

Quadrature-Aware Complex-Linear Neural Operator for Boundary-to-Field Prediction in Resonant Acoustics

Muhammad Idrees Khan* Hua-Dong Yao

Department of Mechanical Engineering, Chalmers University of Technology
Gothenburg 41296, Sweden

Abstract

Repeated prediction of acoustic fields from spatially distributed boundary excitation is computationally expensive when each source realization requires a new wave simulation. This work introduces a quadrature-aware complex-linear boundary operator (CLBO) that maps complex normal velocity on a vibrating surface to complex pressure at receiver locations. The model couples learned source and receiver basis functions through an explicit complex surface-quadrature contraction, so the boundary excitation enters linearly by construction. This preserves complex superposition, homogeneity, and zero response to zero excitation, while representing the source through coordinates, normals, and quadrature weights rather than a fixed flattened input vector. Reference data were generated using a verified three-dimensional multiple-relaxation-time (MRT) lattice Boltzmann solver and stored in a solver-agnostic boundary-to-field format. CLBO was compared with a fixed-sensor complex DeepONet under matched case splits and optimization settings, with additional tests of structural consistency, receiver-coordinate interpolation, source discretization, source-family holdout, label efficiency, physics-informed ablations, unseen source mixtures, and computational cost. Across five training seeds, CLBO achieved a mean complex relative field error of 0.184 ± 0.00771 , compared with 0.367 ± 0.00742 for DeepONet. Its measured source-superposition error was 1.31×10^{-7} , and its mean error on newly simulated mixed-source cases was 0.237, compared with 0.415 for DeepONet. Inference was 1.83×10^4 faster than the reference calculation for the reported query size. These results show that enforcing the known complex-linear boundary-to-field structure improves physical consistency and generalization under distributed acoustic excitation.

Keywords: neural operator; complex linearity; boundary excitation; resonant acoustics; lattice Boltzmann method; surface quadrature

1 Introduction

Many acoustic and vibroacoustic analyses require repeated evaluation of the pressure field generated by a spatially distributed vibrating boundary. Such boundary-to-field predictions are needed when assessing how changes in wall vibration, actuator patterns, source placement, operating frequency, or receiver location affect the resulting interior sound field. Once the

*Corresponding author: Muhammad Idrees Khan

geometry, medium properties, operating frequency, and passive boundary conditions are fixed, the acoustic stage is a complex-linear operator from prescribed normal velocity to pressure. Finite-element, boundary-element, modal, and time-domain wave solvers provide high-fidelity solutions, but repeated calculations over source fields, frequencies, and receiver sets can dominate design exploration, uncertainty studies, and surrogate-based optimization [1, 2, 3, 4, 5].

Operator learning seeks to replace repeated numerical solves by learning maps between input and output functions. The Deep Operator Network (DeepONet) uses branch and trunk networks to represent an operator from input-function samples to coordinate queries [6]; neural-operator frameworks use learned integral kernels and include low-rank, graph, and Fourier parameterizations [7, 8]; geometry-aware variants address irregular discretizations and varying domains [9]. Physics-informed neural networks and physics-informed neural operators introduce governing-equation residuals into optimization [10, 11], although such residual losses can introduce conditioning and loss-balancing difficulties [12]. More broadly, machine learning has become a common tool for data-driven modeling, closure modeling, reduced-order modeling, and control in fluid mechanics [13].

Several related ideas are important for the present problem but do not by themselves enforce its complete structure. Sum aggregation and point-set architectures provide permutation-invariant representations of unordered inputs [14, 15, 16]; kernel neural operators use numerical quadrature for geometrically flexible function-space approximation [17]; recent low-rank kernel models learn separable or singular-function representations [18]; and boundary-integral neural methods reduce partial-differential-equation (PDE) learning to boundary quantities [19, 20]. In acoustics, neural and physics-informed models have been used for parameterized propagation, continuous neural acoustic fields, sound-field reconstruction, room responses, radiation, and acoustic transfer [21, 22, 23, 24, 25, 26, 27]. These studies establish the promise of learned wave surrogates, but generic nonlinear source encoders do not in general guarantee exact complex superposition. Moreover, fixed-sensor vector encodings, such as the baseline used here, do not represent the source surface through explicit quadrature weights.

This work introduces a quadrature-aware complex-linear boundary operator, denoted CLBO. Learned nonlinear networks represent source- and receiver-dependent basis functions, while the complex source velocity enters only through a weighted surface contraction. The architecture is therefore nonlinear in coordinates and frequency but exactly complex-linear in the distributed source field. It is also invariant to consistent permutations of source samples and accepts a variable number of source quadrature points. The present contribution lies in integrating low-rank source–receiver bases, permutation-invariant surface aggregation, and explicit quadrature into a complex boundary-to-field surrogate whose source algebra follows linear acoustics by construction. The resulting formulation preserves the known dependence on prescribed boundary velocity while enabling direct tests of generalization to newly simulated complex source combinations. Recent boundary-integral and low-rank neural operators further motivate this structure, but address different PDEs, geometries, or training objectives [18, 20].

The contribution is evaluated in a resonant three-dimensional rectangular cavity using reference data generated by a multiple-relaxation-time (MRT) lattice Boltzmann method (LBM). The paper makes four contributions:

1. a low-rank boundary-to-field neural operator that preserves complex source superposition, homogeneity, and zero-input consistency by construction;
2. a surface representation using source coordinates, normals, and quadrature weights rather than a fixed flattened boundary vector;
3. a direct generalization test in which new complex source mixtures are simulated with MRT-LBM and evaluated against both learned models; and
4. a verified evaluation workflow from reference calculations to standardized cases, fixed case-level splits, training, evaluation, and ablation studies.

The study concerns one fixed cavity and a linear time-harmonic acoustic regime in nondimensional lattice units. The data interface is solver-agnostic: another verified numerical method or measurement system can provide the labels when it supplies the same boundary and field quantities. A trained checkpoint nevertheless represents the geometries and passive boundary conditions covered by its training data unless those quantities are introduced explicitly as model inputs.

2 Boundary-to-field problem

2.1 Frequency-domain acoustics

Let $\Omega \subset \mathbb{R}^3$ be a bounded acoustic domain with boundary $\Gamma = \Gamma_v \cup \Gamma_r$, where Γ_v is a vibrating surface and Γ_r is rigid. With the convention

$$p(\mathbf{x}, t) = \Re \{ \widehat{p}(\mathbf{x}, \omega) e^{i\omega t} \}, \quad (1)$$

the complex pressure satisfies

$$\nabla^2 \widehat{p} + k^2 \widehat{p} = 0 \quad \text{in } \Omega, \quad (2)$$

$$\frac{\partial \widehat{p}}{\partial n} = -i\rho_0\omega \widehat{v}_n \quad \text{on } \Gamma_v, \quad (3)$$

$$\frac{\partial \widehat{p}}{\partial n} = 0 \quad \text{on } \Gamma_r, \quad (4)$$

where $k = \omega/c_0$ and $\widehat{v}_n = \widehat{\mathbf{u}} \cdot \mathbf{n}$ denotes the prescribed normal velocity, positive in the outward-normal direction. Away from exact lossless cavity eigenfrequencies, or when the finite damping present in the reference calculation is included, Eq. (4) defines a complex-linear map

$$\mathcal{A}_\omega : \widehat{v}_n(\mathbf{s}) \mapsto \widehat{p}(\mathbf{x}, \omega). \quad (5)$$

Consequently,

$$\mathcal{A}_\omega(av_1 + bv_2) = a\mathcal{A}_\omega(v_1) + b\mathcal{A}_\omega(v_2), \quad a, b \in \mathbb{C}. \quad (6)$$

2.2 Discrete surface representation

A sampled source surface is represented by

$$\mathcal{S}_h = \{(\mathbf{s}_j, \mathbf{n}_j, A_j, v_j)\}_{j=1}^{N_s}, \quad (7)$$

where $\mathbf{s}_j$ is a source location, $\mathbf{n}_j$ is the associated normal, $A_j > 0$ is a quadrature weight, and $v_j \in \mathbb{C}$ is the sampled normal velocity. A direct learned kernel would take the form

$$\hat{p}(\mathbf{x}, \omega) \approx \sum_{j=1}^{N_s} A_j K_\theta(\mathbf{x}, \mathbf{s}_j, \mathbf{n}_j, \omega) v_j, \quad (8)$$

with cost proportional to $N_s N_r$ for N_r receiver queries. This expression is analogous to a learned boundary-integral or Kirchhoff–Helmholtz transfer representation, but with the Green-type kernel replaced by a trainable source–receiver kernel. Classical and modern acoustic boundary-element formulations motivate this boundary-only view while also highlighting the dense-kernel cost that the present low-rank factorization avoids [4, 28, 29]. The proposed model factorizes this interaction into source and receiver bases.

3 Quadrature-aware complex-linear boundary operator

3.1 Coordinate and frequency features

Coordinates and frequencies are normalized using statistics computed exclusively from the training split. The same feature map is used for source and receiver coordinates. For a normalized scalar coordinate q , Fourier features are

$$\gamma(q) = [q, \sin(2^0 \pi q), \cos(2^0 \pi q), \dots, \sin(2^{B-1} \pi q), \cos(2^{B-1} \pi q)]. \quad (9)$$

The implementation configuration used in this study uses 6 coordinate bands and 7 frequency bands. Fourier feature mappings are used to improve representation of oscillatory coordinate dependence [30]. Complex quantities are stored as paired real and imaginary channels, while their products are evaluated with exact complex arithmetic implemented over those channels; this differs from using unconstrained complex-valued hidden layers [31].

3.2 Low-rank source and receiver bases

The source network produces a complex basis

$$\boldsymbol{\psi}_\theta(\mathbf{s}_j, \mathbf{n}_j, \omega) \in \mathbb{C}^R, \quad (10)$$

and the receiver network produces

$$\boldsymbol{\phi}_\theta(\mathbf{x}_q, \omega) \in \mathbb{C}^R. \quad (11)$$

The latent source coefficients are obtained by surface quadrature,

$$z_r = \sum_{j=1}^{N_s} A_j \boldsymbol{\psi}_{\theta,r}(\mathbf{s}_j, \mathbf{n}_j, \omega) v_j, \quad (12)$$

and pressure is decoded as

$$\hat{p}_\theta(\mathbf{x}_q, \omega) = \frac{1}{\sqrt{R}} \sum_{r=1}^R \phi_{\theta,r}(\mathbf{x}_q, \omega) z_r. \quad (13)$$

The factorized evaluation cost is $\mathcal{O}((N_s + N_r)R)$ after feature-network evaluation, rather than $\mathcal{O}(N_s N_r)$ for a dense source-receiver kernel. Figure 1 summarizes the corresponding network structure.

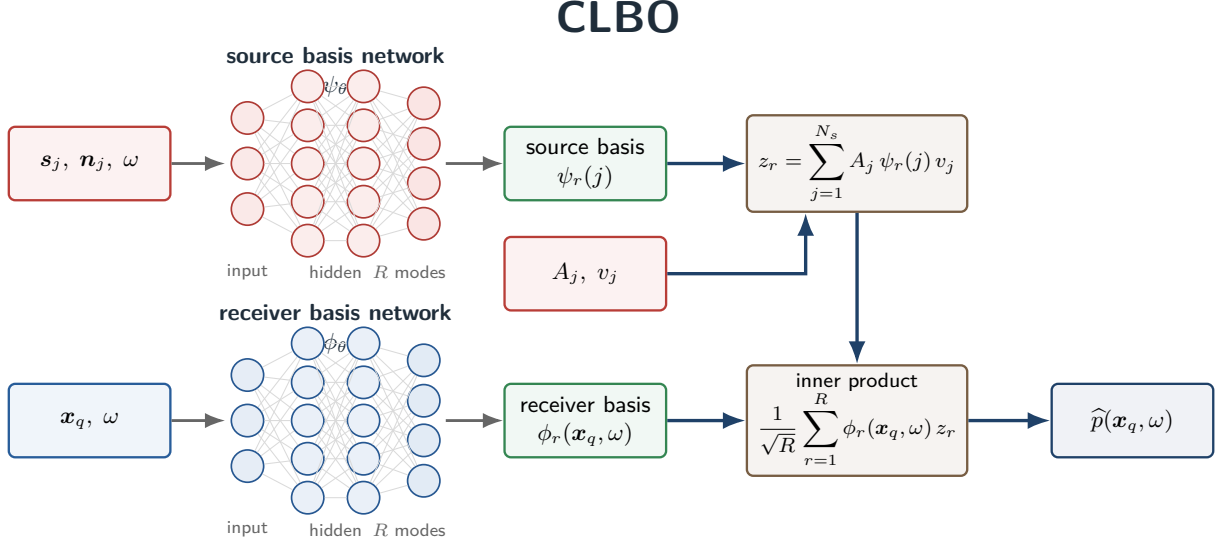


Figure 1: Architecture of the proposed CLBO. A source basis network maps each source location, normal, and frequency to R complex basis functions after normalized coordinate and frequency features. The prescribed boundary velocity enters only through the weighted surface sum in Eq. (12). A receiver basis network maps each query location and frequency to R complex basis functions, and the predicted pressure follows from the low-rank inner product in Eq. (13). The multilayer blocks are schematic; the implemented encoders use depth 5, hidden width 256, and latent rank 96.

3.3 Exact structural properties

Complex linearity. For fixed source coordinates, normals, quadrature weights, frequency, and receiver coordinates, ψ_θ and ϕ_θ do not depend on source amplitude. Substitution of $av_1 + bv_2$ into Eq. (12) and distribution of the finite sum therefore yield

$$\hat{\mathcal{A}}_\theta(av_1 + bv_2) = a\hat{\mathcal{A}}_\theta(v_1) + b\hat{\mathcal{A}}_\theta(v_2). \quad (14)$$

The equality is exact in real arithmetic and is limited only by floating-point roundoff in implementation.

Zero-input consistency. Setting $v_j = 0$ for all source nodes gives $z_r = 0$ for every rank component and therefore $\hat{p}_\theta = 0$ without requiring training examples at zero excitation.

Permutation invariance. A common permutation of $\mathbf{s}_j$, $\mathbf{n}_j$, A_j , and v_j leaves Eq. (12) unchanged because the source aggregation is a sum.

Discretization interface. The input dimension is not tied to one number of source nodes. Changing the source discretization changes the finite quadrature in Eq. (12); accuracy then depends on both the learned kernel approximation and the quality of the source quadrature.

3.4 Relation to prior operator structures

The architecture shares ingredients with several established model classes. Because the vibrating boundary is represented by coordinates, normals, areas, and complex values, the source input is an unordered geometric point set rather than only a sampled vector. Its source summation has the permutation-invariant form studied in Deep Sets and is related to point-set and attention-based set encoders [14, 15, 16]; its source–receiver factorization is related to low-rank neural-operator kernels [7, 18]; and its explicit integration weights are related to quadrature-based kernel operators [17]. Geometry-informed operators additionally address variable domains and discretization convergence by encoding the domain itself [9], whereas the present model is conditioned on source and receiver coordinates inside one fixed cavity. Boundary-integral neural methods embed known integral equations or Green kernels into training [19, 20]; in contrast, CLBO learns the acoustic transfer kernel from verified source–field pairs while hard-coding only the linear dependence on prescribed complex boundary velocity. Accordingly, the contribution is framed as a structure-preserving acoustic boundary-to-field formulation, supported by tests of exact source algebra and generalization to independently simulated complex source combinations.

3.5 Fixed-sensor DeepONet baseline

The baseline follows the conventional branch–trunk construction [6]. Its branch network receives the ordered real and imaginary source-velocity samples at 256 fixed branch sensors. When the number of source samples differs, the stored one-dimensional ordering is linearly resampled to that branch-sensor count. The trunk network receives receiver coordinates and frequency, and a complex latent contraction produces pressure. A zero-source branch response is subtracted so that the baseline also returns zero for zero excitation. The branch remains nonlinear in source velocity and does not guarantee complex superposition or invariance to arbitrary source-node permutations. Both models use the same latent rank, hidden width, depth, coordinate features, optimizer schedule, data splits, and evaluation metrics. Because the input encoders differ, equal widths do not imply identical parameter counts; parameter counts and inference costs are therefore reported explicitly, and conclusions are based on both predictive error and structural tests rather than capacity alone. Figure 2 shows the baseline layout used in the comparison.

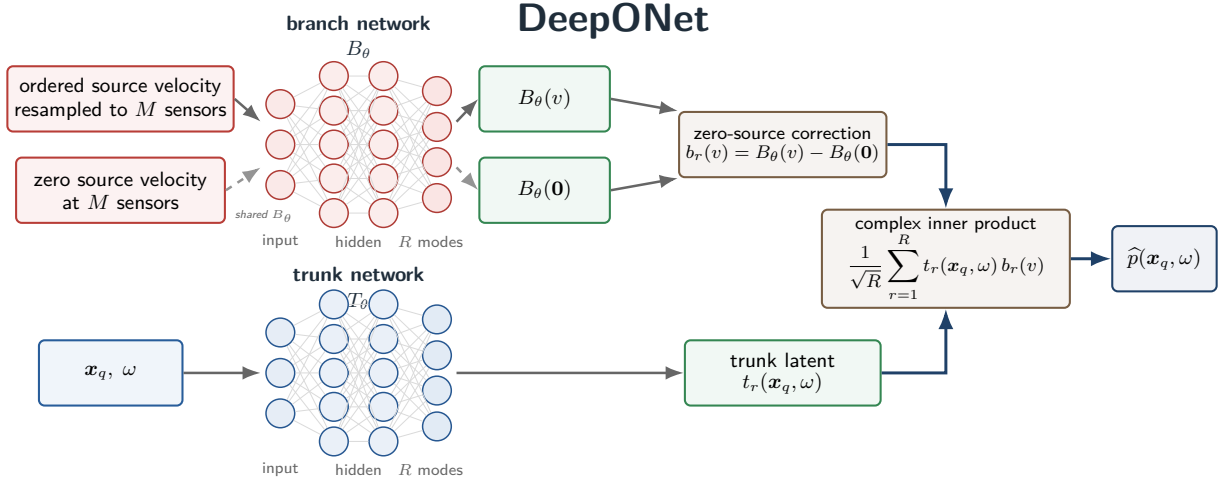


Figure 2: Architecture of the fixed-sensor DeepONet baseline. The branch network encodes ordered real and imaginary source-velocity samples resampled to 256 fixed branch sensors when $N_s \neq M$. The same branch network is evaluated at zero input and subtracted so that zero wall velocity yields zero branch coefficients. The trunk network maps normalized receiver coordinates, frequency, and Fourier features to R complex basis functions. Branch and trunk latents are combined by the same low-rank inner product as in Eq. (13), but source amplitude remains inside the nonlinear branch encoder. The multilayer blocks are schematic; the implemented networks use depth 5, hidden width 256, and latent rank 96.

4 Reference-data generation and verification

Reference cases are generated in the fixed rectangular cavity shown in Fig. 3. The minimum- x wall is prescribed as the vibrating boundary, the remaining walls are rigid, and complex pressure is sampled at interior receiver locations.

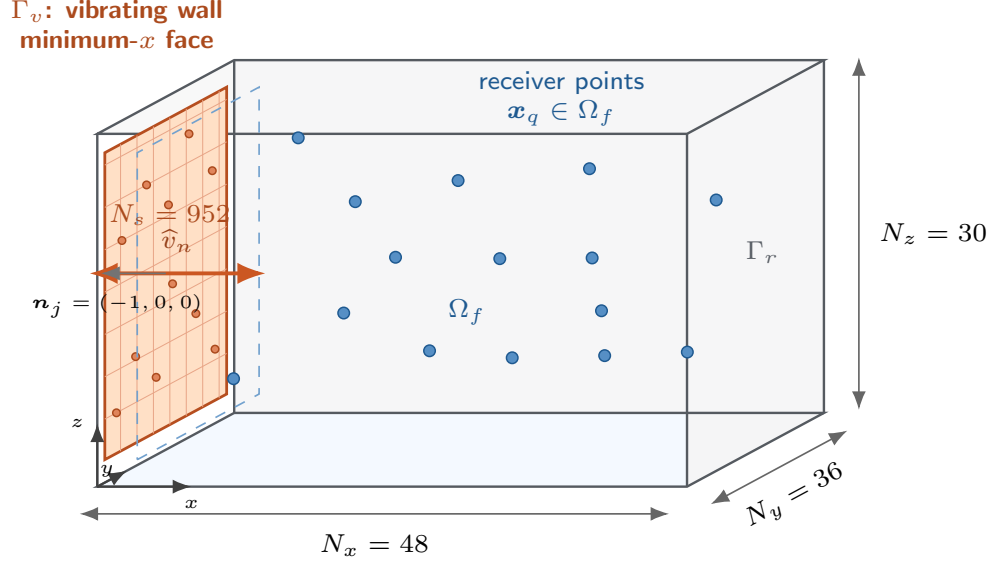


Figure 3: Geometry and sampling layout used for reference-data generation. The rectangular $48 \times 36 \times 30$ lattice nodes lattice cavity has prescribed normal velocity on the minimum- x wall Γ_v , rigid remaining walls Γ_r , $N_s = 952$ source nodes on the vibrating wall, and interior receiver points $\mathbf{x}_q \in \Omega_f$.

4.1 MRT-LBM formulation

Reference cases are generated with a three-dimensional 19-velocity (D3Q19) MRT-LBM. The collision step follows the standard MRT-LBM formulation [32, 33]. LBM has also been benchmarked in broader direct-numerical-simulation/large-eddy-simulation (DNS/LES) flow settings [34, 35, 36], while acoustic-wave propagation and damping in LBM are addressed by acoustic-specific studies cited below. The collision step is performed in moment space,

$$\mathbf{m}(\mathbf{x}, t) = \mathbf{M} \mathbf{f}(\mathbf{x}, t), \quad (15)$$

$$\mathbf{m}^*(\mathbf{x}, t) = \mathbf{m}(\mathbf{x}, t) - \mathbf{S} [\mathbf{m}(\mathbf{x}, t) - \mathbf{m}^{\text{eq}}(\mathbf{x}, t)], \quad (16)$$

$$\mathbf{f}^*(\mathbf{x}, t) = \mathbf{M}^{-1} \mathbf{m}^*(\mathbf{x}, t), \quad (17)$$

$$f_i(\mathbf{x} + \mathbf{c}_i \Delta t, t + \Delta t) = f_i^*(\mathbf{x}, t). \quad (18)$$

Here $\mathbf{M}$ is the moment transformation, $\mathbf{S}$ is diagonal, and conserved density and momentum moments are not relaxed. LBM acoustic-wave propagation and damping have been studied in prior acoustic LBM work [37, 38]. Low-dispersion and low-dissipation MRT-LBM formulations have also been developed specifically for computational aeroacoustics [39]. In the present reference solver, MRT collision is used to control numerical dispersion, dissipation, and stability.

The outer lattice shell represents the boundary. Five faces use halfway bounce-back and one face receives a harmonic moving-wall correction. A half-cosine ramp reduces broadband start-up content. Complex pressure is recovered by fitting the time history after a specified

transient interval to the target harmonic. All quantities in the present dataset are reported in nondimensional lattice units; no physical sound-pressure level is inferred without a separate dimensional mapping.

4.2 Verification protocol

The verification suite checks lattice identities and equilibrium preservation, a separate closed-duct piston benchmark that directly tests the moving-wall boundary condition without complex scaling, the first longitudinal rigid-cavity resonance on the $48 \times 36 \times 30$ training grid, the associated mode shape, small-amplitude source superposition, and a multi-resolution study on rectangular cavities with the same aspect ratio. For a rigid rectangular cavity with dimensions L_x, L_y, L_z , the analytical frequencies are

$$f_{lmn} = \frac{c_0}{2} \sqrt{\left(\frac{l}{L_x}\right)^2 + \left(\frac{m}{L_y}\right)^2 + \left(\frac{n}{L_z}\right)^2}. \quad (19)$$

The corresponding pressure mode is proportional to

$$\Phi_{lmn}(x, y, z) = \cos\left(\frac{l\pi x}{L_x}\right) \cos\left(\frac{m\pi y}{L_y}\right) \cos\left(\frac{n\pi z}{L_z}\right). \quad (20)$$

Mode-shape comparisons allow one optimal complex scale because an eigenfunction is defined only up to complex amplitude and phase; they therefore assess cavity eigenstructure rather than the absolute amplitude and phase of the wall forcing, which is checked separately by the closed-duct piston benchmark. Figure 4 summarizes the resonance check on the training grid and the normalized grid-resolution study across coarser and finer rectangular cavities.

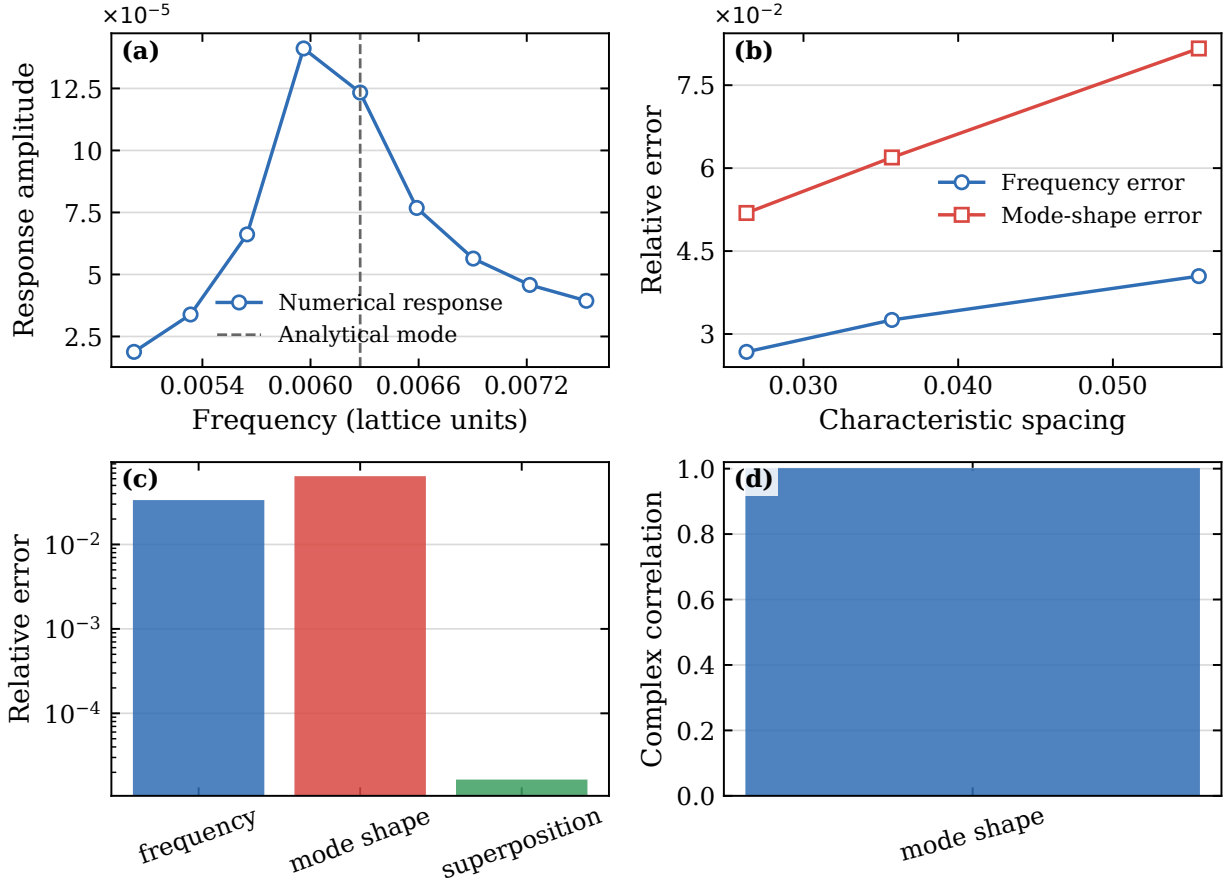


Figure 4: Verification of the MRT-LBM reference solver. Frequencies and response amplitudes are reported in nondimensional lattice units; relative errors and correlations are dimensionless. Panel (a) shows the forced response amplitude on the $48 \times 36 \times 30$ training cavity as the excitation frequency is swept about the first longitudinal mode; the dashed vertical line marks the analytical resonance frequency. Panel (b) reports relative frequency error and mode-shape error versus normalized characteristic grid spacing, defined as the inverse of the shortest fluid-node count, for three rectangular resolutions ($32 \times 24 \times 20$, $48 \times 36 \times 30$, and $64 \times 48 \times 40$); the middle resolution corresponds to the training configuration. Panel (c) lists the resonance frequency error, mode-shape error, and small-amplitude superposition error measured on the training grid. Panel (d) gives the complex correlation between the numerical and analytical mode shapes after optimal complex scaling.

Thus, the reference-data pipeline uses a two-stage validation hierarchy: analytical acoustic benchmarks verify the MRT-LBM solver for canonical cases, and the learned operators are then evaluated against this verified MRT-LBM reference over the full boundary-excitation dataset.

4.3 Dataset configuration

The principal dataset contains 640 complete source-frequency cases. Each case stores source coordinates, normals, areas, complex velocity, frequency, receiver coordinates, complex pressure, 256 interior collocation points, and 128 rigid-wall collocation points. Table 1 summarizes the resolved generation configuration.

Table 1: Resolved reference-data configuration. Frequencies and field quantities are expressed in lattice units.

Quantity	Value
Geometry	rectangular cavity
Lattice shape	$48 \times 36 \times 30$ lattice nodes
Vibrating boundary	minimum- x wall
Source nodes on reference wall	952
MRT relaxation parameter τ	0.56
Time steps per case	7200
Harmonic-fit start step	4200
Source-ramp duration	750
Cases	640
Discrete frequencies	16
Frequency range	0.00391–0.0235 cycles per lattice time step
Receiver queries per case	512
Source families	Gaussian, Fourier, smooth random modal; uniform wall motion used only for piston verification
Peak source-velocity range	2×10^{-5} – 8×10^{-5} lattice velocity units
Stored pressure	complex lattice pressure

5 Learning and evaluation protocol

5.1 Case-level splitting and normalization

Complete source-frequency cases are split before model training. The default split uses 80% of cases for training, 10% for validation, and 10% for testing. All receiver samples from one case remain in the same split. A single fixed case-level split is reused across models for the principal comparison. Coordinate, frequency, source-velocity, and pressure statistics are computed from the training split only. Each case stores 512 receiver queries. The configured training cap is 768 receiver points, so all stored receivers are used during both training and evaluation.

The main comparison is repeated for five seeds (2026, 1234, 2345, 3456, 4567). Results are reported as mean and standard deviation across seeds, with paired per-case distributions retained for statistical comparison. The source-family holdout is a separate seed 2026

secondary study in which the smooth random modal source family is excluded from training and validation and used only for testing.

5.2 Training objective

The primary loss is a masked complex mean-square error over receiver queries,

$$\mathcal{L}_{\text{data}} = \frac{1}{2N} \sum_{q=1}^N \|\hat{\mathbf{p}}_{\theta,q} - \mathbf{p}_q\|_2^2, \quad (21)$$

where $\mathbf{p} = (\Re p, \Im p)$. The principal model comparison uses supervised training only ($\lambda_H = \lambda_R = \lambda_V = 0$) so that the architectural effect is not confounded by derivative-based penalties.

Physics-informed variants are evaluated as ablations. The training implementation supports normalized residuals for the Helmholtz equation, the rigid-wall Neumann condition, and the prescribed vibrating-wall condition,

$$\mathcal{L} = \lambda_d \mathcal{L}_{\text{data}} + \lambda_H \mathcal{L}_H + \lambda_R \mathcal{L}_R + \lambda_V \mathcal{L}_V. \quad (22)$$

The principal model comparison uses $\lambda_H = \lambda_R = \lambda_V = 0$. The physics-residual ablation reported in Table 5 and Fig. 9 activates only the interior Helmholtz and rigid-wall terms ($\lambda_H = 2.25 \times 10^{-3}$, $\lambda_R = 10^{-3}$, $\lambda_V = 0$) in the physics-residual training configuration. During preliminary weight calibration, adding $\mathcal{L}_V$ consistently degraded validation performance relative to the Helmholtz–rigid configuration, so the vibrating-wall residual was omitted from the reported analysis. The residual weights and ramp schedules are held fixed within each ablation.

5.3 Model and optimization settings

Table 2: Common model and optimization settings for the principal comparison.

Setting	Value
Latent rank	96
Hidden width	256
Hidden layers per network	5
Activation	hyperbolic tangent
Coordinate Fourier bands	6
Frequency Fourier bands	7
DeepONet branch sensors	256
Training receiver cap	768 points
Cases per batch	2
Maximum epochs	200
Optimizer	AdamW
Base learning rate	2×10^{-4}
Minimum learning rate	2×10^{-6}
Weight decay	10^{-6}
Warm-up	10 epochs
Gradient clipping	1.0
Early-stopping patience	60 epochs
Training precision	single precision (<code>float32</code>)

5.4 Evaluation studies

The evaluation is organized around distinct questions rather than one aggregate random-split score.

1. **Primary held-out cases:** accuracy on test source-frequency cases using all stored receiver queries.
2. **Receiver interpolation:** both models are retrained using only a deterministic subset of receiver coordinates in each case, and errors are evaluated only on the held-out receiver coordinates.
3. **Algebraic structure:** complex superposition, homogeneity, zero-input response, and source-node permutation.
4. **Unseen source combinations:** held-out source fields at a common frequency are combined using random complex coefficients; for each combined boundary excitation, a fresh MRT-LBM reference solution is generated and evaluated on a common receiver cloud.
5. **Source discretization:** the same source-generation rule sampled on the native, coarsened, and nonuniform source-wall node sets while the physical surface is held fixed.

6. **Source-family generalization:** the smooth random modal source family is excluded from training and validation and used only for testing.
7. **Data efficiency and physics ablations:** training with 10%, 25%, 50%, and 100% of the available training cases, with optional residual penalties.
8. **Computational performance:** synchronized inference timing at fixed source and receiver counts, together with reference-solver time, parameter count, and peak memory.

5.5 Metrics

The primary metric is the complex relative field error

$$\epsilon_p = \frac{\|\hat{p} - p\|_2}{\|p\|_2}, \quad (23)$$

evaluated over valid receiver samples in a case or over a pooled evaluation vector formed by concatenating all valid receiver points in a split. Secondary reported metrics are complex correlation,

$$\rho_c = \frac{\left| \sum_q p_q^* \hat{p}_q \right|}{\left(\sum_q |p_q|^2 \right)^{1/2} \left(\sum_q |\hat{p}_q|^2 \right)^{1/2}}, \quad (24)$$

amplitude-ratio mean absolute error in decibels,

$$\epsilon_{A,\text{dB}} = \frac{1}{N_m} \sum_{q \in \mathcal{M}} \left| 20 \log_{10} \frac{|\hat{p}_q| + \varepsilon}{|p_q| + \varepsilon} \right|, \quad (25)$$

and phase mean absolute error

$$\epsilon_\varphi = \frac{1}{N_m} \sum_{q \in \mathcal{M}} \left| \arg(\hat{p}_q p_q^*) \right|. \quad (26)$$

Phase errors are reported in degrees in the result tables and figures. The mask $\mathcal{M}$ contains valid receiver samples whose reference amplitude is within 60 dB of the maximum reference amplitude of the evaluated field or pooled evaluation vector. Table 3 reports pooled split values for ϵ_p , $\epsilon_{A,\text{dB}}$, and ϵ_φ , and the mean of per-case values for ρ_c . Algebraic superposition is measured by

$$\epsilon_{\text{lin}} = \frac{\|\hat{\mathcal{A}}(av_1 + bv_2) - a\hat{\mathcal{A}}(v_1) - b\hat{\mathcal{A}}(v_2)\|_2}{\|a\hat{\mathcal{A}}(v_1) + b\hat{\mathcal{A}}(v_2)\|_2 + \varepsilon}. \quad (27)$$

The algebraic test above verifies implementation structure but does not by itself establish predictive accuracy on a new physical field. A separate mixed-source test constructs

$$v_{\text{mix}} = av_1 + bv_2, \quad (28)$$

from held-out source fields at the same frequency, carries out a new MRT-LBM reference simulation, and evaluates

$$\epsilon_{\text{mix}} = \frac{\|\hat{p}(v_{\text{mix}}) - p_{\text{LBM}}(v_{\text{mix}})\|_2}{\|p_{\text{LBM}}(v_{\text{mix}})\|_2}. \quad (29)$$

The paired difference between the fixed-sensor DeepONet baseline and CLBO errors is summarized with a 95% percentile bootstrap interval over the identical mixed cases, using 10,000 resamples of the paired case-level differences [40].

6 Results

6.1 Reference-solver verification

Figure 4 reports the checks that were completed before the 640-case generator was enabled. In panel (a), the numerical response peaks near $f = 0.0061$ while the analytical first longitudinal mode lies at $f = 0.0063$; the resulting relative frequency error is 0.0325. Panel (b) shows that both frequency and mode-shape errors decrease as the rectangular cavity resolution is refined from $32 \times 24 \times 20$ to $64 \times 48 \times 40$ nodes, with the $48 \times 36 \times 30$ training resolution in between. The horizontal coordinate is a normalized characteristic grid spacing, $1/\min(N_x - 2, N_y - 2, N_z - 2)$, rather than a change to the unit LBM lattice spacing; fitting frequency error against this spacing gives an observed order of 0.549.

Panel (c) summarizes the training-grid verification metrics: mode-shape relative error 0.0625 and superposition error 1.59×10^{-5} . Panel (d) confirms strong spatial agreement between numerical and analytical mode shapes, with complex correlation 0.998.

A separate closed-duct piston validation directly checks the prescribed normal-velocity sign and phasor convention without fitting a complex scale. At $f/f_1 = 0.4$, the complex relative pressure error was 0.182, the amplitude relative error was 0.115, and the phase root-mean-square error (RMSE) was 8.78° . Because the analytical duct solution is lossless whereas the MRT-LBM calculation at $\tau = 0.56$ has finite viscosity and finite-grid dissipation, exact inviscid amplitude agreement is not expected; the result is therefore interpreted as a finite-grid sign, phase, and amplitude-consistency check.

This piston benchmark corresponds to the uniform normal-velocity limit of the vibrating wall; nonuniform Gaussian, Fourier, and smooth random modal source fields are assessed through the verified MRT-LBM reference data rather than by comparison with the one-dimensional piston solution.

6.2 Primary predictive accuracy

Across five seeds, CLBO achieved a complex relative field error of 0.184 ± 0.00771 , while the fixed-sensor DeepONet obtained 0.367 ± 0.00742 . The corresponding amplitude-ratio errors were 2.32 and 3.78, and the phase errors were 17.2 and 28.9. Table 3 reports the full comparison. Figure 5 visualizes one held-out test case chosen near the median per-case error on the test split. The MRT-LBM reference and the supervised CLBO prediction are evaluated at the same receiver coordinates and displayed on the fixed- z plane with the densest receiver sampling. Panel (a) shows the reference magnitude and panel (b) the predicted magnitude; the two panels share a common color scale. Panel (c) maps the pointwise complex error normalized by the case root-mean-square reference amplitude. Panels (d) and (e) show reference and predicted phase, and panel (f) the absolute wrapped phase error, masked where the amplitude mask $\mathcal{M}$ applies. Figure 6 summarizes the same held-out evaluation in

aggregate. Panel (a) pools per-case errors over five training seeds and therefore sits slightly above the pooled values when hard resonant cases are present. Panel (b) reports the global test-set error obtained by concatenating all receiver points in the split; the markers match Table 3 and the early-stopping criterion.

Table 3: Primary held-out-case performance. Relative L_2 values are mean $\pm$ standard deviation across independent training seeds; other reported metrics are seed means. Amplitude error is an amplitude-ratio error and is not a physical sound pressure level (SPL).

Model	Relative L_2	Complex corr.	Amp. MAE (dB)	Phase MAE (deg)	Parameters
Fixed-sensor DeepONet	0.367 ± 0.00742	0.841	3.78	28.9	770432
CLBO	0.184 ± 0.00771	0.913	2.32	17.2	653952

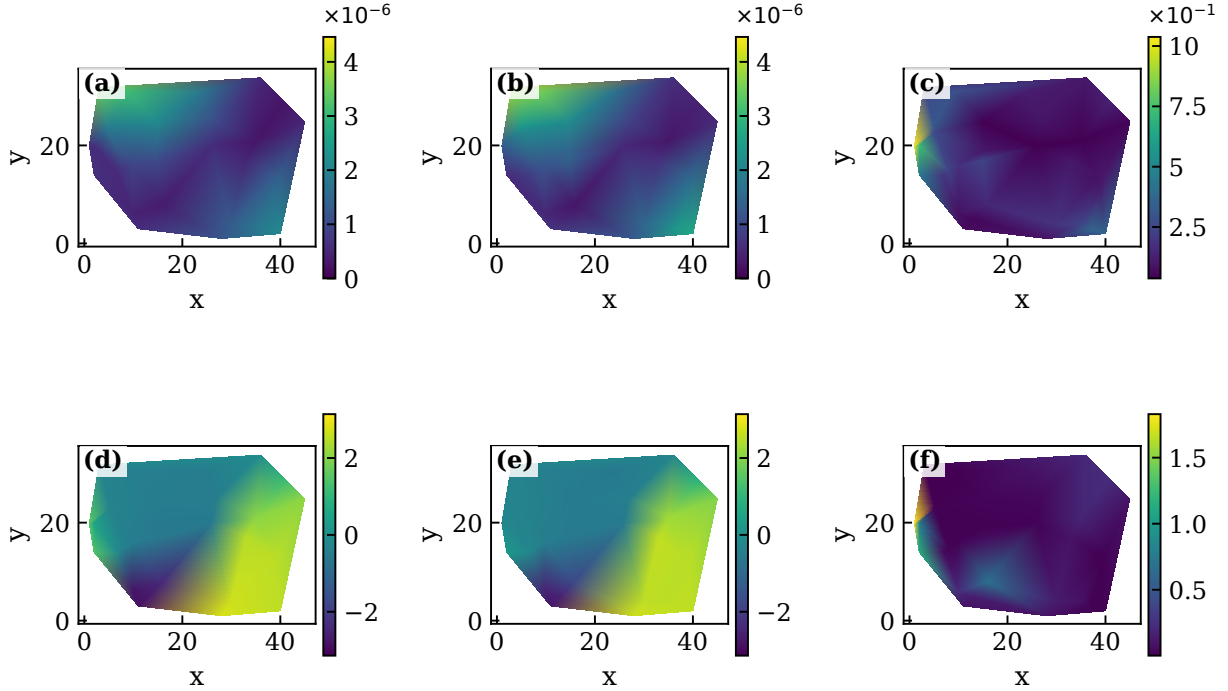


Figure 5: Complex pressure predictions for one held-out test case near the median per-case test error. Fields are shown on the fixed- z receiver plane with the densest sampling and rendered by triangulation over scattered receiver locations; they are not a structured interior grid. Panels (a) and (b) compare MRT-LBM reference and CLBO predicted magnitude on a shared color scale. Panel (c) shows the pointwise complex error magnitude normalized by the case root-mean-square reference amplitude. Panels (d)–(f) show reference phase, predicted phase, and absolute wrapped phase error in radians, respectively, masked where the amplitude mask $\mathcal{M}$ applies.

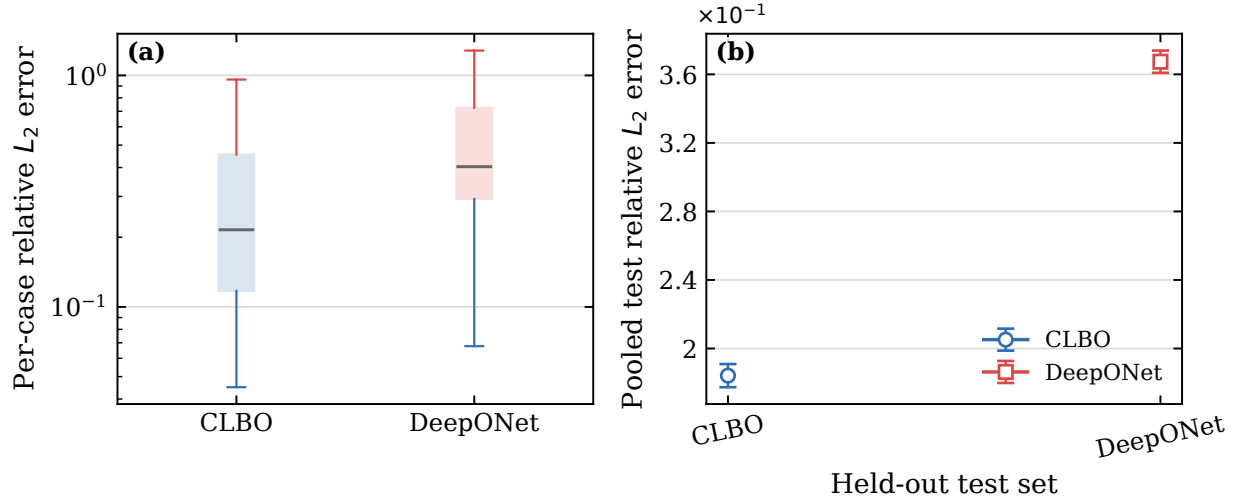


Figure 6: Primary model comparison on the shared held-out test split. Both models were trained on the 640-case $48 \times 36 \times 30$ lattice nodes dataset with identical source fields, receiver clouds, optimization settings, and supervised-only losses. Panel (a) shows the per-case complex relative field error for all 10% test cases, pooled over five training seeds; each box summarizes the distribution over held-out cases and training seeds. Panel (b) shows the global test-set relative error obtained by concatenating all receiver points in the test split and averaging over the same seeds; it uses the same pooled relative- L_2 definition as the validation metric used for early stopping and is the test metric reported in Table 3. Markers give the seed-averaged pooled error and bars indicate 95% confidence intervals across seeds.

On five newly simulated mixed-source cases, evaluated with the seed 2026 primary supervised checkpoints, the mean relative errors were 0.237 for CLBO and 0.415 for DeepONet. CLBO achieved a paired win rate of 100%. The mean paired advantage, defined as $\epsilon_{\text{DeepONet}} - \epsilon_{\text{CLBO}}$, was 0.178 with a 95% percentile bootstrap interval of $[0.117, 0.249]$ over the five paired mixed cases. Because each target was generated by a new MRT-LBM reference simulation, this test measures predictive generalization to unseen complex source mixtures rather than only algebraic consistency of model outputs.

Table 4: Generalization to newly simulated complex source mixtures. Pairs are drawn from the held-out test split at common frequency and source mesh; each mixture is simulated once with MRT-LBM and evaluated on the same receiver cloud. Both models use the seed 2026 primary supervised checkpoints. The manuscript reports the paired error advantage with a 95% percentile bootstrap interval over paired case-level differences; win rate counts mixtures with lower relative L_2 error.

Model	Mixed cases	Mean relative L_2	Median relative L_2	Paired win rate	Mean complex corr.
Fixed-sensor DeepONet	5	0.415	0.474	0%	0.919
CLBO	5	0.237	0.177	100%	0.961

6.3 Receiver interpolation

Receiver-coordinate interpolation was evaluated by retraining both models with only a deterministic subset of receiver locations included in the supervised loss. The remaining receiver locations were excluded during training and used only for evaluation. This test measures whether the learned surrogate represents a continuous pressure field over receiver coordinates rather than only fitting the receiver locations used in training.

Across 64 held-out test cases, CLBO reduced the mean held-out-receiver relative error from 0.581 for DeepONet to 0.332. The median error was also reduced, from 0.427 for DeepONet to 0.245 for CLBO. Figure 7 shows the per-case error distributions on the held-out receiver coordinates.

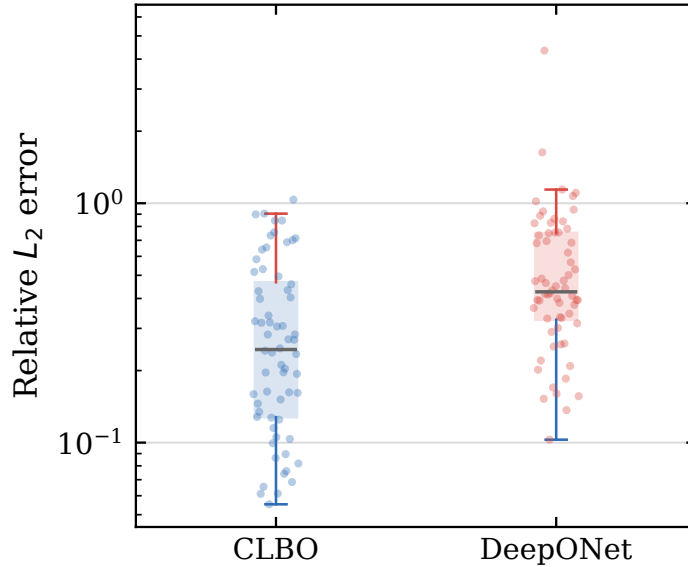


Figure 7: Receiver interpolation on held-out receiver coordinates. Both models were retrained using only a deterministic subset of receiver locations in each case and evaluated on the remaining unseen receiver locations inside the same cavity. Each point is one held-out test case ($n = 64$). CLBO gives lower mean and median relative field error than the fixed-sensor DeepONet baseline.

6.4 Exact structure and source discretization

The measured CLBO superposition error was 1.31×10^{-7} , compared with 0.406 for DeepONet. The zero-input and source-permutation errors of CLBO were 0 and 9.10×10^{-8} , respectively. These measurements confirm the algebraic guarantees of Eqs. (12)–(13) at implementation precision. Figure 8 reports the same tests in panel (a).

When the same source-generation rule was sampled on alternative source-wall node subsets, CLBO obtained a source-sampling transfer error of 1.00, whereas the fixed-sensor baseline obtained 2.33. The experiment used the same physical domain and receiver cloud while changing the source-wall sampling pattern. Panel (b) of Figure 8 summarizes the spacing-resolved remeshing errors.

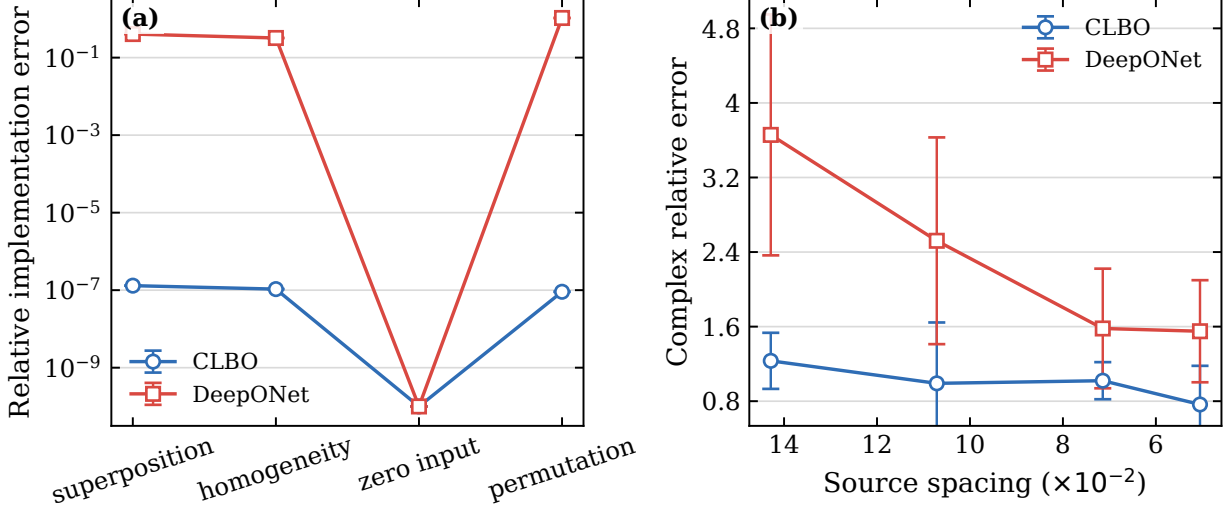


Figure 8: Algebraic structure and source-wall sampling transfer on the primary supervised checkpoints. Panel (a) shows superposition, homogeneity, zero-input, and permutation implementation errors measured with the same inference interface used in training; exact zero-input responses are shown at the logarithmic floor (10^{-10}). Panel (b) shows complex relative error on alternative source-wall node subsets at fixed geometry, excluding the native training discretization; characteristic source-wall sampling spacing decreases from left to right. Markers in panel (b) give means over remeshed test cases and bars indicate 95% confidence intervals across cases.

6.5 Source-family holdout

The source-family holdout evaluates generalization to a source-field family not seen during training. In this secondary seed 2026 study, the smooth random modal source family is removed from both training and validation. The resulting training and validation splits contain only Gaussian and Fourier source fields, while the held-out test split contains 18 smooth random modal cases.

The mean per-case complex relative errors were 0.402 for CLBO and 0.917 for DeepONet. This result indicates that CLBO gives lower error than the fixed-sensor DeepONet baseline when evaluated on the unseen smooth random modal source family.

6.6 Ablations and data efficiency

With 10% of the training cases, CLBO and DeepONet obtained errors of 1.04 and 1.00, respectively. Table 5 reports the full training-fraction grid. Figure 9 summarizes the same evaluation graphically. Panel (a) shows how test error decreases with training-set fraction for CLBO and the fixed-sensor DeepONet. Panel (b) compares three full-data CLBO settings: supervised-only training, an inference-time ablation that removes quadrature weights from the source contraction (0.307), and training with Helmholtz and rigid-wall physics residuals (0.183). The vibrating-wall residual was excluded from the reported physics-residual training configuration after preliminary calibration showed it degraded validation performance. These

ablations separate the effect of architectural linearity, numerical quadrature, and derivative-based regularization.

Table 5: Ablation and data-efficiency results on the fixed test manifest. Entries are complex relative field errors. Columns labeled 10%, 25%, and 50% denote the fraction of the 512-case training split used during training: whole training cases were drawn once at random (seed 2026) while validation and test splits stayed fixed. Supervised CLBO and DeepONet entries at 100% report the five-seed primary test means from Table 3. The quadrature-removal and physics-residual CLBO variants were evaluated at full training data only (seed 2026); the physics-residual row uses Helmholtz and rigid-wall penalties only ($\lambda_V = 0$).

Variant	10% of train	25% of train	50% of train	100% of train
Fixed-sensor DeepONet	1.00	0.882	0.626	0.367
CLBO without areas	—	—	—	0.307
CLBO	1.04	0.644	0.381	0.184
CLBO with Helmholtz and rigid-wall residuals	—	—	—	0.183

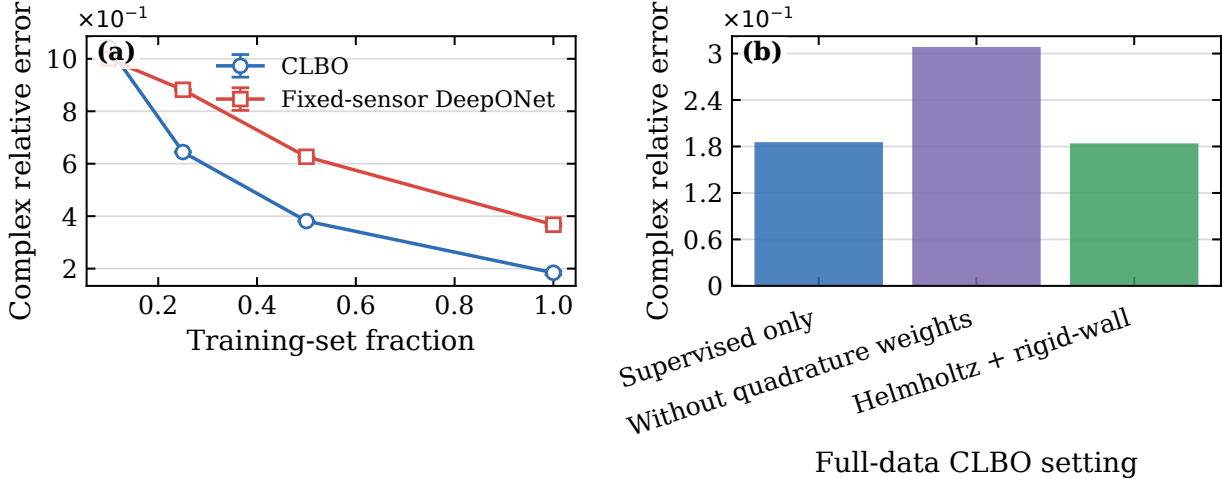


Figure 9: Data efficiency and controlled ablations on the fixed test manifest. Panel (a) shows complex relative error versus training-set fraction for CLBO and the fixed-sensor DeepONet; points at 10%, 25%, and 50% are single-seed (2026) data-efficiency training evaluations, and the 100% endpoints report five-seed primary test means. Panel (b) compares full-data CLBO settings: the five-seed supervised-only primary mean, an inference-time source-area ablation that sets quadrature weights to one using the seed 2026 supervised checkpoint, and a seed 2026 model trained with Helmholtz and rigid-wall physics residuals ($\lambda_V = 0$).

6.7 Computational performance

On CPU laptop (WSL2), single precision (`float32`) inference without GPU acceleration, with 952 source nodes and 512 receiver queries on one held-out test case, CLBO required 14.5 ms per case and the fixed-sensor DeepONet required 7.04 ms. A separate MRT-LBM reference

benchmark, re-running the same case with the data-generation solver, required 2.65×10^5 ms. The resulting speedups relative to MRT-LBM were 1.83×10^4 for CLBO and 3.8×10^4 for DeepONet. Neural timings exclude file input/output, use 10 warm-up forward passes followed by 100 timed forward passes, and apply device synchronization when applicable; the reference timing uses one untimed warm-up solve followed by two timed full lattice simulations. Table 6 lists the measured per-case times and speedups. Figure 10 summarizes the same benchmark graphically: panel (a) compares accuracy against per-case neural inference time at the full receiver count, panel (b) shows the scaling of neural runtime with receiver-query count while the MRT-LBM reference remains a fixed full-case solve, and panel (c) compares the trainable parameter counts of the supervised checkpoints.

Table 6: Computational cost at 952 source nodes and 512 receiver queries on one held-out test case, using the seed 2026 primary supervised checkpoints. Neural timings exclude file input/output and report the mean of 100 timed forward passes after 10 warm-up passes, with device synchronization applied when applicable. The MRT-LBM reference time is a separate benchmark using one untimed warm-up solve followed by two timed full lattice solves on the same case. Peak device memory was not recorded on CPU.

Method	Time per case	Peak memory	Speed relative to MRT-LBM
MRT-LBM reference	2.65×10^5 ms	N/A (CPU)	$1 \times$
Fixed-sensor DeepONet	7.04 ms	N/A (CPU)	3.8×10^4
CLBO	14.5 ms	N/A (CPU)	1.83×10^4

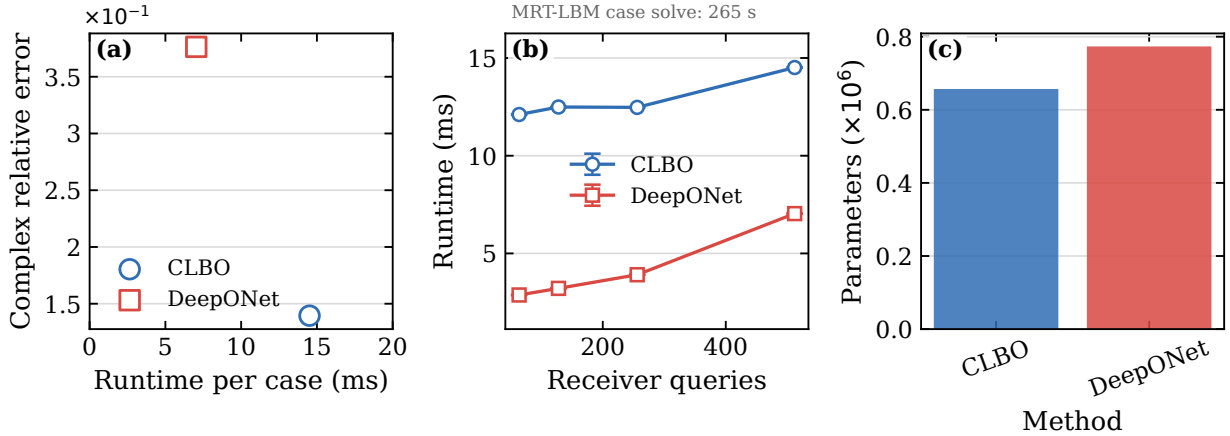


Figure 10: Accuracy and computational cost from the inference benchmark on the primary supervised checkpoints. Panel (a) compares complex relative error against per-case neural inference time at the full receiver count (512 queries, 952 source nodes). Panel (b) shows how neural runtime scales with receiver-query count; the MRT-LBM reference solve time is a single full lattice simulation and is annotated because it does not scale with query count. Panel (c) reports trainable parameter counts for the supervised checkpoints; peak device memory was not measured because the benchmark was run on central processing unit (CPU).

7 Discussion

The central distinction between CLBO and the fixed-sensor baseline is structural rather than merely architectural capacity. This emphasis is consistent with the broader move toward operator models that encode discretization, geometry, or integral structure rather than treating every physical relation as an unconstrained regression problem [7, 9, 17]. A generic nonlinear branch network must infer the superposition law from sampled source fields. In CLBO, that law is satisfied independently of training-set coverage because source amplitude appears only in the linear quadrature contraction. The measured error in Eq. (27) therefore tests numerical implementation rather than learned behavior. The mixed-source experiment in Eq. (29) is the complementary predictive test: it determines whether the hard constraint improves accuracy on new source amplitudes and phases when the target field is generated independently.

Quadrature serves a separate purpose. Exact linearity alone does not guarantee invariance to a changed source mesh; the discrete sum must also approximate the same continuous boundary integral. Coordinates and areas provide the required interface, while the remeshing study measures the remaining learned-kernel and quadrature errors. The result should be interpreted as transfer across the tested discretizations, not as universal mesh independence.

The fixed-sensor DeepONet is an appropriate reference for the cost of replacing explicit source integration with a nonlinear function encoder. It is deliberately tied to an ordered sensor vector and therefore represents the common deployment condition in which the source mesh is fixed. The comparison isolates whether exact source structure is beneficial when both models use the same training cases, receiver trunk representation, latent rank, and optimization budget. These results should be interpreted within the scoped boundary-excitation setting: CLBO gains its advantage by embedding the complex-linear source algebra of the acoustic operator, whereas the fixed-sensor DeepONet baseline must approximate that structure from data.

Physics-informed residuals are secondary to the architectural contribution. They can regularize low-data training, but they add first- and second-order automatic-differentiation cost and introduce loss-balancing choices. In the present study, only the interior Helmholtz and rigid-wall residuals were retained in the reported physics ablation; the prescribed vibrating-wall term degraded validation performance during preliminary calibration and was set to zero. The supervised CLBO is therefore the principal model, and residual terms are retained only when the ablation demonstrates a consistent improvement.

The present limitations are explicit. The acoustic problem is linear and time harmonic, the medium is quiescent, and one trained model represents one fixed cavity and passive boundary configuration. The reference labels use nondimensional lattice units. The workflow itself is not tied to MRT-LBM: the stored data contract contains only surface geometry, complex boundary values, operating frequency, receiver coordinates, and complex target fields. Replacing the label generator does not change the model interface, but new geometries or boundary conditions require representative training data or an explicitly geometry-conditioned extension.

A natural extension is to couple the present complex-linear boundary operator with a geometry encoder, such as a point-cloud, graph, signed-distance, mesh-based, or multiscale geometric representation. Mesh- and graph-based learned simulators provide one possible

route for encoding irregular geometric discretizations while retaining physical simulation structure [41]. Such a model would condition the learned source and receiver bases on the passive acoustic domain while preserving the exact linear dependence on prescribed boundary velocity. This would extend the present fixed-cavity formulation toward geometry-varying acoustic surrogate modeling for industrial computer-aided engineering (CAE) workflows.

8 Conclusions

A quadrature-aware complex-linear neural operator was developed for distributed boundary-to-field prediction in resonant acoustics. The model combines learned source and receiver bases with an explicit complex surface contraction. This construction guarantees source superposition, homogeneity, zero-input consistency, and source-node permutation invariance while permitting variable source-sample counts.

Using verified MRT-LBM reference cases, CLBO achieved a mean complex relative error of 0.184 ± 0.00771 across five seeds, compared with 0.367 ± 0.00742 for the fixed-sensor DeepONet. Its superposition error was 1.31×10^{-7} . On newly simulated mixed-source cases, the corresponding errors were 0.237 and 0.415. Its measured inference speedup was 1.83×10^4 . The structured evaluations separate ordinary predictive accuracy from exact algebraic behavior, unseen source-combination generalization, source discretization, source-family holdout, label efficiency, and computational cost. The resulting workflow provides a structured boundary-to-field learning formulation whose data interface is independent of the method used to obtain verified reference fields.

Data availability

The data supporting the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgements

This work was supported by Vinnova–Fordonstrategisk forskning och innovation (FFI) in the project "3D Virtual Platform for Digitalization of Holistic Acoustic Environment in Cabs of Heavy-Duty Vehicles (OCTAVE)" under Grant No. P2024-01011.

References

- [1] Philip M. Morse and K. Uno Ingard. *Theoretical Acoustics*. Princeton University Press, 1968.
- [2] Heinrich Kuttruff. *Room Acoustics*. CRC Press, 6 edition, 2016.
- [3] Frank Ihlenburg. *Finite Element Analysis of Acoustic Scattering*, volume 132 of *Applied Mathematical Sciences*. Springer, New York, 1998. [doi:10.1007/b98828](https://doi.org/10.1007/b98828).

- [4] T. W. Wu, editor. *Boundary Element Acoustics: Fundamentals and Computer Codes*, volume 7 of *Advances in Boundary Elements*. WIT Press, 2000.
- [5] Doug L. James, Jernej Barbič, and Dinesh K. Pai. Precomputed acoustic transfer: Output-sensitive, accurate sound generation for geometrically complex vibration sources. *ACM Transactions on Graphics*, 25(3):987–995, 2006. doi:[10.1145/1141911.1141983](https://doi.org/10.1145/1141911.1141983).
- [6] Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 3:218–229, 2021. doi:[10.1038/s42256-021-00302-5](https://doi.org/10.1038/s42256-021-00302-5).
- [7] Nikola Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Learning maps between function spaces with applications to PDEs. *Journal of Machine Learning Research*, 24(89):1–97, 2023. URL: <http://jmlr.org/papers/v24/21-1524.html>.
- [8] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations*, 2021. URL: <https://arxiv.org/abs/2010.08895>.
- [9] Zongyi Li, Nikola Kovachki, Chris Choy, Boyi Li, Jean Kossaifi, Shourya Prakash Otta, Mohammad Amin Nabian, Maximilian Stadler, Christian Hundt, Kamyar Azizzadenesheli, and Anima Anandkumar. Geometry-informed neural operator for large-scale 3d PDEs. In *Advances in Neural Information Processing Systems*, volume 36, pages 35836–35854, 2023. URL: https://proceedings.neurips.cc/paper_files/paper/2023/file/70518ea42831f02afc3a2828993935ad-Paper-Conference.pdf.
- [10] Maziar Raissi, Paris Perdikaris, and George Em Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019. doi:[10.1016/j.jcp.2018.10.045](https://doi.org/10.1016/j.jcp.2018.10.045).
- [11] Zongyi Li, Hongkai Zheng, Nikola Kovachki, David Jin, Haoxuan Chen, Burigede Liu, Kamyar Azizzadenesheli, and Anima Anandkumar. Physics-informed neural operator for learning partial differential equations, 2024. [arXiv:2111.03794](https://arxiv.org/abs/2111.03794), doi:[10.1145/3648506](https://doi.org/10.1145/3648506).
- [12] Aditi Krishnapriyan, Amir Gholami, Shandian Zhe, Robert Kirby, and Michael W. Mahoney. Characterizing possible failure modes in physics-informed neural networks. In *Advances in Neural Information Processing Systems*, volume 34, pages 26548–26560, 2021. URL: https://proceedings.neurips.cc/paper_files/paper/2021/file/df438e5206f31600e6ae4af72f2725f1-Paper.pdf.
- [13] Steven L. Brunton, Bernd R. Noack, and Petros Koumoutsakos. Machine learning for fluid mechanics. *Annual Review of Fluid Mechanics*, 52:477–508, 2020. doi:[10.1146/annurev-fluid-010719-060214](https://doi.org/10.1146/annurev-fluid-010719-060214).

- [14] Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabás Póczos, Ruslan Salakhutdinov, and Alexander J. Smola. Deep sets. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/f22e4747da1aa27e363d86d40ff442fe-Paper.pdf.
- [15] Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. PointNet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 652–660, 2017. URL: https://openaccess.thecvf.com/content_cvpr_2017/html/Qi_PointNet_Deep_Learning_CVPR_2017_paper.html.
- [16] Juho Lee, Yoonho Lee, Jungtaek Kim, Adam R. Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 3744–3753. PMLR, 2019. URL: <https://proceedings.mlr.press/v97/lee19d.html>.
- [17] Matthew Lowery, John Turnage, Zachary Morrow, John D. Jakeman, Akil Narayan, Shandian Zhe, and Varun Shankar. Kernel neural operators for scalable, memory-efficient, geometrically flexible operator learning, 2024. URL: <https://arxiv.org/abs/2407.00809>, [arXiv:2407.00809](https://arxiv.org/abs/2407.00809).
- [18] Noam Koren, Ralf J. J. Mackenbach, Ruud J. G. van Sloun, Kira Radinsky, and Daniel Freedman. SVD-NO: Learning PDE solution operators with SVD integral kernels, 2026. [doi:10.1609/aaai.v40i1.37016](https://doi.org/10.1609/aaai.v40i1.37016).
- [19] Zhiwei Fang, Sifan Wang, and Paris Perdikaris. Learning only on boundaries: A physics-informed neural operator for solving parametric partial differential equations in complex geometries, 2024. [doi:10.1162/neco_a_01647](https://doi.org/10.1162/neco_a_01647).
- [20] Wenzhen Qu, Yan Gu, Shengdong Zhao, and Fajie Wang. Boundary integrated neural networks (BINNs) for acoustic radiation and scattering, 2023. URL: <https://arxiv.org/abs/2307.10521>, [arXiv:2307.10521](https://arxiv.org/abs/2307.10521).
- [21] Nikolas Borrel-Jensen, Allan P. Engsig-Karup, and Cheol-Ho Jeong. Physics-informed neural networks for one-dimensional sound field predictions with parameterized sources and impedance boundaries. *JASA Express Letters*, 1(12):122402, 2021. [doi:10.1121/10.0009057](https://doi.org/10.1121/10.0009057).
- [22] Nikolas Borrel-Jensen, Somdatta Goswami, Allan P. Engsig-Karup, George Em Karniadakis, and Cheol-Ho Jeong. Sound propagation in realistic interactive 3d scenes with parameterized sources using deep neural operators. *Proceedings of the National Academy of Sciences*, 121(2):e2312159120, 2024. [doi:10.1073/pnas.2312159120](https://doi.org/10.1073/pnas.2312159120).
- [23] Andrew Luo, Yilun Du, Michael J. Tarr, Joshua B. Tenenbaum, Antonio Torralba, and Chuang Gan. Learning neural acoustic fields. In *Advances in Neural Information Processing Systems*, 2022. URL: <https://arxiv.org/abs/2204.00628>.

- [24] Xutong Jin, Sheng Li, Guoping Wang, and Dinesh Manocha. Neuralsound: Learning-based modal sound synthesis with acoustic transfer. *ACM Transactions on Graphics*, 41(4):121:1–121:15, 2022. doi:[10.1145/3528223.3530184](https://doi.org/10.1145/3528223.3530184).
- [25] Fei Ma, Sipei Zhao, and Ian S. Burnett. Sound field reconstruction using a compact acoustics-informed neural network. *The Journal of the Acoustical Society of America*, 156(3):2009–2021, 2024. doi:[10.1121/10.0029022](https://doi.org/10.1121/10.0029022).
- [26] Xenofon Karakostas, Diego Caviedes-Nozal, Antoine Richard, and Efen Fernandez-Grande. Room impulse response reconstruction with physics-informed deep learning. *The Journal of the Acoustical Society of America*, 155(2):1048–1059, 2024. arXiv:https://pubs.aip.org/asa/jasa/article-pdf/155/2/1048/19508065/1048_1_10.0024750.pdf, doi:[10.1121/10.0024750](https://doi.org/10.1121/10.0024750).
- [27] Xutong Jin, Bo Pang, Chenxi Xu, Xinyun Hou, Guoping Wang, and Sheng Li. NAT: Neural acoustic transfer for interactive scenes in real time, 2025. doi:[10.1109/TVCG.2025.3617802](https://doi.org/10.1109/TVCG.2025.3617802).
- [28] Earl G. Williams. *Fourier Acoustics: Sound Radiation and Nearfield Acoustical Holography*. Academic Press, 1999.
- [29] Elwin van ’t Wout, Seyyed R. Haqshenas, Pierre G  lat, Timo Betcke, and Nader Saffari. Benchmarking preconditioned boundary integral formulations for acoustics. *International Journal for Numerical Methods in Engineering*, 2021. URL: <https://arxiv.org/abs/2104.04594>, doi:[10.1002/nme.6777](https://doi.org/10.1002/nme.6777).
- [30] Matthew Tancik, Pratul P. Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T. Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. In *Advances in Neural Information Processing Systems*, volume 33, pages 7537–7547, 2020. URL: https://proceedings.neurips.cc/paper_files/paper/2020/file/55053683268957697aa39fba6f231c68-Paper.pdf.
- [31] Chiheb Trabelsi, Olexa Bilaniuk, Ying Zhang, Dmitriy Serdyuk, Sandeep Subramanian, Jo  o Felipe Santos, Soroush Mehri, Negar Rostamzadeh, Yoshua Bengio, and Christopher J. Pal. Deep complex networks. In *International Conference on Learning Representations*, 2018. URL: <https://openreview.net/forum?id=H1T2hmZAb>.
- [32] Pierre Lallemand and Li-Shi Luo. Theory of the lattice boltzmann method: Dispersion, dissipation, isotropy, galilean invariance, and stability. *Physical Review E*, 61(6):6546–6562, 2000. doi:[10.1103/PhysRevE.61.6546](https://doi.org/10.1103/PhysRevE.61.6546).
- [33] Dominique d’Humi  res, Irina Ginzburg, Manfred Krafczyk, Pierre Lallemand, and Li-Shi Luo. Multiple-relaxation-time lattice boltzmann models in three dimensions. *Philosophical Transactions of the Royal Society A*, 360:437–451, 2002. doi:[10.1098/rsta.2001.0955](https://doi.org/10.1098/rsta.2001.0955).

- [34] Huidan Yu, Sharath S. Girimaji, and Li-Shi Luo. DNS and LES of decaying isotropic turbulence with and without frame rotation using lattice boltzmann method. *Journal of Computational Physics*, 209(2):599–616, 2005. doi:[10.1016/j.jcp.2005.03.022](https://doi.org/10.1016/j.jcp.2005.03.022).
- [35] Shyam S. Chikatamarla, Christos E. Frouzakis, Iliya V. Karlin, Ananias G. Tomboulides, and Konstantinos B. Boulouchos. Lattice boltzmann method for direct numerical simulation of turbulent flows. *Journal of Fluid Mechanics*, 656:298–308, 2010. doi:[10.1017/S0022112010001388](https://doi.org/10.1017/S0022112010001388).
- [36] Muhammad Idrees Khan, Sauro Succi, and Giacomo Falcucci. Validating the boltzmann approach to the large-eddy simulations of forced homogeneous incompressible turbulence. *International Journal of Modern Physics C*, page 2750064, 2026. doi:[10.1142/S0129183127500641](https://doi.org/10.1142/S0129183127500641).
- [37] Erlend Magnus Viggen. Viscously damped acoustic waves with the lattice boltzmann method. *Philosophical Transactions of the Royal Society A*, 369(1944):2246–2254, 2011. doi:[10.1098/rsta.2011.0040](https://doi.org/10.1098/rsta.2011.0040).
- [38] Thomas Astoul, Gauthier Wissocq, Jean-François Boussuge, Alois Sengissen, and Pierre Sagaut. Lattice boltzmann method for computational aeroacoustics on non-uniform meshes: A direct grid coupling approach. *Journal of Computational Physics*, 447:110667, 2021. doi:[10.1016/j.jcp.2021.110667](https://doi.org/10.1016/j.jcp.2021.110667).
- [39] Hui Xu and Pierre Sagaut. Optimal low-dispersion low-dissipation LBM schemes for computational aeroacoustics. *Journal of Computational Physics*, 230(13):5353–5382, 2011. doi:[10.1016/j.jcp.2011.03.040](https://doi.org/10.1016/j.jcp.2011.03.040).
- [40] Bradley Efron and Robert J. Tibshirani. *An Introduction to the Bootstrap*. CRC Press, 1994. doi:[10.1201/9780429246593](https://doi.org/10.1201/9780429246593).
- [41] Tobias Pfaff, Meire Fortunato, Alvaro Sanchez-Gonzalez, and Peter W. Battaglia. Learning mesh-based simulation with graph networks. In *International Conference on Learning Representations*, 2021. URL: <https://arxiv.org/abs/2010.03409>.