POLITECNICO MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE E DELL'INFORMAZIONE

A space-time discontinuous Galerkin method for the wave equation

Francesco Songia, Enrico Zardi
Advisors: Ilario Mazzieri, Gabriele Ciaramella

Project in Numerical Analysis for Partial Differential Equations
Academic year 2021-22

Keywords: wave equation, space-time, discontinuos Galerkin, domain decomposition, restricted additive Schwarz

Abstract
In this project we propose a space-time discontinuous Galerkin method for the 1D wave equation in a rectangular domain. We focus our attention to domain decomposition techniques and study at the numerical level the behaviour of domain decomposition methods and preconditioners directly in a space-time grid. In particular, we use the Restrictive Additive Schwarz (RAS) method as a solver and as a preconditoner for a Krylov method. Moreover, we introduce a pipelined version of RAS to reduce the resources used, taking advantage of the fact that in space-time domains the correct information travels in the time direction. Finally, in order to compare the different methods, we perform numerical experiments focusing on iterations and number of solved subdomains employed to reach the solution. We study the effects of these methods also on the damped version of the wave equation.

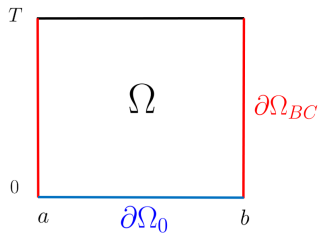
1 Introduction

The goal of this project is to solve the wave equation considering time as a spatial variable and applying a DG finite element method on a space-time grid. Then, to extend existing domain decomposition techniques to this problem to exploit the advantages of parallelization. This work motivates why space-time approaches are valid alternatives of treating time-dependent problems with respect to classic finite difference methods. We work with a simple cartesian grid but this could be applied on a more general polygonal space-time one.
In Section 2, we discretize the problem with discontinuous Galerkin method and we characterize the different formulations that we will use. In Section 3, we present domain decomposition techniques based on a restricted additive Schwarz method, and we compare these techniques by performing numerical experiments in Section 3.3.

2 DG Discretization

Let us consider the 1D wave equation in a rectangular space-time domain $\Omega = (a, b) \times (0, T)$. Assume that the wave speed is constant and equal to c. We assign initial conditions on $\partial\Omega_0$ and Dirichlet boundary condition on $\partial\Omega_{BC}$ to $u = u(x, t)$.

$$\begin{cases} u_{tt} - c^2 u_{xx} = f & \text{in } \Omega, \\ u(x,0) = u_0(x) & x \in [a,b], \\ u_t(x,0) = w_0(x) & x \in [a,b], \\ u(a,t) = g(t) & t \in [0,T], \\ u(b,t) = h(t) & t \in [0,T]. \end{cases} \tag{1}$$



To discretize this problem we first split the wave equation in two equations adding the variable $w = w(x, t)$:

$$\begin{aligned} w_t - c^2 u_{xx} &= f, \\ w - u_t &= 0. \end{aligned} \tag{2}$$

This technique allows us to deal with the initial condition on u_t.
In order to write the weak formulation we start subdividing our square domain Ω in cartesian finite elements $\Omega_k \in \mathcal{T}_h$, where $\mathcal{T}_h$ is the triangulation on Ω. We consider space on the x-axis and time on the y-axis.

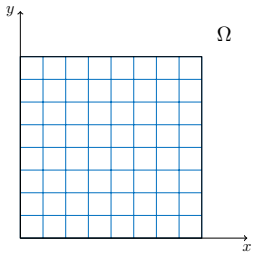


Figure 1: Example of triangulation $\mathcal{T}_h$.

We define now some sets which will help us in the DG framework. Let $\mathcal{E}^0$ be the collection of the interior edges shared between any two elements in $\mathcal{T}_h$ and by $\mathcal{E}^\partial$ the collection all the boundary edges.

$$\begin{aligned} \mathcal{E}^0 &:= \{e \colon e = \partial\Omega_i \cap \partial\Omega_j, \forall\, \Omega_i, \Omega_j \in \mathcal{T}_h\}, \\ \mathcal{E}^\partial &:= \{e \colon e = \partial\Omega_k \cap \partial\Omega, \forall\, \Omega_k \in \mathcal{T}_h\}. \end{aligned}$$

As a consequence the set of all the edges is $\mathcal{E} := \mathcal{E}^0 \cup \mathcal{E}^\partial$. We define also $\mathcal{E}^S$ as the set collecting all the vertical edge and $\mathcal{E}^T$ the set containing all the horizontal edges except the top one at the final time step.

$$\begin{aligned} \mathcal{E}^S &:= \{e \in \mathcal{E} \colon \boldsymbol{n}_e \cdot \boldsymbol{e}_1 = \pm 1\}, \\ \mathcal{E}^T &:= \{e \in \mathcal{E} \colon \boldsymbol{n}_e \cdot \boldsymbol{e}_2 = -1\}, \end{aligned}$$

where $\boldsymbol{n}_e$ is the normal associated to the edge and $\boldsymbol{e}_1$, $\boldsymbol{e}_2$ the canonical basis vector in $\mathbb{R}^2$. We notice that we can associate two opposite normal vectors to an interior edge, which represents the edge of two adjacent element Ω_i and Ω_j. However, boundary edges have only one normal and therefore the $\boldsymbol{n}_e \cdot \boldsymbol{e}_2 = -1$ condition discards only the horizontal edge at the final time step.

Let $\mathcal{E}^{T_0}$ be the set collecting all the boundary edges at the initial time step, and $\mathcal{E}^{\partial S}$ be the set containing all the vertical boundary edges.

$$\mathcal{E}^{T_0} \coloneqq \mathcal{E}^{\partial} \cap \mathcal{E}^{T},$$
$$\mathcal{E}^{\partial S} \coloneqq \mathcal{E}^{\partial} \cap \mathcal{E}^{S}.$$

Finally, we need a space V of test functions. Considering the triangulation $\mathcal{T}_h$ we take $V = V_{DG}$ as the space of test function for our problem.

$$V_{DG} \coloneqq \{v \in L^2(\Omega) \colon v|_{\Omega_k} \in \mathbb{P}^r(\Omega_k), \forall \Omega_k \in \mathcal{T}_h\}.$$

We notice that a function $\psi \in V_{DG}$ can jump across $\mathcal{E}^0$ but is *single-valued* on $\mathcal{E}^{\partial}$. We also define the average and jump operator for a function $\psi \in V_{DG}$ on the edge $e \coloneqq \partial\Omega_1 \cap \partial\Omega_2$. Considering $\psi_i \coloneqq \psi|_{\partial\Omega_i}$ we have

$$\{\!\{\psi\}\!\} = \frac{1}{2}(\psi_1 + \psi_2), \quad [\![\psi]\!] = \psi_1 \boldsymbol{n}_1 + \psi_2 \boldsymbol{n}_2,$$

where $\boldsymbol{n}_i$ is the normal outwards of $\partial\Omega_i$. At the boundary we have $\{\!\{\psi\}\!\} = \psi$ and $[\![\psi]\!] = \psi\,\boldsymbol{n}$.
We are ready to write the weak formulation of problem (2). Let $V = V_{DG}$ and $u, w \in V$,

$$\sum_{\Omega_k \in \mathcal{T}_h} \int_{\Omega_k} w_t v \, d\Omega - c^2 \sum_{\Omega_k \in \mathcal{T}_h} \int_{\Omega_k} u_{xx} v \, d\Omega = \sum_{\Omega_k \in \mathcal{T}_h} \int_{\Omega_k} f v \, d\Omega \qquad \forall v \in V, \tag{3}$$

$$\sum_{\Omega_k \in \mathcal{T}_h} \int_{\Omega_k} u_t q \, d\Omega - \sum_{\Omega_k \in \mathcal{T}_h} \int_{\Omega_k} w q \, d\Omega = 0 \qquad \forall q \in V. \tag{4}$$

We now focus on the second term of (3) and perform integration by parts:

$$\sum_{\Omega_k \in \mathcal{T}_h} \int_{\Omega_k} u_{xx} v \, d\Omega = -\sum_{\Omega_k \in \mathcal{T}_h} \int_{\Omega_k} u_x v_x \, d\Omega + \sum_{\Omega_k \in \mathcal{T}_h} \left(\int_{\Gamma_{2k}} u_x v \; n_x \, dl + \int_{\Gamma_{1k}} u_x v \; n_x \, dl \right), \tag{5}$$

where Γ_{1k} and Γ_{2k} are respectively the left and right vertical boundary associated to the element Ω_k.

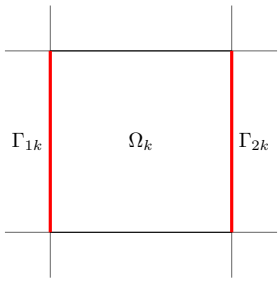


Figure 2: Γ_{2k} on the right vertical border and Γ_{1k} on the left vertical border of the element Ω_k.

The last term in (5) can be written using the jump and average operators as

$$\sum_{\Omega_k \in \mathcal{T}_h} \left(\int_{\Gamma_{2k}} u_x v \, n_x \, dl + \int_{\Gamma_{1k}} u_x v \, n_x \, dl \right) = \sum_{\Gamma_i \in \mathcal{E}^S} \int_{\Gamma_i} \{\!\{ u_x \}\!\} [\![v]\!] \, dl + \sum_{\Gamma_i \in \mathcal{E}^0 \cap \mathcal{E}^S} \int_{\Gamma_i} [\![u_x]\!] \{\!\{ v \}\!\} \, dl. \quad (6)$$

The last term in (6) is null since for exact solution the jump on internal faces of u_x is zero.
To simplify the notation we write $\{\!\{ u_x \}\!\}$ instead of $\{\!\{ \nabla u \}\!\}$ since for these integrals on the vertical edge the only component remaining is u_x.

Using (6) (5) in (3) we obtain a first possible DG weak formulation

$$\sum_{\Omega_k \in \mathcal{T}_h} \int_{\Omega_k} w_t v \, d\Omega + c^2 \sum_{\Omega_k \in \mathcal{T}_h} \int_{\Omega_k} u_x v_x \, d\Omega - c^2 \sum_{\Gamma_i \in \mathcal{E}^S} \int_{\Gamma_i} \{\!\{ u_x \}\!\} [\![v]\!] \, dl = \sum_{\Omega_k \in \mathcal{T}_h} \int_{\Omega_k} f v \, d\Omega \qquad \forall v \in V, \quad (7)$$

$$\sum_{\Omega_k \in \mathcal{T}_h} \int_{\Omega_k} u_t q \, d\Omega - \sum_{\Omega_k \in \mathcal{T}_h} \int_{\Omega_k} w q \, d\Omega = 0 \qquad \forall q \in V. \quad (8)$$

2.1 Stability terms

To this formulation we add four stability terms which don not alter the consistency of the numerical method:

- A term for symmetry in (7)
$$-c^2 \sum_{\Gamma_i \in \mathcal{E}^S} \int_{\Gamma_i} \{\!\{ v_x \}\!\} [\![u]\!] \, dl.$$

- A space stability term in (7) with μ large enough to be chosen. This term penalizes high jumps of the solution on the space interfaces between elements.
$$\mu \sum_{\Gamma_i \in \mathcal{E}^S} \int_{\Gamma_i} [\![u]\!] \cdot [\![v]\!] \, dl.$$

- Two stabilisation terms in time in (7) and (8) respectively. This term also penalizes high jumps of the solution on the lower time interface of each element.
$$\sum_{\Gamma_i \in \mathcal{E}^T} \int_{\Gamma_i} [\![w]\!] v \, dl, \qquad \sum_{\Gamma_i \in \mathcal{E}^T} \int_{\Gamma_i} [\![u]\!] v \, dl.$$
by $[\![\psi]\!] v$ we indicate $(\psi^+ - \psi^-) \, v^+$ where ψ^+ and ψ^- are the bases functions in the above and bottom element sharing an horizontal edge.

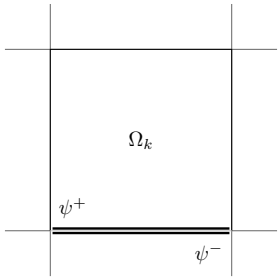


Figure 3: ψ^+ associated to Ω_k, ψ^- associated to its neighbour below.

Finally, we reach this final formulation.

$$\sum_{\Omega_k\in\mathcal{T}_h}\int_{\Omega_k} w_t v\,d\Omega + c^2\sum_{\Omega_k\in\mathcal{T}_h}\int_{\Omega_k} u_x v_x\,d\Omega - c^2\sum_{\Gamma_i\in\mathcal{E}^S}\int_{\Gamma_i}\{\!\{u_x\}\!\}[\![v]\!]\,dl - c^2\sum_{\Gamma_i\in\mathcal{E}^S}\int_{\Gamma_i}\{\!\{v_x\}\!\}[\![u]\!]\,dl$$
$$\sum_{\Gamma_i\in\mathcal{E}^T}\int_{\Gamma_i}[\![w]\!]v\,dl + \mu\sum_{\Gamma_i\in\mathcal{E}^S}\int_{\Gamma_i}[\![u]\!]\cdot[\![v]\!]\,dl = \sum_{\Omega_k\in\mathcal{T}_h}\int_{\Omega_k} f v\,d\Omega \qquad \forall v\in V, \quad (9)$$

$$\sum_{\Omega_k\in\mathcal{T}_h}\int_{\Omega_k} u_t q\,d\Omega - \sum_{\Omega_k\in\mathcal{T}_h}\int_{\Omega_k} wq\,d\Omega\sum_{\Gamma_i\in\mathcal{E}^T}\int_{\Gamma_i}[\![u]\!]q\,dl = 0 \qquad \forall q\in V. \quad (10)$$

2.2 Boundary terms

It is important to clarify how the terms with jumps and averages behave at the boundary and understand how the boundary conditions are weakly imposed. Thus we show how those terms are explicitly computed at the boundary in the following:

$$\sum_{\Gamma_i\in\mathcal{E}^{\partial S}}\int_{\Gamma_i}\{\!\{u_x\}\!\}[\![v]\!]\,dl = \sum_{\Gamma_i\in\mathcal{E}^{\partial S}}\int_{\Gamma_i} u_x v\,dl, \quad (11)$$

$$\sum_{\Gamma_i\in\mathcal{E}^{\partial S}}\int_{\Gamma_i}\{\!\{v_x\}\!\}[\![u]\!]\,dl = \sum_{\Gamma_i\in\mathcal{E}^{\partial S}}\int_{\Gamma_i} v_x(u-g)\,n_x\,dl = \sum_{\Gamma_i\in\mathcal{E}^{\partial S}}\int_{\Gamma_i} v_x u\,n_x\,dl - \sum_{\Gamma_i\in\mathcal{E}^{\partial S}}\int_{\Gamma_i} v_x g\,n_x\,dl. \quad (12)$$

The last term of (12) is a known quantity and goes to the right hand side imposing the spatial boundary condition on u.

$$\mu\sum_{\Gamma_i\in\mathcal{E}^{\partial S}}\int_{\Gamma_i}[\![u]\!]\cdot[\![v]\!]\,dl = \mu\sum_{\Gamma_i\in\mathcal{E}^{\partial S}}\int_{\Gamma_i}(u-g)v\,n_x\,dl = \mu\sum_{\Gamma_i\in\mathcal{E}^{\partial S}}\int_{\Gamma_i} uv\,n_x\,dl - \mu\sum_{\Gamma_i\in\mathcal{E}^{\partial S}}\int_{\Gamma_i} gv\,n_x\,dl. \quad (13)$$

The last term of (13) is a known quantity and goes to the right hand side. Also here we have a weakly imposition on u of the spatial boundary condition.

$$\sum_{\Gamma_i\in\mathcal{E}^{T_0}}\int_{\Gamma_i}[\![u]\!]v\,dl = \sum_{\Gamma_i\in\mathcal{E}^{T_0}}\int_{\Gamma_i}(u_0-u)v\,dl = \sum_{\Gamma_i\in\mathcal{E}^{T_0}}\int_{\Gamma_i} u_0 v\,dl - \sum_{\Gamma_i\in\mathcal{E}^{T_0}}\int_{\Gamma_i} uv\,dl. \quad (14)$$

The first term of (14) is a known quantity and goes to the right hand side and impose the initial condition weakly on u. A similar result is obtained if we compute the term associated to w, in that case we weakly impose the initial condition on u_t.

2.3 IP and IPH formulation

Some of the terms in (9) can be grouped in an Interior Penalty (IP) term corresponding to a symmetric bilinear form $a_{\mathrm{IP}}\colon V\times V\to\mathbb{R}$,

$$a_{\mathrm{IP}}(u,v) \coloneqq c^2\sum_{\Omega_k\in\mathcal{T}_h}\int_{\Omega_k} u_x v_x\,d\Omega - c^2\sum_{\Gamma_i\in\mathcal{E}^S}\int_{\Gamma_i}\{\!\{u_x\}\!\}[\![v]\!]\,dl$$
$$- c^2\sum_{\Gamma_i\in\mathcal{E}^S}\int_{\Gamma_i}\{\!\{v_x\}\!\}[\![u]\!]\,dl + \mu\sum_{\Gamma_i\in\mathcal{E}^S}\int_{\Gamma_i}[\![u]\!]\cdot[\![v]\!]\,dl \qquad \forall v\in V. \quad (15)$$

By adding an additional consistent term affecting the interior edges we obtain the Hybridizable Interior Penalty (IPH) formulation:

$$a_{\mathrm{IPH}}(u,v) \coloneqq c^2 \sum_{\Omega_k \in \mathcal{T}_h} \int_{\Omega_k} u_x v_x \, d\Omega - c^2 \sum_{\Gamma_i \in \mathcal{E}^S} \int_{\Gamma_i} \{\!\{u_x\}\!\} [\![v]\!] \, dl - c^2 \sum_{\Gamma_i \in \mathcal{E}^S} \int_{\Gamma_i} \{\!\{v_x\}\!\} [\![u]\!] \, dl$$
$$+ \frac{\mu}{2} \sum_{\Gamma_i \in \mathcal{E}^S} \int_{\Gamma_i} [\![u]\!] \cdot [\![v]\!] \, dl - \frac{1}{2\mu} \sum_{\Gamma_i \in \mathcal{E}^0 \cap \mathcal{E}^S} \int_{\Gamma_i} [\![u_x]\!] \cdot [\![v_x]\!] \, dl \qquad \forall v \in V. \quad (16)$$

We consider these formulations since we will see that they are linked to different transmission conditions when applying domain decomposition methods. These formulations are studied in [1] and in [3] for reaction diffusion problems in spatial domain, and we will investigate their behaviour in a space-time framework.

2.4 Algebraic formulation

It is possible to write the remaining terms in (9) (10) compactly by defining some operators:

$$\mathrm{b}(w,v) \coloneqq \sum_{\Omega_k \in \mathcal{T}_h} \int_{\Omega_k} w_t v \, d\Omega \quad \forall w, v \in V, \quad (17)$$

$$\mathrm{c}(w,v) \coloneqq \sum_{\Gamma_i \in \mathcal{E}^T} \int_{\Gamma_i} [\![w]\!] v \, dl \quad \forall w, v \in V, \quad (18)$$

$$\mathrm{m}(w,v) \coloneqq \sum_{\Omega_k \in \mathcal{T}_h} \int_{\Omega_k} wv \, d\Omega \quad \forall w, v \in V, \quad (19)$$

$$\mathrm{F}(v) \coloneqq \sum_{\Omega_k \in \mathcal{T}_h} \int_{\Omega_k} fv \, d\Omega \qquad \forall v \in V. \quad (20)$$

Therefore (9) and (10) become

$$\mathrm{b}(w,v) + \mathrm{a}_{\mathrm{IP/IPH}}(u,v) + \mathrm{c}(w,v) = \mathrm{F}(v) \qquad \forall v \in V, \quad (21)$$
$$\mathrm{b}(u,q) - \mathrm{m}(w,q) + \mathrm{c}(u,q) = 0 \qquad \forall q \in V. \quad (22)$$

Each form written can be associated to matrix and vector multiplication compactly written as

$$\begin{bmatrix} A_{IP/IPH} & B+C \\ B+C & -M \end{bmatrix} \begin{bmatrix} U \\ W \end{bmatrix} = \begin{bmatrix} F \\ 0 \end{bmatrix} + \begin{bmatrix} F_{BC1} \\ F_{BC2} \end{bmatrix}, \quad (23)$$

where F_{BC1} and F_{BC2} are the vectors arising from o the boundary condition.

2.5 Damped wave equation

It is possible to apply the DG space-time approach also to the damped version of the wave equation:

$$\begin{cases} u_{tt} - c^2 u_{xx} + k u_t = f \quad \text{in } \Omega, \\ u(x,0) = u_0(x) \quad x \in [a,b], \\ u_t(x,0) = w_0(x) \quad x \in [a,b], \\ u(a,t) = g(t) \quad t \in [0,T], \\ u(b,t) = h(t) \quad t \in [0,T], \end{cases} \quad (24)$$

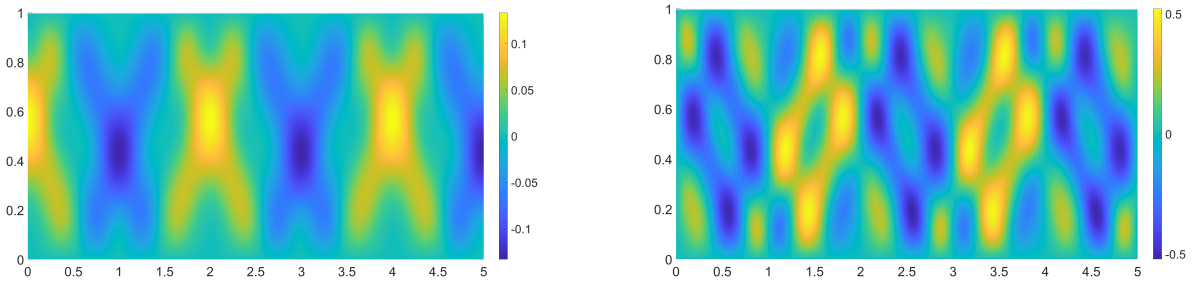
where k is the *damping constant*.
In the end the weak formulation is similar to the classical version of the wave equation

$$\mathrm{b}(w,v) + \mathrm{a}_{\mathrm{IP/IPH}}(u,v) + \mathrm{c}(w,v) + k\mathrm{m}(w,v) = \mathrm{F}(v) \qquad \forall v \in V,$$
$$\mathrm{b}(u,q) - \mathrm{m}(w,q) + \mathrm{c}(u,q) = 0 \qquad \forall q \in V.$$

2.6 Numerical test

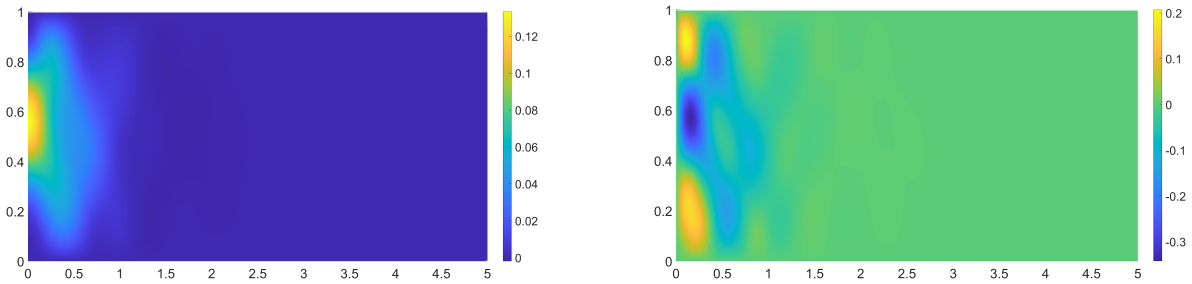
Having stated the weak formulations and their algebraic structure we perform some numerical tests. For the problem (24) we solve the linear system arising from the algebraic formulation with Matlab. The domain considered is $\Omega = (0,1) \times (0,T)$, $T = 5$ and a mesh of 20 and 100 elements in space and time respectively.

$$\begin{cases} u_{tt} - u_{xx} + k u_t = 0 & \text{in } \Omega, \\ u(x,0) = x^2(1-x)\sin(\pi x)^2 & x \in [0,1], \\ u_t(x,0) = 0 & x \in [0,1], \\ u(a,t) = u(b,t) = 0 & t \in [0,5]. \end{cases} \tag{25}$$



(a) displacement u. (b) velocity w.

Figure 4: IP formulation for the wave equation ($k = 0$), μ=400



(a) displacement u. (b) velocity w.

Figure 5: IP formulation for the damped wave equation $k = 5$, μ=400.

A reasonable choice of the DG stability coefficient for achieve stability for this problem is $\mu = 400$.

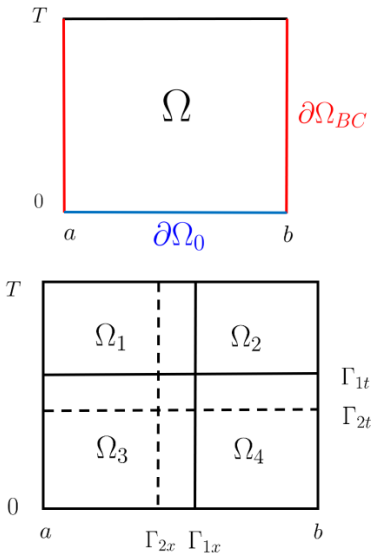
3 Domain decomposition

The goal of this section is to study, through numerical tests, the behaviour of domain decomposition methods, usually used just on a space grid, on a space-time cartesian domain. This lays the foundation for a future development where all the existing technologies for spatial domain decomposition will be applied on space-time polygonal elements.

We consider the wave equation in 1D discretized with a space-time approach using Discontinuos Galerkin method. This discretization leads to a solution of a linear system $A\boldsymbol{u} = \boldsymbol{f}$ for all the space-time domain $\Omega = (a,b) \times (0,T)$.

To formalize the problem, we decompose the domain Ω considering one division in time and space obtaining Ω_i with $i = 1, .., M$ where $M = 4$ in the following figure.
Let $\boldsymbol{z}_i := (u_i, w_i)$ and let $\mathcal{L}$ be the operator corresponding to (1). We consider M different subproblems of the form:

$$\begin{cases} \mathcal{L}\boldsymbol{z}_i = \boldsymbol{f} & \Omega_i, \\ \boldsymbol{z}_i = \boldsymbol{z}_{i+2} & \Gamma_{it} \quad i = 1, 2, \\ \mathcal{B}(\boldsymbol{z}_i) = \mathcal{B}(\boldsymbol{z}_{i+1}) & \Gamma_{ix} \quad i = 1, 3, \\ \boldsymbol{z}_i(x, 0) = (u_0(x), w_0(x)) & \partial\Omega_i \cap \partial\Omega_0, \\ u_i(a, t) = g(t), \ u_i(b, t) = h(t) & \partial\Omega_i \cap \partial\Omega_{BC}. \end{cases} \tag{26}$$



3.1 Space-time decomposition methods

For the solution of the linear system $A\boldsymbol{u} = \boldsymbol{f}$ we will follow three different approaches:

- Restricted Additive Schwarz (RAS) method that changes the restriction matrices of Additive Schwarz (AS) method [2];
- GMRES method with the preconditioner identified by RAS;
- Pipelined RAS that modify RAS based on the idea that the solution evolves in the space time domain in the time direction.

3.1.1 RAS as an iterative method and as a preconditioner for GMRES

The discretization of the partial differential equation leads to a linear system of equations (27) that can be solved with a stationary iterative method (28). Let M be the number of subdomains in which we decompose the domain. R_j and R_j^T, with $j = 1, ..., M$, are rectangular matrices that correspond to the restriction and prolongation operators, respectively. The local matrices A_j are related to A through the restriction and prolongation matrices: $A_j = R_j A R_j^T$.

$$A\boldsymbol{u} = \boldsymbol{f}. \tag{27}$$

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^n + P^{-1}(\boldsymbol{f} - A\boldsymbol{u}^n). \tag{28}$$

For the Additive Schwarz method P_{as} is used as the preconditioner of (28)

$$P_{as}^{-1} = \sum_{j=1}^{M} R_j^T A_j^{-1} R_j. \tag{29}$$

Additive Schwarz methods suffers in convergence [2] since the prolongation operator considers the contribution of both the neighbouring subdomains. Therefore, in the overlapping regions the final solution will be the sum of the two solutions computed in the subdomains associated to the overlap. To fix this issue Restricted Additive Schwarz method weights the two solutions in the overlap. Those weight factors are in the prolongation matrix $\tilde{R}^T$. This weight is added both in the time and space overlapping and we fixed it to 0.5.

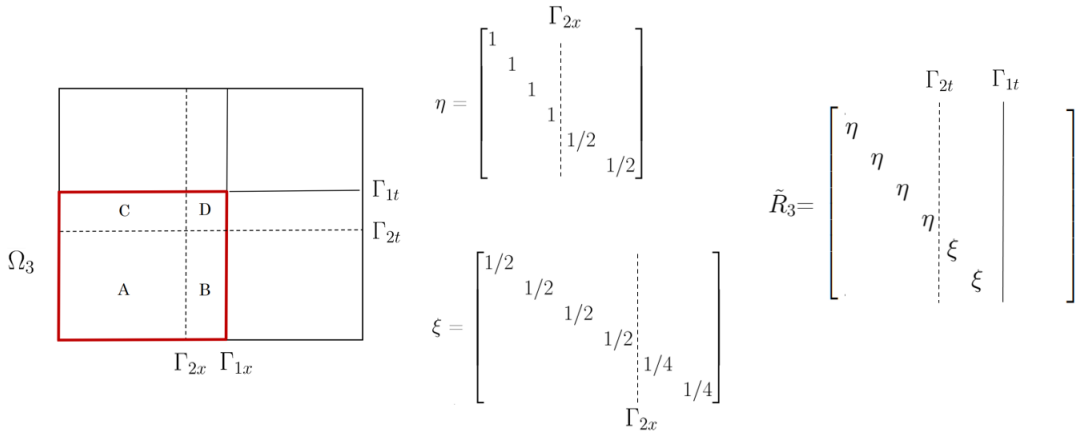


Figure 6: Restriction matrix $\tilde{R}_3$. We can observe the different weights corresponding to the different overlapped zone B,C and D. For example, D is common to all subdomains and therefore its contribute is weighted by 4.

Now we can define the RAS preconditioner:

$$P_{ras}^{-1} = \sum_{j=1}^{M} \tilde{R_j}^T A_j^{-1} R_j. \tag{30}$$

Given our cartesian domain we have two degrees of freedom regarding the decomposition in subdomains: the number of subdomains and the overlap magnitude. The domain is divided based on the number of threads available and then each subdomain is extended in the neighbouring subdomains by few elements to create the overlapping. The overlap is often limited because with an increasing overlap the method solves a bigger zone more times. For example, for a domain $\Omega = (0, 1) \times (0, 5)$, we usually choose 2 subdomains in space and 10 in time with an overlap of 0.2 and 0.05 in space and time respectively.
The RAS method can be used as preconditioner to accelerate the convergence of Krylov methods, like GMRES. A good preconditioner must be a good approximation of A to reduce the magnitude of the condition number. In domain decomposition method with restriction and prolongation operator we give a good representation of A. We keep the idea of decomposing the domain to pass a good preconditioner to an efficient Krylov method.

It is important to observe that the preconditioning step is fully parallelizable as it involves the solution of M independent system, one for each local matrix A_j.

3.1.2 Pipelined RAS

For both the approaches presented before the preconditioner solves all subdomain problems at each iteration. We are not using the fact that in space-time domains the correct information travels in the time direction. This means that the first subdomains in time will be the first to be exactly solved. When we compute the inverse of the preconditioner times the residual, the problem is solved at each

subdomain Ω_j, but at the beginning it will be impossible to have a correct solution in the part of the domain corresponding to the last time instants. Considering only few subdomains at the time permits to avoid useless computation to solve system for which is impossible to obtain a good solution. From this idea we built a Pipelined space-time RAS method, it has the same framework of RAS method but the preconditioner P_{pipe} considers only K subdomains:

$$P_{pipe}^{-1} = \sum_{j=1}^{K} \tilde{R}_j^{\ T} A_j^{-1} R_j. \tag{31}$$

We have the freedom to decide which subdomains are used at each iteration, this must be chosen according to how fast the wave propagation moves from the initial time and according to the velocity of the convergence. We will refer to the subdomains considered at each iteration as subdomains' window S.

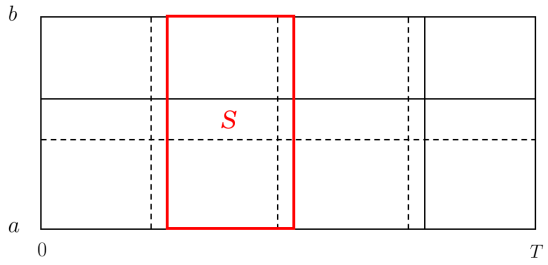


Figure 7: Subdomains' window S. Imagine dividing the entire domain in the time direction and consider only the subdomains in the red zone S. Then, using the division in space, assign the subdomains at each thread. Notice that the domain is not oriented as usual but there is time on the horizontal axis.

At first S is set at the the initial time. Then we run the iterations until the residual in the left part of the zone S reaches a defined tolerance and then move S in time. In this case we have to choose a fixed time interval that corresponds to a certain number of subdomains in time, then we checked if all the subdomains at the left boundary of S are exactly solved and, only after, shift the subdomains' window to the right. This is a very conservative way to proceed since we can be blocked in one window for many iterations and we could lose time in vain.
To face this problem we implemented a more 'aggressive' version of Pipelined RAS: we do not wait until all the subdomains in the left part of S are solved and we update the right boundary after a defined number of iterations. However, we continue to check the left part and we shift it to the right when we achieved a fixed tolerance. In this way the zone of highlighted subdomains S is freer to move, we are not blocked in only one zone and the right side can evolve faster exploring all the subdomains. Then if it moves too fast it will be necessary to wait for all the previous subdomains to be exactly solved but certainly the method will solve less subdomains with respect to the original RAS.

The use of this approach with a moving pipeline brings benefits to the parallel resolution. We can choose the number of space subdomains according to the number of threads available and then assign to each of them one space subdomain. In this way S is equally divided and with one iteration we will solve all the subdomains' window. Compared to the previous method we assign at each thread much less subdomains to solve without lose quality in the convergence.

3.2 Trasmission conditions

IP and IPH formulations are associated to different transmission conditions operators $\mathcal{B}(\cdot)$ between subdomains.
For the analysis of our problem, we have followed [3] which study the effect of IP and IPH transmission

condition on convergence. In order to investigate if this effect still holds in a space-time domain, we want to apply the same ideas and a similar proof on our problem.
Starting from (26), we apply RAS method without overlapping subdomains and we study the corresponding transmission conditions of the two formulations. We recall that this is not our typical framework since we usually consider overlap. A possible further development would be to prove the transmission conditions with overlapping subdomains. However, it is reasonable to expect the same transmission conditions in both cases, so we proceed with the proof of transmission conditions in a non-overlapping decomposition.

3.2.1 Trasmission conditions for IP and IPH

To obtain the transmission condition operator $\mathcal{B}(\cdot)$ on the space interface, we will study only the bilinear form $a(\cdot,\cdot)$ corresponding to the space derivative in (1). This form is the one which characterizes the two different formulations IP and IPH. Let Ω be a domain decomposed without overlap and let Γ be the space interface between two subdomains. The space vertical edges are denoted by $\mathcal{E}^S$ and $\mathcal{E}^{S0}$ correspond to the internal one. With $(\cdot,\cdot)_{\Omega_i}$ we refer to $\int_{\Omega_i} d\Omega$ and with $\langle\cdot,\cdot\rangle_{\mathcal{E}_i^S}$ we refer to $\int_{\mathcal{E}_i^S} dl$. Let first consider IP formulation on the subdomain Ω_i

$$\begin{aligned}\frac{1}{c^2}a_{\mathrm{IP}}(u_i,v_i) &= (u_{ix}v_{ix})_{\Omega_i}\langle\{\!\{u_x\}\!\},[\![v]\!]\rangle_{\mathcal{E}_i^S} - \langle\{\!\{v_{ix}\}\!\},[\![u_i]\!]\rangle_{\mathcal{E}_i^S} + \mu\langle[\![u_i]\!],[\![v_i]\!]\rangle_{\mathcal{E}_i^S} \qquad \forall u_i,v_i\in V(\Omega_i)\\ &= (u_{ix}v_{ix})_{\Omega_i} - \langle\ \{\!\{u_{ix}\}\!\},[\![v_i]\!]\rangle_{\mathcal{E}_i^S\setminus\Gamma} - \langle\ \{\!\{v_{ix}\}\!\},[\![u_i]\!]\rangle_{\mathcal{E}_i^S\setminus\Gamma} + \mu\langle[\![u_i]\!],[\![v_i]\!]\rangle_{\mathcal{E}_i^S\setminus\Gamma}\\ &\quad + \int_\Gamma -u_i\frac{\partial v_i}{\partial n_i} - v_i\frac{\partial u_i}{\partial n_i} + \mu u_i v_i dl \qquad \forall u_i,v_i\in V(\Omega_i).\end{aligned}$$

Integrating back the volume term, we arrive at

$$\begin{aligned}\frac{1}{c^2}a_{\mathrm{IP}}(u_i,v_i) &= (-u_{ixx},v_i)_{\Omega_i} + \langle\ \{\!\{u_{ix}\}\!\},[\![v_i]\!]\rangle_{\mathcal{E}_i^S} - \langle\ \{\!\{u_{ix}\}\!\},[\![v_i]\!]\rangle_{\mathcal{E}_i^S\setminus\Gamma} + \langle\ \{\!\{v_i\}\!\},[\![u_{ix}]\!]\rangle_{\mathcal{E}_i^{S0}}\\ &\quad -\langle\ \{\!\{v_{ix}\}\!\},[\![u_i]\!]\rangle_{\mathcal{E}_i^S\setminus\Gamma} + \langle\mu[\![u_i]\!],[\![v_i]\!]\rangle_{\mathcal{E}_i^S\setminus\Gamma} + \int_\Gamma -u_i\frac{\partial v_i}{\partial n_i} - v_i\frac{\partial u_i}{\partial n_i} + \mu u_i v_i dl \qquad \forall u_i,v_i\in V(\Omega_i)\\ &= (-u_{ixx},v_i)_{\Omega_i} + \langle\ \{\!\{v_i\}\!\},[\![u_{ix}]\!]\rangle_{\mathcal{E}_i^{S0}} + \langle[\![u_i]\!],\mu[\![u_i]\!] - \ \{\!\{v_{ix}\}\!\}\rangle_{\mathcal{E}_i^S\setminus\Gamma}\\ &\quad + \int_\Gamma -u_i\frac{\partial v_i}{\partial n_i} - v_i\frac{\partial u_i}{\partial n_i} + v_i\frac{\partial u_i}{\partial n_i} + \mu u_i v_i dl \qquad \forall u_i,v_i\in V(\Omega_i)\\ &= (-u_{ixx},v_i)_{\Omega_i} + \langle\ \{\!\{v_i\}\!\},[\![u_{ix}]\!]\rangle_{\mathcal{E}_i^{S0}} + \langle[\![u_i]\!],\mu[\![u_i]\!] - \ \{\!\{v_{ix}\}\!\}\rangle_{\mathcal{E}_i^S\setminus\Gamma}\\ &\quad + \int_\Gamma u_i(\mu v_i - \frac{\partial v_i}{\partial n_i})\ dl \qquad \forall u_i,v_i\in V(\Omega_i).\end{aligned}$$

If we however restrict u to the neighboring subdomain Ω_j, such that $\Gamma = \partial\Omega_i \setminus \partial\Omega_j$, we obtain

$$\frac{1}{c^2}a_{\mathrm{IP}}(u_j,v_i) = \int_\Gamma u_j(\mu v_i - \frac{\partial v_i}{\partial n_i})\ dl.$$

Now consider the problem: given $u_j\in H^2(\Omega_j)$ and $\boldsymbol{f}\in[L^2(\Omega)]^2$, find $u_i\in H^2(\Omega_i)$ such that

$$(\mathcal{L}(u_i,0),(v_i,0)) + (\mathcal{L}(u_j,0),(v_i,0)) = (\boldsymbol{f},(v_i,0)) \qquad \forall v_i\in V(\Omega_i).$$

Then one can use Thm.1 in [4] and conclude that all the jumps and the averages are equal to zero since $u_i,v_i\in H^2(\Omega_i)$. Moreover, regarding the integrals over Ω_i, $\mathcal{L}u_i$ cancels out with the force term and it remains only the integral over the interface Γ

$$\int_\Gamma u_i(\mu v_i - \frac{\partial v_i}{\partial n_i}) - u_j(\mu v_i - \frac{\partial v_i}{\partial n_i})\ dl = 0 \qquad \forall v_i\in V(\Omega_i).$$

Finally, we obtain the transmission condition operator $\mathcal{B}(\cdot)$ for IP formulation

$$u_i = u_j \qquad on\ \Gamma.$$

For IPH, by following the same steps as above and following in [3] (pag 39), we obtain a Robin transmission condition

$$\mu u_i + \frac{\partial u_i}{\partial n_i} = \mu u_j + \frac{\partial u_j}{\partial n_i} \qquad on\ \Gamma.$$

The DG stability coefficient μ appears only in the IPH transmission condition as the Robin parameter, instead for IP we obtain a Dirichlet condition. It is proved in [3], by analyzing the convergence factor with Fourier series, that there exists an optimal choice of μ for convergence. We want to investigate through numerical tests how μ influences convergence also in our case.
It is reasonable to expect that for high values of μ a Robin condition behave as a Dirichlet one since the term with the normal derivative could have a lower magnitude and can be neglected with respect than the other one. This is the case in our problem that requires high values for the stability coefficient, therefore the formulations are similar and we don't expect the possible advantages for a Robin condition. However, we will further investigate differences in number of iteration and solved subdomains between the two formulations to understand if a Robin condition is more effective in terms of convergence velocity.

3.3 Numerical tests

In this section we perform different numerical tests to understand the effect of different ways to decompose the domain and different choices of parameters. Through these choices we can investigate the differences among the three methods presented.
We study the effect of the DG coefficient of stability μ, the number of subdomains and the IP or IPH formulation on the solution and on the three methods presented above. The coefficient μ can be varied only between a range of values that leads to a convergence to a stable solution. For the choices of the subdomains we have to select a number that takes the best advantages from the number of thread available.
Furthermore, we also study the damped version of the wave equation (24) to observe differences while using these methods.

In the following tests we will consider the same problem (25) of section 2.6 with $\Omega = [0,1] \times [0,5]$ divided in 20 elements in space and 100 in time.
The domain is decomposed in subdomains with the equal number of elements, therefore the cost of solving each of them is the same. We will consider minimal overlap in both space and time and with different choices of the number of subdomains in space NX and in time NT:

- $NX = 2; NT = 5, 10, 20$
- $NX = 5; NT = 20$

The tolerance required for convergence is 10^{-10} and we have chosen different criterion for the evaluation of the residual that will consequently affect the number of iteration. For RAS and GMRES we define the preconditioned relative residual in $\|\cdot\|_{L_2(\Omega)}$ and for Pipelined RAS we considered the residual in $\|\cdot\|_{L_\infty(\Omega)}$.

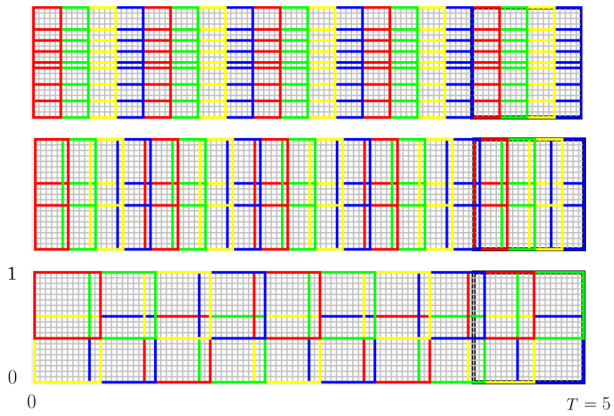


Figure 8: Different choices of subdomains. $NX = 5, 2, 2$ and $NT = 20, 20, 10$ respectively.

3.3.1 Number of subdomains

Usually the choices of the number of subdomains in which we decompose the domain is strictly related to the number of threads available. In any case, we can observe that increasing the number of subdomains leads to an increased number of iterations.
We observe the same behaviour for all the three methods but we can compare only RAS and GMRES since they have the same stopping criterion.

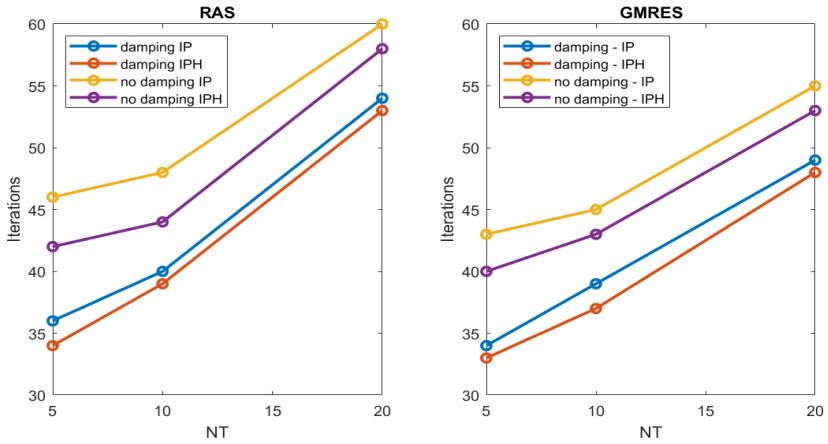


Figure 9: Number of time subdomain vs iteration for RAS and GMRES. The stability coefficient μ is fixed to 400. We observe that the solution of the damped version is reached in less iterations.

From [5] we report a general result that compare residuals and iterations of Krylov methods and stationary iterative methods.

Theorem (Preconditioned Krylov methods versus stationary iterative methods)**.** *Consider a splitting $A = M - N$ with M invertible and the corresponding stationary method:*

$$\boldsymbol{u}_{k+1} = M^{-1}N\boldsymbol{u}_k + M^{-1}\boldsymbol{f} = (I - M^{-1}A)\boldsymbol{u}_k + M^{-1}\boldsymbol{f}. \qquad (a)$$

Consider a Krylov method minimizing the residual applied to the preconditioned system

$$M^{-1}A\boldsymbol{u} = M^{-1}\boldsymbol{f} \qquad (b)$$

(both initialized by the same vector $\boldsymbol{u}_0$). Define the corresponding preconditioned residuals as $r_k^{stat} := M^{-1}\boldsymbol{f} - M^{-1}A\boldsymbol{u}_k^{stat}$ and $r_k^{kry} := M^{-1}\boldsymbol{f} - M^{-1}A\boldsymbol{u}_k^{kry}$. Then we have that $\|r_k^{kry}\|_2 \leq \|r_k^{stat}\|_2$ for any $k = 0, 1, 2, \ldots, n$. In other words, a stationary iterative method (a) based on M can never perform fewer iterations than a Krylov method minimizing the residual applied to (b).

As we can observe in Figure 9 but also in the following figures, what is stated in the theorem is confirmed since we can observe that the iterations when using GMRES are fewer than the ones used on the corresponding problem by RAS method.

3.3.2 Stability coefficient μ

In Figure 10 we can observe how the coefficient of stability μ affects the number of iterations. This is principally linked to the coercivity costant that increases with μ and leads to a slower convergence. From this plot we can appreciate also the difference in the formulations: IPH has a Robin transmission condition that is usually faster as seen in [3]; however, the effect is reduced since for stability reasons, we need μ to be high. It is reasonable to expect that for values lower than 400 the difference between IP and IPH will be more evident. Also in this case the same behaviour is observed with the Pipelined RAS method.

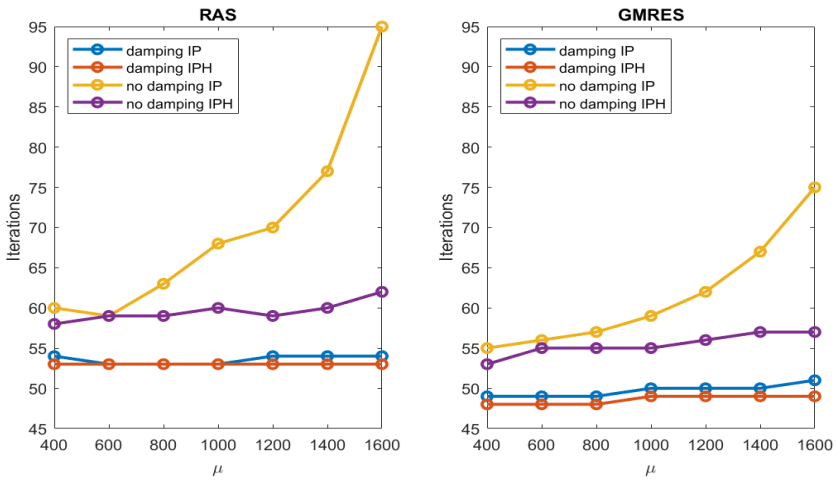


Figure 10: Impact of μ on iteration for RAS and GMRES. Decomposition choice: $NX = 2$, $NT = 20$.

3.3.3 Solved subdomains

The main difference in these methods can be highlighted with the number of solved subdomains. As the number of solved subdomains we consider how many times a subdomain of the decomposition is solved. For RAS or GMRES this number is equal to the total number of iterations times the total number of subdomains since at each iterations these methods use all the subdomains to compute the preconditioner. Instead, Pipelined RAS uses less computations and this has a great repercussions on the parallelization process, we can assign at each threads fewer subdomains to solve. We can observe in Figure 11 and in Table 1 the incredible advantage of use a Pipelined RAS method for all the combinations of formulations, μ and presence of damping.

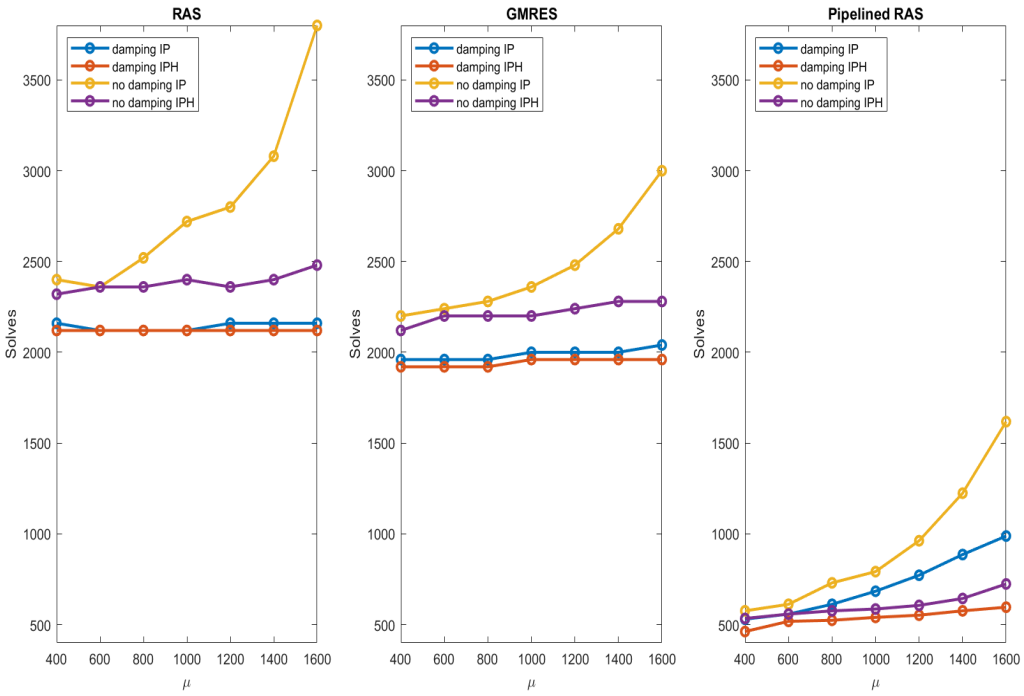


Figure 11: Number of subdomains solved for the different methods. Decomposition choice: $NX = 2$, $NT = 20$.

	RAS	GMRES	Pipe.RAS
damping IP	2160	1960	530
damping IPH	2120	1920	462
no damping IP	2400	2200	576
no damping IPH	2320	2120	534

Table 1: Solved subdomains for $\mu = 400$. Decomposition choice: $NX = 2$, $NT = 20$.

3.3.4 Residuals evolution

Looking at the residuals we can better understand how the convergence is affected by the different methods, formulations and number of subdomains. Clearly we can see how the damping version of the wave equation has a better evolution in residuals since does not present a residual growth along the iterations. This behaviour is affected also by the formulation (for IPH is mitigated) and by the number of subdomains. This is coherent also with the analysis on the increasing number of iterations when using IP with many subdomains.
The arise of the residual growth along the iterations, the remarkable number of iterations and the convergence not reached in some cases are all visible only when using RAS method. The use of P_{ras} computed in all subdomains seems to be a weak choice in this framework, instead GMRES that uses RAS as a preconditioner is more robust and has acceptable results in every situation.

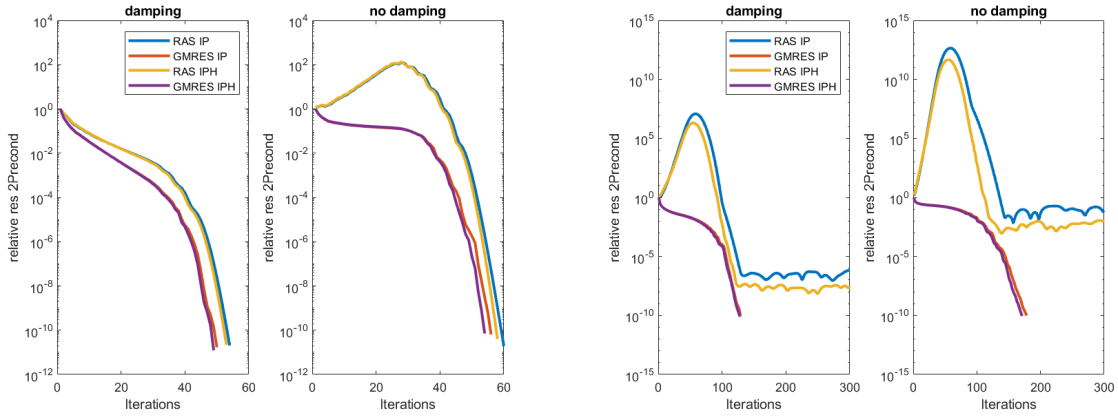


Figure 12: Residuals evolution for RAS and GMRES. Decomposition choice: $NX = 2$, $NT = 20$ in the first column and $NX = 5$, $NT = 20$ in the second one.

The idea of RAS can be brought back to life when using the pipelined version. In the case reported in Figure 13, we obtain the same convergence as GMRES but with more attention on the resources used: the smaller number of solved subdomains leads to a more lighter and powerful parallelization. We can observe how the residual travels with the moving subdomains' window S and in this way at each iteration we do not take in account residuals from distant subdomains that are not yet solved. We can appreciate also how the left boundary of S moves to the right once its corresponding subdomains have reached the tolerance.

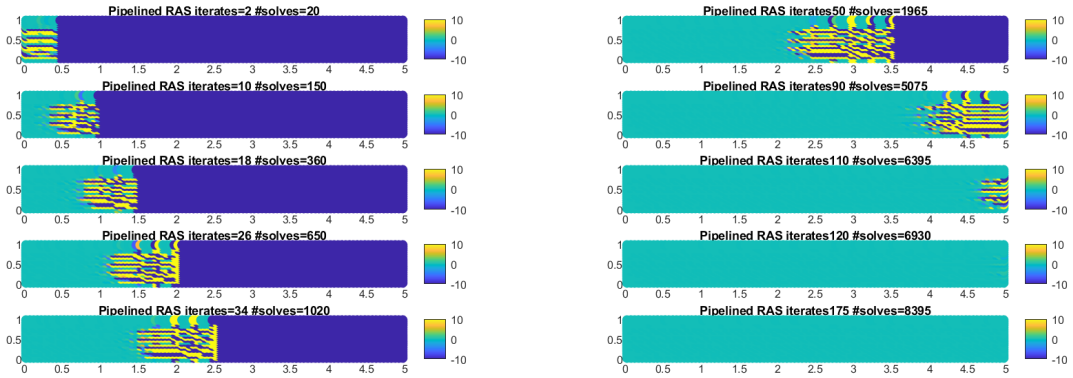


Figure 13: Residuals evolution for RAS Pipelined. Decomposition choice: $NX = 5$, $NT = 20$.

3.4 Characteristic line and subdomains shape

A further comment that we have not studied through numerical tests is the impact of the wave speed c, namely the slope of the characteristic lines, on the performance of the proposed domain decomposition methods. When dealing with domain decomposition techniques the choice in the number and shape of the subdomains can depend on different parameters, for example on the number of threads available for computing. Moreover, for this problem the choice of subdomains division can be driven also by considering the relation between the subdomains' size and the characteristic line arising from the wave equation.

In problem (1) the characteristic lines are straight lines with slope depending on the value of c. Considering in our case rectangular subdomains, Figure 14, the characteristic lines highlight a triangular region in which the solution depends only by the values at the bottom edge of that subdomain. The part of the subdomain out of the triangle will need information coming also from horizontal adjacent subdomains. Therefore several iterations of the domain decomposition technique are required to exchange useful information at the boundary and compute a solution in the region upper the triangle.

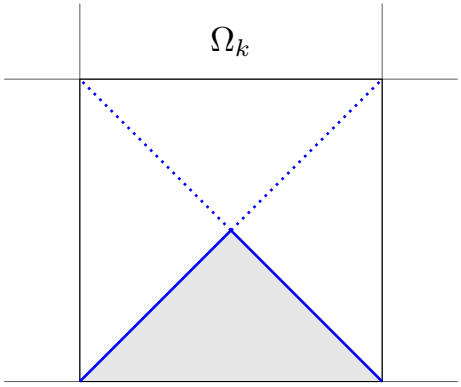
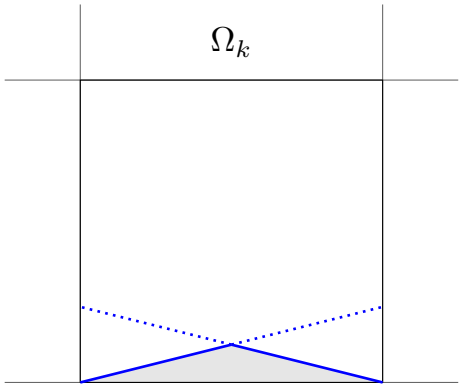
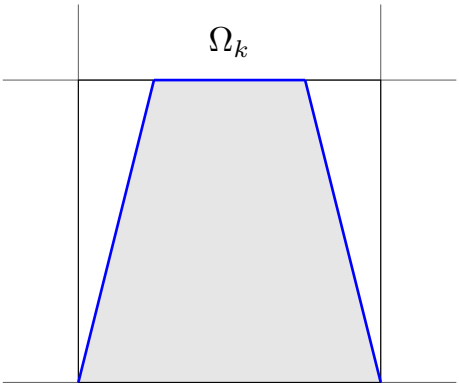


Figure 14: The blue lines are the characteristic lines which enclose a triangle shape in the subdomain Ω_k

We can identify two extreme cases which are not optimal. In Figure 15a the characteristic lines form a very low triangle and there is a great region in the upper part where the solution would need a lot of iteration to reach convergence because at the beginning the iterative method propagates mostly errors. On the other hand, it is not optimal either if it occurs a combination of characteristic lines and subdomain size such that there is not a triangular region as we can see in Figure 15b where the lines intersect the top edge before touching. In this scenario we expect a low number of iteration to reach a solution since most of the subdomain solution needs on local information, however at each iteration the flow of useful information is blocked at the top of the subdomain. For this reason, in the end, the overall convergence in the whole domain Ω is slower. A solution could be adopting a bigger subdomain size, for which we can include in it all the dependency domain.



(a) very tall subdomain when compared to the characteristic lines



(b) characteristic lines intersect the top edge of the subdomain

Figure 15: Two examples of possible configuration of characteristic lines in a subdomain

A good choice of subdomain size is therefore the one in Figure 14 where the characteristic lines are the diagonals of the subdomain. In this case a good portion of the subdomain is covered by the

triangular local dependency domain. Hence, all the useful information is not blocked. Finally, in [6] it is studied a new technique regarding the impact of the characteristic line that defines ad-hoc polygonal subdomains avoiding useless error propagation.

4 Conclusion

In this project we have presented a space-time approach for the wave equation in a simple framework. Starting from a discontinuous Galerkin discretization we have employed domain decomposition techniques based on a restricted additive Schwarz method. We focus our attention on the efficiency of these methods in order to solve this problem through parallel computing techniques. Pipelined RAS permits to solve the problem assinging at each thread much less subdomains to solve reducing the computational resources required.
This work lays the foundation for possible further developments where these domain decomposition techniques could be applied on more general domains with a polygonal space-time grid [7]. Moreover, RAS and its pipelined version could be used on others time dependent equations, always using a space-time approach. For example, it could be interesting to compare the behaviour of the wave equation with the one of the heat equation. Finally, a generalization in more space dimensions could be investigate.

References

[1] *A space-time discontinuous Galerkin method for the elastic wave equation*, Paola F. Antonietti, Ilario Mazzieri, Francesco Migliorini. Journal of Computational Physics, Volume 419, 2020

[2] *Optimized multiplicative, additive and restricted additive Schwarz preconditioning*, A. St-Cyr, M.J. Gander and S.J. Thomas, SIAM J. Sci. Comput., Vol. 29, No. 6, pp. 2402-2425, 2007

[3] *Analysis of Schwarz methods for discontinuous Galerkin discretizations*, HAJIAN, Soheil. Thèse de doctorat : Univ. Genève, 2015, no. Sc. 4795

[4] *Stabilization mechanisms in discontinuous Galerkin finite element methods*, F. Brezzi, B. Cockburn, L.D. Marini, and E. Suli. Computer Methods in Applied Mechanics and Engineering, 195 (2006), pp. 3293 - 3310.

[5] *Iterative methods and preconditioners for systems of linear equations*, Gabriele Ciaramella, Martin J Gander. Society for Industrial and Applied Mathematics, 2022.

[6] *Unmapped tent pitching schemes by waveform relaxation*, Gabriele Ciaramella, Martin J. Gander, Ilario Mazzieri. MOX Report Collection, 2022.

[7] *hp-Version space-time discontinuous Galerkin methods for parabolic problems on prismatic meshes*, Andrea Cangiani, Zhaonan Dong, Emmanuil H. Georgoulis. arXiv, 2016.

Acknowledgements

We would like to thank our tutors, Ilario Mazzieri and Gabriele Ciaramella for their continuous and special support throughout this work.