Journal of Sound and Vibration 643 (2026) 120018

Contents lists available at ScienceDirect

Journal of Sound and Vibration

journal homepage: www.elsevier.com/locate/jsvi

Decomposed boundary-based Fourier Neural Operator (DB-FNO) method for efficient parametric acoustic analysis with frequency-dependent boundary conditions

Ruoyan Li [a,b], Wenjing Ye [a,*], Yijun Liu [b]

[a] Department of Mechanical and Aerospace Engineering, The Hong Kong University of Science and Technology, Hong Kong SAR, PR China
[b] Department of Mechanics and Aerospace Engineering, Southern University of Science and Technology, Shenzhen, PR China

ARTICLE INFO

Keywords:
Surrogate model
Acoustic wave
Boundary element method
Fourier neural operators

ABSTRACT

Acoustic wave modeling across parametric domains—spanning evolving geometries, frequencies, and frequency-dependent boundary conditions—remains computationally intensive for traditional solvers. The Boundary Fourier Neural Operator (B-FNO) accelerates the boundary element method by learning the boundary-to-solution map, but its memory and efficiency scale prohibitively in three-dimensional configurations. In this paper, we introduce the Decomposed Boundary Fourier Neural Operator (DB-FNO), a tensor-compressed surrogate that retains the boundary-centric architecture while decomposing latent tensors into separable rank-1 tensors. This reduction converts two-dimensional Fourier transforms into successive one-dimensional operations, lowering complexity from $O(NlogN)$ to $O(\sqrt{N}\,logN)$, where N is the total number of boundary elements. The model handles frequency-dependent boundary conditions, enabling realistic engineering problems such as submarine scattering and phased-array transducers radiation to be solved in minutes on a single GPU. Comprehensive benchmarks covering scattering and radiation problems demonstrate comparable accuracy and significantly higher efficiency than B-FNO. DB-FNO thus opens a scalable data-driven pathway to acoustic simulation and design in unbounded domains.

1. Introduction

Acoustic waves are foundational to numerous modern technologies, ranging from medical imaging systems for tumor diagnosis to underwater sonar arrays mapping oceanic terrains [1,2]. In practical applications, parametric analysis—a systematic evaluation of system behavior under varying parameters—is often indispensable. For instance, in the design of acoustic metamaterials, iterative wave simulations are required to assess the performance of evolving structural designs during optimization processes. Concurrently, frequency sweep analyses are employed to characterize system responses across operational bandwidths, ensuring robustness under diverse spectral conditions. These requirements impose a severe computational load. The situation is further complicated when material properties and boundary conditions such as impedance exhibit dynamic dependencies on wave frequency.

Traditional simulation methods for acoustic waves such as finite element methods (FEM) and boundary element methods (BEM) discretize spatial domains or surfaces into meshes, solving the discretized Helmholtz equation using iterative linear solvers. Among

* Corresponding author.
E-mail address: mewye@ust.hk (W. Ye).

https://doi.org/10.1016/j.jsv.2026.120018
Received 24 September 2025; Received in revised form 30 May 2026; Accepted 19 July 2026
Available online 20 July 2026

them, the BEM is well regarded is an effective method for acoustic wave modeling particularly for problems with unbounded domains due to its reduced dimensionality. However, the fully populated system matrix in conventional BEM can still lead to computational challenges. To overcome the prohibitive computational overhead of the conventional BEM—which scales as $O(N^2)$ in memory storage and $O(N^3)$ in direct solution complexity —various accelerated BEM paradigms have been developed over the past three decades. The Fast Multipole Method (FMM) reduces the computational complexity of the boundary integral convolution to $O(NlogN)$ or $O(N)$ by recursively clustering elements and analytically expanding the Helmholtz fundamental solution into multipole and local expansions [3–5]. Concurrently, Hierarchical Matrix ($\mathscr{H}$-matrix) [6] methods combined with the Adaptive Cross Approximation (ACA) provide a purely algebraic, kernel-independent compression alternative [7]. By identifying geometrically well-separated element clusters, $\mathscr{H}$-matrices compress far-field interaction submatrices into low-rank representations using outer-product forms, bypassing the need for analytical series expansions. Despite these advancements, the reliance of FMM and ACA on iterative solvers introduces serious convergence vulnerabilities in highly complex or high-frequency domains. To address this iterative stagnation, recent work has shifted toward fast direct boundary element methods. Utilizing $\mathscr{H}$-matrices coupled with fast decomposition technique [8] and the Sherman-Morrison-Woodbury [9,10] inversion formula, these fast direct solvers factorize the system explicitly with quasi-linear complexity, establishing a highly robust framework for broadband analysis [11–14].

Although accelerated techniques like fast multipole BEM, ACA BEM, and pre-corrected fast Fourier transform BEM [15] have reduced the computational cost to $O(NlogN)$, where N is the total number of boundary elements/degrees of freedom (DOFs), conducting large-scale wave analysis across different problem settings using BEM remains resource-intensive. This is because these conventional approaches require repeated, costly mesh generation and linear system solving for each new problem setting, leading to prohibitive costs even with these state-of-the-art accelerators.

Various reduced order modeling techniques have been developed to reduce the computational cost of parametric wave analysis, mostly focusing on frequency variation. In [16–19], project-based model order reduction (MOR) methods were developed for conventional BEM systems, aiming to reduce the computational cost of multi-frequency acoustic wave analysis. A key step in this approach is to construct a series of frequency-independent matrices by expanding frequency-dependent BEM kernels, which is non-trivial especially for problems with complex boundary conditions [20]. To overcome this challenge, a matrix-free MOR method was proposed [20]. This method operates on the transfer functions between system inputs and outputs, circumventing complications arising from complex boundary conditions and enabling integration with existing acceleration techniques. Nevertheless, these conventional projection-based reduced order modelling methods were primarily developed for problems with fixed domains, extending them to solve design problems involving continuously changing domains is challenging.

Recent advancements in machine learning, particularly neural operators, present a compelling alternative for addressing computational challenges in parametric wave analysis [21–23]. Among these, Fourier Neural Operators (FNOs) have emerged as one of leading candidates by leveraging spectral transformations to learn mappings between function spaces, bypassing traditional iterative solvers [24]. However, a critical limitation persists in their implementation: FNOs require a volume integration operating on the entire domain, inherently restricting their applicability to problems involving unbounded geometries. This constraint undermines both computational efficiency and model generalizability, particularly for wave-structure interaction problems where boundary-driven phenomena dominate. To resolve this issue, the Boundary Fourier Neural Operator (B-FNO) was developed, integrating principles from acoustic BEM with FNO architectures [25]. Unlike traditional FNOs that convolve kernels over volumetric domains, the B-FNO restricts operations to boundaries, aligning with the natural formulation of acoustic scattering and radiation problems. By establishing direct mappings between boundary geometries, physical parameters, and boundary integral equation (BIE) solutions, the B-FNO retains the accuracy of BEM while achieving orders-of-magnitude acceleration for parametric analyses involving domain or frequency variations. In [25], a comparison was made between B-FNO and FMM-accelerated BEM. The results demonstrated that B-FNO exhibits more stable behavior and requires substantially fewer iterations particularly in problems with high condition numbers. Consequently, the online inference cost of B-FNO remains entirely invariant to matrix ill-conditioning, frequency sweeps, or geometric complexity.

Despite these advancements, the initial B-FNO framework was validated only for frequency-independent boundary conditions (BCs). Furthermore, while the method has demonstrated high efficiency compared to conventional simulation methods, the scalability of the model for large-scale 3D problems is limited due to rapid memory increase and the computational cost of Fourier transform, which scales as $O(NlogN)$ where N represents DOFs. To overcome these challenges, this work introduces the Decomposed B-FNO (DB-FNO), a novel framework that achieves superior scalability through the strategic application of low-rank tensor decomposition. Although tensor decomposition is a recognized model reduction technique recently applied to improve the efficiency of FNO for finite domains [26,27], its integration within the boundary-centric paradigm of B-FNO for unbounded problems has not been previously explored. By decomposing high-dimensional latent states into separable rank-1 tensors, DB-FNO reduces the computational complexity of 2D Fourier transforms to a series of efficient 1D operations while simultaneously reducing the model's parameter count. This dual benefit dramatically lowers memory consumption and accelerates Fourier computations.

Crucially, this new framework not only retains the boundary-centric formulation of its predecessor but also extends its applicability to frequency-dependent BCs. Among all frequency-dependent BCs, one key type of BC is the impedance boundary conditions (IBCs). IBCs represent more realistic boundaries than idealized boundaries, which partially absorb and reflect incident waves. Despite their importance, IBCs present significant challenges for surrogate models in acoustic simulations. Most existing MOR and data-driven approaches for acoustic problems with IBCs still treat the impedance as frequency-independent [28,29]. Although the work presented in [20] and [30] incorporate frequency-dependent IBCs, their projection-based methods or neural-network frameworks cannot accommodate changes in the computational domain. In contrast, the DB-FNO not only handles evolving geometries, but also maintains high accuracy even under frequency-dependent IBCs, such as those encountered in submarine stealth coatings designed for sonar evasion [31,32].

The remainder of this article is organized as follows. Section 2 begins by deriving the boundary integral formulation of the Helmholtz equation, contextualizing its synergy with Fourier-based neural operators. The architecture of the DB-FNO is then introduced including its boundary-adaptive Fourier layers, implicit frequency encoding, and tensor decomposition strategies. Section 3 benchmarks the proposed DB-FNO framework against the B-FNO, utilizing conventional BEM solutions as the ground-truth dataset, demonstrating superior accuracy and efficiency across diverse acoustic scenarios, including frequency-dependent impedance models. Section 4 extends the methodology to real-world challenges, such as submarine scattering and phased array transducers, highlighting its adaptability in multi-physics environments. Finally, Section 5 concludes with a discussion of broader implications, current limitations, and future directions for operator learning in wave-dominated systems.

2. Methods

2.1. Boundary integral formulation of acoustic wave equation

The type of problem we focus on this work is acoustic wave analysis; however, the method proposed is also applicable to elastic and electromagnetic wave analysis. The linear acoustic equation in the frequency domain is governed by the Helmholtz equation:

$$\nabla^2\phi + k^2\phi + Q\delta(\mathbf{x}, \mathbf{x}_Q) = 0, \ \forall \mathbf{x} \in D \tag{1}$$

where ϕ is the complex acoustic pressure at location $\mathbf{x}$, k is the wavenumber, and $Q\delta(\mathbf{x}, \mathbf{x}_Q)$ represents a point sound source at point $\mathbf{x}_Q$ inside domain D, with amplitude of Q.

There are three types of boundary conditions typically prescribed on the boundary S of domain D: (1) sound pressure is given by $\phi = \overline{\phi}$; (2) particle velocity is given by $q \equiv \frac{\partial\phi}{\partial n} = \overline{q} = i\omega\rho v_n$; and (3) the impedance is given by $\phi = Zv_n$, where the overbar symbol denotes the given value, ρ is the density of acoustic medium, v_n is the normal velocity of the acoustic medium, and Z is the specific acoustic impedance. For exterior problems, besides abiding the boundary conditions, the field at infinity should comply with the subsequent Sommerfeld radiation condition:

$$\lim_{R\to\infty}\left[R\left|\frac{\partial\phi}{\partial R} - ik\phi\right|\right] = 0,$$

where R denotes the radius of an encompassing sphere around the structure and ϕ designates the radiated wave in a radiation issue or the scattered wave during a scattering predicament.

Applying the Green's second identity and the property of Dirac function, Eq. (1) can be formulated into a boundary integral equation, termed as the conventional boundary integral equation (CBIE):

$$c(\boldsymbol{x})\phi(\boldsymbol{x}) = \int_S [G(\boldsymbol{x},\boldsymbol{y},\omega)q(\boldsymbol{y}) - F(\boldsymbol{x},\boldsymbol{y},\omega)\phi(\boldsymbol{y})]dS(\boldsymbol{y}) + \phi^I(\boldsymbol{x}) + QG(\boldsymbol{x},\boldsymbol{x}_Q,\omega) \tag{2}$$

$$G(\boldsymbol{x},\boldsymbol{y},\omega) = \frac{1}{4\pi r}e^{ikr},$$

$$F(\boldsymbol{x},\boldsymbol{y},\omega) = \frac{1}{4\pi r^2}(ikr - 1)r_{,j}n_j(y)e^{ikr},$$

where $\phi^I(\boldsymbol{x})$ represents the incident wave, which is absent in radiation problems, $\boldsymbol{r}$ is the distance between a source point $\boldsymbol{x}$ to a field point $\boldsymbol{y}$. Kernel G, also recognized as the 'fundamental solution' or 'Green's Function', describes the response at a certain field point due to a solitary source at a source point. Kernel F is its normal derivative. The CBIE essentially performs a convolution of kernels G and F along the boundary. $c(\boldsymbol{x})$ is a coefficient with its value depends on the location of $\boldsymbol{x}$:

$$c(\boldsymbol{x}) = \begin{cases} 1, \forall \boldsymbol{x} \in D; \\ \frac{1}{2}, \forall \boldsymbol{x} \in S \ (\text{smooth}); \end{cases}$$

With the given boundary conditions, Eq. (2) with $c(\boldsymbol{x}) = 1/2$ is first solved for unknown boundary quantities. The pressure inside the domain is then found by computing the integral in Eq. (2) with $c(\boldsymbol{x}) = 1$ and the known boundary values.

One critical issue of CBIE for exterior acoustic problems is the non-uniqueness at fictitious frequencies, which are related to the eigenfrequencies of the corresponding interior acoustic problems. A dual formulation consisting of the CBIE and its normal derivative was proposed by Burton and Miller to resolve this issue [32]. Taking the derivate of the CBIE with respected to the normal direction at the source point $\boldsymbol{x}$, hypersingular BIE (HBIE) is obtained. Then, the Burton-Miller (B-M) formulation can be written symbolically as:

$$\text{CBIE} + \beta\ \text{HBIE} = 0,$$

where β is a coupling coefficient. In this paper, $\beta = -ih$ is used, with h being the typical element size in the BEM mesh.

Eq. (2) is numerically solved through discretization of the boundary S of domain D, yielding a linear system of equations expressed as $\mathbf{A}\boldsymbol{q} = \boldsymbol{b}$. Here, $\boldsymbol{q}$ represents the discretized solution vector, $\mathbf{A}$ denotes the system matrix whose entries are computed by evaluating kernel integrals (e.g., Green's function and its derivatives) over boundary elements, and $\boldsymbol{b}$ is the known vector derived from boundary condition enforcement and source term integration. The system can be solved via direct methods, which provide exact solutions at a computational cost with the complexity of $O(N^3)$, or iterative methods such as the Generalized Minimal Residual Method (GMRES) [33], a Krylov subspace technique particularly suited for sparse, non-symmetric systems. While direct methods guarantee stability for well-posed problems, iterative approaches are preferred for large-scale simulations due to their reduced memory and computational time.

2.2. Basic framework of the B-FNO

FNO is a neural operator that maps a parameter space to the corresponding solution space in a finite domain. The key idea of the FNO is to add a non-local integral operation to the usual network updating procedure in each hidden layer, shown as:

$$v_{t+1}(\boldsymbol{x}) := \sigma(Wv_t(\boldsymbol{x}) + (\mathscr{K}(a;\theta)v_t)(\boldsymbol{x})),\ \forall x \in D$$

where v_t is the high-dimensional representation of the solution at the t-th hidden layer, W and σ stand for a linear transformation and non-linear activation function, a represents the problem parameters, D is the latent domain, θ denotes the network parameters and $(\mathscr{K}(a;\theta)v_t)(\boldsymbol{x})$ is the non-local integral defined as $(K(a;\theta)v_t)(x) = \int_D \kappa_\theta(x-y,a)v_t(y)dy$. The convolutional nature of the integral allows it to be computed via fast Fourier Transform (FFT):

$$(K(a;\theta)v_t)(x) = \mathscr{F}^{-1}(\mathscr{F}(\kappa_\theta)\cdot\mathscr{F}(v_t))(x) = \mathscr{F}^{-1}(R_\theta\cdot(\mathscr{F}v_t))(x) \tag{3}$$

where $\mathscr{F}$ denotes the Fourier transform of a function and $\mathscr{F}^{-1}$ is its inverse. R_θ is a tensor containing the Fourier coefficients of the kernel function κ_θ, which is determined through learning via network updates. During the computation, high-order modes are filtered out, resulting in model speed-up. Additionally, this filtration aids in removing high-frequency noises present in the feature information.

In [25], the B-FNO was proposed to solve a range of wave scattering problems characterized by different parameters. The parameter space considered encompasses both the varying structures and incident wave frequencies. Fig. 1 shows the architecture of the B-FNO network model for 3D acoustic problems. The inputs of the network are vectors containing the Cartesian coordinates of boundary points $\boldsymbol{x} \in S$ used to define the problem geometry. The frequency input is the incident wave pressure at each boundary point $\phi^I(\boldsymbol{x}) = e^{i\boldsymbol{k}\cdot\boldsymbol{x}}$, where $\boldsymbol{k}$ is the angular wave vector. The output of the network is the solution for boundary unknowns. For example, the output in wave scattering problems is the sound pressure on the domain boundary $\phi(\boldsymbol{x})$, $\boldsymbol{x} \in S$. After the boundary unknowns are output by the B-FNO, Eq. (2) is employed to calculate the wave field at any point inside the domain, that is $\phi(\boldsymbol{x})$, $\boldsymbol{x} \in D$.

First, the input data—comprising typically unevenly distributed boundary points and their incident pressure values—is passed through a fully connected neural network (FC1). In this work, the total number of boundary points is selected as $N = N_1 \times N_2$. As depicted in Fig. 1, this operation yields a high-dimensional latent representation $v_0(\boldsymbol{x}) \in \mathbb{R}^{d_v\times N_1\times N_2}$, defined on a uniformly distributed grid $N_1 \times N_2$ and with a channel size of d_v. The high-dimensional latent representation $v_0(\boldsymbol{x})$ is then update iteratively according to Eq. (3). Specifically, the FFT is first employed to compute the Fourier coefficients of $v_t(\boldsymbol{x})$ along the two grid dimensions. Subsequently, the high-order modes are filtered out, and the remaining coefficients are multiplied with the kernel coefficient tensor R_θ. Inverse FFT is then performed on the product to obtain the updated $v_t(\boldsymbol{x})$, denoted as $v_{t+1}(\boldsymbol{x})$. Finally, after the final Fourier layer, the high-dimensional representation is projected to the boundary solution $\phi(\boldsymbol{x})$, $\boldsymbol{x} \in S$ through another fully connected neural network, FC2.

The non-local integral operator $(\mathscr{K}(a;\theta)v_t)(\boldsymbol{x})$ in the B-FNO inherits the framework of the original FNO but restricts its operations

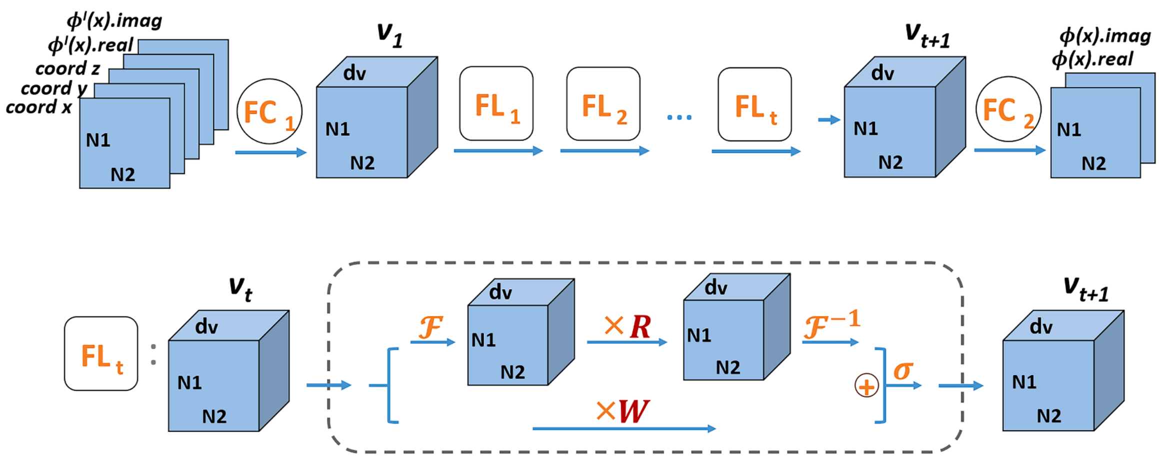

Fig. 1. Schematic of the B-FNO architecture.

to boundary geometries. The B-FNO employs a data-driven approach to learn parameterized kernel functions directly from training data, contrasting with BIEs that rely on analytically derived kernels, which are Helmholtz fundamental solutions. It circumvents the need for explicit singular/hypersingular integral regularization while demonstrating superior convergence behavior.

2.3. Basic framework of DB-FNO

The idea of applying tensor decomposition is based on the separable property of FFT. Let us take the discrete Fourier transformation (DFT) as example. Consider a two-dimensional discrete signal with dimensions of $N_1 \times N_2$. If we apply canonical polyadic decomposition (CPD) on this signal, that is, $x[m,n] = \sum_{p=1}^{P} u_p[m] w_p[n]$, the discretized FFT of x can then be transferred as a product of 1D FFT as shown below:

$$X[k,l] = \sum_{m=0}^{N_1-1}\sum_{n=0}^{N_2-1} x[m,n] e^{-j2\pi\left(\frac{km}{M}+\frac{ln}{N}\right)} = \sum_{m=0}^{N_1-1}\sum_{n=0}^{N_2-1}\sum_{p=1}^{P} u_p[m] w_p[n] e^{-j2\pi\left(\frac{km}{N_1}+\frac{ln}{N_2}\right)} = \sum_{p=1}^{P}\sum_{n=0}^{N_2-1}\left(\sum_{m=0}^{N_1-1} u_p[m] e^{-j2\pi\frac{km}{N_1}}\right) w_p[n] e^{-j2\pi\frac{ln}{N_2}}$$

$$= \sum_{p=1}^{P} U_p[k]\left(\sum_{n=0}^{N_2-1} w_p[n] e^{-j2\pi\frac{ln}{N_2}}\right) = \sum_{p=1}^{P} U_p[k] W_p[l],$$

where $k \in [0, N_1 - 1]$, $l \in [0, N_2 - 1]$. The resulting time complexity becomes $O(PN_1 logN_1) + O(PN_2 logN_2)$. Let the total number of boundary points be $N = N_1 \times N_2$. Assuming an approximately square mapping where $N_1 \approx N_2 \approx \sqrt{N}$, the DB-FNO time complexity scales as $O\left(P\sqrt{N}\, logN\right)$. This is significantly lower than the original B-FNO complexity of $O(NlogN)$ when $P \ll \sqrt{N}$. The feasibility of this approach relies on the inherent low-rank nature of many engineering problems, a property that has been successfully exploited by the CPD in numerous applications to achieve such computational savings. Hence, in the DB-FNO, the latent tensor v_t is represented by a product of P rank-1 tensors as $v_t = \sum_{p=1}^{P} v_{t,p,1} \cdot v_{t,p,2}{}^{T}$.

Similarly, the decomposition can also be established on inverse Fourier transform (IDFT). To achieve this, besides the decomposition of the latent tensor, we also separate the learnable kernel R_θ in the B-FNO into $R_1 \cdot R_2$ in the DB-FNO. The domain integral Eq. (3) is converted into:

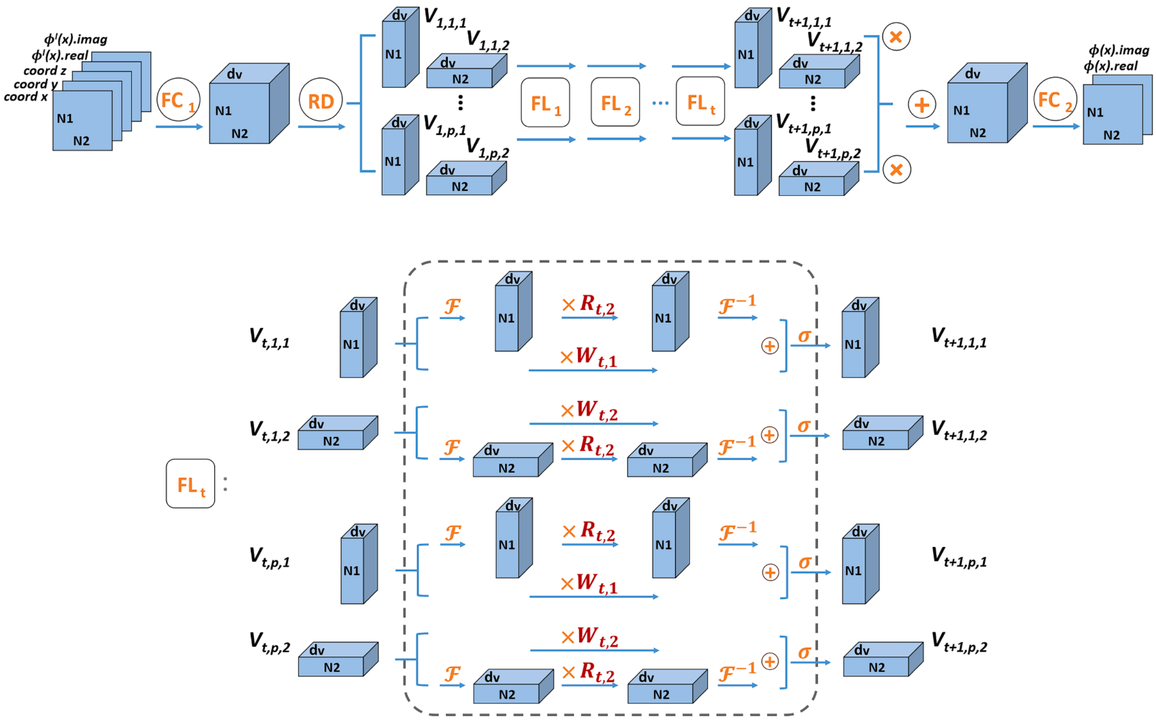

Fig. 2. Schematic of DB-FNO architecture.

$$
\begin{aligned}
(K(a;\theta)v_t)(\boldsymbol{x}) &= \mathscr{F}^{-1}\left(R_1 \cdot R_2 \cdot \left(\sum_{p=1}^{P} (\mathscr{F}u_p)\cdot(\mathscr{F}w_p)\right)\right)(\boldsymbol{x}) \\
&= \mathscr{F}^{-1}\left(\sum_{p=1}^{P} (R_1 \cdot \mathscr{F}u_p)\cdot(R_2 \cdot \mathscr{F}w_p)\right)(\boldsymbol{x}) \\
&= \left(\sum_{p=1}^{P} \mathscr{F}^{-1}(R_1 \cdot \mathscr{F}u_p)\cdot \mathscr{F}^{-1}(R_2 \cdot \mathscr{F}w_p)\right)(\boldsymbol{x}),
\end{aligned}
$$

where $\mathscr{F}$ denotes the Fourier transform of a function and $\mathscr{F}^{-1}$ is its inverse. R_1 and R_2 are tensors containing the Fourier coefficients of the kernel function κ, which is determined through learning via network updates. Therefore, the iterative update of each 1D vector in the DB-FNO is defined as Eq. (4):

$$
v_{t+1,p,d} = \sigma\left(W_{t,d}v_{t,p,d} + (\mathscr{K}(a;\theta)v_{t,p,d})\right) = \sigma\left(W_{t,d}v_{t,p,d} + \mathscr{F}^{-1}\left(R_{t,d}\mathscr{F}\left(v_{t,p,d}\right)\right)\right), \tag{4}
$$

where $p \in [1, P]$ and $d = 1,\ 2$. The $W_{t,d}$ and σ stand for a linear transformation and non-linear activation function, a represents the problem parameters, θ denotes the network parameters. Notably, both $W_{t,d}$ and $R_{t,d}$ are independent with the term p, which means the different terms shared the same weight tensor and Fourier coefficient tensor. This key feature greatly reduces the number of model parameters and thus improves the efficiency. The decomposed Fourier layer is illustrated in lower parts in Fig. 2.

The input 2D function is first lifted to a higher dimension representation v_0 with d_v channels by a fully connected layer $FC1$. Then a convolutional layer RD is applied to obtain the P terms rank-1 tensors $v_{1,p,d}(\boldsymbol{x})$ where $p = [1,\ldots,P]$ and $d = [1,\ 2]$. This dimensional reduction layer utilizes two parallel convolutional operations with kernel sizes of $1 \times N_2$ and $N_1 \times 1$ to extract the vertical and horizontal features from the high-dimensional latent representation simultaneously. These rank-1 tensors are iteratively updated by going through a series of Fourier layers $v_{t,p,d}(\boldsymbol{x})$ to $v_{t+1,p,d}(\boldsymbol{x})$ according to Eq. (4). After the final t-th Fourier layer, the output $u(\boldsymbol{x})$ is the obtained via a linear projection layer $FC2$, which is a fully connected layer:

$$
u(\boldsymbol{x}) = FC_2\left(\sum_{p=1}^{P} v_{t+1,p,1}(\boldsymbol{x}_1) \times v_{t+1,p,2}(\boldsymbol{x}_2)\right),
$$

The overall process is similar to the original FNO, but reduces the time complexity from the $O(NlogN)$ to the most $O(PN)$. The B-FNO and DB-FNO are compared layer by layer as in Table 1.

3. Benchmark study

This section evaluates the accuracy, computational efficiency, and parametric sensitivity of the proposed method through four numerical case studies. Two cases focus on scattering analyses involving Helmholtz resonators (HR), while the remaining two address radiation problems with geometrically varying objects. The ground-truth dataset is generated using a conventional BEM solver.

It is important to note that BEM solutions are numerical approximations and therefore contain inherent discretization and integration errors. In this study, the surfaces are discretized into constant triangular elements. To ensure the fidelity of the ground-truth data, the mesh density for all cases was strictly maintained at a minimum of 6 to 10 elements per acoustic wavelength. With this discretization standard, the BEM solutions achieve the numerical convergence, serving as a reliable high-fidelity baseline for training and validating the neural operator.

The primary metric used to evaluate the performance of the DB-FNO model is the relative root mean square error (rRMSE). The rRMSE quantifies the additional approximation error introduced by the neural network prediction ϕ_{pred} compared to the BEM ground truth ϕ_{BEM}, defined mathematically as:

$$
\mathrm{rRMSE} = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left|\phi_{pred}(\boldsymbol{x}_i) - \phi_{BEM}(\boldsymbol{x}_i)\right|^2}}{\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left|\phi_{BEM}(\boldsymbol{x}_i)\right|^2}} \times 100\%
$$

Table 1
The time complexity of different steps in B-FNO and DB-FNO.

	B-FNO	DB-FNO
Lifting layer	$O(N)$	$O(N)$
Decomposing layer		$O(PN)$
Linear transform	$O(N)$	$O(\sqrt{N})$
FFT	$O(NlogN)$	$O(P\sqrt{N}\,logN)$
Kernel Multiplication	$O(N)$	$O(PN)$
Inverse FFT	$O(NlogN)$	$O(P\sqrt{N}\,logN)$
Projection layer	$O(N)$	$O(PN)$

where N is the total number of boundary elements. By normalizing the L2 error against the ground truth magnitude, the rRMSE provides a relative and scale-invariant metric. This choice ensures a fair and equitable performance evaluation across diverse operating frequencies and varying excitation amplitudes.

To ensure full interpretability of the model's performance, the predictive accuracy throughout the subsequent benchmark studies is reported in the format of $\mu \pm \epsilon$. In this specific notation, the baseline value μ represents the mean rRMSE across all test samples, while the combined sum $\mu + \epsilon$ dictates the absolute maximum error observed within the entire test dataset. As will be demonstrated in the following subsections, the DB-FNO's approximation error is at least an order of magnitude smaller than the underlying BEM discretization error, the overall system error is still fundamentally dictated by the BEM mesh density.

3.1. Case 1: Plane wave scattering by a 2 × 2 HR matrix

An 2 × 2 array of Helmholtz resonators with varying dimensions, detailed in Table 2 and illustrated in Fig. 3 (c), is subjected to a 500 Hz plane wave propagating along the negative z-axis. The rectangular structure spans 1.5 meters in length, 1.2 meters in width, and 0.6 meters in height. Its surface is discretized into 4,880 triangular constant elements. Fig. 4 illustrates the surface mesh and the boundary acoustic pressure for a sample structure, obtained from the BEM simulation.

The input to the B-FNO and DB-FNO model consists of five arrays: three 61 × 48 arrays storing the Cartesian coordinates for the 4,880 triangular element centroids, and two additional arrays storing the incident wave pressure at each centroid. The ground-truth dataset generated from the BEM consists of 6,561 training samples and 5,000 testing samples. The training dataset uniformly samples the parameter space of geometric variations, and the testing samples are randomly distributed.

3.2. Case 2: Point source scattering by a linear HR array

The linear array of four identical HR shown in Fig. 3(b) is irradiated by a 400 Hz point source located at a point with coordinates of (0.95, 2.8, 2). The dimensions of each resonator neck, along with the global geometry of the structure measuring 5.6 meters by 1.9 meters by 1.45 meters, are parameterized in Table 2 and visualized in Fig. 3 (b). Fig. 5 illustrates the surface mesh and the boundary acoustic pressure for a sample structure, obtained from the BEM simulation.

The input to the B-FNO and DB-FNO model consists of five arrays: three 144 × 114 arrays storing the incident wave pressure at each centroid. The ground-truth dataset generated from the BEM. We conducted two sets of experiments: one with 3,822 training samples and 300 test samples, and another with 3,080 training samples and 1000 test samples. The training dataset uniformly samples the parameter space of geometric variations, and the testing samples are randomly distributed.

3.3. Case 3: Pulsating sphere matrix with frequency variation

This example involves pulsating spheres arranged into a 10 by 10 matrix. In each instance, one sphere is missing, resulting in 99 spheres per configuration and a total of 100 different configurations. Simultaneously, the radiation frequency varies between 200 Hz and 400 Hz. The boundary condition for each sphere is set to a sound speed of -0.1 m/s. Each sphere is discretized into 432 triangular elements, resulting a total of 42,768 elements per configuration. Fig. 6 illustrates part of the surface mesh and the total boundary acoustic pressure for a sample structure, obtained from the BEM simulation.

The input to the B-FNO and DB-FNO model consists of five arrays: three 198 × 216 arrays storing the Cartesian coordinates for the 42,768 triangular element centroids, and two additional arrays storing the wave pressure axis at each centroid, which is assumed to incident along -z axis. The ground-truth dataset generated from the BEM. We conducted three sets of experiments: one with 3,000 training samples and 1,500 test samples, one with 6,000 training samples and 3,000 test samples, and another with 10,000 training samples and 5,000 test samples. The training dataset uniformly covers the frequency range, but stochastically samples from geometric variation space. The testing samples are randomly distributed.

3.4. Case 4: Speaker array with hemispherical actuation

This example investigates a 3 by 8 speaker matrix, where in each instance, four speakers are missing, resulting in 20 speakers per geometry and a total of 10,626 different geometries. The geometry and surface mesh of each speaker are illustrated in Fig. 7, with only the hemispherical part of the geometry vibrating. The boundary condition for the hemispherical part is set to a sound speed of 0.001 m/s, while the remaining parts are treated as hard walls. And the operating frequency is at 10000Hz. The surface of the whole structures is discretized into 78,680 triangular elements.

The input to the B-FNO and DB-FNO model consists of five arrays: three 281 × 280 arrays storing the Cartesian coordinates for the

Table 2
The dimensions and varying window sizes of Helmholtz Resonators.

HRs	w2 (cm)	d2 (cm)	h2 (cm)	w1 (cm)	d1 (cm)	h1 (cm)
2 × 2 varied HRs	25∼45	15∼30	20	60	45	32.5
4 × 1 identical HRs	50∼130	30∼80	50	150	100	75

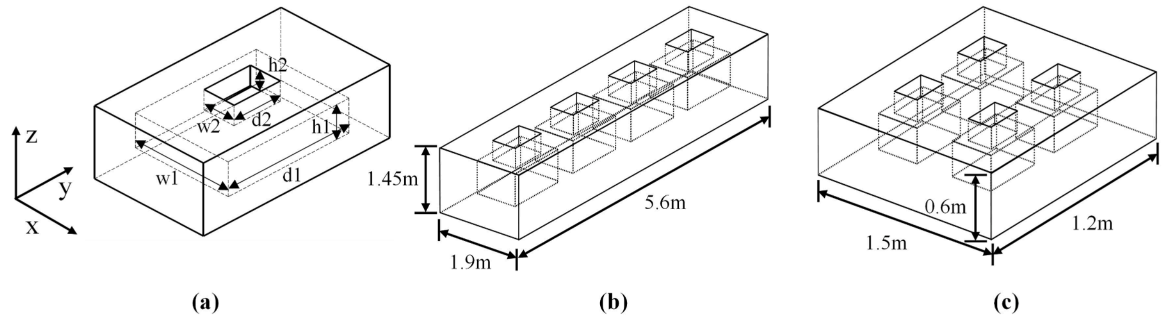

Fig. 3. (a) Schematic of a typical Helmholtz resonator structure and its geometric parameters; (b) a linearly arranged identical Helmholtz resonators; (c) four Helmholtz resonators with varying dimensions arranged in a 2 × 2 matrix.

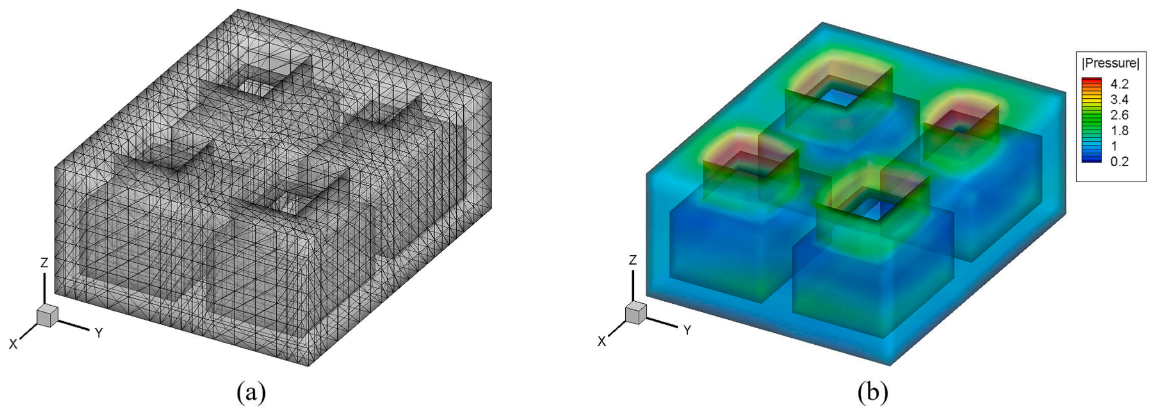

Fig. 4. (a) Schematic illustration of the discretized 2 by 2 varied HRs matrix. (b) The acoustic pressure on the boundary in this case calculated by the BEM at 500Hz.

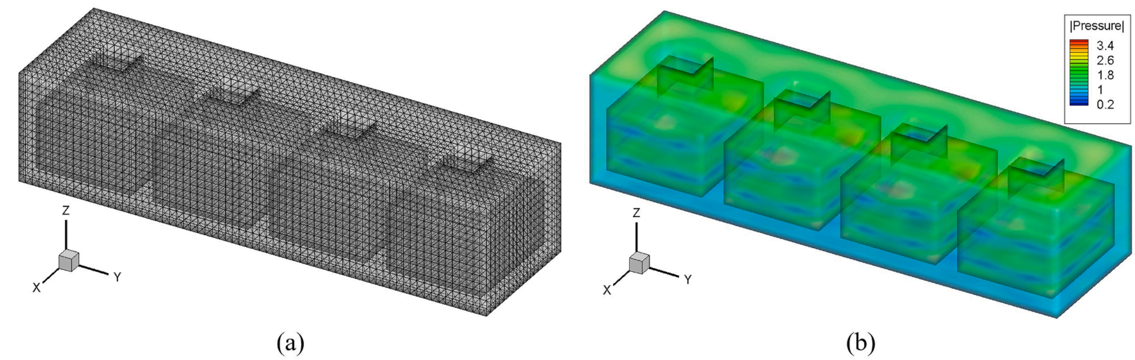

Fig. 5. (a) Schematic illustration of the discretized 4 by 1 identical HRs matrix. (b) The acoustic pressure on the boundary in this case calculated by the BEM at 400Hz.

78,680 triangular element centroids, and two additional arrays storing the wave pressure axis at each centroid, which is assumed to incident along -z axis. The ground-truth dataset generated from the BEM. We conducted two sets of experiments: one with 3,000 training samples and 3,000 test samples, and another with 5,000 training samples and 5,000 test samples. The training dataset stochastically samples from geometric variation space. The testing samples are randomly distributed.

3.5. Accuracy in relation to decomposed terms and layers

Tensor decomposition theory posits that increasing the number of decomposition terms generally improves approximation accuracy by capturing higher-dimensional features. Consequently, in the DB-FNO framework, the relative root mean square error (rRMSE)

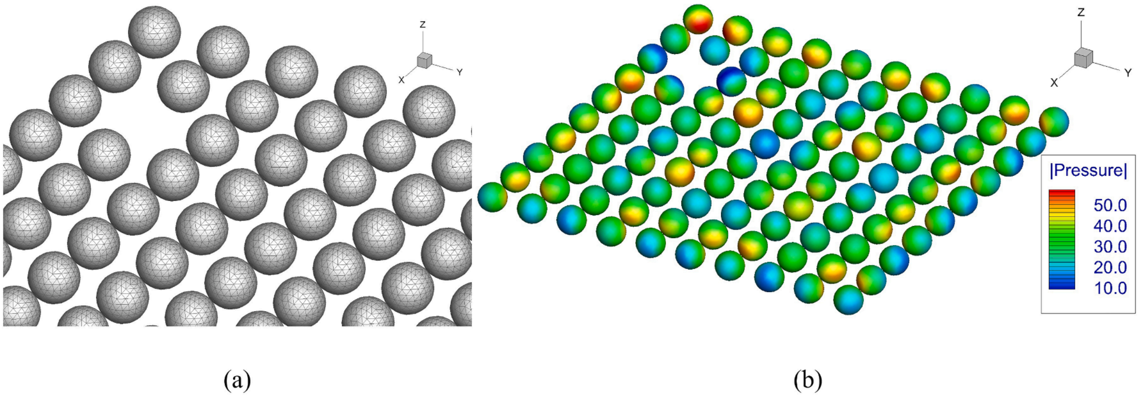

Fig. 6. (a) Schematic illustration of a part of the discretized sphere matrix. The surface mesh of each sphere is identical in the entire matrix. (b) The acoustic pressure on the boundary of each sphere calculated by the BEM in the case where the sphere in the second row the third column is missing, and the rest of spheres are pulsating at 200Hz.

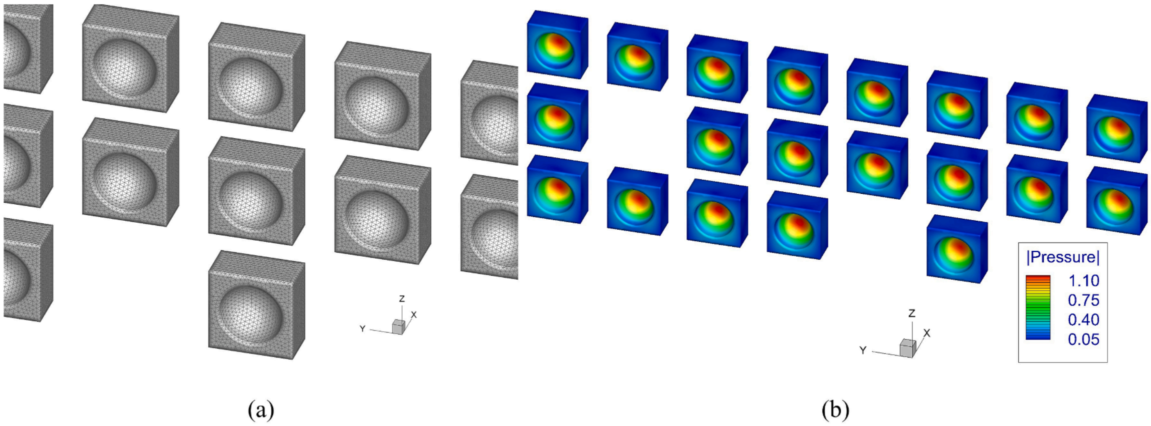

Fig. 7. (a) Schematic illustration of a part of the discretized speaker matrix. The surface mesh of each speaker in identical in the entire matrix. (b) Simulated acoustic pressure on the boundary of each sphere obtained from the BEM for the case where four speakers are missing, and the rest spheres are pulsating at 10000Hz.

of the test set is anticipated to decrease monotonically with additional terms. Fig. 8 (a) confirms this trend, despite slight fluctuations in some cases, and demonstrates a progressive error reduction as the number of terms grows. However, the results reveal case-specific optimality: while the speaker matrix exhibits continued error reduction up to $p = 30$, other cases achieve minimal error at $p = 10$ decomposition terms with variations within 0.01%. This suggests that $p = 10$ is sufficient for these scenarios.

A parallel analysis of Fourier layers reveals analogous behavior, though with amplified sensitivity. As shown in Fig. 8 (b), accuracy gains are most pronounced when increasing from 2 to 4 layers, particularly for the 4×1 Helmholtz resonator case, where error decreases greatly. Across other cases, 4 Fourier layers consistently suffice to achieve high accuracy. Given the consideration of time and memory costs associated with increasing the number of Fourier layers, and similar to B-FNO, 4 Fourier layers are adopted universally in subsequent studies.

3.6. Accuracy compared to B-FNO

Table 3 compares the accuracy of the original and DB-FNO frameworks under identical training conditions, including same sample sizes and Fourier layer configurations. Entries left blank in the "Decomposed Terms" column indicate the use of the original Fourier Neural Operator. The results of four examples are presented with the listed decomposed terms corresponding to configurations achieving optimal accuracy across hyperparameter trials. As previously noted, variations in the number of decomposition terms yield minimal accuracy fluctuations.

For the 2×2 HR and speaker matrix cases, the DB-FNO exhibits negligible accuracy degradation compared to the original model. In contrast, the 4×1 HR case shows a more pronounced performance gap, with the decomposed variant underperforming by 0.016% in rRMSE. Nevertheless, the accuracy of the D-BFNO in this case is accepted. The sphere matrix case stands out as exceptional: across both 6,000 and 10,000 training samples, the DB-FNO performs slightly better than the original model, achieving a 0.007% to 0.010%

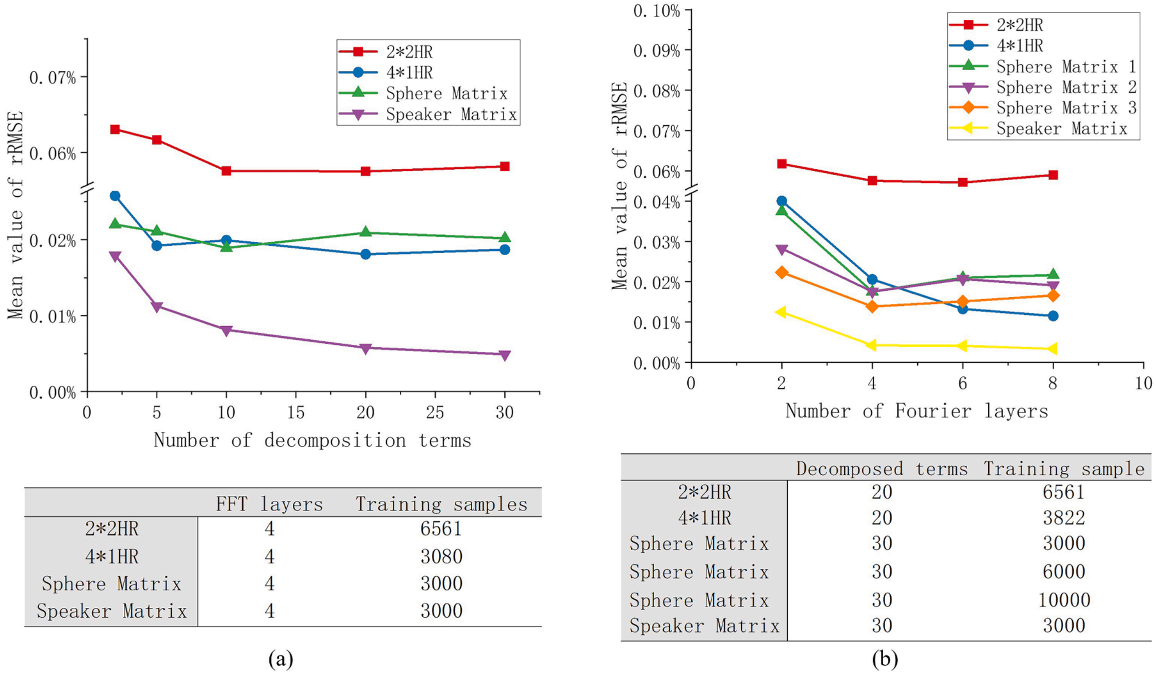

Fig. 8. (a) The mean value of rRMSE among all test samples with increasing number of decomposition terms, 4 cases are plotted in the figure. (b) The mean value of rRMSE among all test samples with increasing number of Fourier layers, 4 cases are plotted in the figure, and the sphere matrix case with different size of training samples are presented.

Table 3
The rRMSE of the 4 examples with different number of training samples, different number of Fourier layers and different number of decomposed terms.

Numerical examples	Training samples	Fourier layers	Decomposed terms	rRMSE
2*2HR	6561	4		0.052%±0.029%
			20	0.058%±0.026%
4*1HR	3822	4		0.005%±0.005%
			20	0.021%±0.029%
Sphere Matrix	6000	4		0.024%±0.017%
			30	0.018%±0.013%
	10000			0.017%±0.013%
			30	0.014%±0.010%
Speaker Matrix	3000	4		0.003%±0.001%
			20	0.006%±0.001%
	5000			0.002%±0.001%
			30	0.004%±0.001%

improvement in accuracy. Overall, these results show that the accuracy of the DB-FNO is comparable to that of B-FNO.

3.7. Efficiency compared to B-FNO

The inference time reported in this section is averaged over at least 1,000 independent runs to ensure statistical reliability. Fig. 9 (a) compares the efficiency of the two methods for the 2×2 and 4×1 HR cases, both configured with 20 decomposition terms. Fig. 9 (b) presents the results of the sphere matrix and speaker matrix cases, each employing 30 decomposition terms. Across all scenarios, the training dataset size remains constant. The computational advantages of DB-FNO scale markedly with increasing input size and Fourier layer.

For instance, in the 2×2 HR case with the smallest input size of 61×80 and 2 Fourier layers, the B-FNO requires 2.08 ms per inference, whereas the DB-FNO achieves a 16% reduction at 1.74 ms. This efficiency gap amplifies dramatically for large-scale problems: in the speaker matrix case with the largest input size of 280×281 and 8 Fourier layers, the B-FNO consumes 83.58 ms, while the DB-FNO completes inferences in merely 6.55 ms—a 12.76 times speedup.

The way training time grows for B-FNO and DB-FNO follows the same pattern we observe during inference: the larger the input size and the number of Fourier layers, the larger DB-FNO's speed advantage over B-FNO becomes. On the largest speaker-matrix case, one

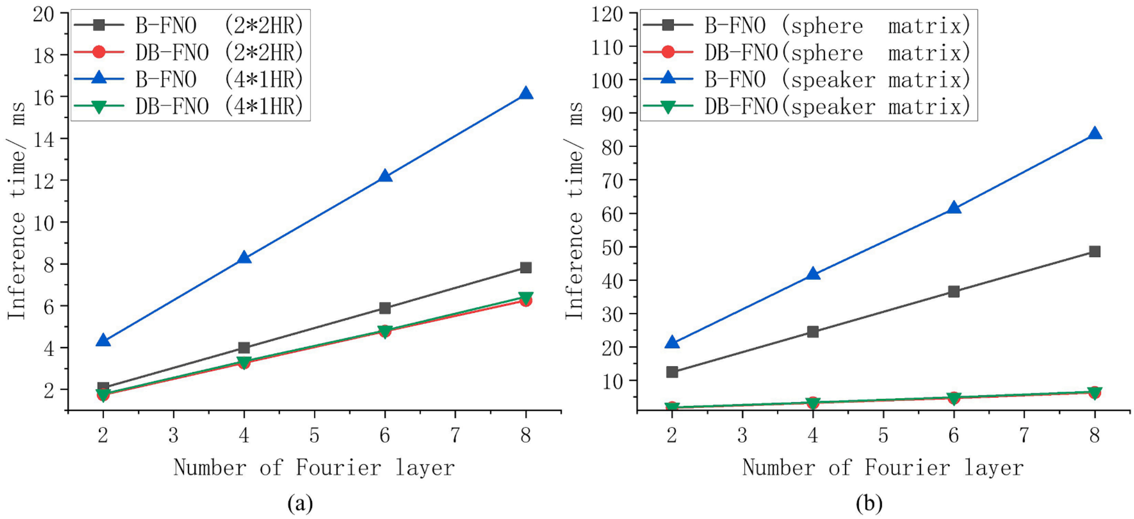

Fig. 9. (a) The inference time with increasing number of Fourier layers, for the 2 by 2 arranged HR matrix case and the 4 by 1 arranged HR matrix case, and 20 terms for DB-FNO. (b) The inference time with increasing the number of Fourier layers, for the sphere matrix and speaker matrix, and 30 terms for DB-FNO.

epoch with 3000 training samples takes B-FNO around 3700s, whereas DB-FNO finishes in only 10s. These results underscore the DB-FNO's superior scalability, establishing it as a valuable method for large-scale engineering applications that demand high computational efficiency.

3.8. Time in relation to problem size

The inference times reported in Table 4 correspond to configurations with 4 Fourier layers and 20 decomposition terms. Across four numerical test cases, the DB-FNO demonstrates minimal execution time variation on the same GPU, whereas the B-FNO exhibits significant runtime escalation as input size increases. The previous analysis of time complexity concluded that the DB-FNO has a complexity of $O(\sqrt{N}\,logN)$, reduced from the B-FNO's $O(NlogN)$. The execution times in the Table 4 align with the time complexity analysis for the B-FNO, as shown in Fig. 10 (a), but not for the DB-FNO. This discrepancy arises because of the small problem size in comparison with other parameters.

To elucidate this discrepancy, we define key parameters: N_t is the number of Fourier layers; N_h is the hidden channel dimension after the lifting step; N_p is the number of decomposed terms; N_e is the dimension of the reshaped 2D input, approximately equal to $\sqrt{N}$.

For each Fourier layer, the FFT and IFFT operations incur a time complexity of $O(N_h N_p N_e \log N_e)$, while weight multiplication scales as $O(N_h{}^2 N_p N_e)$. In all test cases, $N_h(64)$ and $N_p(20)$ are fixed, resulting in a constant product of 1280. Since the largest problem dimension $N_e(280)$ is smaller than $N_h N_p$, the overall execution time is dominated by $N_h N_p$, not N_e. This explains the nearly constant inference time observed across the four test cases. When N_e continues to increase beyond a certain threshold, it exerts measurable influence on inference time, as validated by tests with larger dimension sizes presented in Table 5. The actual time complexity shown in Fig. 10 (b) conform with the theoretical analysis, which is around $O(N_e \log N_e)$.

However, as the problem scale expands significantly, hardware memory limitations begin to dominate the runtime. At an extreme configuration where the square root element size reaches 8000, the inference time exhibits a severe, non-linear spike, abruptly increasing to approximately 8 seconds. This non-linear escalation is driven by a fundamental shift in how the hardware accesses data. For moderate sizes up to 4000, the intermediate computational data generated by the network fits entirely within the limited but ultra-fast local memory workspace of the GPU processor. Beyond this threshold, the massive data footprint exceeds this fast local workspace, forcing the hardware to continually transfer intermediate variables back and forth from the larger but much slower global memory storage. This constant data relocation creates a severe transmission bottleneck that dominates the total execution time. By requiring significantly less memory for intermediate representations, the DB-FNO prevents these heavy data transfer delays at large scales,

Table 4
The inference time of B-FNO and DB-FNO for examples with increasing problem size.

Examples	Number of elements (N)	Square root of the number of elements ($\sqrt{N}$)	Time of DB-FNO/ ms	Time of B-FNO/ ms
2*2HR	4880	69.9	3.78	3.98
4*1HR	16416	128.1	3.93	8.25
Sphere Matrix	42768	206.8	4.16	24.46
Speaker Matrix	78680	280.5	4.27	41.56

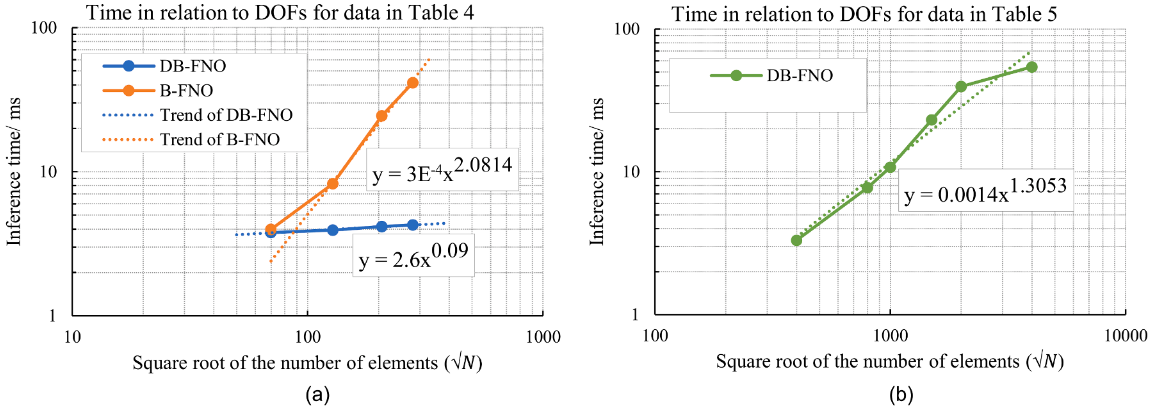

Fig. 10. (a) Computational Time vs. DOFs for DB-FNO and B-FNO Models, the presented data are from Table 4; (b) Computational Time vs. DOFs for DB-FNO Models, the presented data are from Table 5.

Table 5
The inference time of DB-FNO with the square root of increasing number of elements.

	400	800	1000	1500	2000	4000	8000
Time/ms	3.32	7.75	10.79	23.15	39.56	54.28	8187

sustaining high-speed inference across geometric domains that would otherwise saturate the hardware capability.

3.9. The number of parameters in DB-FNO compared with B-FNO

To quantify model complexity, we analyze the components within Fourier layers, including the weight tensor W, convolution kernel R_θ, and latent tensor v. For the B-FNO, the total parameter count scales as $O\left(N_t N_h^2 {N_{modes}}^{N_D} + N_t N_h {N_{modes}}^2\right)$, where N_t, N_h, N_D, and N_{modes} denote the number of Fourier layers, hidden channels, problem dimensions, and the number of retained Fourier modes respectively. In this work, both the original and DB-FNO utilize full Fourier modes, making N_{modes} equivalent to the number of boundary elements. In contrast, the DB-FNO reduces parameter complexity to $O\left(N_t N_h^2 N_{modes} N_D + N_t N_h N_p N_{modes}\right)$, where N_p represents the number of decomposed terms. This reformulation achieves significant parameter reduction, directly translating to improved GPU memory efficiency—a critical advantage for large-scale simulations.

Tables 6 and 7 validate this theoretical analysis. Entries with zero decomposed terms correspond to the B-FNO, while configurations with up to 30 terms demonstrate the DB-FNO's superior memory efficiency. Notably, increasing Fourier layers or input size amplifies the DB-FNO's advantages. Furthermore, varying N_p exerts minimal impact on total parameters, as illustrated in both tables. This insensitivity arises because the DB-FNO shares weight and Fourier coefficient tensors across all terms, as illustrated in Fig. 2, ensuring that the dominant term $O\left(N_t N_h^2 N_{modes} N_D\right)$ remains unaffected by N_p.

To translate these parameter counts into practical hardware requirements, we estimate the memory footprint for the largest benchmark, the speaker matrix with 78,680 boundary elements. The original 4-layer B-FNO, which contains approximately 553 million parameters, requires roughly 3 GB of VRAM for single-sample inference. During training, tracking gradients, optimizer states, and mini-batch activations pushes this requirement beyond 12 GB. By applying decomposition, the 4-layer DB-FNO with 30 terms reduces the parameter count to 18.3 million, requiring only 74 MB for weight storage. Consequently, single-sample inference consumes less than 200 MB of VRAM, and peak memory during training remains around 1 GB. This efficiency ensures that the DB-FNO can be trained and deployed comfortably on standard, commercial-grade GPUs. Conventional BEM requires constructing a fully populated N

Table 6
The number of parameters within the B-FNO and DB-FNO models for the 2*2HR case and 4*1HR case, with different number of Fourier layers and decomposed terms.

Number of decomposed terms	2*2HR: 61 × 80		4*1HR: 144 × 114	
	Number of Fourier layers		Number of Fourier layers	
	4	8	4	8
0	32,149,314	64,263,490	111,296,482	222,557,826
10	4,491,672	8,981,400	8,359,466	16,715,818
20	4,493,102	8,982,830	8,362,066	16,718,418
30	4,494,532	8,984,260	8,364,666	16,721,018

Table 7
The number of parameters within the B-FNO and DB-FNO models for the sphere matrix case and speaker matrix case, with different number of Fourier layers and decomposed terms.

Number of decomposed terms	Sphere Matrix: 198 × 216		Speaker Matrix: 281 × 280	
	Number of Fourier layers		Number of Fourier layers	
	4	8	4	8
0	301,552,770	603,070,402	553,102,530	1,106,169,922
10	13,472,834	26,940,994	18,323,968	36,641,792
20	13,476,994	26,945,154	18,329,598	36,647,422
30	13,481,154	26,949,314	18,335,228	36,653,052

× N complex system matrix, demanding approximately 100 GB of system RAM solely for allocation—a severe memory bottleneck. Modern accelerated solvers employ fast multipole methods or hierarchical matrix techniques to avoid evaluating all matrix entries and thus compress this memory footprint significantly. However, these BEM solvers must repeatedly assemble a new system for every individual frequency or geometric variation during parametric sweeps.

3.10. Computational cost compared to accelerated BEM solvers

To fully evaluate the practical efficiency of the DB-FNO framework, it is necessary to compare its overall computational lifecycle against modern accelerated boundary element solvers. Table 8 provides a breakdown of the computational costs for the large-scale Speaker Matrix case, divided into dataset generation, training, and inference. The baseline physical simulations are conducted using two advanced techniques, specifically a fast multipole BEM using the FastBEM Acoustics® software and a H-matrix based fast direct solver detailed in reference [14] with multi-thread CPU acceleration. The training and inference of DB-FNO is on GPU as stated before.

In Table 8, the time spent on training data generation is 500 hours when using the H-matrix based fast direct BEM solver and 75 hours when using FMM-BEM. By equating the total time required for these modern accelerated solvers with the cumulative computational time of the DB-FNO lifecycle, the break-even points for this specific speaker matrix case can be accurately established. Specifically, the neural operator achieves a net time reduction after approximately 3,000 evaluations compared to both the fast multipole BEM and the hierarchical matrix direct solver. This break-even dynamic directly dictates the typical parameter ranges and the expected number of evaluations for which the DB-FNO is practically suited. The proposed framework is not intended to replace conventional BEM for single-run or low-volume simulations. Instead, it is specifically designed for computationally exhaustive tasks such as topology optimization of acoustic metamaterials, uncertainty quantification, or high-resolution wideband frequency sweeps. In such applications, a single trained DB-FNO model is typically evaluated tens to hundreds of thousands of times, far surpassing the break-even threshold and providing an exponentially growing net reduction in computational time.

Furthermore, the efficiency of the DB-FNO is significantly enhanced by its continuous parameter resolution capabilities. To illustrate the relationship between sampling density and parameter resolution, consider the speaker matrix case. Although there are 10,626 possible geometric combinations for this specific array, the DB-FNO was successfully trained to a high degree of accuracy using a subset of only 3,000 samples. Despite this sparse sampling, the neural operator fundamentally learns a continuous boundary-to-solution mapping. Consequently, the parameter resolution during the inference stage remains extremely high, enabling the evaluation of unseen geometric configurations or extremely fine frequency steps without the need for an exhaustively dense training dataset. Conventional surrogate methods, such as interpolating spatial result maps, are straightforward to implement but suffer from the curse of dimensionality. For these methods, maintaining accuracy demands dense sampling of the parameter space, which becomes computationally prohibitive for high-dimensional problems.

4. Numerical examples

This section demonstrates the application of the DB-FNO to predict wave phenomena in realistic physical systems.

4.1. Radiation example

Tonpilz transducers, also known as "mushroom head" transducers, are a type of piezoelectric transducer commonly used in sonar systems. Due to their high efficiency and high-power output, Tonpilz transducers are widely utilized in the field of underwater

Table 8
Computational lifecycle cost comparison for the Speaker Matrix case.

Phase	Fast multipole BEM	H-matrix based fast direct BEM	DB-FNO
Dataset Generation	N/A (Direct Calculation)	N/A (Direct Calculation)	500/75 hours (3000 samples)
Model Training	N/A	N/A	3 hours (1,000 epochs)
Inference (per sample)	90s	600 s	4.27 ms

acoustics, including underwater communication, detection, and imaging systems. When a Tonpilz transducer receives an electrical signal, the piezoelectric ceramic element, such as a cylinder made of PZT, undergoes mechanical deformation. This deformation drives the vibration of the front and rear mass blocks. The vibration of these mass blocks induces pressure fluctuations in the surrounding medium, which propagate as sound waves.

In phased array configurations, beam steering is achieved by applying phase-shifted voltages across transducer elements. For the 3 $\times$ 3 array depicted the Fig. 11 (b), the voltage for the first row of transducers is set to V_0, while the second and third rows receive $V_0e^{j\varphi_0}$ and $V_0e^{j2\varphi_0}$, respectively, where j denotes the imaginary unit and φ_0 is the phase shift. Modulating φ_0 controls the beam's emission angle: increasing φ_0 steers the beam azimuthally, while decreasing it redirects the beam oppositely. Precise phase adjustments enable focal beamforming, enhancing acoustic intensity and detection sensitivity in target directions.

The transducer array is simulated in COMSOL Multiphysics® as a coupled electrostatics-solid mechanics-acoustics system. As illustrated in Fig. 11 (a), the steel tail mass is fixed at its outer surface, while PZT disks are excited with harmonic electrical signals. The solid mechanics module resolves structural deformations across all components, and the electrostatics interface computes electric field distributions within the PZT disks. These domains are bidirectionally coupled via the piezoelectric effect, linking mechanical stresses and strains to electric displacements. Static frequency-domain simulations compute the system's response under harmonic excitation, with $V_0 = \sqrt{2}$, phase shifts φ_0 ranging from $\pi/200$ to $\pi/2$, and operating frequencies spanning 10–11 kHz.

This study employs the BEM to simulate acoustic radiation from nine transducers embedded inside a cuboidal domain with fixed external dimensions of 140 mm length, 140 mm width and 40 mm height. The bottom of nine head mass are assigned with the velocity boundary conditions derived from coupled solid-electrostatic simulations, as depicted in Fig. 12 (c), while the remaining surfaces are modeled as acoustically rigid hard-wall boundaries. The surface of transducers is discretized into 27,650 boundary elements, with input data reshaped into a two-dimensional array of dimensions 158*175. The simulation frequencies align with those used in prior solid mechanics and electrostatics analyses, spanning 10–11 kHz.

A critical distinction between this transducer array application and earlier benchmark studies is that, the global geometry remains strictly unchanged, while the boundary velocities dynamically adapt to variations in operating frequency and phase shift. To accommodate this unique configuration, acoustic velocity profiles must be incorporated as primary input channels. However, this raises a fundamental question regarding optimal feature selection: are spatial coordinates still necessary when the geometry is fixed? To investigate this, a systematic evaluation of input configurations was conducted. This experiment aims to identify the optimal input channel configuration for maximizing predictive accuracy in fixed-geometry acoustic problems with dynamically varying boundary conditions. The configurations and their resulting accuracies are detailed in Table 9.

This study evaluates the performance of the original and DB-FNO across six input configurations as shown in Table 9, with both models configured identically: 4 Fourier layers, 64 hidden channels, half Fourier modes, 8,000 training samples uniformly spanning frequency and phase shift ranges, and 2,000 randomized test samples. Specifically, this grid is defined uniformly per dimension, comprising 100 evenly spaced frequency points from 10kHz to 11kHz, and 80 evenly spaced phase shift values from $\pi/200$ to $\pi/2$ radians. Input channels are defined as follows:

- Spatial coordinates x: (x, y, z);
- Velocity profiles v: $(v.real,\ v.imag)$ derived from coupled electrostatics-solid mechanics simulations;
- Frequency-encoded terms f: $\left(e^{ikx}.real,\ e^{ikx}.imag\right)$ with wavenumber k tied to operating frequency;

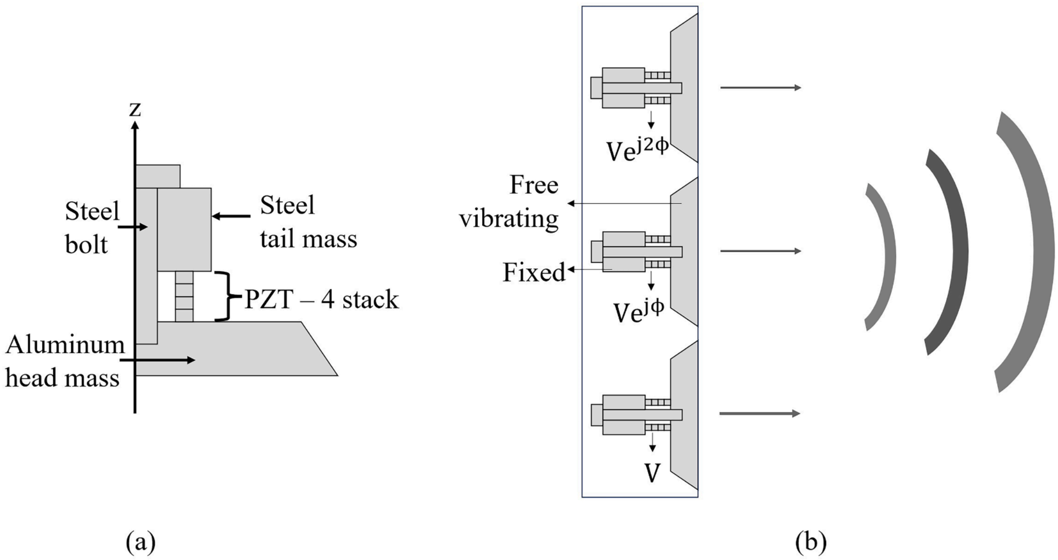

Fig. 11. (a) Axisymmetric plane diagram for a single transducer structure. (b) 2D schematic illustration for the whole transducer device (side view).

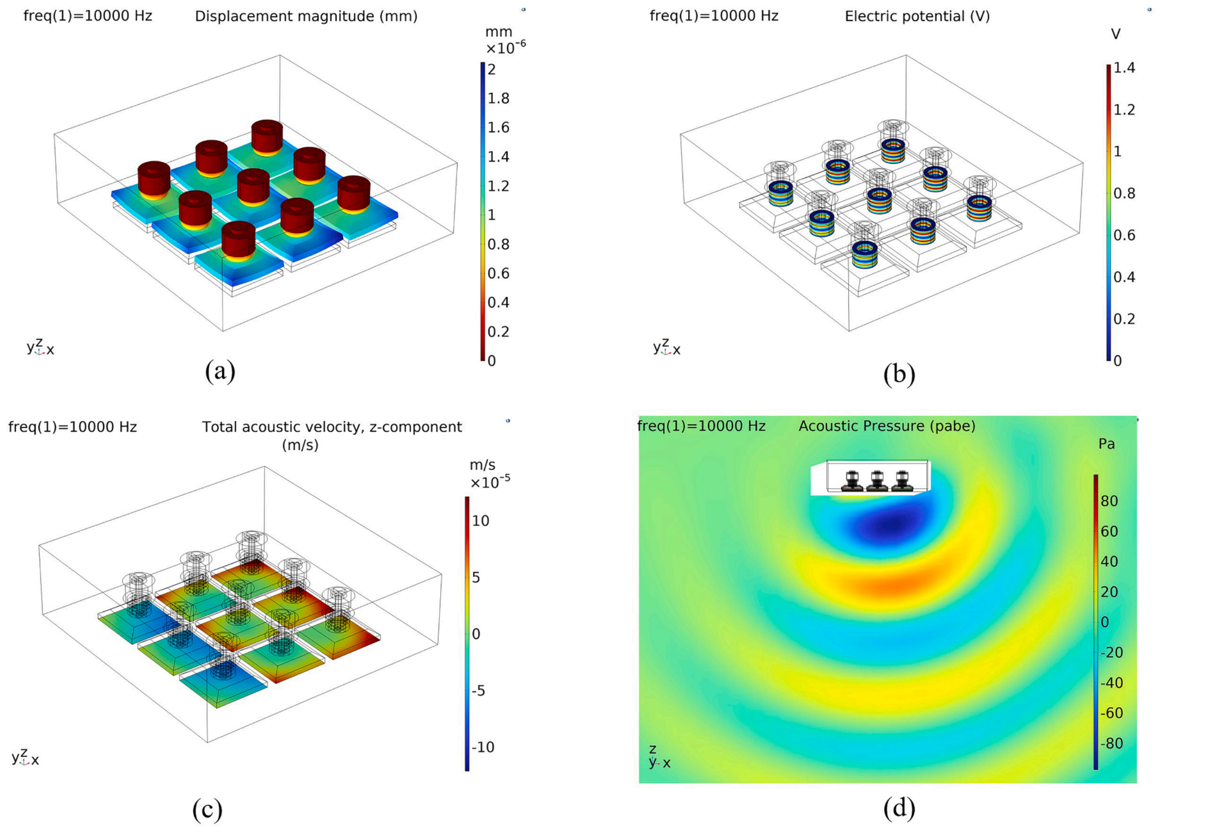

Fig. 12. (a) Displacement field obtained from structural analysis; (b) Electric potential field obtained from electrostatics simulation; (c) The boundary condition of the acoustic BEM field; (d) The acoustic radiation pressure field obtained from BEM simulation.

Table 9
The accuracy varied with different combination of input information, and the comparison of original and DB-FNO.

Input information	Number of input channel	Decomposed terms	rRMSE
$\boldsymbol{x}$, $\boldsymbol{v}$, f	7	0	0.1937%±0.0513%
		20	0.1926%±0.0513%
$\boldsymbol{v}$, f	4	0	0.1985%±0.0501%
		20	0.0665%±0.0511%
$\boldsymbol{x}$, $\boldsymbol{v}$, f, φ	7	0	0.1937%±0.0513%
		20	0.1613%±0.0779%
$\boldsymbol{v}$, f, φ	4	0	0.0154%±0.0038%
		20	0.0520%±0.0275%
$\boldsymbol{x}$, $\boldsymbol{v}$	5	0	0.1934%±0.0513%
		20	0.1931%±0.0514%
$\boldsymbol{v}$	2	0	0.0146%±0.0038%
		20	0.0113%±0.0036%

- Phase-shift terms φ: $\left(e^{ikx\cdot\varphi_0}.real, e^{ikx\cdot\varphi_0}.imag\right)$ incorporating voltage phase differences;

As demonstrated in Table 9, the DB-FNO consistently outperforms the baseline model across almost all input combinations. However, a critical finding from this ablation study reveals a counterintuitive inverse relationship between input complexity and predictive accuracy. Specifically, augmenting the inputs with theoretically relevant parameters, such as Cartesian coordinates or explicit frequency terms alongside the velocity, triggers a cascade of performance degradation.

This degradation occurs because the boundary velocity fields inherently encode the complete spectral-spatial state of this fixed-geometry acoustic system through implicit frequency and phase shift coupling. When additional input information is introduced, it acts as computational noise. This redundancy complicates the learning process by expanding the parameter space of the network, forcing the optimizer to navigate a more complex loss landscape. Furthermore, expanding the input channels inherently requires a more complex network architecture and sophisticated optimization techniques to prevent overfitting. Consequently, rather than providing the network with actionable physical insights, these redundant inputs prevent the model from fully leveraging the essential boundary data, leading to suboptimal training convergence and reduced prediction accuracy.

The velocity-only configuration achieves superior predictive accuracy 0.0113% rRMSE, reducing errors by nearly an order of magnitude compared to other input schemes. Figs. 13 and 14 visualizing the maximum and mean rRMSE cases across test samples for the optimal velocity-input case.

4.2. Submarine wave scattering with frequency-dependent IBCs

Submarine acoustic scattering research is critical for naval defense, underwater communications, and marine exploration. Understanding sound wave interactions with submarine surfaces enables improved object detection, stealth technology, and sonar performance. Submarine surface impedance depends on material properties, coatings, and structural features, making precise IBC modeling essential for predicting acoustic signatures and designing countermeasures. In this example, we deploy the DB-FNO for efficient and accurate modeling of submarine acoustic scattering with frequency-dependent IBCs.

We analyze a scaled submarine model with a length of 46.5m under 50–300 Hz frequency excitation. The surface impedance Z increases monotonically from 500,000 to 505,000 Pa·s/m across this band—a frequency-dependent characteristic critical for realistic scattering predictions. A unit-amplitude plane wave incidents along the positive x-axis, generating boundary pressure patterns visualized in Fig. 15. With the geometry fixed, we streamline inputs to complex incident wave pressures $(\phi^I.real,\ \phi^I.imag)$, eliminating redundant spatial coordinates as validated in Section 4.1. The surface mesh with 12,450 elements is reshaped into a 150×83 array to align with the DB-FNO architecture.

Training leverages 300 samples uniformly covering the frequency-impedance parameter space, supplemented by 100 randomized test cases. Configured with 4 Fourier layers, 64 hidden channels, and 20 decomposition terms, the model achieves 0.073% mean rRMSE after 2,000 epochs. Fig. 16 confirms exceptional agreement between predicted and BEM-calculated boundary pressures, while the descending loss curve suggests further gains with extended training.

To explore enhancement opportunities, we explicitly embedded impedance normalization $\left(\phi^I.real \cdot \frac{Z}{Z_0},\ \phi^I.imag \cdot \frac{Z}{Z_0}\right)$, hypothesizing that increased reflectivity would amplify pressure magnitudes. However, this modification yielded negligible improvement—evidence that frequency-dependent impedance effects are implicitly captured through wave propagation physics, rendering explicit scaling redundant. In this case, we also compare the accuracy of original and decomposed FNO and different input in Table 10. The DB-FNO achieved obvious higher accuracy than the B-FNO, which is likely due to the reduced model complexity and thus more effective training.

5. Conclusion

The proposed DB-FNO has demonstrated its effectiveness across a number benchmark examples and two realistic physical problems, positioning it as a promising approach to solving 3D large-scale parametric wave problems with notable efficiency and accuracy. By harmonizing tensor decomposition techniques with B-FNO, this framework effectively tackles efficiency challenges in acoustic simulations, particularly in scenarios involving complex geometries, frequency-dependent boundary conditions, and industrial-scale applications. The success of this methodology lies in its ability to reconcile the computational demands of traditional numerical methods with the learning capabilities of modern machine learning, thereby opening new avenues for high-fidelity large-scale wave analysis.

At the core of the DB-FNO's innovation is its ability to drastically reduce computational complexity without compromising predictive accuracy. Although the B-FNO is powerful, as the scale of the 3D problem increased, its time and memory requirements tend to increase significantly. By decomposing high-dimensional latent tensors into rank-1 components and leveraging parameter sharing

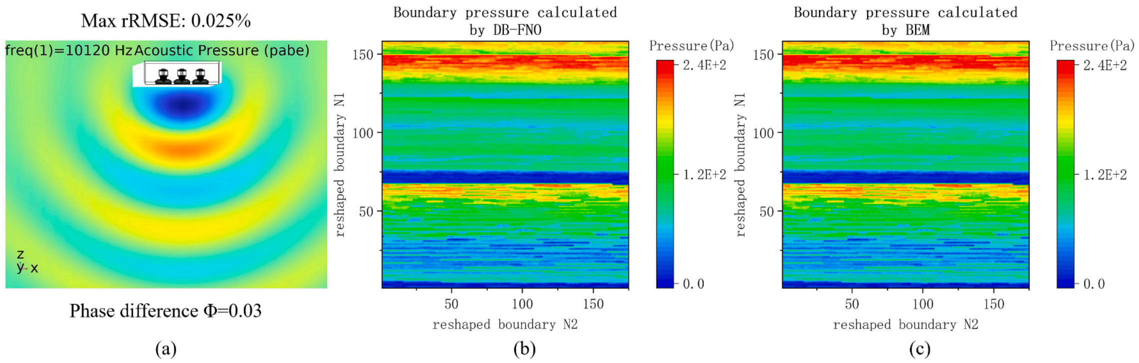

Fig. 13. (a) The case with max value of rRMSE among all test samples is 0.025%. The operating frequency is 10120Hz, and the phase difference phi is 0.03. (b) The absolute pressure on the 2D reshaped spatial array calculated by DB-FNO. (c) The absolute pressure on the 2D reshaped spatial array calculated by the BEM.

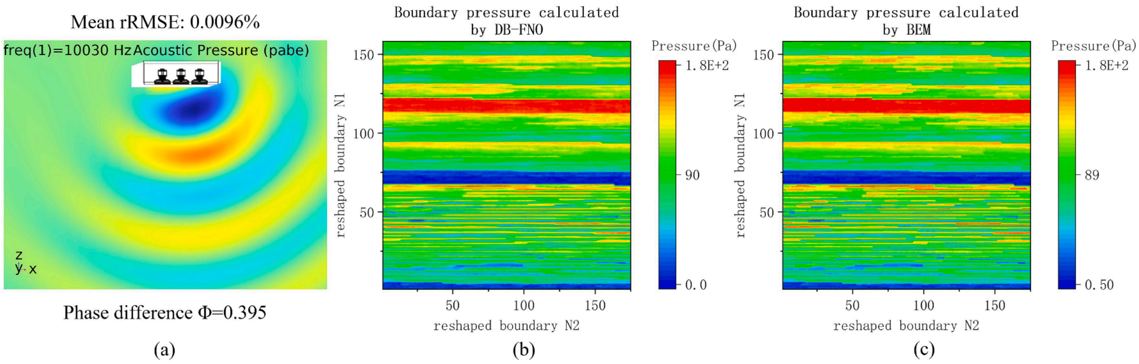

Fig. 14. (a) The case with mean value of rRMSE among all test samples is 0.0096%. The operate frequency is 10030Hz, and the phase difference phi is 0.395. (b) The absolute pressure on the 2D reshaped spatial array calculated by DB-FNO. (c) The absolute pressure on the 2D reshaped spatial array calculated by the BEM.

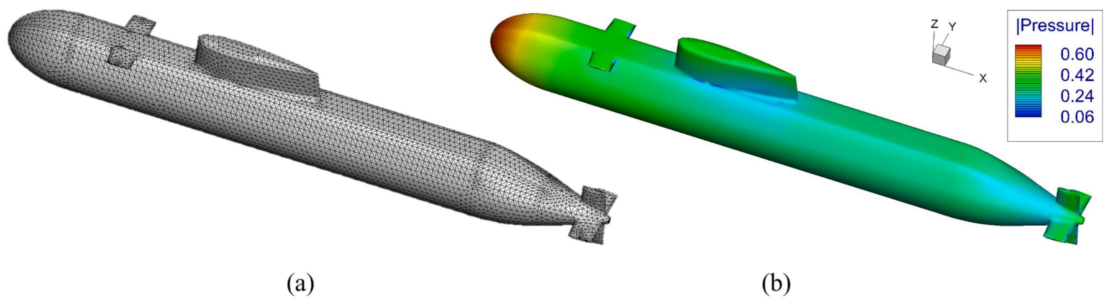

Fig. 15. (a) Discretized mesh of the submarine with the total length of 46.5m. (b) Absolute acoustic pressure on the surface of submarine calculated by BEM, subject to a plane wave at 50Hz along with positive x-axis, and the impedance boundary condition Z is 500,000 Pa·s/m.

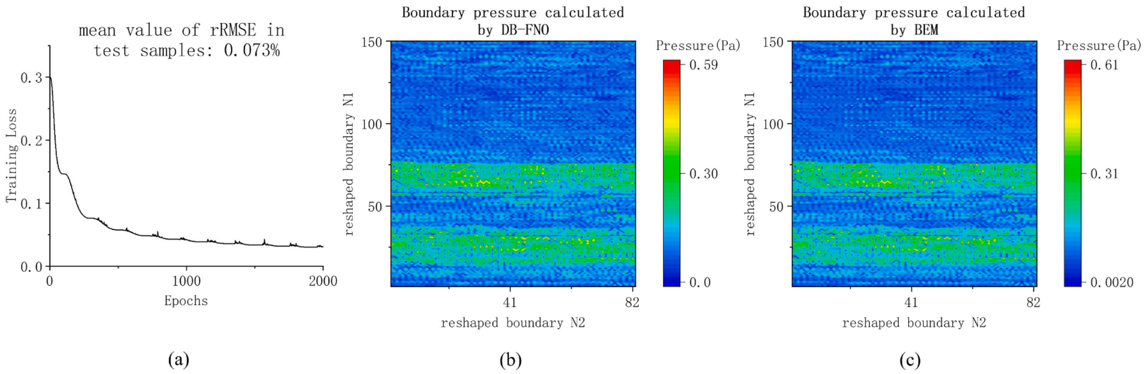

Fig. 16. (a) Loss curve of DB-FNO with the input of ϕ^I .real and ϕ^I.imag. (b) The absolute pressure on the 2D reshaped spatial array calculated by DB-FNO. (c) The absolute pressure on the 2D reshaped spatial array calculated by BEM.

Table 10
The accuracy of original and DB-FNO for different input information.

Input channels	Decomposed Terms	rRMSE
$(\phi^I.real,\ \phi^I.imag)$	0	0.2118%±0.0350%
$(\phi^I.real,\ \phi^I.imag)$	20	0.0726%±0.0071%
$\left(\phi^I.real\cdot\frac{Z}{Z0},\ \phi^I.imag\cdot\frac{Z}{Z0}\right)$	0	0.2114%±0.0334%
$\left(\phi^I.real\cdot\frac{Z}{Z0},\ \phi^I.imag\cdot\frac{Z}{Z0}\right)$	20	0.0722%±0.0068%

across decomposed terms, the proposed framework reduces time complexity from $O(N^2\log N)$ to $O(N^2)$, while simultaneously slashing memory consumption. In benchmark examples such as the speaker matrix case with 78,680 elements, the decomposed B-FNO achieves a remarkable 12.76 times speedup compared to its non-decomposed counterpart. Such performance enhancements are critical for industrial applications where large-scale rapid iteration is essential.

Beyond computational efficiency, the DB-FNO demonstrates remarkable robustness and generalization across diverse acoustic scenarios. In benchmark studies spanning Helmholtz resonator array and pulsating objects matrices, the framework consistently achieves rRMSE below 0.2%, with some cases reaching as low as 0.002%. This capability is further validated in real-world applications, such as modeling phased transducer arrays with continuously varying velocity boundary conditions and modeling submarine surfaces with IBCs. For challenging frequency-dependent IBCs, the framework achieves a mean rRMSE of 0.073%, underscoring its ability to generalized boundary conditions.

Despite these advancements, there is still much work to be done to enhance the method's capabilities. While the framework excels in static or parametrically varying geometries, extending it to dynamically deforming boundaries, such as those encountered in fluid-structure interaction problems represents a significant challenge. Future work could explore hybrid approaches that combine the DB-FNO with graph neural networks (GNNs) to handle dynamic meshes or topological transformations.

Moreover, several promising directions could further refine and expand the DB-FNO framework. First, incorporating physics-informed loss functions could enhance generalization to unseen parametric regimes. This would be particularly valuable for problems with sparse experimental data, such as ultra-high-frequency acoustic scattering or non-linear wave propagation. Second, integrating the multi-grid decomposition strategy from [27] into DB-FNO could further reduce the problem size, especially for very large-scale three-dimensional boundary value problems. Third, extending the framework to transient wave analysis—a domain currently dominated by time-stepping methods—would unlock new applications in shock wave modeling, seismic event prediction, and real-time acoustic diagnostics.\

CRediT authorship contribution statement

Ruoyan Li: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Wenjing Ye:** Writing – review & editing, Supervision, Project administration, Funding acquisition, Conceptualization. **Yijun Liu:** Writing – review & editing, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] M. Fatemi, J.F. Greenleaf, Imaging the viscoelastic properties of tissue. Imaging of Complex Media with Acoustic and Seismic Waves, Springer, 2002, pp. 257–276.

[2] A. Sakhnevych, A. Genovese, A. Maiorano, F. Timpone, F. Farroni, An ultrasound method for characterization of viscoelastic properties in frequency domain at small deformations, Proc. Inst. Mech. Eng., Part C 235 (23) (2021) 7180–7191.

[3] V. Rokhlin, Rapid solution of integral equations of scattering theory in two dimensions, J. Comput. Phys. 86 (2) (1990) 414–439.

[4] V. Rokhlin, Diagonal forms of translation operators for the Helmholtz equation in three dimensions, Appl. Comput. Harmon. Anal. 1 (1) (1993) 82–93.

[5] Y.J. Liu, Fast Multipole Boundary Element Method: Theory and Applications in Engineering, Cambridge University Press, 2009.

[6] M. Bebendorf, Hierarchical Matrices: A Means to Efficiently Solve Elliptic Boundary Value Problems (Lecture Notes in Computational Science and Engineering), Springer-Verlag, Berlin, 2008.

[7] A. Brancati, M. Aliabadi, I. Benedetti, Hierarchical adaptive cross approximation GMRES technique for solution of acoustic problems using the boundary element method, Comput. Model. Eng. Sci. (CMES) 43 (2) (2009) 149.

[8] P.-G. Martinsson, J.A. Tropp, Randomized numerical linear algebra: Foundations and algorithms, Acta Numerica 29 (2020) 403–572.

[9] J. Sherman, W.J. Morrison, Adjustment of an inverse matrix corresponding to a change in one element of a given matrix, Ann. Math. Stat. 21 (1) (1950) 124–127.

[10] M.A. Woodbury, Inverting modified matrices, Statistical Research Group, 1950.

[11] L. Greengard, D. Gueyffier, P.-G. Martinsson, V. Rokhlin, Fast direct solvers for integral equations in complex three-dimensional domains, Acta Numerica 18 (2009) 243–275, https://doi.org/10.1017/S0962492906410011.

[12] S. Ambikasaran, E. Darve, An O(NlogN) fast direct solver for partial hierarchically semi-separable matrices, J. Sci. Comput. 57 (3) (2013) 477–501, https://doi.org/10.1007/s10915-013-9714-z.

[13] J. Lai, S. Ambikasaran, L. Greengard, A fast direct solver for high frequency scattering from a large cavity in two dimensions, SIAM J. Sci. Comput. 36 (6) (2014) 887–903, https://doi.org/10.1137/140964904.

[14] R. Li, Y. Liu, W. Ye, A fast direct boundary element method for 3D acoustic problems based on hierarchical matrices, Eng. Anal. Bound. Elem. 147 (2023) 171–180.

[15] J. Xiao, W. Ye, Y. Cai, J. Zhang, Precorrected FFT accelerated BEM for large-scale transient elastodynamic analysis using frequency-domain approach, Int. J. Numer. Methods Eng. 90 (1) (2012) 116–134.

[16] D. Panagiotopoulos, E. Deckers, W. Desmet, Krylov subspaces recycling based model order reduction for acoustic BEM systems and an error estimator, Comput. Methods Appl. Mech. Eng. 359 (2020) 112755.

[17] X. Xie, Y. Liu, An adaptive model order reduction method for boundary element-based multi-frequency acoustic wave problems, Comput. Methods Appl. Mech. Eng. 373 (2021) 113532.
[18] D. Panagiotopoulos, W. Desmet, E. Deckers, Parametric model order reduction for acoustic boundary element method systems through a multiparameter Krylov subspaces recycling strategy, Int. J. Numer. Methods Eng. 123 (22) (2022) 5546–5569.
[19] S. Lefteriu, M.S. Lenzi, H. Beriot, M. Tournour, W. Desmet, Fast frequency sweep method for indirect boundary element models arising in acoustics, Eng. Anal. Bound. Elem. 69 (2016) 32–45.
[20] Y. Li, O. Atak, S. Jonckheere, W. Desmet, Accelerating boundary element methods in wideband frequency sweep analysis by matrix-free model order reduction, J. Sound. Vib. 541 (2022) 117323.
[21] A. Alguacil, M. Bauerheim, M.C. Jacob, S. Moreau, Deep learning surrogate for the temporal propagation and scattering of acoustic waves, AIAA J. 60 (10) (2022) 5890–5906.
[22] N. Mehtaj, S. Banerjee, Scientific machine learning for elastic and acoustic wave propagation: neural operator and physics-guided neural network, Sensors 25 (12) (2025) 3588.
[23] S.Y. Lee, C.-S. Park, K. Park, H.J. Lee, S. Lee, A Physics-informed and data-driven deep learning approach for wave propagation and its scattering characteristics, Eng. Comput. 39 (4) (2023) 2609–2625.
[24] Z. Li et al., "Fourier neural operator for parametric partial differential equations," arXiv preprint arXiv:2010.08895, 2020.
[25] R. Li, W. Ye, Y. Liu, A boundary-based Fourier neural operator (B-FNO) method for efficient parametric acoustic wave analysis, Eng. Comput. (2025) 1–18.
[26] K. Li, W. Ye, D-FNO: A decomposed Fourier neural operator for large-scale parametric partial differential equations, Comput. Methods Appl. Mech. Eng. 436 (2025) 117732.
[27] J. Kossaifi, N. Kovachki, K. Azizzadenesheli, and A. Anandkumar, "Multi-grid tensorized fourier neural operator for high-resolution pdes," arXiv preprint arXiv: 2310.00120, 2023.
[28] X. Xie, W. Wang, K. He, G. Li, Fast model order reduction boundary element method for large-scale acoustic systems involving surface impedance, Comput. Methods Appl. Mech. Eng. 400 (2022) 115618.
[29] X. Xie, W. Wang, H. Wu, M. Guo, Data-driven analysis of parametrized acoustic systems in the frequency domain, Appl. Math. Model. 124 (2023) 791–805.
[30] N. Borrel-Jensen, A.P. Engsig-Karup, C.-H. Jeong, Physics-informed neural networks for one-dimensional sound field predictions with parameterized sources and impedance boundaries, JASa Express. Lett. 1 (12) (2021).
[31] Z. Zhang, Y. Zhao, N. Gao, Recent study progress of underwater sound absorption coating, Eng. Rep. 5 (7) (2023) e12627.
[32] M. Ranjbar, M.U. Bayer, Design of acoustic coating for underwater stealth in low-frequency ranges, J. Braz. Soc. Mech. Sci. Eng. 46 (3) (2024) 140.
[33] Y. Saad, M.H. Schultz, GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Stat. Comput. 7 (3) (1986) 856–869.