented. For the duality pairing of the Calderón preconditioner, an approach introduced by Stevenson and van Venetië was presented, and used in the two dimensional implementations.

The influence of the two aforementioned preconditioning techniques on the iteration count was compared. Additionally, the RAS method was considered as a stand-alone iterative method.

A two dimensional Calderón preconditioned system was used in testing the efficiency of the parallel GMRES implementations. After revisiting different parallelization schemes, parallel implementations of the GMRES method were compared to each other and to a different non-symmetric solver, the Induced Dimension Reduction (IDR) method. Possible parallelization of the IDR method was also explored.

The convergence of the numerical implementations to the exact solutions were described with respect to the Euclidean norm. This is rather unconventional in a (space-time) BEM setting. However, as the purpose of the numerical experiments regarded validation of the implementation, the Euclidean norm sufficed.

The proposed semi-analytical approach regarding the approximation of the Galerkin matrix entries, significantly increases the accuracy of the estimate. Even though only uniform grids are considered in the numerical experiments, this approach remains valid for non-uniform meshes. A downside to this method is that the imposed restrictions on the triangulation entail a significant loss of generality. It is recommended to investigate the potential gain in computational efficiency versus the imposed restrictions of this approach, against more general approaches, such as proposed by Manson and Tausch [13].

Both Calderón preconditioning and RAS preconditioning reduced the iteration count of the GMRES method until convergence significantly. The spectral condition number of the Calderón preconditioned weakly singular matrix, compared to its non-preconditioned counterpart, suggests asymptotic convergent behaviour. The RAS preconditioning is observed to be the most efficient preconditioner, when the domain is decomposed in two uniform domains. However, it should be noted that this type of preconditioning requires computing the inverse of two potentially large matrices, which is a computationally heavy procedure. The contribution of implementing overlap is negligible.

As a basic iterative method, RAS converges. The difference in iteration count between preconditioned GMRES and RAS increases as the mesh size increases, in favour of GMRES (lower iteration count). We believe that, apart from looking at iteration counts, future research should also look at the computation-time (e.g. until convergence) of the methods and their parallel counterparts.

Parallelizing matrix vector products using the row-wise decomposition algorithm, results into a more efficient parallel GMRES implementation, compared to using checkerboard decompositions. Note however that, given 2 processes, checkerboard decomposition actually coincides with a column-wise decomposition. The implementation of parallel GMRES shows an increasing computational-time related efficiency regarding degree of parallelism, for higher levels of refinement. However, the implementation is outperformed by IDR(s) in all cases. This could be due to higher computational cost of GMRES until convergence, or due to sub-optimal programming of the parallel GMRES implementation. Additionally, parallelisation of the IDR(s) method does not lead to a speedup, which suggests further research is required regarding the IDR(s) method.

All time-related measurements are done on a personal computer (4-core). Naturally, an attempt was made to create a stable environment for running the measurements. However, this way remains error prone. For qualitatively better results, it is recommended to run similar experiments in a truly isolated environment.

References

[1] M. Costabel. “Boundary Integral Operators for the Heat Equation”. In: *Integral Equations and Operator Theory* 13 (1990), pp. 498–552.

[2] M. Costabel. *Principles of Boundary Element Methods*. Technische Hochschule Darmstadt. 1986.

[3] S. Dohr. “Distributed and Preconditioned Space–Time Boundary Element Methods for the Heat Equation”. PhD thesis. Technischen Universität Graz, 2019.

[4] V. Dolean. *An Introduction to Schwarz Methods*. University Lecture slides. Université de Nice and University of Geneva, 2011.

[5] E. Efstathiou. “Why Restricted Schwarz Converges Faster than Additive Schwarz”. In: *BIT Numerical mathematics* 52 (2003), pp. 949–959.

[6] M. Geller and E.W. Ng. “A Table of Integrals of the Exponential Integral”. In: *Mathematics and Mathematical Science* 73B.3 (1969).

[7] V. P. Gergel. *Introduction to Parallel Programming*. University Lecture slides. University of Nizhni Novgorod, 2005.

[8] R. Hiptmair. “Operator Preconditioning”. In: *An international Journal computers & mathematics with applications* 52 (2006), pp. 699–706.

[9] R. Hiptmair, C. Jerez-Hanckes, and C. Urzua. *Optimal Operator Preconditioning for Boundary Elements and Open Curves*. Eigenössische Technische Hochschule, 2013.

[10] W. B. Johnson and J. Lindenstrauss. *Basic concepts of Geometry in Banach spaces*. Vol. 1. Elsevier, 2001.

[11] J. van Kan, G. Segal, and F. Vermolen. *Numerical methods in Scientific Computing*. 2nd ed. Delft Academic Press / VSSD, 2014.

[12] J. L. Lions and E. Magenes. *Non-Homogeneous Boundary Value Problems and Applications*. Vol. 2. Springer-Verlag Berlin Heidelberg New York, 1972.

[13] N. Manson and J. Tausch. “Quadrature for parabolic Galerkin BEM with moving surfaces”. In: *Computers and Mathematics with Applications* 77 (2019), pp. 1–14.

[14] J. van Neerven. *An Introduction to Functional Analysis*. University lecture notes. University of Delft. 2020.

[15] P. J. Noon. “The Single Layer Heat Potential and Galerkin Boundary Element Methods for the Heat Equation”. PhD thesis. University of Maryland, 1988.

[16] A. Reinarz. “Sparse Space-time Boundary Element Methods for the Heat Equation”. University of Reading, 2015.

[17] Y. Saad. *Iterative Methods for Sparse Linear Systems*. 2nd ed. Society for Industrial and Applied Mathematics, 2003.

[18] S. A. Sauter and C. Schwab. *Boundary element methods*. Springer series in Computational Mathematics 39. Springer, 2011.

[19] P. Sonneveld and M. van Gijzen. *IDR(s): A Family of Simple and Fast Algorithms for Solving Large Linear Nonsymmetric Systems*. University of Delft, 2007.

[20] O. Steinbach. *Numerical Approximation Methods for Elliptic Boundary Value Problems*. 1st ed. Springer, 2003.

[21] O. Steinbach. “On a generalized L2 projection and some related stability estimates in Sobolev spaces”. In: *Numerische Mathematik* 52 (2000), pp. 775–786.

[22] O. Steinbach. *Stability Estimates for Hybrid Coupled Domain Decomposition Methods*. Springer, 2003.

[23] R. Stevenson and R. van Venetië. *Uniform Preconditioners for Problems of Negative Order*. Hausdorff Research Institute for Mathematics, 2018.

[24] L. N. Trefethen and D. Bau. *Numerical Linear Algebra*. Society of Industrial Applied Mathematics, 1997.

[25] C. Vuik and D.J.P. Lahaye. *Scientific Computing (wi4201)*. University lecture notes. University of Delft. 2019.