MOX-Report No. 27/2026

A hybrid reduced-order and high-fidelity discontinuous Galerkin Spectral Element framework for large-scale PMUT array simulations

Antonietti, P. F.; Abdalla, O. M. O.; Garroni, M. G.; Mazzieri, I.; Parolini, N.

MOX, Dipartimento di Matematica
Politecnico di Milano, Via Bonardi 9 - 20133 Milano (Italy)

mox-dmat@polimi.it https://mox.polimi.it

A hybrid reduced-order and high-fidelity discontinuous Galerkin Spectral Element framework for large-scale PMUT array simulations

Paola F. Antonietti[§], Omer M.O. Abdalla[*], Michelangelo G. Garroni[§], Ilario Mazzieri[§], and Nicola Parolini[§]

[§]MOX-Laboratory for Modeling and Scientific Computing, Department of Mathematics, Politecnico di Milano, Piazza Leondardo da Vinci 32, 20133 Milan, Italy.

[*]Department of Civil and Environmental Engineering, Politecnico di Milano, Piazza Leondardo da Vinci 32, 20133 Milan, Italy.

March 6, 2026

Abstract

Piezoelectric Micromachined Ultrasonic Transducers (PMUTs) are essential for next-generation ultrasonic sensing and imaging due to their bidirectional electromechanical behavior, compact design, and compatibility with low-voltage electronics. As PMUT arrays grow in size and complexity, efficiently modeling their coupled electromechanical-acoustic behavior becomes increasingly challenging. This work presents a novel computational framework that combines model order reduction with a Discontinuous Galerkin Spectral Element Method (DG-SEM) paradigm to simulate large PMUT arrays. Each PMUT's mechanical behavior is represented using a reduced set of vibration modes, which are coupled to an acoustic domain model to describe the full array. To further improve efficiency, a secondary acoustic domain is connected via DG interfaces, enabling non-conforming mesh refinement, with variable approximation order, and accurate wave propagation. The framework is implemented in the SPectral Elements in Elastodynamics with Discontinuous Galerkin (SPEED) software, an open-source, parallelized platform leveraging domain decomposition, high-order polynomials, METIS graph partitioning, and MPI for scalable performance. The proposed methodology addresses key challenges in meshing, supporting high-fidelity simulations for both PMUT transmission and reception phases. Numerical results demonstrate the framework's accuracy, scalability, and efficiency for large PMUT array simulations.

Keywords: PMUT, model order reduction, wave propagation, parallel computing

1 Introduction

Piezoelectricity is the property of certain materials to generate an electric potential when subjected to mechanical stress and, conversely, to deform elastically when an electrical voltage is applied. These phenomena are referred to as the direct and inverse piezoelectric effects, respectively. In Piezoelectric Micromachined Ultrasonic Transducers (PMUTs), this coupled mechanical behaviour enables both ultrasonic wave emission and reception: an applied alternating voltage induces diaphragm vibrations that radiate acoustic waves, while incident pressure waves deform the membrane and generate electrical signals. Thin piezoelectric films such as Lead Zirconate Titanate (PZT), Aluminum Nitride (AlN), and Scandium-doped Aluminum Nitride (ScAlN) are commonly integrated into multilayer diaphragms to achieve efficient electromechanical transduction [16].

With technological progress, systems increasingly demand higher miniaturization, complexity, and automation. Ultrasonic devices have followed this trend, particularly through the development of Micromachined Ultrasonic Transducers (MUTs) using Micro-Electro-Mechanical Systems (MEMS) technology, which allow compact, high-performance systems [6, 13]. Among them, PMUTs have gained significant attention due to their advantageous electrical properties and compatibility with low-voltage electronics [16]. Their Complementary Metal-Oxide Semiconductor compatibility enables integration of sensing and control electronics on a single substrate, supporting the development of miniaturized and efficient ultrasonic systems [6]. Figure 1 provides a representation of a multilayered PMUT stack cross-section.

Compared to bulk piezoelectric transducers, PMUTs benefit from batch microfabrication, resulting in compact arrays with improved yield and reduced cost. In contrast to capacitive MUTs, they do not require high bias voltages and exhibit lower impedance, simplifying electronic interfacing and improving energy efficiency [6, 13].

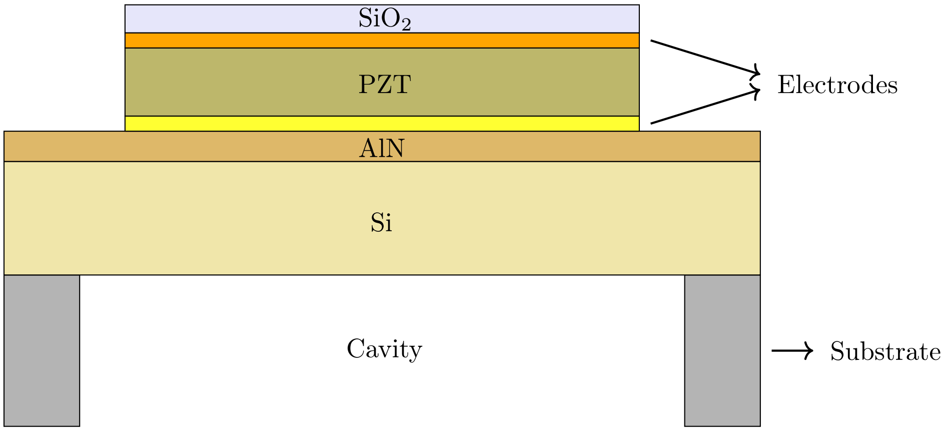


Figure 1: Cross-section of a representative PMUT stack, illustrating the multilayer structure in which each layer has a defined thickness and distinct material properties.

These advantages have enabled PMUT arrays to find applications in medical imaging, fingerprint sensing, sonar, non-destructive testing, flow measurement, and underwater communication [6, 13, 16].

As PMUT arrays grow in geometric complexity and face increasingly demanding operational constraints, the numerical models required to accurately simulate their behaviour become significantly more computationally inten sive. Consequently, there is a pressing need to develop innovative modeling strategies that deliver fast, accurate, and reliable tools capable of keeping up with the rapid evolution of PMUT technology [2].

The multiphysics nature of PMUT operation—combining structural deformation, piezoelectric coupling, electrical excitation, and acoustic radiation—makes modeling particularly challenging. Early analytical formulations based on classical plate and membrane theories provided closed-form estimates of resonance frequencies and modal behavior, but these simplified approaches cannot capture multilayer stacks, non-uniform electrode layouts, residual stresses, or fluid loading effects encountered in practical devices [6, 22].

To overcome these limitations, high-fidelity Finite Element Method (FEM) simulations have become the reference tool for detailed PMUT analysis. Three-dimensional multiphysics FEM models simultaneously solve structural mechanics and piezoelectric equations, enabling accurate prediction of mode shapes, displacement amplitudes, and electromechanical coupling coefficients across geometric and material parameters [12, 23]. Symmetry-based reductions and parametric sweeps are frequently employed to optimize diaphragm geometry and piezoelectric layer thickness [9, 24]. For realistic performance evaluation, structural models are often coupled with acoustic domains through pressure acoustics or fluid–structure interaction formulations. These coupled simulations allow estimation of radiated pressure fields, radiation impedance, directivity patterns, and bandwidth in air- or liquid-loaded conditions [10, 19, 22, 26]. Such analyses are crucial for air-coupled sensing and imaging arrays, where acoustic loading significantly affects resonance frequency and damping characteristics. Beyond linear frequency-domain analysis, nonlinear and transient multiphysics simulations have been explored to investigate complex dynamic phenomena. Time-domain coupling of piezoelectric, structural, and fluid equations has been used to model acoustic streaming and micropump actuation in ScAlN-based devices, revealing nonlinear mode interactions and even chaotic behavior under specific excitation regimes [15, 25]. These studies highlight the limitations of purely linear modal approximations when operating at large amplitudes.

While full 3D multiphysics FEM provides high predictive accuracy, it becomes prohibitively computationally expensive for large parameter sweeps and extended PMUT arrays. To address this issue, reduced-order modeling strategies have been developed. Modal superposition techniques project the electromechanical response onto a truncated basis of dominant eigenmodes extracted from high-fidelity simulations, significantly reducing computational cost while preserving accuracy [9, 16]. Equivalent circuit models based on Mason or Butterworth–Van Dyke formalisms [8] offer further simplification by translating electromechanical behavior into electrical analogs suitable for system-level analysis [7, 24]. At the array level, modeling complexity increases due to mutual acoustic coupling, mechanical crosstalk, and phase synchronization effects. Hybrid analytical–numerical formulations and reduced-dimensional acoustic models have been proposed to approximate inter-element interactions without explicitly resolving all structural degrees of freedom [7, 18, 27, 29]. Large-domain acoustic simulations typically incorporate absorbing boundary treatments such as perfectly matched layers (PMLs) to prevent artificial reflections [11, 14, 28].

Despite substantial progress, key challenges remain: balancing computational efficiency and predictive accuracy for large arrays, accounting for fabrication variability and residual stresses, capturing nonlinear behavior at high excitation levels, and ensuring stable acoustic–structure coupling in time-domain simulations. These limitations motivate the development of scalable numerical frameworks that combine model reduction with high-performance wave propagation solvers.

The main objective of this study is to explore computational strategies that can efficiently handle large array simulations involving an extensive number of degrees of freedom (DOFs). The main idea is to consider an efficient model order reduction approach derived from [16] to reduce the computational complexity associated with simulating large PMUT arrays. The proposed novel framework is based on first solving the three-dimensional eigenvalue piezoelectric problem on a reference PMUT, from which a reduced set of significant vibration modes is extracted. The mechanical response of each transducer is then expressed as a linear combination of these modal shapes. The modal displacements are interpolated within a purely acoustic domain that models the PMUT arrays as bi-dimensional membranes. The proposed framework is implemented and validated in SPEED [4], an open-source software developed for large-scale wave propagation phenomena in complex three-dimensional media. Our framework integrates the Discontinuous Galerkin Spectral Element Method (DG-SEM) with domain decomposition and high-order polynomial approximation to ensure geometric flexibility and high-order accuracy. Mesh partitioning and parallel communication rely on METIS [1] and MPI libraries, while explicit time integration ensures excellent scalability and performance on distributed-memory architectures.

The primary challenges addressed include an optimized mesh generation and partitioning strategy, as well as the optimization of algorithmic operations to ensure both high-performance computing and accuracy in the simulations. By improving these aspects, the proposed approach lead to scalable, high-fidelity modeling of complex PMUT arrays both in transmission (TX) and reception (RX) phases.

The remainder of this article is organized as follows. Section 2 formulates the acoustic problem coupled with the modal equations of the PMUT, addressing both the transmission and reception scenarios. This section also includes a discussion on Perfectly Matched Layer (PML) [11], highlighting its role in eliminating spurious reflections at the boundaries of the computational domain.

Section 3 presents the numerical discretization of the coupled problem in both space (using the Discontinuous Galerkin Spectral Element Method) and time with Newmark scheme. Section 4 presents the domain decomposition strategy employed for efficient mesh partitioning, which is critical for handling large-scale PMUT arrays. Section 5 focuses on the implementation optimization designed to accelerate computations involving non-conforming meshes. Section 6 presents representative numerical results that validate the methodology and demonstrate its performance for large PMUT array simulations, and includes the evaluation of the scalability of the proposed approach on parallel computing architectures. Finally, in Section 7 we draw some conclusions and discuss future perspectives.

2 Multi-physics problem formulation for large PMUT array simulations

In this section, we introduce the general formulation of the coupled problem. Let $\Omega = \Omega_{\text{in}} \cup \Omega_{\text{out}} \cup \Omega_{\text{PML}}$ be a three-dimensional open Lipschitz domain, and $T \in \mathbb{R}^+$ be the final time. The boundary of the acoustic domain Ω is decomposed as $\partial\Omega = \Gamma_{\text{PMUT}} \cup \Gamma_{\text{N}} \cup \Gamma_{\text{ABC}}$, consisting of disjoint $\Gamma_{\text{PMUT}}, \Gamma_{\text{N}}$ and Γ_{ABC} parts. Γ_{PMUT} is the portion of the boundary where the action of PMUT membranes is accounted for, Γ_{N} is where homogeneous Neumann conditions are imposed, and Γ_{ABC} is the remaining part of the boundary where absorbing condition are considered. Additionally, Γ_{I} represents an internal interface between the sub-domain Ω_{in} and Ω_{out}, cf. Figure 2. The splitting of Ω_{in} and Ω_{out} enables the treatment of the two subdomains as potentially non-conforming regions while simultaneously allowing the assignment of distinct material properties within each subdomain. Moreover, the inclusion of the subdomain Ω_{PML} ensures the proper implementation of a Perfectly Matched Layer, thereby enabling the effective absorption of outgoing waves and the mitigation of artificial reflections at the truncated computational boundary.

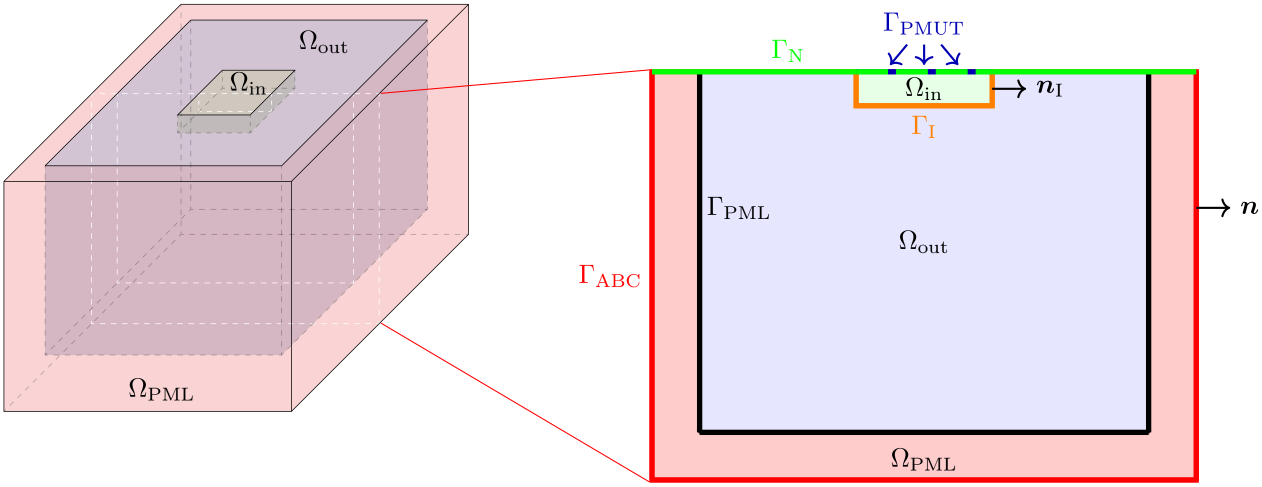


Figure 2: A three-dimensional view of the domain decomposition (left) and the corresponding two-dimensional slice with boundary labels (right) for (1).

We consider the following problem

$$\begin{cases} p_{\text{tt}} + \alpha p_{\text{t}} + \beta p = \nabla \cdot (c^2 \nabla p) + \nabla \cdot \boldsymbol{\Phi} - \gamma \psi, & \text{in } \Omega \times (0,T]\,, \\ \boldsymbol{\Phi}_{\text{t}} = Z_1 \boldsymbol{\Phi} + c^2 Z_2 \nabla p + c^2 Z_3 \nabla \psi, & \text{in } \Omega \times (0,T]\,, \\ \psi_{\text{t}} = p, & \text{in } \Omega \times (0,T]\,, \\ p_{\text{t}} = p = \psi = 0, & \text{in } \Omega \times \{0\}\,, \\ \boldsymbol{\Phi} = \boldsymbol{0}, & \text{in } \Omega \times \{0\}\,, \\ \nabla p \cdot \boldsymbol{n} = \dfrac{1}{c} p_{\text{t}}, & \text{on } \Gamma_{\text{ABC}} \times [0,T]\,, \\ \nabla p \cdot \boldsymbol{n} = 0, & \text{on } \Gamma_{\text{N}} \times [0,T]\,, \\ \nabla p \cdot \boldsymbol{n} = -\kappa \displaystyle\sum_m \ddot{q}_m \boldsymbol{U}_m \cdot \boldsymbol{n}, & \text{on } \Gamma_{\text{PMUT}} \times [0,T]\,, \\ p_{|\Omega_{\text{in}}} = p_{|\Omega_{\text{out}}}, & \text{on } \Gamma_{\text{I}} \times [0,T]\,, \\ \left(\nabla p_{|\Omega_{\text{in}}} - \nabla p_{|\Omega_{\text{out}}} \right) \cdot \boldsymbol{n}_{\text{I}} = 0, & \text{on } \Gamma_{\text{I}} \times [0,T]\,, \end{cases} \tag{1}$$

where $p(\boldsymbol{x}, t)$ denotes the acoustic pressure, $\boldsymbol{\Phi}(\boldsymbol{x}, t)$ represents the (vector) variable that introduces additional diffusion to the problem, and $\psi(\boldsymbol{x}, t)$ is an auxiliary variable enabling the formulation of the PML damping in a purely local differential form in time. The term $\ddot{q}_m$ is computed from the following ordinary differential equation

$$\begin{cases} \ddot{q}_m = -\omega_m^2 q_m + \displaystyle\int_{\Gamma_{\text{PMUT}}} p \boldsymbol{U}_m \cdot \boldsymbol{n} \, \mathrm{d}\sigma + \eta_m \phi, & \text{in } (0,T]\,, \\ \phi = \phi_0(t) \mathbb{1}_{\{0 \le t \le \bar{t}\}}(t) + \dfrac{1}{C} \displaystyle\sum_m \eta_m q_m \mathbb{1}_{\{\bar{t} < t \le T\}}(t), & \text{in } [0,T]\,, \\ q_m = \dot{q}_m = 0, & \text{at } t = 0, \end{cases} \tag{2}$$

where $\boldsymbol{U}_m$ encodes the mechanical displacement of the PMUTs with the modal coordinates q_m, while $\phi(t)$ is a low-voltage activation function.

The acoustic problem is thus coupled with the piezo-electro-mechanical problem describing the behavior of PMUTs. Due to the large number of membranes typically employed in realistic scenarios, solving the full physics of the PMUTs may be computationally unaffordable. Thus, it is preferable to consider model order reduction techniques, where the behavior of bi-dimensional PMUT membranes is described by ordinary differential equations. Here, we consider the model discussed in [16], whose aim is to reduce the constitutive law of piezoelectricity by solving an eigenvalue problem to represent the displacement nodal values on a reduced basis of modal vectors.

Notice that the equations describing the PMUTs' contribution in (1)-(2) can be expanded as follows:

$$\begin{cases} \ddot{q}_{k,m} = -\omega_{k,m}^2 q_{k,m} + \displaystyle\int_{\gamma_k} p\boldsymbol{U}_{k,m}\cdot\boldsymbol{n}\,\mathrm{d}\sigma + \eta_{k,m}\phi_k, & \text{in } (0,T]\,, \\ \phi_k = \phi_{k,0}(t)\mathbb{1}_{\{0\le t\le \bar{t}\}}(t) + \dfrac{1}{C}\displaystyle\sum_{m} \eta_{k,m}q_{k,m}\mathbb{1}_{\{\bar{t}<t\le T\}}(t), & \text{in } [0,T]\,, \\ \nabla p\cdot\boldsymbol{n} = -\kappa\displaystyle\sum_{k,m}\ddot{q}_{k,m}\boldsymbol{U}_{k,m}\cdot\boldsymbol{n}, & \text{on } \gamma_k\times[0,T]\,, \\ q_{k,m}=\dot{q}_{k,m}=0, & \text{at } t=0, \end{cases} \tag{3}$$

where $k \in \{1,\dots,N_p\}$, and $m \in \{1,\dots,N_m\}$ represent the k-th PMUT and the m-th mode, respectively, while $\bigcup_{1\le k\le N_p} \gamma_k = \Gamma_{\mathrm{PMUT}}$. Precisely, if $\boldsymbol{U}_k(\mathbf{x},t)$ denotes the mechanical displacement of the k-th PMUT in the reduced-order approach, one has

$$\boldsymbol{U}_k(\mathbf{x},t) = \sum_{m=1}^{N_m} q_{k,m}(t)\,\boldsymbol{U}_{k,m}(\mathbf{x}),$$

where $\boldsymbol{U}_{k,m}$ is the m-th mode shape, and $q_{k,m}(t)$ is the corresponding generalized modal coordinate. Projecting the mechanical balance equations onto each retained mode, one obtains the ordinary differential equation for $q_{k,m}$ described in (3). The coefficient $\omega_{k,m}$ is the natural frequency of mode m for the k-th PMUT, p is the acoustic pressure on the PMUT's surface γ_k, and $\eta_{k,m}$ is the electro-mechanical coupling coefficient relating the applied potential ϕ_k to mode m. Due to their nature, PMUTs are able to capture both the transmission (TX) and reception (RX) phases. Setting $\bar{t} \in (0,T)$, for $t \le \bar{t}$ the voltage $\phi_{k,0}$ is applied to produce an acoustic pressure, while for $t > \bar{t}$ it is directly produced by the pressure on the surface γ_k. The constant C in (3) models the piezoelectric capacitance. For any $k \in \{1,...,N_p\}$, the system (3) represents the coupling condition at the boundary γ_k, i.e., at Γ_{PMUT}. In (3), $\kappa = \rho c^2$ is the bulk modulus and $\rho = \rho(\boldsymbol{x})$ is the acoustic density, c being the sound speed of the medium.

The first three equations of (1) describe the acoustic wave propagation and incorporate a perfectly matched layer, i.e. Ω_{PML}, which effectively attenuates nonphysical reflections at the boundary. The derivation of the PML conditions can be found in [11, 14]. The scalars $\alpha = (\zeta_1+\zeta_2+\zeta_3)$, $\beta = (\zeta_1\zeta_2+\zeta_2\zeta_3+\zeta_3\zeta_1)$, $\gamma = \zeta_1\zeta_2\zeta_3$, and the matrices

$$\begin{aligned} Z_1 &= \mathrm{diag}(-\zeta_1,-\zeta_2,-\zeta_3), \\ Z_2 &= \mathrm{diag}(\alpha-2\zeta_1,\alpha-2\zeta_2,\alpha-2\zeta_3), \\ Z_3 &= \mathrm{diag}(\zeta_2\zeta_3,\zeta_3\zeta_1,\zeta_1\zeta_2), \end{aligned}$$

are related to the specific damping profile $\zeta_i(x_i), i \in \{1,2,3\}$ selected in the three spatial directions. The idea of the PML equations is to provide additional damping and diffusivity to the acoustic equation to shrink the waves' amplitude before they touch the external boundary of the PML, i.e., Γ_{ABC}. In this paper, in agreement with [11], we choose the following damping profile

$$\zeta_i(x_i) = \tilde{\zeta}_i\left(\frac{(|x_i|-h_i)}{\ell_i} - \frac{\sin\left(\frac{2\pi(|x_i|-h_i)}{\ell_i}\right)}{2\pi}\right)\mathbb{1}_{\{h_i\le|x_i|\le h_i+\ell_i\}}(x_i),$$

where, for each $i-th$ direction, $i \in \{1,2,3\}$, h_i denotes half of the thickness of the propagation domain and ℓ_i the thickness of the PML. The function $\zeta_i(x_i)$ vanishes with continuity at the interface Γ_{PML} with the purely acoustic domain, and the constant $\tilde{\zeta}_i$ is related to the sound speed c in the acoustic medium, the thickness of the PML, and the relative reflection coefficient R of the acoustic wave at the PML interface, as follows

$$\tilde{\zeta}_i = \frac{c}{\ell_i}\log\left(\frac{1}{R}\right), \quad i = 1,2,3.$$

The inverse of the reflection coefficient R has to be selected large enough to avoid the wave being reflected at the interface, but it has to be kept small enough to have a less steep function $\zeta_i(x_i)$ in the transition region between the interface Γ_{PML} and the end of the PML. In the following, $R \in [10^{-5}, 10^{-4}]$. A first-order absorbing boundary condition is imposed on Γ_{ABC}, while a hard wall boundary condition is set on Γ_{N}. Continuity of the pressure and of its normal derivative is imposed on Γ_{I}. The terms $\boldsymbol{n}$, and $\boldsymbol{n}_{\mathrm{I}}$ represent the unit outer normal to the domain Ω and to the domain Ω_{in}, respectively. The full set of unknowns and parameters defined in equations (1)-(2) is reported in Table 1.

Unknown	Unit
p	[Pa]
ψ	[Pa s]
$\boldsymbol{\Phi}$	[Pa m s^{-2}]
q_m	[m]
ϕ	[V]

Acoustic parameter	Unit
α, Z_1, Z_2	[s^{-1}]
β, Z_3	[s^{-2}]
γ	[s^{-3}]
c	[m s^{-1}]

PMUTs parameter	Unit
ω_m	[rad s^{-1}]
η_m	[C m^{-1}]
κ	[Pa]
C	[F]
ϕ_0	[V]

Table 1: The three tables report the unknowns and the parameters of equations (1)–(2) with the corresponding units of measurement.

3 Space discretization: Discontinous Galerkin Spectral Element Method

The numerical discretization of (1) is based on a Discontinuous Galerkin Spectral Element Method, which allows both for high-order polynomials of SEM and grid flexibility of the DG method [4].

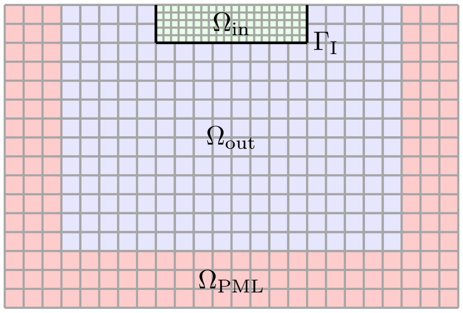


Figure 3: Example of mesh employed in the DG-SE approach. Different mesh sizes and polynomial degrees are employed for Ω_{in} and Ω_{out}, resulting in a non-conforming mesh.

We next discuss the derivation of the DG-SEM formulation by assuming, without loss of generality, that $\Omega_{\text{PML}} = \emptyset$. Let $\Omega = \Omega_{\text{in}} \cup \Omega_{\text{out}}$ be a polygonal domain and consider its partition $\mathcal{T}_h = \mathcal{T}_{\text{in}} \cup \mathcal{T}_{\text{out}}$, s.t. $\mathcal{T}_{\text{in}} \cap \mathcal{T}_{\text{out}} = \emptyset$ with $\mathcal{T}_{\text{in}}$ and $\mathcal{T}_{\text{out}}$ containing hexahedral elements K_{in} and K_{out} of different granularity h_{in} and h_{out}, respectively. By defining N_{in}, and N_{out}, the number of elements in each partitions, one has $\mathcal{T}_{\text{in}} = \left\{K^j_{\text{in}}\right\}_{j=1}^{N_{\text{in}}}$ and $\mathcal{T}_{\text{out}} = \left\{K^j_{\text{out}}\right\}_{j=1}^{N_{\text{out}}}$, respectively. Let $\hat{K} = [-1,1]^3$ be the reference element, and let $\boldsymbol{X}_{K_{\text{a}}} : \hat{K} \to K_{\text{a}}$ be a trilinear invertible map with positive Jacobian $\boldsymbol{J}_{K_{\text{a}}}$ such that $\forall\, K_{\text{a}} \in \mathcal{T}_{\text{a}}$, $K_{\text{a}} = \boldsymbol{X}_{K_{\text{a}}}(\hat{K})$, for a $= \{\text{in}, \text{out}\}$, with $\boldsymbol{X}^{-1}_{K_{\text{a}}} : K_{\text{a}} \to \hat{K}$. Next, we define the internal faces as $F = \overline{K}^j_{\text{in}} \cap \overline{K}^k_{\text{out}}$ for certain $j \in \{1, ..., N_{\text{in}}\}$ and $k \in \{1, ..., N_{\text{out}}\}$ and collect them in $\mathcal{F}^I$. Similarly, we define the boundary faces $F = \overline{K}^{j_{\text{a}}}_{\text{a}} \cap \overline{\partial\Omega}$ for certain $j_{\text{a}} \in \{1, ..., N_{\text{a}}\}$ and collect them in $\mathcal{F}^B$. Moreover, we split $\mathcal{F}^B$ as $\mathcal{F}^B = \mathcal{F}^B_{\text{N}} \cup \mathcal{F}^B_{\text{ABC}} \cup \mathcal{F}^B_{\text{PMUT}}$, where Neumann, absorbing, and coupling conditions (with PMUT membranes) are imposed, respectively. Note that boundary faces are compliant with this subdivision.
Consider now two adjacent elements $K^- \in \mathcal{T}_{\text{out}}$ and $K^+ \in \mathcal{T}_{\text{in}}$, sharing a face $F \in \mathcal{F}^I$ and define $\boldsymbol{n}^+$ and $\boldsymbol{n}^-$ to be the outward unit normal vectors to the face F from K^+ and K^-, respectively. Figure 4 provides a visual representation of two adjacent elements.

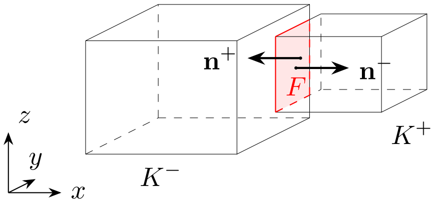


Figure 4: Representation of two adjacent hexahedra K^+ and K^- with the shared face F and the corresponding normal vectors $\boldsymbol{n}^+$ and $\boldsymbol{n}^-$.

For each face $F \in \mathcal{F}^I$ and for any sufficiently regular scalar function u and vector $\boldsymbol{v}$ we define the jump operator

$[\![\cdot]\!]$ and the average operator $\{\!\!\{\cdot\}\!\!\}$ by

$$\{\!\!\{u\}\!\!\} = \frac{1}{2}(u^+ + u^-), \quad [\![u]\!] = u^+\boldsymbol{n}^+ + u^-\boldsymbol{n}^-,$$

$$\{\!\!\{\boldsymbol{v}\}\!\!\} = \frac{1}{2}(\boldsymbol{v}^+ + \boldsymbol{v}^-), \quad [\![\boldsymbol{v}]\!] = \boldsymbol{v}^+ \cdot \boldsymbol{n}^+ + \boldsymbol{u}^- \cdot \boldsymbol{n}^-,$$

where ± denotes the trace on u and $\boldsymbol{v}$ taken within the interior of $K^\pm$, respectively. Similarly, for each face $F \in \mathcal{F}^B$ we set $\{\!\!\{u\}\!\!\} = u$, $[\![u]\!] = u\boldsymbol{n}$ and $\{\!\!\{\boldsymbol{v}\}\!\!\} = \boldsymbol{v}$, $[\![\boldsymbol{v}]\!] = \boldsymbol{v} \cdot \boldsymbol{n}$.
For $\mathrm{a} = \{\mathrm{in}, \mathrm{out}\}$ we consider the space

$$V_{\mathrm{a}}^{r_{\mathrm{a}}} = \left\{ v \in \mathcal{C}^0\left(\overline{\Omega}_{\mathrm{a}}\right) : v\big|_{K_{\mathrm{a}}} \circ \boldsymbol{X}_{K_{\mathrm{a}}}^{-1} \in \mathbb{Q}_{r_{\mathrm{a}}}(\hat{K}), \quad \forall K_{\mathrm{a}} \in \mathcal{T}_{\mathrm{a}} \right\},$$

where $\mathbb{Q}_{r_{\mathrm{a}}}(\hat{K})$ is the space of polynomials of degree less than or equal to $r_{\mathrm{a}} \geq 1$ in each coordinate direction. Next, we introduce

$$V^{\mathrm{DG}} = \left\{ v \in L^2(\Omega) : v\big|_{\Omega_{\mathrm{a}}} \in V_{\mathrm{a}}^{r_{\mathrm{a}}} \right\},$$

and provide the semi-discrete formulation: $\forall\, t \in (0, T]$ find $p_h = p_h(t) \in V^{\mathrm{DG}}$ s.t.

$$m(\ddot{p_h}, v_h) + a(p_h, v_h) + d(\dot{p}_h, v_h) = -\kappa \sum_{m=1}^{N_m} \int_{\Gamma_{\mathrm{PMUT}}} \ddot{q}_m \boldsymbol{U}_m \cdot \boldsymbol{n} v_h \,\mathrm{d}\sigma, \quad \forall\, v_h \in V^{\mathrm{DG}}, \tag{4}$$

with $p_h(0) = \dot{p}_h(0) = 0$ and where, for any $1 \leq m \leq N_m$, it holds

$$\ddot{q}_m(t) + \omega_m^2 q_m(t) = \int_{\Gamma_{\mathrm{PMUT}}} p_h(t) \boldsymbol{U}_m \cdot \boldsymbol{n} \,\mathrm{d}\sigma + \eta_m \phi(t) \quad \forall\, t \in (0, T]. \tag{5}$$

For any $\omega, v \in V^{\mathrm{DG}}$, the bilinear forms in (4) are defined as:

$$m(\omega, v) = \sum_{K_{\mathrm{s}} \in \mathcal{T}_{\mathrm{in}} \cup \mathcal{T}_{\mathrm{out}}} \int_{K_{\mathrm{s}}} \omega v \,\mathrm{d}\Omega, \tag{6}$$

$$\begin{aligned} a(\omega, v) = & \sum_{K_{\mathrm{s}} \in \mathcal{T}_{\mathrm{in}} \cup \mathcal{T}_{\mathrm{out}}} \int_{K_{\mathrm{s}}} c^2 \nabla\omega \cdot \nabla v \,\mathrm{d}\Omega - \sum_{F \in \mathcal{F}^I} \int_F \{\!\!\{c^2 \nabla\omega\}\!\!\} \cdot [\![v]\!] \,\mathrm{d}\sigma \\ & - \sum_{F \in \mathcal{F}^I} \int_F \{\!\!\{c^2 \nabla v\}\!\!\} \cdot [\![\omega]\!] \,\mathrm{d}\sigma + \sum_{F \in \mathcal{F}^I} \int_F \gamma_F \, [\![\omega]\!] \cdot [\![v]\!] \,\mathrm{d}\sigma, \end{aligned} \tag{7}$$

$$d(\omega, v) = \sum_{F \in \mathcal{F}^{\mathrm{B}}_{\mathrm{ABC}}} \int_F c\omega v \,\mathrm{d}\sigma, \tag{8}$$

where the parameter $\gamma_F = \alpha \bar{c}^2 \dfrac{\max(r_{\mathrm{in}}, r_{\mathrm{out}})^2}{\min(h_{\mathrm{in}}, h_{\mathrm{out}})}$, with $\bar{c}$ being the harmonic average of c on F and with α a sufficiently large positive parameter at our disposal, see, e.g., [5].

3.1 Algebraic formulation and time integration

Let $\{\psi_i\}_{i=1}^{N_h}$ be the Lagrangian basis functions for V^{DG}, cf. [20], and set $v_h(\boldsymbol{x}) = \psi_i(\boldsymbol{x})$, and $p_h(\boldsymbol{x}, t) = \sum_{j=1}^{N_h} p_j(t)\psi_j(\boldsymbol{x})$ in (6)–(8). Rearranging the terms in matrix form, the semi-discrete algebraic formulation of problem (4)-(5) reads: for any $t \in (0, T]$ find $\boldsymbol{p}(t) = (p_1(t), ..., p_{N_h}(t))^T \in \mathbb{R}^{N_h}$ and $\boldsymbol{q}(t) = (q_1(t), ..., q_m(t))^T \in \mathbb{R}^{N_m}$, s.t.

$$M\ddot{\boldsymbol{p}}(t) + K\boldsymbol{p}(t) + C\dot{\boldsymbol{p}}(t) = \boldsymbol{f}(\ddot{\boldsymbol{q}}(t)), \tag{9}$$

$$\ddot{\boldsymbol{q}}(t) + W\boldsymbol{q}(t) = \boldsymbol{g}(\boldsymbol{p}(t), \phi(t)), \tag{10}$$

with $\boldsymbol{p} = \dot{\boldsymbol{p}} = \boldsymbol{0}$, $\boldsymbol{q} = \dot{\boldsymbol{q}} = \boldsymbol{0}$. In (9), $M \in \mathbb{R}^{N_h \times N_h}$ is the diagonal mass matrix associated with $m(\cdot, \cdot)$, i.e., the matrix representation of the L^2 inner product, evaluated with reduced integration, $K \in \mathbb{R}^{N_h \times N_h}$ is the stiffness matrix related to $a(\cdot,\cdot)$, $C \in \mathbb{R}^{N_h \times N_h}$ is the matrix encoding the absorbing conditions $d(\cdot,\cdot)$, while $\boldsymbol{f} \in \mathbb{R}^{N_h}$ is the vector describing the forcing term acting on Γ_{PMUT}, i.e.,

$$\boldsymbol{f}_i(t) = -\kappa \sum_{m=1}^{N_m} \ddot{q}_m(t) \int_{\Gamma_{\mathrm{PMUT}}} \boldsymbol{U}_m(t,\boldsymbol{x}) \cdot \boldsymbol{n}\psi_i(\boldsymbol{x})\,\mathrm{d}\sigma, \quad 1 \leq i \leq N_h. \tag{11}$$

In (10), $W \in \mathbb{R}^{N_m \times N_m}$ is the matrix $W = \mathrm{diag}(\omega_1^2, ..., \omega_{N_m}^2)$ and $\boldsymbol{g} \in \mathbb{R}^{N_m}$ is such that

$$g_m(t) = \sum_{j=1}^{N_h} p_j(t) \int_{\Gamma_{\mathrm{PMUT}}} \psi_j(\boldsymbol{x})\boldsymbol{U}_m(\boldsymbol{x}) \cdot \boldsymbol{n}\,\mathrm{d}\sigma + \eta_m \phi(t) \quad 1 \leq m \leq N_m. \tag{12}$$

Next, we let $S \in \mathbb{N}^+$ be the number of sub-intervals of length Δt in which the time interval $[0,T]$ is divided in such a way that $t^n = n\Delta t, n \in \{0,\dots,S\}$. The equations (9)-(10) can be discretized with the following second-order predictor-corrector Newmark scheme, shown in Algorithm 1. Note that a similar scheme is employed when a PML boundary is introduced in the computational model.

The Newmark time-integration method is employed with $\gamma = 1/2$ and $\beta = 0$. For this choice, the scheme is second-order accurate, and the corresponding update relations reduce to the explicit leapfrog formulation. Since the method is explicit, it is conditionally stable under the CFL condition $\Delta t \lesssim h_{\min}$, being $h_{\min}$ the minimum diameter among the mesh elements [21].

Algorithm 1 Newmark time integration scheme

```
Set $\boldsymbol{q}^0 = \dot{\boldsymbol{q}}^0 = \ddot{\boldsymbol{q}}^0 = \mathbf{0}$ and $\boldsymbol{p}^0 = \dot{\boldsymbol{p}}^0 = \ddot{\boldsymbol{p}}^0 = \mathbf{0}$ and compute the voltage $\phi^0$
for $n = 0, ..., S$ do
    Predictor (PMUT): set $\boldsymbol{q}^{n+1} \leftarrow \boldsymbol{q}^n + \Delta t\,\dot{\boldsymbol{q}}^n + \frac{1}{2}\Delta t^2\,\ddot{\boldsymbol{q}}^n$
    Predictor (PMUT): set $\dot{\boldsymbol{q}}^{n+1} \leftarrow \dot{\boldsymbol{q}}^n + \frac{1}{2}\Delta t\,\ddot{\boldsymbol{q}}^n$
    Solve (PMUT):          $\ddot{\boldsymbol{q}}^{n+1} \leftarrow -W\boldsymbol{q}^{n+1} + \boldsymbol{g}(\boldsymbol{p}^n, \phi^n)$
    Corrector (PMUT):      $\dot{\boldsymbol{q}}^{n+1} \leftarrow \dot{\boldsymbol{q}}^{n+1} + \frac{1}{2}\Delta t\,\ddot{\boldsymbol{q}}^{n+1}$
    Predictor (acoustic): set $\boldsymbol{p}^{n+1} \leftarrow \boldsymbol{p}^n + \Delta t\,\dot{\boldsymbol{p}}^n + \frac{1}{2}\Delta t^2\,\ddot{\boldsymbol{p}}^n$,
    Predictor (acoustic): set $\dot{\boldsymbol{p}}^{n+1} \leftarrow \dot{\boldsymbol{p}}^n + \frac{1}{2}\Delta t\,\ddot{\boldsymbol{p}}^n$
    Solve (acoustic):      $\ddot{\boldsymbol{p}}^{n+1} \leftarrow M^{-1}(\boldsymbol{f}(\ddot{\boldsymbol{q}}^{n+1}) - K\boldsymbol{p}^{n+1} - C\dot{\boldsymbol{p}}^{n+1})$
    Corrector (acoustic):  $\dot{\boldsymbol{p}}^{n+1} \leftarrow \dot{\boldsymbol{p}}^{n+1} + \frac{1}{2}\Delta t\,\ddot{\boldsymbol{p}}^{n+1}$
    Update: $\boldsymbol{q}^n \leftarrow \boldsymbol{q}^{n+1}$,   $\dot{\boldsymbol{q}}^n \leftarrow \dot{\boldsymbol{q}}^{n+1}$,   $\ddot{\boldsymbol{q}}^n \leftarrow \ddot{\boldsymbol{q}}^{n+1}$
    Update: $\boldsymbol{p}^n \leftarrow \boldsymbol{p}^{n+1}$,   $\dot{\boldsymbol{p}}^n \leftarrow \dot{\boldsymbol{p}}^{n+1}$,   $\ddot{\boldsymbol{p}}^n \leftarrow \ddot{\boldsymbol{p}}^{n+1}$
end for
```

4 Parallel implementation strategies for PMUT simulations

In the context of High-Performance Computing, achieving competitive runtimes for large-scale simulations is of paramount importance. This requires fully leveraging the computational capabilities of parallel codes on multi-core clusters. A key challenge in this context is the efficient distribution of computational workload across cores, which often involves adopting domain-partitioning strategies tailored to the specific requirements of the simulation. In this work, we address a particularly challenging load-balancing problem arising from the need to account for multiple constraints simultaneously. Specifically, our goal is to balance the computational cost associated with the

i) boundary conditions for the PMUTs-array;

ii) acoustic propagation in $\Omega_{\mathrm{out}} \cup \Omega_{\mathrm{PML}}$;

iii) connectivity setup for non matching grids on Γ_{I}.

Additionally, we aim to ensure that the membranes representing the PMUTs, namely the boundary Γ_{PMUT}, are not split across multiple cores, as such fragmentation can degrade both performance and efficiency due to increased inter-process communication and loss of structural coherence. To this end, we propose a tailored partitioning strategy that respects the structural integrity of the PMUTs while distributing the simulation load evenly across processing units. This approach enhances the scalability of the simulation framework and ensures efficient utilization of HPC resources without compromising the accuracy of the underlying discretization.

4.1 Setup of the computational domain structure

The computational domain considered in the present case study consists of three main regions. The first one, which is referred to as *inner domain*, Ω_{in}, contains the PMUT array and constitutes the region of primary geometrical and physical interest. The second one, which is referred to as *outer domain*, Ω_{out}, represents the acoustic wave propagation region. The third one, Ω_{PML}, describes the external diffusive layer where Perfectly Matched Layer conditions are applied in order to mimic an unbounded medium.

For geometric fidelity and accuracy requirements associated with the vibrating membranes, the grid in Ω_{in} must be highly refined. Imposing a mesh of comparable resolution throughout Ω_{out} would lead to a prohibitive number of elements—potentially on the order of billions—thus rendering the computation intractable with standard computational resources. To reduce the overall number of degrees of freedom while preserving the required accuracy near the membranes, a coarser discretization is therefore adopted in Ω_{out}.

Accordingly, the inner and the outer subdomains are meshed independently, allowing for local refinement in Ω_{in} and a coarser mesh in the propagation region Ω_{out}. Conformity is instead imposed between Ω_{out} and Ω_{PML} to ensure a consistent treatment of the absorbing layer. The use of distinct discretizations in Ω_{in} and Ω_{out} results in a globally nonconforming mesh. To properly couple the two regions, an internal interface Γ_{I} is introduced between Ω_{in} and Ω_{out}, enabling the matching of the finer and coarser meshes across the subdomain boundary.

4.1.1 Load balancing strategy for $\Omega_{\text{out}} \cup \Omega_{\text{PML}}$

The two domains Ω_{out} and Ω_{PML} are partitioned directly using Metis 5.1.0 [1]. The idea is to exploit multiple times a Metis routine, namely `METIS_PartMeshDual`, that receives as input the set of hexahedra composing the two regions and returns the ownership of the mesh elements to the computational cores. The routine can be directly applied to the full set of mesh elements composing Ω_{out} and Ω_{PML} if the primary interest is to balance the number of hexahedra among the unit processes. However, in most cases, we will obtain an unbalanced workload for the processors, since not all processors will compute the integral terms on $\mathcal{F}^I$ in (7), cf. Figure 5 (left).

To overcome this bottleneck, we first split Ω_{out} into two subdomains: Ω_{DG} containing the hexahedral elements that share a face with Γ_{I} and the remaining ones. Next, to equally distribute elements over processors, we call three times `METIS_PartMeshDual`: one for Ω_{PML}, one for Ω_{DG} and one for the remaining part $\Omega_{\text{out}} \setminus \Omega_{\text{DG}}$. Figure 5 shows the Ω_{DG} layer connected to the outer domain Ω_{out} (middle) and a schematic representation of the Metis partition (right).

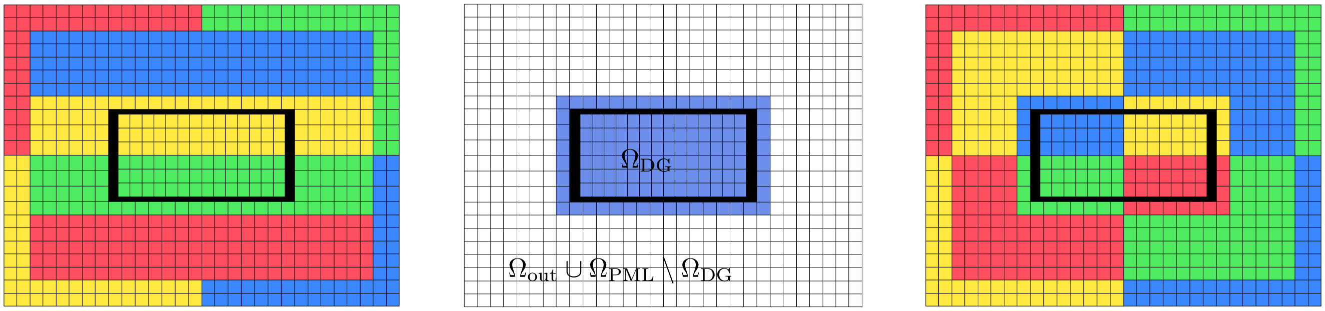

Ω_{DG}
$\Omega_{\text{out}} \cup \Omega_{\text{PML}} \setminus \Omega_{\text{DG}}$

Figure 5: Schematic illustration of the outer domain decomposition: non-optimized partitioning (left), DG layer Ω_{DG} (middle), optimal partitioning (right). The set of processors is $\{P_1, P_2, P_3, P_4\}$ = {🔵, 🟡, 🔴, 🟢}.

4.1.2 Mesh generation and load balancing strategy for Ω_{in}

The concept behind the mesh generation for the inner domain is to define a set of fundamental unit blocks that can be combined to form the complex membrane pattern of the PMUT array. In general, the membranes are not arranged in a Cartesian x-y grid but are only aligned along the x-direction, as shown in Figure 6. If we define *row* the set of membranes aligned along x and *column* the set of PMUTs that winds along y, then each row in the array is offset relative to the previous one by a distance determined by the pitch of the PMUT, d_p, that is, the distance of the centers of two neighboring membranes, see Figure 6.

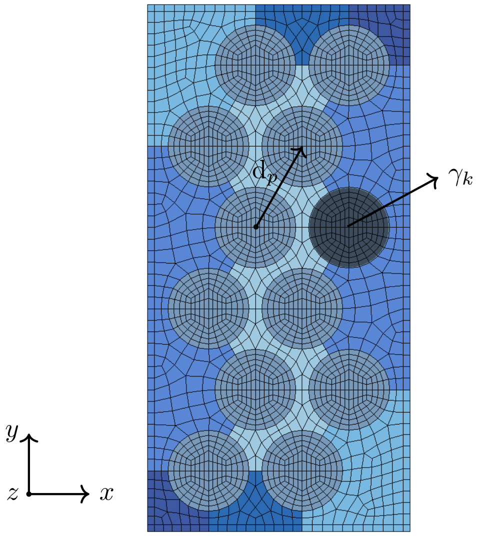


Figure 6: Top view of a minimal example of inner domain Ω_{in}. The elementary structures originating Ω_{in} are depicted with different colors. The circular regions represent the membranes γ_k belonging to Γ_{PMUT}, while the remaining part is Γ_{N}.

While element partitioning for $\Omega_{\text{out}} \cup \Omega_{\text{PML}}$ is performed by employing the Metis library, for the inner domain Ω_{in}, we consider a different strategy. Indeed, we need a sufficiently refined mesh for Ω_{in} to accurately capture the geometry of the PMUT membranes and to properly resolve the applied voltage input signal on them. On the other hand we need to balance among processors: (i) the hexahedral elements that share a face with Γ_{I}, (ii) the number of PMUT membranes, and (iii) the remaining elements. Moreover, an additional issue to face in simulations involving a large computational domain is generating and extracting the mesh (to be partitioned) in a single step due to RAM limitations. To account for all these constraints, the main approach proposed in this study is to cut the mesh into smaller blocks and provide a tool to merge them into a single mesh directly within the code, since the of membranes involved is extremely large, reaching more than ten thousands PMUTs. Consequently, generating the complete mesh in a single step is often impractical due to memory limitations and time resources.

The proposed strategy, which simultaneously considers the constraints imposed by PMUT arrays and the requirements of load balancing, is to generate a set of elementary files, $\{f_s^{\text{el}}\}$, with $s \in \{1, \ldots, N_{\text{sub}}^{\text{el}}\}$, each containing the mesh of a subdomain of the inner region Ω_{in}. These subdomains are then assigned to individual computational processes, which collectively reconstruct the complete array structure. Because PMUT arrays are designed according to a predefined pattern, multiple processes can adopt the same elementary file, applying an appropriate coordinate shift to give rise to the global arrangement. Figure 7 (left) provides an example of how those blocks appear.

In this way, four main constraints are directly satisfied: the PMUTs are distributed among different cores, keeping them entirely on the same process unit, the load balance of the inner domain is guaranteed, the DG faces on the finer side are well balanced, and the RAM limitation is overcame. Figure 6 illustrates a minimal example of the inner domain, along with the elementary structures used in its construction.

Next, we detailed the procedure developed to address the mesh generation and partitioning in Ω_{in}, when the number of PMUTs N_p is sufficiently large. Let $N_{p,\text{row}}$ denote the number of PMUTs in a row, and let $N_{\text{block}} = N_{p,\text{row}} + 1$ represent the number of elementary blocks required to form a complete row in the inner domain. The procedure begins by verifying that the number of computational cores N_{proc} is an integer multiple of N_{block}, i.e. $N_{\text{proc}} = M N_{\text{block}}$ with $M \in \mathbb{N}^+$. Upon failure, the system raises an exception in order to avoid unfavorable workload imbalances, particularly within the finer regions of the mesh.

Subsequently, a connectivity structure A_i, for $i = 0, ..., N_{\text{proc}} - 1$, is constructed for each process i to define adjacency relationships. This ensures that each process i has precise knowledge of its neighboring domains labeled in A_i, and thereby enabling efficient reconstruction of the global mesh connectivity. As an example, we can retrieve the mesh depicted in Figure 6 by setting $N_{\text{block}} = 3$, and $M = 3$ in Figure 7. The corresponding blocks $B_{n,m}$ and the core-ownership $L_{n,m}$ of each block, for $n = 1, ..., N_{\text{block}}$, $m = 1, ..., M$, can be grouped in the following matrices, respectively:

$$B = \begin{bmatrix} B_{3,1} & B_{3,2} & B_{3,3} \\ B_{2,1} & B_{2,2} & B_{2,3} \\ B_{1,1} & B_{1,2} & B_{1,3} \end{bmatrix}, \quad L = \begin{bmatrix} 2 & 5 & 8 \\ 1 & 4 & 7 \\ 0 & 3 & 6 \end{bmatrix},$$

from which the adjacency structure A_i, $\forall i = 0, \ldots, 8$, is derived

$$\begin{aligned} A_0 &= \begin{bmatrix} 1 & 3 & 4 \end{bmatrix}, & A_1 &= \begin{bmatrix} 0 & 2 & 3 & 4 & 5 \end{bmatrix}, & A_2 &= \begin{bmatrix} 1 & 4 & 5 \end{bmatrix}, \\ A_3 &= \begin{bmatrix} 0 & 1 & 4 & 6 & 7 \end{bmatrix}, & A_4 &= \begin{bmatrix} 0 & 1 & 2 & 3 & 5 & 6 & 7 & 8 \end{bmatrix}, & A_5 &= \begin{bmatrix} 1 & 2 & 4 & 7 & 8 \end{bmatrix}, \\ A_6 &= \begin{bmatrix} 3 & 4 & 7 \end{bmatrix}, & A_7 &= \begin{bmatrix} 3 & 4 & 5 & 6 & 8 \end{bmatrix}, & A_8 &= \begin{bmatrix} 4 & 5 & 7 \end{bmatrix}. \end{aligned}$$

So far, all the elementary files f_i^{el} are independent of each other, due to the fact that nodes, faces, and hexahedra have a local numeration starting from 1. Then, we need an algorthm to merge all elementary blocks and define a global numbering for the mesh elements. We start by writing all nodes lying on shared interfaces $\Gamma^{\mathrm{s}}_{n,m}$, for $n = 1, ..., N_{\mathrm{block}}$, $m = 1, ..., M$, cf. Figure 7 (right), in specific files S_i, $i = 0, ..., N_{\mathrm{proc}} - 1$. A quicksort algorithm orders the nodes according to the coordinates x, y, and z; in this way, the procedure for matching the nodes' IDs of different interfaces $\Gamma^{\mathrm{s}}_{n,m}$ that represent the same physical surface is fast and affordable. Notice that this step must be performed sequentially, and the connectivity structures A_i provide a way to speed up the procedure. Rank zero keeps its original identifiers, while all other processes receive the identifiers of shared nodes from neighboring ranks with smaller IDs, ensuring that each common node is assigned a unique identifier and duplicates are removed. This procedure leads to non-continuous local numbering due to missing identifiers. A final global renumbering step is then performed across all mesh files to restore continuity and establish a consistent global numbering of nodes, faces, and elements. The full procedure is synthetically shown in Algorithm 2.

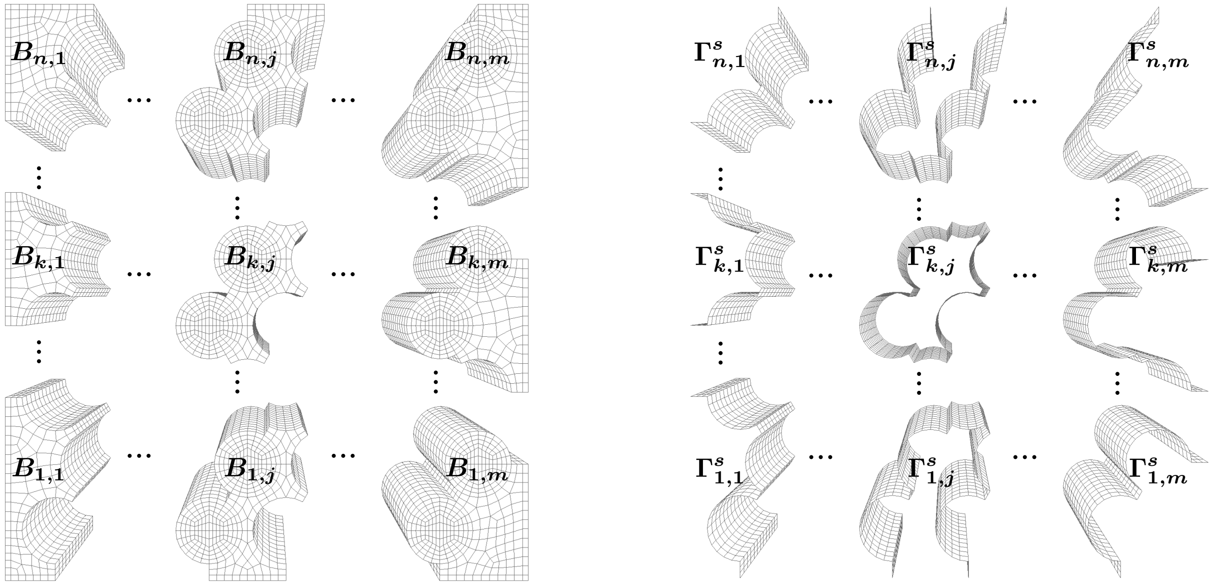


Figure 7: Schematic representation of the mesh partitioning into blocks $B_{n,m}$ (left) and the corresponding interfaces $\Gamma^s_{n,m}$, with $n = 1, \ldots, N_{\mathrm{block}}$ and $m = 1, \ldots, M$, used for block merging (right). The blocks illustrate the content of the elementary files f_s^{el} owned by the computational processes. Repetition of these blocks along the x- and y-directions allows the computational domain to be extended to an arbitrary size.

Algorithm 2 Inner domain partition and merging algorithm

```
Require: $N_{\text{proc}}$ multiple of $N_{\text{block}}$
for $i \in \{0, \dots, N_{\text{proc}} - 1\}$ do
    Construct core-connectivity structure $A_i$
    Assign to core $i$ its mesh file $f_i^{\text{el}}$
    Renumber nodes, faces, and elements of $f_i^{\text{el}}$
    Sort and store blocks' interface nodes in file $S_i$
    for $j$ in $A_i$ s.t. $j < i$, $i \neq 0$ do
        Match nodes in file $S_i$ and $S_j$
        Retain node IDs of file $S_j$ and discard IDs in $S_i$
    end for
end for
Renumber nodes according to global IDs
```

5 Fast assembly of interface terms for non-matching grids

The nonconformity between the inner and outer discretizations entails the construction of an interface mesh on $\Gamma_{\text{I}} = \partial\Omega_{\text{in}} \cap \partial\Omega_{\text{out}}$, defined as the geometric intersection of two hexahedral meshes with different resolutions. The number of resulting DG interfaces increases with the number of membranes under consideration. Efficiently identifying and matching interface elements becomes a critical task before the numerical solution of the governing equations can be carried out. While this strategy provides clear advantages in terms of geometric flexibility and reduction of the number of degrees of freedom, it also introduces significant challenges. In particular, a fast neighbor-matching algorithm is required to determine pairs of elements sharing DG interfaces, with the dual objective of reducing preprocessing overhead and improving performance during the time-stepping loop. In this work, we address these issues and propose an optimized methodology for interface construction and matching on Γ_{I}, significantly reducing the computational burden associated with the intersection process. Note that the intersection faces $F \in \mathcal{F}^I$ between the coarse and the fine boundaries are defined starting from the quadrature nodes located on the surface with the finer resolution. These quadrature nodes therefore determine the connectivity between elements (without explicitly computing the geometric intersection of the faces), which is a crucial aspect for the evaluation of the integrals in (7), cf. [17]. In particular, we focus our attention to the terms in (7) of the form

$$\int_F g^-(\boldsymbol{x}) f^+(\boldsymbol{x}) \, \mathrm{d}\sigma, \tag{13}$$

being $F = \partial K^+ \cap \partial K^-$ and where superscript ± denotes the restriction of polynomial functions g and f to K^+ or K^-, respectively. To compute the integral in (13), we recall that every element K of the mesh is the image through a trilinear map $\boldsymbol{X}_K$ of the reference element $\hat{K} = (-1, 1)^3$. Therefore, using the change of variables $\boldsymbol{x} = \boldsymbol{X}_{K^\pm}(\hat{\boldsymbol{x}})$ it holds

$$\int_F g^-(\boldsymbol{x}) f^+(\boldsymbol{x}) \, \mathrm{d}\sigma = \int_{\hat{F}} g^-(\boldsymbol{X}_{K^-}(\hat{\boldsymbol{x}}^-)) f^+(\boldsymbol{X}_{K^+}(\hat{\boldsymbol{x}}^+)) |J_{\boldsymbol{X}_{K^+}}(\hat{\boldsymbol{x}}^+)| \, \mathrm{d}\hat{\sigma},$$

being $\hat{F}$ (a portion of) a face of $\partial\hat{K}$ mapped through $\boldsymbol{X}_{K^-}$ or $\boldsymbol{X}_{K^+}$ to F, see Figure 8. The above integral is thus evaluated using a suitable quadrature rule with N_q nodes and $\{\omega_q\}_{q=1}^{N_q}$ weights as follows

$$\begin{aligned} &\int_{\hat{F}} g^-(\boldsymbol{X}_{K^-}(\hat{\boldsymbol{x}}^-)) f^+(\boldsymbol{X}_{K^+}(\hat{\boldsymbol{x}}^+)) |J_{\boldsymbol{X}_{K^+}}(\hat{\boldsymbol{x}}^+)| \, \mathrm{d}\hat{\sigma} \\ &\qquad \approx \sum_{q=1}^{N_q} g^-(\boldsymbol{X}_{K^-}(\hat{\boldsymbol{x}}_q^-)) f^+(\boldsymbol{X}_{K^+}(\hat{\boldsymbol{x}}_q^+)) |J_{\boldsymbol{X}_{K^+}}(\hat{\boldsymbol{x}}_q^+)| w_q = \sum_{q=1}^{N_q} \hat{g}^-(\hat{\boldsymbol{x}}_q^-) \hat{f}^+(\hat{\boldsymbol{x}}_q^+) |J_{\boldsymbol{X}_{K^+}}(\hat{\boldsymbol{x}}_q^+)| w_q, \end{aligned}$$

where w_q, for $q = 1, .., N_q$ are suitable weights and since $g^-(\boldsymbol{x}_q) = \hat{g}^-(\hat{\boldsymbol{x}}_q^-)$ and $f^+(\boldsymbol{x}_q) = \hat{f}^+(\hat{\boldsymbol{x}}_q^+)$.

In our process, we first define $\boldsymbol{x}_q \in F$ as the image of quadrature nodes $\hat{\boldsymbol{x}}_q \in \hat{F}$ through $\boldsymbol{X}_{K^+}$, i.e., $\boldsymbol{x}_q = X_{K^+}(\hat{\boldsymbol{x}}_q^+)$, and then, we map them back by using $\boldsymbol{X}_{K^-}$ to find $\hat{\boldsymbol{x}}_q^- \in \hat{F}$, i.e., $\hat{\boldsymbol{x}}_q^- = \boldsymbol{X}_{K^-}^{-1}(\boldsymbol{x}_q)$, cf. Figure 8. So, in practice, we implicitly define F and $\hat{F}$ through quadrature points.

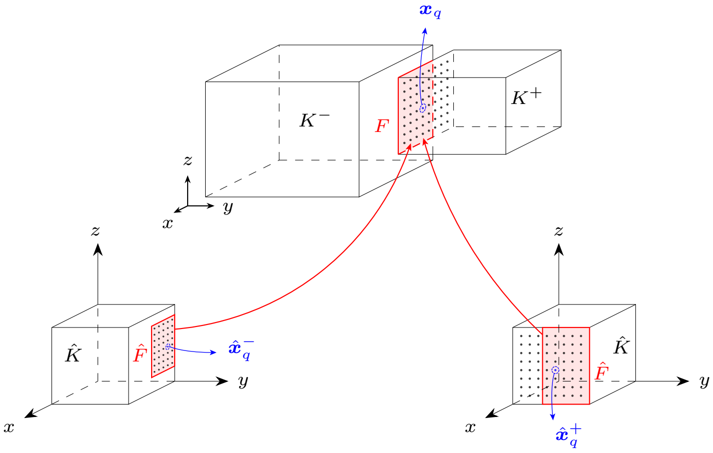


Figure 8: The figure shows the physical hexahedra K^- and K^+, sharing the common face F, together with their quadrature nodes $\boldsymbol{x}_q$, and the corresponding reference hexahedra $\hat{K}^\pm$ and their reference nodes $\hat{\boldsymbol{x}}_q^\pm$.

To accelerate the aforementioned computation, information regarding element faces ∂K^+ and ∂K^- in $\mathcal{F}^I$ is stored in two separate files. A third file contains the coordinates of a single vertex of each face ∂K^- belonging to $\mathcal{T}_{\text{out}}$. The algorithm for face-matching proceeds as follows: the finer face ∂K^+ loop over their quadrature nodes $\{\boldsymbol{x}_q\}_{q=1}^{N_q}$, and look for the coarser face containing the current node $\boldsymbol{x}_j$.

In this scenario, where hundreds of thousands of faces appears, to narrow the number of faces among the preselected ones, we first consider all the coarse faces ∂K^- whose normal $\boldsymbol{n}^-$ is opposite, up to a tolerance, to that of the current fine face $\boldsymbol{n}^+$. Hence, the search for the correct neighboring element is reduced to a smaller set of faces. Then, among the latter, we consider the ones whose vertex V_{K^-} lies within a circular neighborhood $C_r(\boldsymbol{x}_j)$ of the quadrature node of the finer face, with radius $r = \max\{\text{diam}(\partial K^-), \text{diam}(\partial K^+)\} + \varepsilon$, $\varepsilon > 0$. If those conditions are satisfied, the face is selected among the possible ones intersecting ∂K^+.

Finally, to properly find the face ∂K^- sharing the node $\boldsymbol{x}_j$ with ∂K^+, a Newton-Raphson routine checks whether the quadrature point lies within the element K^-, computing its reference position in $[-1,1]^3$. This procedure is shown in Algorithm 3, and Figure 9 provides a visual explanation.

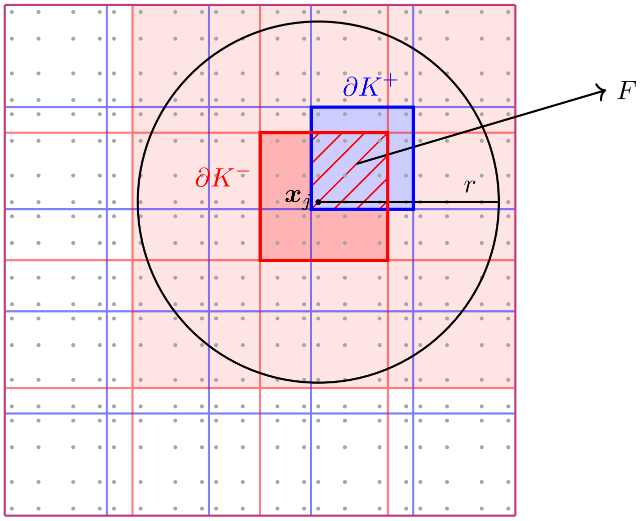


Figure 9: Example of the matching algorithm of DG faces. The red and blue grids refer to the coarse and fine faces, respectively, and the gray dots are the LG quadrature nodes of the finer faces. The blue-filled face is a representative fine face, and the center $\boldsymbol{x}_j$ of the circle is its quadrature node considered; the red-filled faces are those selected by the algorithm. Notice that considering only one vertex of the coarse face instead of all four increases the number of residual faces, but it avoids excessive RAM consumption.

Algorithm 3 Face matching procedure

```
Fix tol $> 0$
for $\partial K^+$ s.t. $\partial K^+ \cap \Gamma_{\mathrm{I}} \neq \emptyset$ do
  for $\boldsymbol{x}_q$ with $q = 1, ..., N_q$ do
    for $\partial K^-$ s.t. $\partial K^- \cap \Gamma_{\mathrm{I}} \neq \emptyset$ do
      if $|\boldsymbol{n}_+ + \boldsymbol{n}_-| <$ tol and $V_{K^-} \in C_r(\boldsymbol{x}_j)$ then
        Compute reference position $\hat{\boldsymbol{x}}_j$ with the Newton-Raphson method
        if $\hat{\boldsymbol{x}}_j \in [-1, 1]^3$ then
          $K^-$ is a neighbour of $K^+$
        end if
      end if
    end for
  end for
end for
```

6 Numerical results

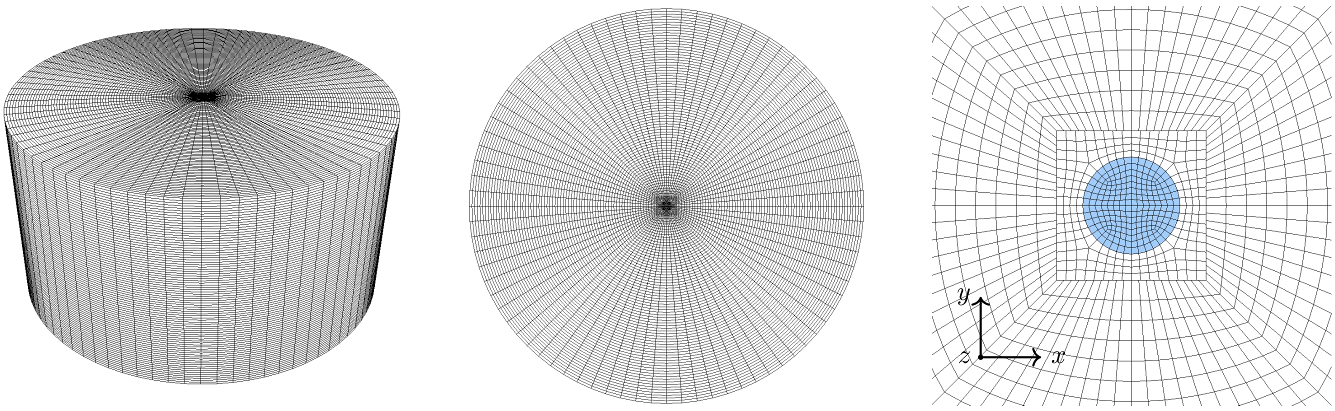
This section presents the numerical results obtained from the PMUT array simulations. We first consider the case of a single PMUT configuration, which serves as a reference case for the proposed computational model. Once the model's consistency is established, we investigate more complex scenarios involving multi-element PMUT arrays. The simulations are performed under two distinct operational conditions: a pure transmission (TX) framework, where the focus is on the emitted acoustic field, and a transmission–reception (TX–RX) framework, which additionally assesses the PMUTs sensing and reception capabilities.

6.1 Single PMUT excitation in TX phase

The first analysis focuses on a single PMUT, i.e. $N_p = 1$ in (3), operating in transmission mode. An alternating low-voltage signal of the form $\phi(t) = A \sin(2\pi f t) g(t)$ is applied to the surface of the PMUT, which is free to vibrate over the upper boundary of the computational domain. The modulation function $g(t)$ is introduced to ensure a smooth onset and offset of the excitation signal, thereby minimizing spurious frequency components and numerical artifacts associated with abrupt signal imposition. The function is defined as follows:

$$g(t) = \exp(1 - \alpha(T_d - t))\, \mathbb{1}_{\{0 \le t < T_d/2\}} + \exp(1 - \alpha(t))\, \mathbb{1}_{\{T_d/2 \le t \le T_d\}}, \tag{14}$$

where $\alpha(\tau) = \left(1 - \dfrac{2\beta(\tau)^{1+2n}}{1 + \beta(\tau)^{2n}}\right)^{-2}$, with $\beta(\tau) = (2\tau/T_d - 1)$ and $n = 10$. The excitation parameters are set to an amplitude of $A = 1.0\,\mathrm{V}$, a driving frequency of $f = 2.5\,\mathrm{MHz}$, and a total duration of $T_d = 5\,\mu\mathrm{s}$. The imposed voltage is plotted in Figure 11. The computational domain comprises the PMUT surrounded by an acoustic medium for wave propagation, which extends for $2000\,\mu\mathrm{m}$ both in the radial and z directions. The radius of the PMUT is $65\,\mu\mathrm{m}$. To avoid spurious reflections, first-order absorbing boundary conditions are imposed along the lateral and bottom boundaries of the acoustic domain, while a hard-wall constraint is imposed at the top surface supporting the PMUT as in (1), cf. Figure 10. A time step $\Delta t = 2 \times 10^{-4}\,\mu\mathrm{s}$ and a second-order polynomial degree in space are selected. The final time is set to $T = 12\,\mu\mathrm{s}$. Water is employed for both the inner and outer domains, hence the material density and the velocity of acoustic waves are $\rho = 1 \times 10^{-3}\,\mathrm{kg\,m^{-3}}$ and $c = 1481\,\mathrm{m\,s^{-1}}$, respectively. The first three axially symmetric modes, i.e., $N_m = 3$ in (3), are considered.



(a) Lateral view.

(b) Top view.

(c) Top view (zoom).

Figure 10: Test case of Section 6.1. Mesh configuration used for the single PMUT simulation. Panels (a–c) show the lateral, top, and zoomed top views, respectively. The PMUT, highlighted in panel (c), is excited by an input voltage signal applied to its upper surface.

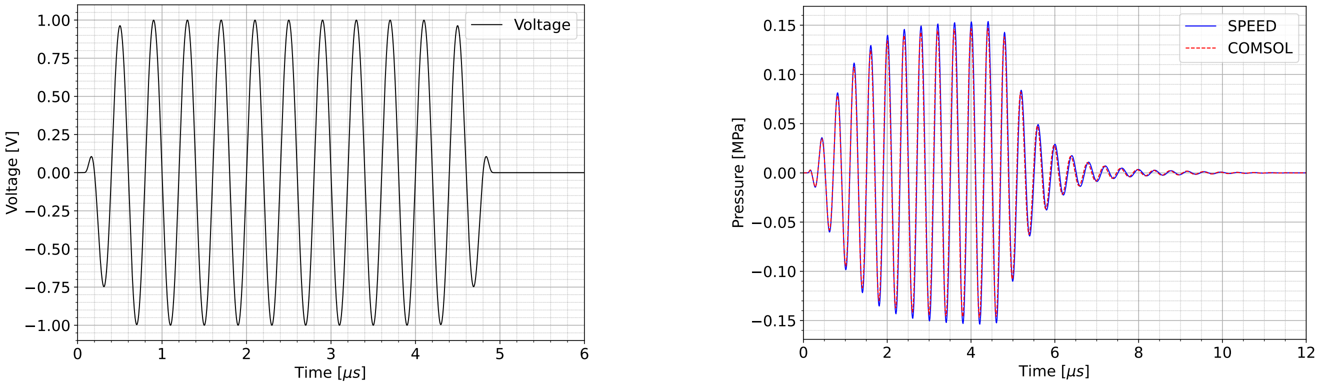


Figure 11: Test case of Section 6.1. Time evolution of the input signal imposed on the PMUT surface (left), and output pressure computed at the PMUT center (right). SPEED and COMSOL signal are in blue and red, respectively.

Figure 11 illustrates the temporal profile of the smooth input voltage applied to the PMUT surface, together with the resulting pressure excitation at the center of the membrane. The results are compared with a COMSOL Multiphysics V6.3 simulation of the PMUT array, where the device is modeled using a hybrid solid-shell approach: thin passive layers are represented using 2D first-order shear deformation theory shell elements, while the PZT layer and the silicon substrate are retained as 3D solid elements. Displacement continuity is enforced across shell–solid interfaces. The front encapsulation is modeled as an acoustic fluid with impedance-matched properties, and the surrounding acoustic domain solves the linearized acoustic wave equation [3]. This configuration provides an accurate approximation of a full 3D model while reducing computational cost, and the comparison demonstrates close correspondence between the two simulation frameworks. Figure 12, instead, provides a visualization of the acoustic wavefield propagation on an x–y plane extracted at a depth of $z = -1000\,\mu\mathrm{m}$, together with the temporal evolution of the pressure at $P = (0\,\mu\mathrm{m}, 0\,\mu\mathrm{m}, -1000\,\mu\mathrm{m})$. The comparison with COMSOL further confirms the consistency and agreement between the two simulations.

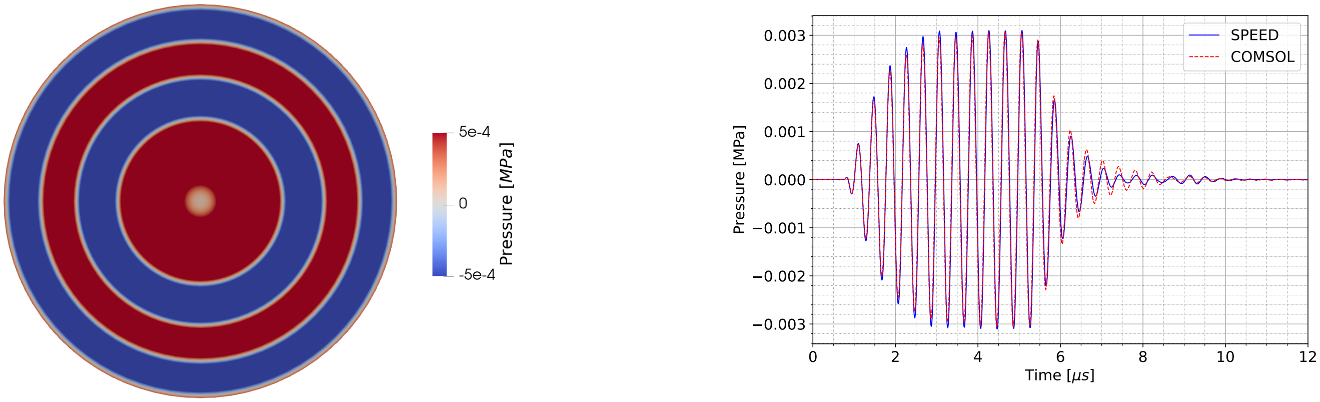
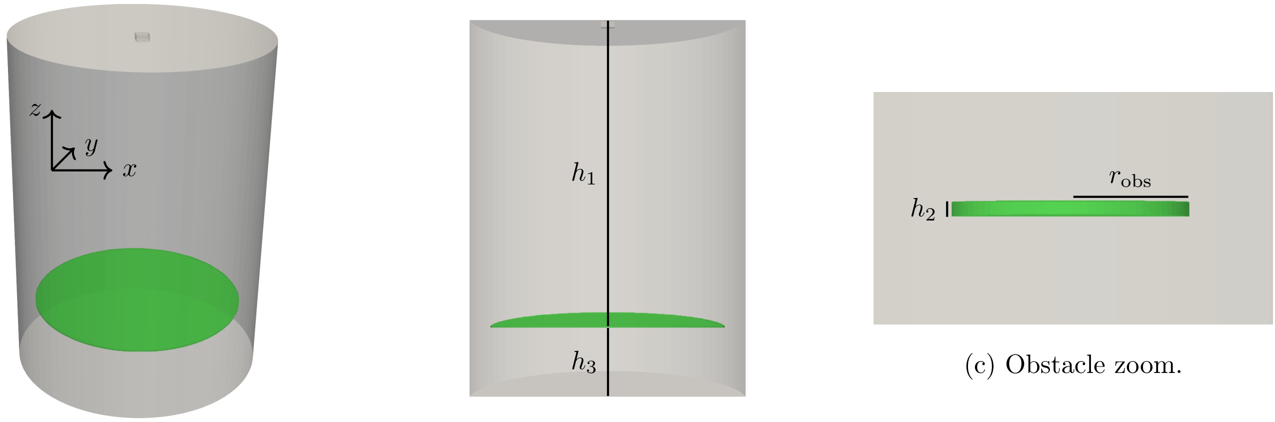


Figure 12: Test case of Section 6.1. Snapshot at $t = 4\,\mu\mathrm{m}$ of a slice in the x-y plane (left), and output pressure computed at the $P = (0\,\mu\mathrm{m}, 0\,\mu\mathrm{m}, -1000\,\mu\mathrm{m})$ for SPEED and COMSOL (right).

6.2 Single PMUT with an obstacle in TX-RX phase

In general, PMUTs are designed to be used in both the transmission (TX) and the receiving (RX) phases, i.e., to trigger the propagation of the acoustic signal up to a specific target and to capture the response that they receive back from the target itself. To this aim, the following simulation deals with both TX and RX phases.



(a) Full geometry.

(b) Thickness of the layers.

(c) Obstacle zoom.

Figure 13: Test case of Section 6.2. Geometry used for the single PMUT simulation with an obstacle in TX-RX phases. Panels (a–c) show the geometry employed, the thickness of the cylinder layers, and a zoom of a representative obstacle for graphical reasons, respectively.

An obstacle representing the target is put inside the acoustic propagation domain as shown in Figure 13. The setup is the following: an AC low-voltage $\phi(t) = A\sin(2\pi f t)h(t)$ is used with

$$h(t) = \exp(1 - \alpha(T_d - t))\, \mathbb{1}_{\{0 \le t \le T_d\}},$$

where $\alpha(\tau)$ is defined as in the test case of Section 6.1, with voltage parameters $A = 2.0\,\mathrm{V}$, $f = 2.5\,\mathrm{MHz}$, and a total duration $T_d = 1\,\mu\mathrm{s}$. The acoustic water domain is a cylinder of radius $r = 2000\,\mu\mathrm{m}$ composed of three layers of height $h_1 = 3c$, $h_2 = 25\,\mu\mathrm{m}$, and $h_3 = 1000\,\mu\mathrm{m}$, respectively, where h_1 is exactly the distance traveled by the acoustic wave in a time equivalent to $3T_d$ and $c = 1481\,\mathrm{m\,s^{-1}}$ is the wave speed in water. In this way, the acoustic pressure wave fully reaches the obstacle with radius $r_{\mathrm{obs}} = 1700\,\mu\mathrm{m}$ placed at a distance h_1 from the PMUT center, and the pressure is received back from the membrane after a time equivalent to $6T_d$ (see Figure 14). From time T_d up to the final time $T = 12\,\mu\mathrm{s}$, the PMUT is left in open-circuit condition with no externally applied voltage. The resulting mechanical vibrations and reflected pressure waves induce a voltage across the electrodes via the direct piezoelectric effect. Absorbing boundary conditions are imposed on the lateral and bottom sides of the outer cylinder, while a homogeneous Neumann condition is enforced on the top side and on the obstacle's surface.

Figure 14 shows the time evolution of both the voltage and the corresponding pressure computed at the PMUT center. The behavior of the PMUT changes at time T_d: the voltage is no more imposed, hence the PMUT ends the transmission phase and enters in receiving phase. From time T_d up to $t = 6T_d$, the voltage amplitude decays since the pressure on the PMUT surface reduces, while after a time $t > 6T_d$ the voltage slightly increases due to the pressure reflected from the obstacle. This trend is also visible in the pressure plot, where an increase in value is shown after the characteristic time $t = 6T_d$, cf. Figure 14 (right). Figure 15 presents two snapshots of the iso-contours of the acoustic pressure; the one on the left is a snapshot at time $t = 3T_d$, where the waves move towards the obstacle represented in green, while the picture on the right highlights the pressure waves that reach the PMUT surface at time $6T_d$ after the reflection.

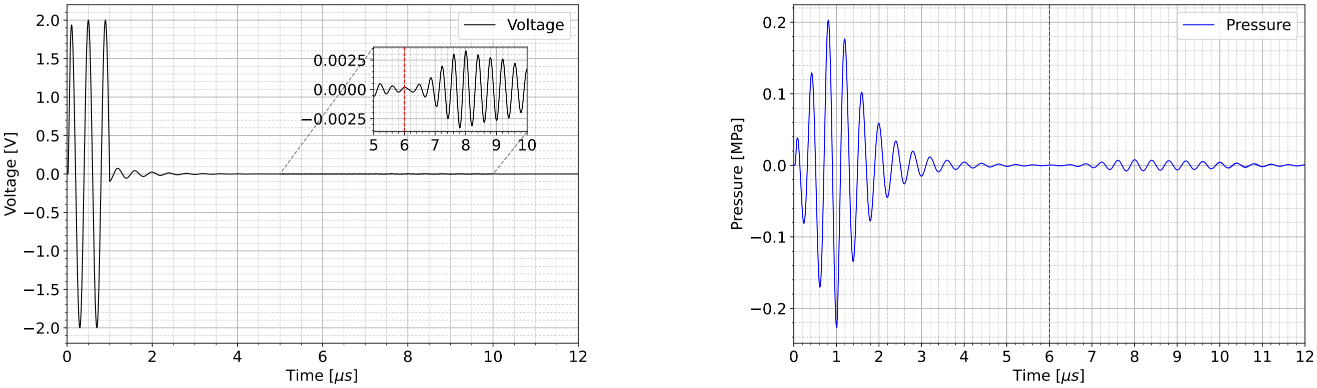


Figure 14: Test case of Section 6.2. Time evolution of the input signal imposed on the PMUT surface (left) with a zoom on the region for $t \in [5, 10]$µs, and output pressure computed at the PMUT center (right). The time instant $t = 6T_d$ is highlighted with a dashed line.

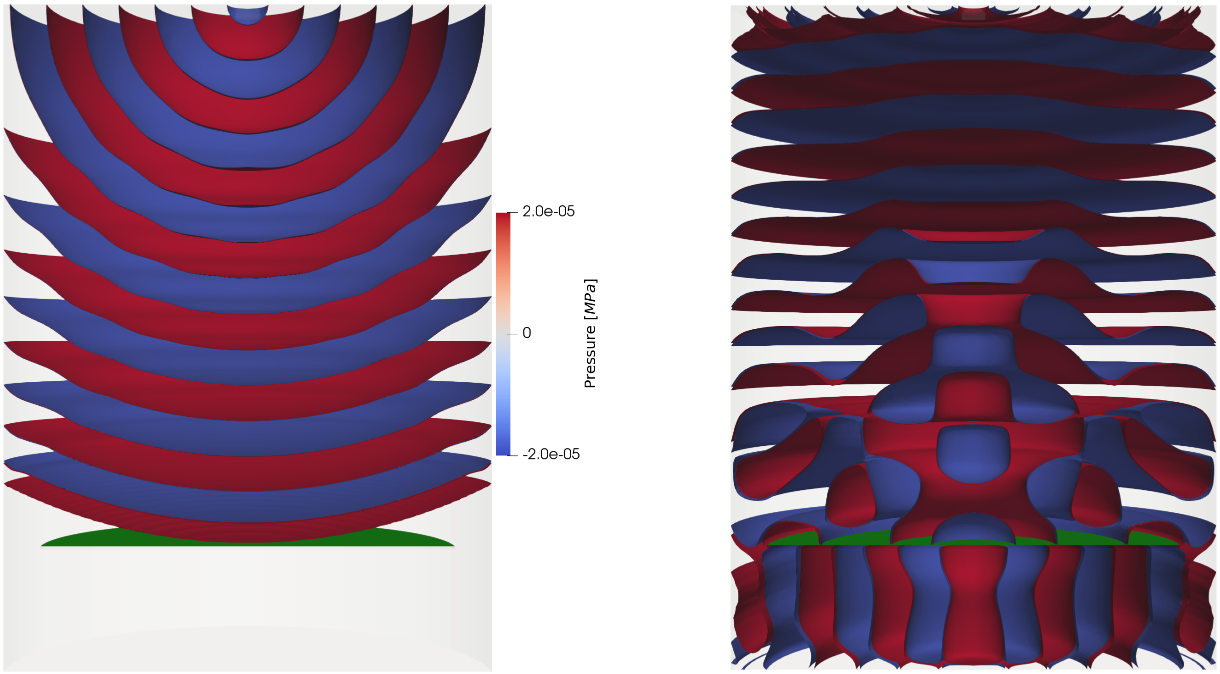


Figure 15: Test case of Section 6.2. Snapshots of the acoustic pressure waves contour plot: on the left, the one hitting the obstacle (in green), on the right, the one that travels back up to the PMUT.

6.3 Multi-array PMUT simulation

In general, absorbing boundary conditions are well-suited for simulations where wave propagation is predominantly orthogonal to the boundary. This approach is particularly effective when the computational domain is spherical or cylindrical, namely when the shape of the boundary matches the shape of the propagating waves. However, classical absorbing conditions can lead to undesirable internal reflections in domains with corners. Figure 16 illustrates the influence of the PML on a single-element array within a computational domain featuring corners. The two images correspond to the same simulation performed with and without Ω_{PML}. In the absence of this layer, wave reflections arising at the domain boundaries lead to noticeable distortions in the wavefront, thereby compromising the accuracy of the numerical solution. Conversely, when the PML is applied in combination with standard absorbing boundary conditions, spurious reflections are effectively suppressed, allowing for a smoother and more physically consistent propagation of acoustic waves within the domain.

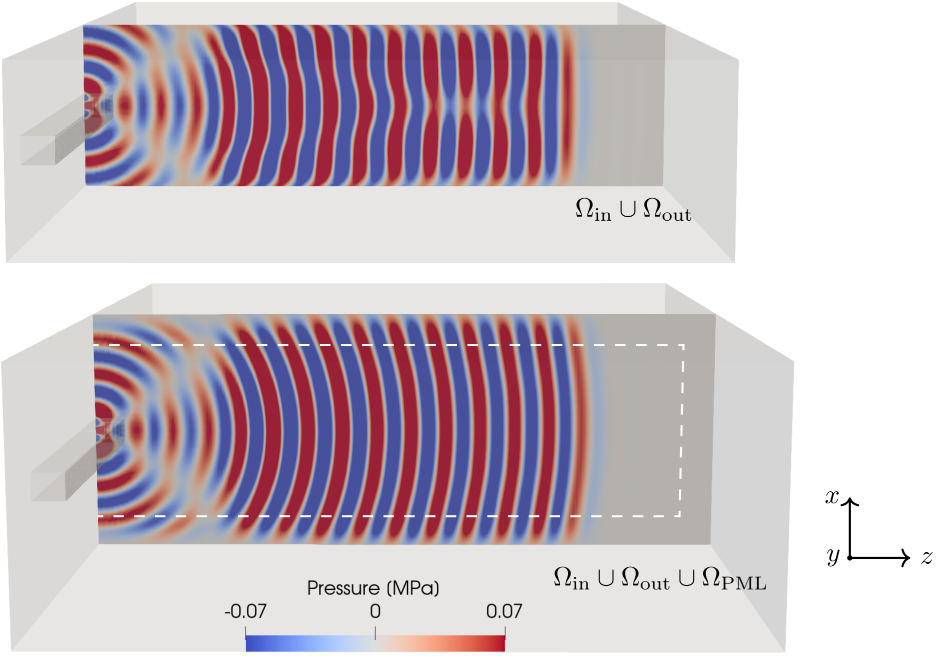

$\Omega_{\text{in}} \cup \Omega_{\text{out}}$
$\Omega_{\text{in}} \cup \Omega_{\text{out}} \cup \Omega_{\text{PML}}$
x
y
z
Pressure (MPa)
-0.07
0
0.07

Figure 16: Test case of Section 6.3. Schematic representation of a slice of the acoustic domain $\Omega_{\text{in}} \cup \Omega_{\text{out}}$ and the surrounding perfectly matched layer region Ω_{PML}. The upper figure corresponds to the simulation without the PML, while the lower figure shows the solution obtained when the PML is included. The dashed line represents the interface Γ_{PML}.

Multi-array simulations involve a large number of membranes. Each PMUT array consists of 184 membranes arranged in two adjacent columns of 92 each, distributed according to the configuration illustrated in Figure 6. This section focuses on the simulation of the 64-element array, cf. Figure 17.

The central idea behind this type of simulation, regardless of the number of arrays employed, is to define an acoustic propagation domain with a Cartesian geometry rather than a spherical or cylindrical one. Using Cartesian coordinates significantly reduces the total number of degrees of freedom required for the mesh compared to alternative geometries. Hard-wall conditions are imposed on the top surface Γ_{N}. A Perfectly Matched Layer encloses the computational domain, with absorbing boundary conditions applied along the outer boundary Γ_{ABC}, see equation (1). Each array operates independently; PMUTs in different arrays may be driven by distinct voltage excitations. In the following simulation, the applied voltage signal follows the expression in equation (14), with parameters $f = 2.5\,\text{MHz}$ and $A = 7.0\,\text{V}$, over a total activation time of $T_d = 4\,\mu\text{s}$. All arrays simultaneously begin and end the transmission phase. The domain consists of two regions described in Section 4, i.e., an inner and an outer domain. In the inner region, the material properties are defined as $\rho_{\text{in}} = 1140\,\text{kg}\,\text{m}^{-3}$ and $c_{\text{in}} = 1130\,\text{m}\,\text{s}^{-1}$. The outer region corresponds to water, characterized by $\rho_{\text{out}} = 1000\,\text{kg}\,\text{m}^{-3}$ and $c_{\text{out}} = 1481\,\text{m}\,\text{s}^{-1}$. The inner domain has a thickness of $h_{\text{in}} = 500\,\mu\text{m}$. The perfectly matched layer extends over a distance equal to one wavelength in water, $\lambda_{\text{w}} = 592.4\,\mu\text{m}$, and the overall dimensions of the domain are $l_x = 22\,874.4\,\mu\text{m}$, $l_y = 15\,570.64\,\mu\text{m}$, and $l_z = 12\,940.4\,\mu\text{m}$ along the x, y, and z directions, respectively. The radius of the PMUT is $r_p = 65\,\mu\text{m}$, and the distance between two adjacent PMUT centers, known as pitch, is $d_p = 150\,\mu\text{m}$. Figure 17 contains the geometrical information of the simulation. The time step of the simulation is $\Delta t = 2 \times 10^{-4}\,\mu\text{s}$, and a second-order polynomial degree in space is considered. The final time is set to $T = 15\,\mu\text{s}$. The number of elements in the mesh is 33.8M, and the degrees of freedom are 262.6M. Figure 18 illustrates the membrane excitation after 0.2 µs and the resulting acoustic wave propagation within the computational domain along the three spatial directions. The pressure field is accurately captured within the computational domain, as shown by the alternating pattern of positive and negative planar pressure waves. Ten probes placed at $P = (0, 0, z)$, with $z \in \{-500\,\mu\text{m}, \ldots, -9500\,\mu\text{m}\}$, allow for evaluating the acoustic pressure signal in both the near and far fields (see Figure 19). The simulation ran on a high-performance cluster consisting of 40 CPU compute nodes. Each CPU node is configured with two AMD EPYC 7413 processors, each featuring 24 cores, yielding a total of 96 threads per node. These nodes are provisioned with 512 GB of RAM each, providing substantial memory capacity to support large-scale simulations and data-intensive computations within a parallel computing framework. By utilizing 21 nodes, and 1935 processes, the simulation required approximately 12 hours to complete.

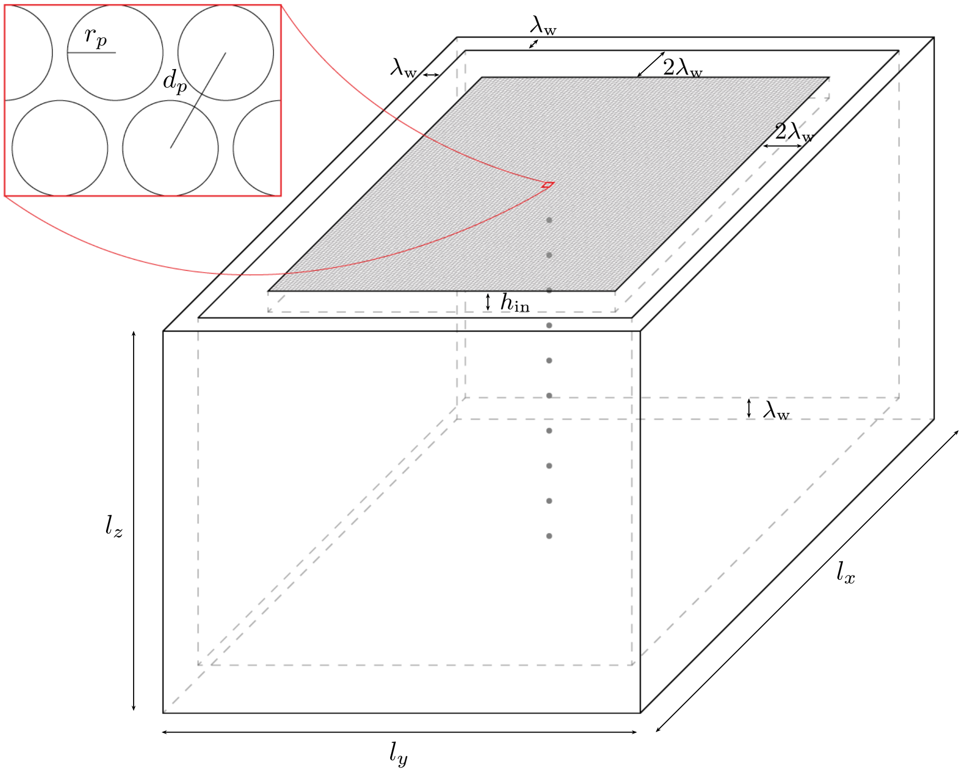


Figure 17: Test case of Section 6.3. Representation of the three-dimensional view of the geometry employed for the 64-elements simulation. The dots represent the probes at which the solution is computed. Notice that the figure is not scaled.

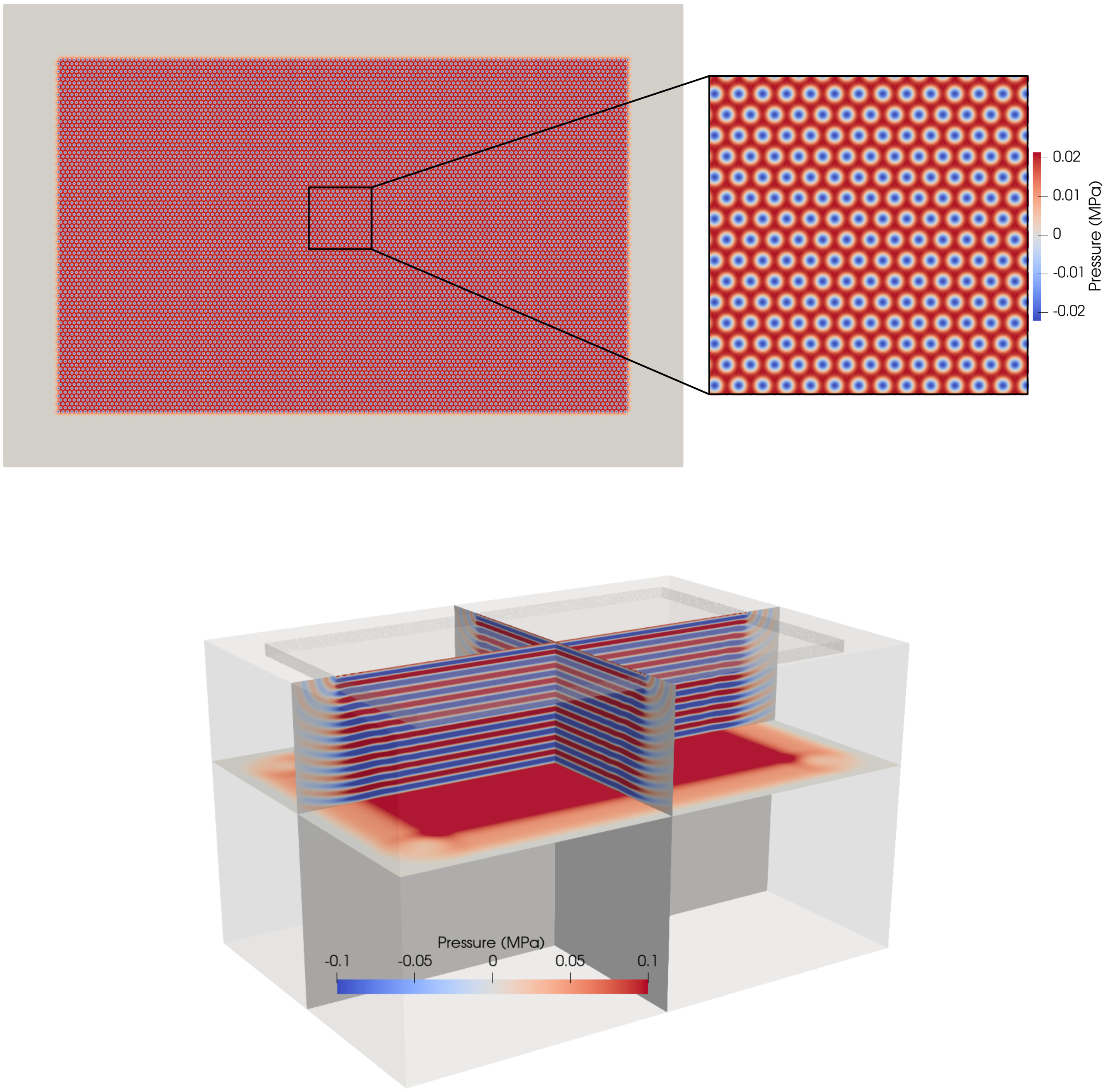


Figure 18: Test case of Section 6.3. The figure shows the snapshot of the surface containing the PMUTs at time $t = 0.2\,\mu\mathrm{s}$ with a zoom of the middle region (top), and the solution at time $t = 4\,\mu\mathrm{s}$ depicted over 3 slices (bottom).

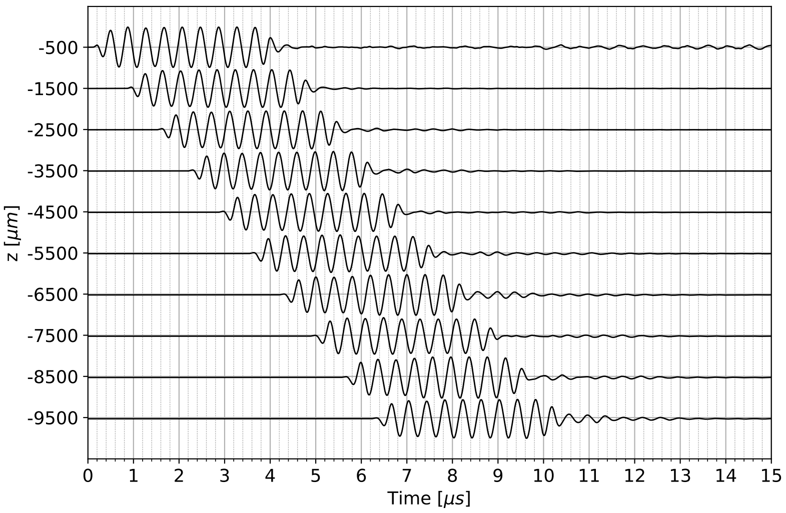


Figure 19: Test case of Section 6.3. The figure shows the acoustic pressure computed at ten equispaced points $P = (0, 0, z)$, with $z \in \{-500\,\mu\text{m}, \ldots, -9500\,\mu\text{m}\}$.

6.4 Performance comparisons and scalability test

A comparative performance analysis was carried out to evaluate the effectiveness of the DG interface optimization described in Section 5 and to quantify the computational and communication overhead associated with the evaluation of the integrals over Γ_{PMUT} in equations (11) and (12), as discussed in Section 4. The first test case employed a computational domain consisting of a 20-element array distributed across 315 processes. Two configurations were investigated: one without DG interface optimization and one with the optimization enabled. In the absence of DG optimization, the setup of 318698 interfaces required approximately 160 000 s. By contrast, enabling DG interface optimization reduced the setup time to approximately 630 s. This represents a reduction of more than two orders of magnitude in interface setup time, underscoring the critical importance of DG optimization for improving scalability and significantly reducing preprocessing overhead. The second test case was performed on a computational domain comprising a single-element array partitioned across 96 processes. In the first configuration, an arbitrary mesh partitioning was employed, with no constraint enforcing the association of an entire PMUT with a single process. Under this configuration, the solution of equation (10) incurred an inter-process communication cost of approximately 0.167 s per timestep, arising solely from the exchange of local contribution to the PMUTs solution on each process with all other processes. This communication overhead substantially exceeded the pure computational cost, thereby indicating poor parallel efficiency.

In the second configuration, the tailored partitioning strategy described in Section 4 was adopted. This approach resulted in an average synchronization waiting time of approximately 0.0008 s per PMUT per timestep for each process, corresponding to a reduction of more than two orders of magnitude. It is worth noting that, in the former configuration, increasing either the number of membranes or the number of processes leads to a corresponding increase in communication time, whereas in the latter configuration, the communication cost remains, on average, constant. This behavior is further illustrated by the scalability analysis presented in the following.

Overall, these results demonstrate that DG interface optimization is essential for reducing preprocessing time in large-scale simulations, while PMUTs performance is highly sensitive to communication patterns and domain decomposition strategies. Inappropriate partitioning can result in communication-dominated runtimes, thereby severely limiting parallel efficiency. These findings underscore the necessity of balanced mesh partitioning and efficient data exchange mechanisms to achieve scalable and efficient parallel performance.

Scalability refers to the ability of a computational system to maintain or improve performance as workload and resources increase. In high-performance and parallel computing, it depends on the efficient use of processors, memory, and communication, requiring well-designed parallel algorithms, effective memory management, and controlled communication overhead to ensure competitive performance. In the specific context of the problem under investigation, the main constraints that influence scalability are the ones discussed in Section 4 and can be categorized into three main factors:

i) PMUTs blocks: the scalability of the system is directly affected by the configuration and number of PMUT array blocks. As the array size increases, the computational load associated with the membrane becomes dominant. Efficient data partitioning and optimized communication between array elements are therefore essential to maintain scalability in large configurations.

ii) Acoustic propagation domain: the computational complexity of simulating the acoustic propagation domain scales with both the spatial resolution and the physical model. High-resolution domains require finer discretization, leading to increased memory usage and processing demands. Load balancing and domain decomposition strategies are of paramount importance to ensure that the computations remain scalable across multiple processes. Moreover, simulations with the PML enlarge the DOFs and hence require an additional complexity to be taken into account.

iii) DG interfaces: they represent another critical scalability constraint. These interfaces govern the communication and numerical flux exchanges between subdomains. As the number of hexahedral elements increases, so does the volume of interfacial data transfer, potentially leading to communication bottlenecks. Maintaining scalability in this context requires efficient inter-process communication schemes and optimized data structures that ease rapid access and exchange of boundary information.

Overcoming these challenges requires a combination of algorithmic optimization, advanced parallelization strategies, and system design to ensure that performance scales effectively with increasing computational resources. A strong scalability test is performed to assess the validity of the approach discussed in the previous sections. Four simulations are conducted within the same computational domain, exploiting the cluster of the Mathematical Department of Politecnico di Milano (DMAT), whose characteristics are reported in the previous section. The number of processes selected is $22, 44, 88$ and 176, respectively. In this way, communication and load balancing are performed both among processes within a single node and between different nodes. Figure 20 (left) shows the comparison between the theoretical and normalized real-time resources required in the four cases. The plot in logarithmic scale shows a very good trend compared to the expected behavior. Moreover, a weak scalability test was assessed over three different simulations proportionally increasing both the number of degrees of freedom and the number of processes, thereby preserving a fixed ratio of approximately 140000 DOFs per process. Figure 20 (right) shows an almost ideal weak scalability, by holding the execution timestep constant.

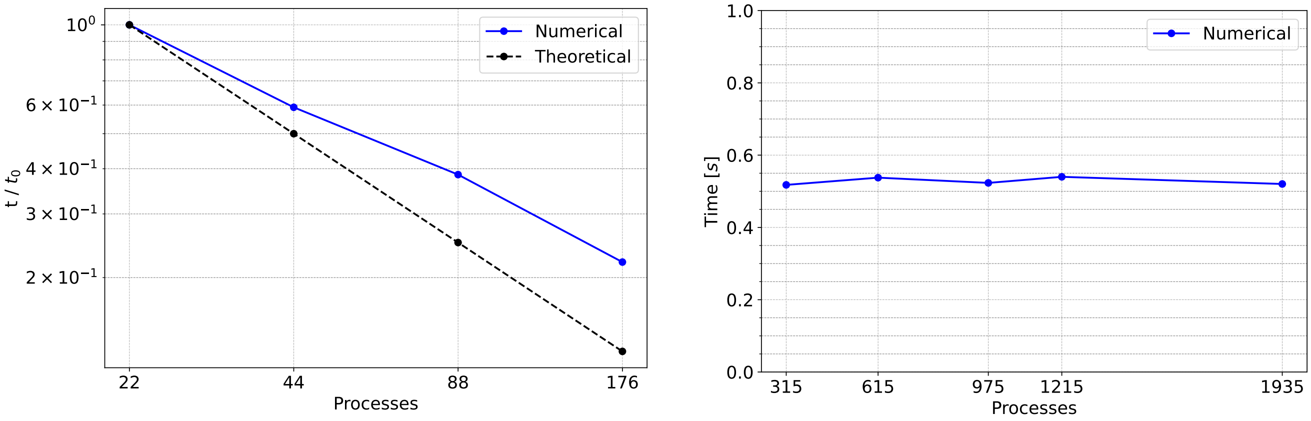


Figure 20: Test case of Section 6.4. The figure shows the theoretical trend (black) and the real one (blue) of the strong scalability test conducted across four different simulations (left). The value on the $y-$axes is normalized with respect to the time t_0 of the first simulation. On the right, the weak scalability test performed on five different simulations.

7 Conclusions and future developements

This work presented a comprehensive methodology for the efficient numerical simulation of large-scale Piezoelectric Micromachined Ultrasonic Transducer arrays, based on a hybrid approach consisting of model order reduction techniques and a high-performance Discontinuous Galerkin Spectral Element framework implemented within the SPectral Elements in Elastodynamics with Discontinuous Galerkin (SPEED) environment. By integrating the mechanical modal behavior of individual PMUTs into an acoustic domain discretized through high-order elements, the proposed approach achieves a balance between accuracy and computational efficiency. The resulting model successfully captures both transmission and reception phenomena, enabling scalable and high-fidelity simulations of complex array configurations. The implemented domain decomposition and optimization strategies proved effective in enhancing computational performance, ensuring that the framework remains suitable for massively parallel simulations involving thousands of transducers. The numerical results confirm the robustness and scalability of the proposed approach, making it a promising tool for the design and optimization of next-generation PMUT-based ultrasonic systems. Beyond the present developments, future work will focus on extending the current framework to account for thermoacoustic effects, which are increasingly relevant as PMUTs operate at higher power densities and within miniaturized environments. Incorporating thermo-mechanical coupling and heat generation due to dielectric and mechanical losses will enable the study of temperature-induced frequency shifts, performance degradation, and thermal noise in PMUT arrays. Moreover, the integration of nonlinear thermoelastic models could provide deeper insight into self-heating phenomena and their influence on both emission and reception behavior. Such extensions would pave the way toward multiphysics simulations capable of predicting the full electromechanical–acoustic–thermal response of PMUT systems under realistic operational conditions. Ultimately, these developments will support the design of more reliable, energy-efficient, and thermally stable ultrasonic microsystems, broadening their applicability in demanding environments such as biomedical diagnostics, industrial sensing, and harsh-condition monitoring.

Acknowledgments

This work is part of the *PROgramma per la realizzazione di dispositivi ad Ultrasuoni e loro promozione Democratica* (PROUD) project funded by the Italian Ministry of Enterprises and Made in Italy (MIMIT): Prog. n.: F/310211/01-05/X56 (Decree n.4187, 19-12-2023). The authors would like to thank STMicroelectronics for their valuable support and contributions within the framework of the PROUD project. This work also benefited from helpful discussions and technical support from all project partners. P.F. Antonietti, M.G. Garroni, I. Mazzieri, and N. Prolini are members of INdAM-GNCS group. The present research is part of the activities of "Dipartimento di Eccellenza 2023-2027". The simulations discussed in this work were, in part, performed on the HPC Cluster of the Department of Mathematics of Politecnico di Milano, which was funded by MUR grant "Dipartimento di Eccellenza 2023-2027".

References

[1] METIS: https://github.com/karypislab/metis.

[2] O. M. O. Abdalla, M. G. Garroni, E. Moretti, G. Massimino, A. Corigliano, A. Frangi, et al. Efficient Modeling and Simulation of Large PMUT Arrays for Biomedical Applications. In *Proceedings of the 2025 26th International Conference on Thermal, Mechanical and Multi-Physics Simulation and Experiments in Microelectronics and Microsystems (EuroSimE 2025)*, pages 1–5. Institute of Electrical and Electronics Engineers Inc., 2025.

[3] O. M. O. Abdalla, G. Massimino, A. S. Savoia, F. Quaglia, and A. Corigliano. Efficient Modeling and Simulation of PMUT Arrays in Various Ambients. *Micromachines*, 13(6), 2022.

[4] P.F. Antonietti, F. Bonaldi, and I. Mazzieri. Simulation of three-dimensional elastoacoustic wave propagation based on a Discontinuous Galerkin spectral element method. *International Journal for Numerical Methods in Engineering*, 121(10):2206–2226, 2020.

[5] A. Artoni, P. F. Antonietti, I. Mazzieri, N. Parolini, and D. Rocchi. A hybrid Finite Volume - Spectral Element Method for aeroacoustic problems. *Computers & Mathematics with Applications*, 155:193–208, 2024.

[6] Y. Birjis, S. Swaminathan, H. Nazemi, G. C. A. Raj, P. Munirathinam, A. Abu-Libdeh, and A. Emadi. Piezoelectric Micromachined Ultrasonic Transducers (PMUTs): Performance Metrics, Advancements, and Applications. *Sensors (Basel)*, 22(23):9151, 2022.

[7] A. Bybi, D. Khouili, C. Granger, M. Garoum, A. Mzerd, and A. Hladky. Experimental Characterization of a Piezoelectric Transducer Array Taking into Account Crosstalk Phenomenon. *International Journal of Engineering and Technology Innovation*, 2020.

[8] S. Chaudhary, J. Singh, and F. Z. Haque. BVD and Mason's modelling of piezoelectric bulk acoustic resonators for high frequency applications. *Materials Today: Proceedings*, 65:2556–2559, 2022.

[9] Guo-Hua F. and Hua Jin L. Piezoelectric Micromachined Ultrasonic Transducers with a Cost-Effective Bottom-Up Fabrication Scheme for Millimeter-Scale Range Finding. *Sensors*, 19:4696, 2019.

[10] Y. Gao, X. Wang, L. Zhao, et al. High frame rate ultrasonic flowmeter based on PMUT array with bidirectional acoustic beams. *Microsystems Nanoeng*, 11, 2025.

[11] M. J. Grote and I. Sim. Efficient PML for the wave equation. In *https://arxiv.org/abs/1001.0319*, 2010.

[12] L. He, W. Xu, Y. Wang, J. Zhou, and J. Ren. Sensitivity-Bandwidth Optimization of PMUT with Acoustical Matching Using Finite Element Method. *Sensors*, 22(6):2307, 2022.

[13] Y. He, H. Wan, X. Jiang, and C. Peng. Piezoelectric Micromachined Ultrasound Transducer Technology: Recent Advances and Applications. *Biosensors (Basel)*, 13(1):55, 2022.

[14] B. Kaltenbacher, M. Kaltenbacher, and I. Sim. A modified and stable version of a perfectly matched layer technique for the 3-d second order wave equation in time domain with an application to aeroacoustics. *Journal of Computational Physics*, 235:407–422, 2013.

[15] Z Luo, J Cai, S. Zhang, Y. Gu, L. Lou, and T. Wu. Nonlinearity of Piezoelectric Micromachined Ultrasonic Transducer Using AlN Thin Film. *IEEE Open Journal of Ultrasonics, Ferroelectrics, and Frequency Control*, 2:96–104, 2022.

[16] G. Massimino, F. Quaglia, A. Corigliano, and A. Frangi. Model order reduction for the analysis of large arrays of piezoelectric micromachined ultrasonic transducers in water. *Applied Acoustics*, 182:108231, 2021.

[17] I. Mazzieri. *Non-conforming high order methods for the elastodynamics equation*. PhD thesis, Politecnico di Milano, 2012. PhD thesis, POLITesi digital repository.

[18] F. Narváez, S. Hosseini, H. Farkhani, and F. Moradi. Analysis and Design of a PMUT-based transducer for Powering Brain Implants. 2021.

[19] Jiong Pan, Chenyu Bai, Qincheng Zheng, and Huikai Xie. Review of Piezoelectric Micromachined Ultrasonic Transducers for Rangefinders. *Micromachines*, 14(2):374, 2023.

[20] F. Rapetti, P. Antonietti, I. Mazzieri, and A. Quarteroni. Non-conforming high order approximations for the elastic wave equation. 01 2012.

[21] P.-A. Raviart and J. M. Thomas. *Introduction à l'analyse numérique des équations aux dérivées partielles.* Masson, 1983.

[22] Kaustav Roy, Joshua En-Yuan Lee, and Chengkuo Lee. Thin-film PMUTs: a review of over 40 years of research. *Microsystems and Nanoengineering*, 9:95, 2023.

[23] S. Sanvito, M. Passoni, D. Giusti, Marco Terenzi, C. Prelini, M. La Mura, and A. Savoia. PMUT-Based System for Continuous Monitoring of Bolted Joints Preload. *Sensors*, 24:4150, 06 2024.

[24] S. Shelton, M. Chan, H. Park, D. Horsley, B. Boser, I. Izyumin, R. Przybyla, T. Frey, M. Judy, K. Nunan, F. Sammoura, and K. Yang. CMOS-compatible AlN piezoelectric micromachined ultrasonic transducers. In *2009 IEEE International Ultrasonics Symposium*, pages 402–405, 2009.

[25] K. Smyth and S. Kim. Experiment and simulation validated analytical equivalent circuit model for piezoelectric micromachined ultrasonic transducers. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 62(4):744 – 765, 2015.

[26] X. Su, X. Ren, H. Wan, X. Jiang, and X. Liu. Simulation and Optimization of Piezoelectric Micromachined Ultrasonic Transducer Unit Based on AlN. *Electronics*, 11:2915, 09 2022.

[27] T. Xu, Z. Zhang, R. Tumolin Rocha, L. Zeng, and C. Xu. Theoretical Validation of a Single-Channel Air-Coupled PMUT with Multi-Frequency Operation for Compressed 3D Spatial Sensing. *IEEE Open Journal of Ultrasonics, Ferroelectrics, and Frequency Control*, 4:37–51, 2024.

[28] Junru Zhang, Mi Zhao, Guoliang Zhang, Junqi Zhang, and Xiuli Du. 3D acoustic scaled boundary perfectly matched layer (SBPML) for acoustic-structure interaction problems. *Engineering Analysis with Boundary Elements*, 164:105765, 2024.

[29] L. Zhao, J. Xia, C. Yang, and Y. Lu. Modeling and experimental studies of mutual acoustic radiation effect on PMUT array. *Sensors and Actuators A: Physical*, 377:115677, 2024.

MOX Technical Reports, last issues

Dipartimento di Matematica
Politecnico di Milano, Via Bonardi 9 - 20133 Milano (Italy)

25/2026 Carrara, D.; Hirschvogel, M.; Bonizzoni, F.; Pagani, S.; Pezzuto, S.; Regazzoni, F.
Shape-informed cardiac mechanics surrogates in data-scarce regimes via geometric encoding and generative augmentation

23/2026 Ballini, E.; Muscarnera, L.; Fumagalli, A.; Scotti, A.; Regazzoni, F.
Elimination-compensation pruning for fully-connected neural networks

26/2026 Dokuchaev, A.; Bonizzoni, F.; Pagani, S.; Regazzoni, F.; Pezzuto, S.
Learning geometry-dependent lead-field operators for forward ECG modeling

21/2026 Bottacini, G.; Torzoni, M.; Manzoni, A.
Neural Markov chain Monte Carlo: Bayesian inversion via normalizing flows and variational autoencoders

17/2026 Caldera, L.; Bottacini, G.; Cavinato, L.
MAGIC-Flow: multiscale adaptive conditional flows for generation and interpretable classification

20/2026 Caldera, L.; Cavinato, L.; Cirone, A.; Cama, I.; Garbarino, S.; Lodi, R.; Tagliavini, F.; Nigri, A.; De Francesco, S.; Cappozzo, A.; Piana, M.; Ieva, F.;
DISARM++: Beyond scanner-free harmonization

19/2026 Caldera, L.; Cavinato, L.; Ieva, F.
Scanner-agnostic MRI harmonization via SSIM-guided disentaglement

15/2026 Zecchi, A. A.; Sanavio, C.; Cappelli, L.; Perotto, S.; Roggero, A.; Succi, S.
Block encoding of sparse matrices with a periodic diagonal structure

14/2026 Agasisti, T.; Cannistrà, M.; Paganoni, A.M.
Nudging communication for students at risk: experimental evidence from an Italian university

13/2026 Dimola, N.; Coclite, A.; Zunino, P.
Neural Preconditioning via Krylov Subspace Geometry