A generic deep learning framework for propagation and scattering of acoustic waves in quiescent flows

Antonio Alguacil, Michaël Bauerheim, Marc Jacob, Stéphane Moreau

A generic deep learning framework for propagation and scattering of acoustic waves in quiescent flows

Antonio Alguacil* and Michael Bauerheim†
Université de Sherbrooke, Sherbrooke, Quebéc, J1K2R1, Canada
ISAE-SUPAERO, Université de Toulouse, 31400 Toulouse, France

Marc C. Jacob‡
Ecole Centrale de Lyon, Université de Lyon, 69130 Ecully, France

Stéphane Moreau§
Université de Sherbrooke, Sherbrooke, Quebéc, J1K2R1, Canada

A deep learning surrogate for the direct numerical prediction of two-dimensional acoustic waves propagation and scattering with obstacles is developed through an auto-regressive spatio-temporal convolutional neural network. A single database of high-fidelity lattice Boltzmann temporal simulations is employed in the training of the network, achieving accurate predictions for long simulation times for a variety of test cases, representative of bounded and unbounded configurations. The capacity of the network to extrapolate outside the manifold of examples seen during the training phase is demonstrated by the obtaining of accurate acoustic predictions for relevant applications, such as the scattering of acoustic waves on an airfoil trailing edge, an engine nacelle or in-duct propagation. The method is tested for two types of input normalizations, coupled with an a-posteriori correction which improves the acoustic energy conservation of the predictions. The use of an adaptive local normalization along with the physics-based energy conservation results in an error reduction for all the studied cases.

I. Introduction

THE propagation of acoustic waves in complex media constitutes a challenging task in the context of aerodynamically generated noise. The presence of complex boundaries, mean flow or background medium inhomogeneities affects the trajectory of propagating waves, through known physical mechanisms such as scattering, dispersion, reflection or absorption. Taking into account such phenomena is required to design quiet aeronautical devices, however creating such pre-design tools with a low cost is still an open challenge. To do so, two types of approaches exist for predicting noise propagation [1]: on the one hand, direct numerical computational aeroacoustics (CAA) calculate simultaneously both acoustic sources and the subsequent wave propagation. Direct computations yield highly accurate results but are highly demanding in terms of computational costs. On the other hand, hybrid methods constitute a more affordable way to perform aeroacoustic predictions. These methods rely on the separate computations of the source region and the propagation into the far field. The propagation step can be achieved either through semi-analytical means (acoustic analogies) or fully numerical approaches (Linearized Euler Equations, Acoustic Perturbation Equations).

The coupling between the source and the propagation regions can become cumbersome, especially when considering the complex media effects on propagation. For example, acoustic analogies for non-homogeneous media, e.g. Lilley's analogy for sheared mean flows [2]), require an additional effort in the computation of the associated Green's function, because of the presence of hydrodynamic modes and critical layers. The same observation applies for complex boundary conditions as in the Ffowcs Williams and Hawkings analogy [3], where new integrals appear when accounting for the presence of moving surfaces. Thus, the more mean flow or boundaries effects are taken into account by the acoustic analogy, the more complex becomes the calculation of the associated Green's function or its subsequent convolution

*PhD Student, Département Aérodynamique, Énergétique et Propulsion, ISAE-SUPAER0, BP 54032, F-31055 Toulouse Cedex 4, antonio.alguacil-cabrerizo@isae-supaero.fr

†Associate Professor, Département Aérodynamique, Énergétique et Propulsion, ISAE-SUPAER0, BP 54032, F-31055 Toulouse Cedex 4

‡Professor, Université de Lyon, École Centrale de Lyon, INSA Lyon, Université Claude Bernard Lyon I, CNRS, Laboratoire de Mécanique des Fluides et d'Acoustique, UMR 5509, F-69134 Ecully Cedex, France, AIAA Member

§Professor, Mechanical Engineering Department, AIAA Lifetime Member

with sources. In the most complex cases, relevant for industrial application, the Green's function can no longer be fully analytically calculated (e.g. annular duct with swirling mean flow analogy [4]), and numerical methods must be employed instead. Therefore, in the general cases of complex mean flows or boundary conditions, propagation methods must rely on expensive numerical methods, using for instance high-order numerical schemes to ensure low dissipation and dispersion properties, which prevents these methods from being applied efficiently at a pre-design stage.

A recent work by the authors [5, 6] developed an alternative method to replicate, at lower cost, a high-fidelity numerical code for the propagation of acoustic waves using a surrogate model: instead of resolving the discretized propagation equations, a neural network was trained with examples of high-fidelity direct Lattice-Boltzmann (LBM) simulations, resulting in a data-driven convolutional neural network for acoustic wave propagation. It was found that such an approach managed to reproduce wave propagation in simple academic cases of 2D quiescent closed domains with reflecting walls. These promising results suggest that a similar approach might be applied in open-domain problems, in the presence of complex boundary conditions (such as ducts or scattering objects). In particular, the method has demonstrated good generalization capabilities (e.g. being able to predict wave propagation of unseen source distribution) while diminishing classical numerical constraints (e.g. the CFL number) allowing faster simulations without loss of accuracy.

Yet, the previous framework [5] implicitly encoded the boundary conditions of the problem, and was thus not applicable to more complex geometries and representative configurations.

Several studies have dealt with the problem of encoding boundary condition information into neural network surrogates for modelling partial differential equations (PDEs). Some make use of built-in tools of convolutional networks, such as periodic padding to encode simple boundary conditions such as periodic conditions [7]. Other works dealing with more complex geometries employ binary inputs [8] or signed distance functions [9] to encode the geometry information. However, such works only tackle steady-state problems, such as RANS predictions [8], or Sound Pressure Level (SPL) estimation [9].

Closer to the problem of interest, works such as Lino *et al*. [10] predict the full spatio-temporal propagation and scattering of seismic waves with a convolutional network. They employ a binary field to encode geometry information and use an auto-regressive strategy to perform the temporal prediction. However, few quantitative results are shown, in particular regarding the ability of the network to perform accurately outside the training dataset. Furthermore, it seems that they train the network on different datasets, depending on the different types boundary conditions. A general database capable of training the neural network for different types of boundaries conditions would be beneficial to such kind of methods, as it would avoid the cost of retraining the network each time. A typical example of such a pathology can be found in Physics-Informed Neural Networks (PINNs) [11], which need to be retrained for each new set of initial or boundary conditions. Some works, such as Moseley *et al*. [12] employ PINNs to solve the wave equation for complex background media [12], and parameterize the sources input position in order to avoid retraining the model for each new set of initial conditions. However, the problem of using a single neural network for various types of boundary conditions, such as reflective and non-reflective acoustic conditions, is still an open question.

Consequently, the present work proposes a unified generic framework based on an deep auto-regressive convolutional neural network to tackle both bounded and unbounded acoustic problems in quiescent flows, with potential scattering by inner objects. The work focuses on two mains aspects: (i) the ability of the trained neural network to perform in various physics regimes dependent on the choice of boundary conditions, some of which differ significantly from the ones present in the chosen database, (ii) the practical choices for pre- and post-processing the data when predicting in the auto-regressive loop. In particular, two types of data normalization are compared, along with a modified Energy Preserving Correction, which was introduced in [6]. The latter intends to improve the conservation of acoustic energy over the temporal neural network predictions, and thus improving significantly its long-term predictions.

First the methodology regarding the choice of neural network architecture and its training are presented in Section II, Then, section III describes the unique dataset employed for training the neural network and its validation. In section IV, several test cases representative of unbounded problems are evaluated using the present framework, and compared with the LBM reference. In particular, Sections IV.C and IV.D explore two additional cases representative of typical industrial applications, namely a NACA airfoil and an engine nacelle. Finally, duct configurations, as well as a completely closed domain, are also studied in Section V.

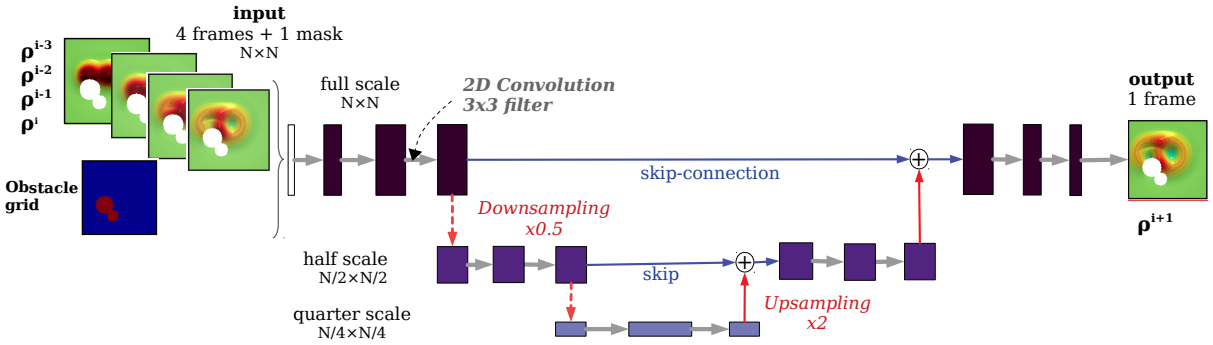


Fig. 1 Schematics of U-Net multi-scale CNN with three scales of convolutions at full, half and quarter resolution. Input is composed of 4 consecutive frames $t = i - 3$, $t = i - 2$, ..., $t = i$ of acoustic density and of a Boolean mask representing the position of the obstacles. The output is the next frame at $t = i + 1$ of resolution NxN voxels. Each rectangular block is proportional to the number of convolutional filters per layer. Skip connections are employed to facilitate the training of the network.

II. Methodology

A. Modeling dynamical systems with a learned surrogate

The objective of this work is to efficiently approximate the complete space-time evolution of a dynamical system with a surrogate, model trained on high-fidelity data. Here, the system of interest is the propagation and scattering of acoustic waves with hard-reflecting obstacles in quiescent flows, which can be described by the following Cauchy problem for the acoustic density ρ wave equation:

$$\begin{aligned} &\frac{\partial^2 \rho(\boldsymbol{x},t)}{\partial t^2} + c_0^2 \Delta \rho(\boldsymbol{x},t) = f(\boldsymbol{x},t), \quad \boldsymbol{x} \in \mathcal{D}, t \in [0,T] \\ &\mathcal{B}(\rho(\boldsymbol{x},t)) = 0, \quad \boldsymbol{x} \in \partial\mathcal{D} \\ &\rho(\boldsymbol{x},0) = \mathcal{I}_0(\boldsymbol{x}) \end{aligned} \tag{1}$$

where ρ is defined in the domain $\mathcal{D}$ with boundaries $\partial\mathcal{D}$. $\mathcal{B}$ is the boundary operator that enforces the boundary conditions (here, $\mathcal{B} = 0$ expresses the absence of sources along the boundaries). Typically for hard-reflecting walls it simply reads $\nabla\rho(\boldsymbol{x} \cdot \boldsymbol{n})|_{\partial\mathcal{D}} = 0$. c_0 is the speed of sound, f a source term and $\mathcal{I}_0$ is the initial condition expressed as a volume distribution.

For an arbitrary boundary condition operator $\mathcal{B}$, Eq. (1) is usually resolved numerically, by discretizing the state variable in space and time and subsequently integrating the time evolution from t to $t + \Delta t$ using an iterative solver, where Δt is the time-step between to iterations.

The goal of the present study is to recast this numerical integration as an optimization problem, in order to learn the space-time evolution of the acoustic density. Discretizing the state variable in space and time, let $\rho^i \in \mathbb{R}^{d_1 \times d_2}$ be the solution of Eq. (1) at time-step i on a uniformly spaced discretization of $\mathcal{D}$ defined by a grid with d_i nodes in each direction.

The learned PDE operator G, defined by its trainable parameters θ, must fulfill the following time-invariant equation,

$$\rho_{i+1} = G(X_{i+1}, \theta) \tag{2}$$

where $X_{i+1} = \{\rho_i, \rho_{i-1}, ..., \rho_{i-k}\}$ is the input of the surrogate model, corresponding to the k previous time-states. Note that following [5], the present study will use 4 input snapshots ($k = 3$).

The operator G is obtained by solving the following optimization problem:

$$\arg\min_{\theta} \quad \mathcal{L}\left(G(X_{i+1}, \theta), \rho_{i+1}\right) \tag{3}$$

where $\mathcal{L}$ is a metric of the distance between the approximated prediction from the learned operator $\hat{\rho}_{i+1} = G(X_{i+1}, \theta)$ and the actual solution ρ_{i+1}. Here ρ_{i+1} is obtained through a high-fidelity numerical simulation database as described

in section III, but could be obtained by other means: analytical solutions (if existing), experimental observation or a blending of the previous.

Such a space-time operator can be learned through a highly non-linear neural network regressor. For high-dimensional databases, such as the ones encountered in Computational Aeroacoustics or Computational Fluid Dynamics, the use of convolutional networks is an efficient solution to simultaneously learn the spatial and time integration, as shown in earlier works [6, 7, 10]. The main advantage with respect to traditional numerical schemes is two-fold: in comparison with explicit schemes, such as Euler or Runge-Kutta integrators, the learned surrogate is not constrained by the classical time-stepping stability constraints [5, 13]. Furthermore, it does not require any matrix inversion as implicit integrators do.

B. Autoregressive spatio-temporal prediction of wave propagation

Once trained, the resulting neural network G is employed in an auto-regressive strategy to predict the complete spatio-temporal evolution of acoustic fields. Although the network is only trained to predict one single time-step ahead from the input data, this auto-regressive strategy is used as a time-integrator to propagate signals up to an arbitrary time horizon.

Formally, the acoustic density at time $t = n\Delta t$ can be calculated as

$$\rho_n = \underbrace{G \circ G \circ ... \circ G}_{n-k-1 \text{ times}}(X_{k+1}) \tag{4}$$

where $X_{k+1} = \{\rho_k, \rho_k, ..., \rho_0\}$ contains the initial condition. In practice, $k = 3$ is chosen, so that the last predicted frame ($i + 1$ in Fig. 1) is added to a new series of inputs (composed of $i - 2$, $i - 1$, i and $i + 1$ in this example).

C. Neural network: U-Net convolutional network

A U-Net convolutional neural network [14] is employed as shown in Fig. 1. Such an architecture is capable of treating large structured image-like inputs with few trainable parameters by sliding small local trainable convolutional filters around the input (typically 3×3 or 5×5 filters). In comparison with classical fully-connected networks, convolutional networks naturally encode neighbourhood information between pixels which is key when resolving Partial Differential Equations. This can be related to the stencils used to calculate the spatial gradients in a discrete grid.

Furthermore, convolutions can be combined with down-sampling operations to increase the receptive field [15], in order to capture long-range spatial information. The U-net architecture combines convolutions with pairs of down- and up-sampling operations to treat the input in a multi-scale way. In this case, max pooling operation along with bi-linear up-sampling is employed. The multi-scale treatment separately processes the different scales of the problem, specialising the different filters to focus either on local or global flow features. Unlike for the Multi-Scale used in previous works [5], it was found that the U-Net network converged quicker than the Multi-Scale one. This can be attributed to the presence of skip connections that facilitate the backward flow of gradients during the optimization of the network. Moreover, U-Net have better inference time performance compared with the Multi-Scale net of [5], which further accelerate the acoustic predictions.

The input to the neural network corresponds to the previously defined X^{i+1} vector, concatenated with an additional obstacle grid g indicating the presence of the reflecting obstacles with a Boolean encoding (ones assigned to obstacles pixels and zeros to fluid ones). Thus, the position of the boundary conditions are explicitly given as inputs to the neural network, as employed in other works using convolutional neural networks in fluid-mechanics related problems [8, 16].

The time-step Δt at which the prediction $t \rightarrow t + \Delta t$ is performed is fixed by the temporal spacing between input and output, thus in order to change its value, a new training is required.

Appendix A details the parametric functions used for each one of the U-Net scales, as well as the number of trainable parameters.

III. Dataset D1: Propagation of Gaussian pulses in unbounded domain with hard wall obstacles

As in [6], a LBM code is used to generate the data for the optimization of the network on the training dataset. Such a dataset, employed to feed inputs into the neural network and compare its outputs with supervised references, is decribed in the following section. A particular attention is paid to the validation of the LBM code on the wave scattering by objects to guarantee the accuracy of the training database.

A. Numerical setup

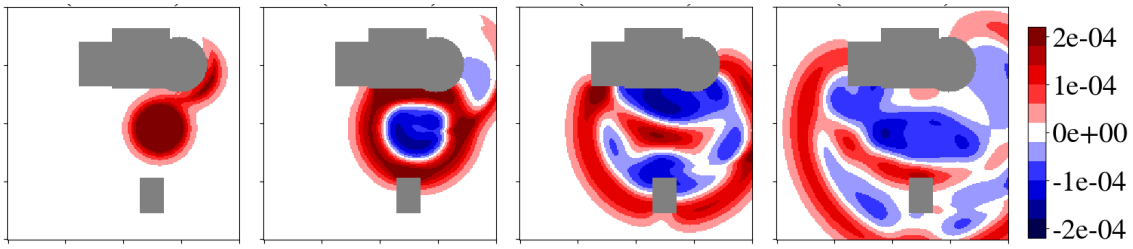


Fig. 2 Example of one simulation from the dataset at times (from left to right) $\tau = 0.03, 0.17, 0.35$ **and** 0.52**. Initial conditions are composed of Gaussian pulses of acoustic density located at random positions. Each initial condition** $(it = 0)$ **is stepped for 300 LBM iterations in total** $(\tau = 0.87)$**. Acoustic density fields are saved every 3 LBM iterations.**

The dataset (indicated as **D1**) consists of 100 2D-simulations of time-propagating acoustic waves on square domains with non-reflecting boundary conditions and random obstacles with hard reflecting walls (cylinders and rectangles) placed inside the calculation domain, as seen in Fig. 2. This dataset aims at studying the capacity of the neural network to predict scattering of acoustic waves by such hard-wall reflecting obstacles. The *physical* dimension of the computational domain is chosen as D (see Fig. 3a), each direction being discretized with $N_{nodes} = 200$. Similar to the dataset used in [6], the wave sources are composed of a random number of Gaussian pulses which are used as initial conditions for the density field, namely:

$$\rho(\boldsymbol{x}, 0) = \rho_0 + \sum_{j=1}^{N_p} \varepsilon_j \exp\left(-\frac{\log 2}{b_j^2} ||\boldsymbol{x} - \boldsymbol{c}_j||\right) \tag{5}$$

where ρ_0 is the mean flow density, N_p is sampled from a uniform distribution $\mathcal{U}[1, 5]$, ε is the pulse amplitude, here fixed to $\varepsilon = 10^{-3}$, b is the pulse half-width, fixed to $b/\Delta x = 12$ and $\boldsymbol{c}_j$ is the pulse center, such that $\boldsymbol{c}_j = \mathcal{U}[0.2D, 0.8D]^2$. Both the number of cylinders and rectangles are drawn from a random uniform discrete distribution $\mathcal{U}[0, 3]$. The rectangle sizes (height and width) vary from $0.01D$ to $0.3D$, while cylinder radii vary from $0.08D$ to $0.15D$. Boundary conditions at the obstacle walls are modelled as bounce-back conditions. This type of boundary is not generally fitted for inclined boundaries, such as the cylinder wall. However, in absence of mean flow, if the wavelength of acoustic waves is sufficiently large with respect to the cell size Δx, then the bounce-back "staircase" will be compact and have no effect on the accuracy of numerical results. This is demonstrated in section III.B, where the LBM method is validated for wave scattering by a cylinder. Non-reflecting boundary conditions are modelled using a perfectly matched layer (PML) that prescribes a damping term in the governing equations in order to attenuate outgoing acoustic modes [17]. The output fields from the LBM simulations are cropped to only feed the physical domain to the neural network.

Results are reported using a normalized time, defined as

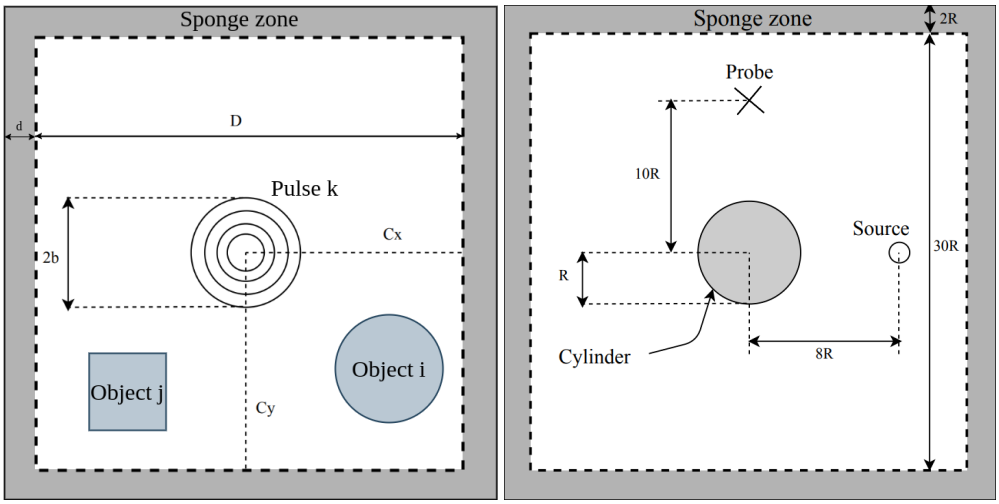
$$\tau = tc_0/D \tag{6}$$

A grid spacing of $\Delta x = D/(N_{nodes} - 1) = 0.5025$ is employed. Notice that in LBM the value of the time-step Δt is proportional to the grid spacing, here $\Delta t = c_s/c_0 \Delta x$. This acts as a Courant-Friedrichs-Lewy (CFL) condition, depending on the sound speed in lattice units c_s, which is determined by the lattice choice, here equal to $(c_s)_{lb} = 1/\sqrt{3} \simeq 0.57$ in the $D2Q9$ lattice [18].

The time τ corresponds to the propagation time of an acoustic wave from one boundary to the other. Each training simulation is stopped at $\tau = 0.5$, when the Gaussian pulse has reached the absorbing layer and density fields are recorded at time-steps that are multiples of $\Delta\tau = 0.0087$ (*i.e.* each 3 LBM iterations). The latter time-step is the one used by the CNN, in order to demonstrate the CNN capability to replicate high-fidelity codes while mitigating their CFL constraint. In practice, the computed LBM fields are packed into groups of 4+1 frames (inputs + target) for the CNN training, as detailed in section II.B.

Note that Convolutional network are translation-invariant by construction [19]. However, they are not invariant by rotation. Thus, data augmentation is performed during the training of the network: input-target pairs are randomly flipped in angles of 0, $\pi/2$, π and $3\pi/2$ radians.

B. Validation of the dataset



(a) **(b)**

Fig. 3 Schematic of the dataset configuration. (a) Sketch of boundary and initial conditions, made of Gaussian pulses of half-width b propagating in a domain of size $D \times D$ with arbitrary cylindrical and rectangular obstacles located inside the physical domain. A sponge layer of width d is added to absorb the outgoing acoustic waves. (b) Configuration used for the validation case: a cylinder of radius R located at the domain center scatters a Gaussian pulse. Measurements are performed at the probe location.

The LBM code is validated with respect to the scattering of a Gaussian pulse by a cylinder. A schematic is shown in Fig. 3b. A rigid cylinder of radius R is located at $(x, y) = (0, 0)$ inside a domain of size $30R \times 30R$, plus an additional $2R$ of sponge zones at each boundary. A density Gaussian pulse of half-width $b/R = 0.4$ is located at $(x, y) = (8R, 0)$ and a probe is located at $(x, y) = (0, 10R)$. A recursive and regularized BGK collision model (rrBGK-LBM) [20] is employed to maintain code stability with low numerical dissipation. Two cases are compared: a first one with the pulse half-width discretized by 12 lattice points (901 lattice points in the domain excluding sponge zones), and a second one with 18 points per half-width (1351 lattice points in the domain excluding sponge zones).

Results are compared to the analytical solution derived by Tam [21]. For a pulse of half-width b and pulse parameter $\alpha = (\log 2)/b^2$ located at (x_s, y_s), the total unknown potential can be divided in two components related respectively to the incident (subscript i) and reflected waves (subscript r):

$$\phi(x, y, t) = \phi_i(x, y, t) + \phi_r(x, y, t) \tag{7}$$

The total pressure field is found as follows:

$$\begin{aligned} p(x, y, t) &= -\frac{\partial \phi}{\partial t} = -\frac{\partial}{\partial t}(\phi_i + \phi_r) \\ &= Re\left\{\int_0^\infty (A_i(x, y, \omega) + A_r(x, y, \omega)e^{-i\omega t}\, d\omega\right\} \end{aligned} \tag{8}$$

where $A_i(x, y, \omega)$ is given by

$$A_i(x, y, \omega) = \frac{1}{2\alpha} e^{-\frac{\omega^2}{(4\alpha)}} J_0(\omega r_s) \tag{9}$$

and $r_s = \sqrt{(x - x_s)^2 + (y - y_s)^2}$. J_k represents the Bessel function of first kind of order k. $A_r(x, y, \omega)$ is given, in polar coordinates (r, θ):

$$A_r(x, y, \omega) = \sum_{k=0}^{\infty} C_k(\omega) H_k^{(1)}(r\omega) \cos(k\theta) \tag{10}$$

where

$$C_k(\omega) = \left(\frac{\omega}{2b} e^{-\frac{\omega^2}{(4\alpha}}\right) \frac{\varepsilon_k}{\pi\omega H_k^{(1)'}(r_0\omega)} \int_0^\pi J_1(\omega r_{s0}) \frac{r_0 - x_s\cos(\theta) - y_s\sin(\theta)}{r_{s0}} \cos(k\omega) d\theta \tag{11}$$

with $\varepsilon_0 = 1$ and $\varepsilon_k = 2$ for $k \neq 0$. r_{s0} is defined as:

$$r_{s0} = \sqrt{r_0^2 - 2r_0 x_s \cos\theta - 2r_0 y_s \sin\theta + x_s^2 + y_s^2} \tag{12}$$

and $H_k^{(1)'}$ is the derivative of the Hankel function of the first kind of order k.

Figure 4 shows the time evolution of the density at the probe location for two different resolutions of the LBM lattice. A resolution of 12 point per half-width is sufficient to capture all the physics of the problem accurately. The first amplitude peak represents the direct propagation of the initial Gaussian pulse, while the other two peaks at $tc_0/D = 8$ and 10 arise from the pulse scattered by the cylinder wall. Therefore, the pulses in the dataset are resolved with $b/\Delta = 12$ points per initial half-width.

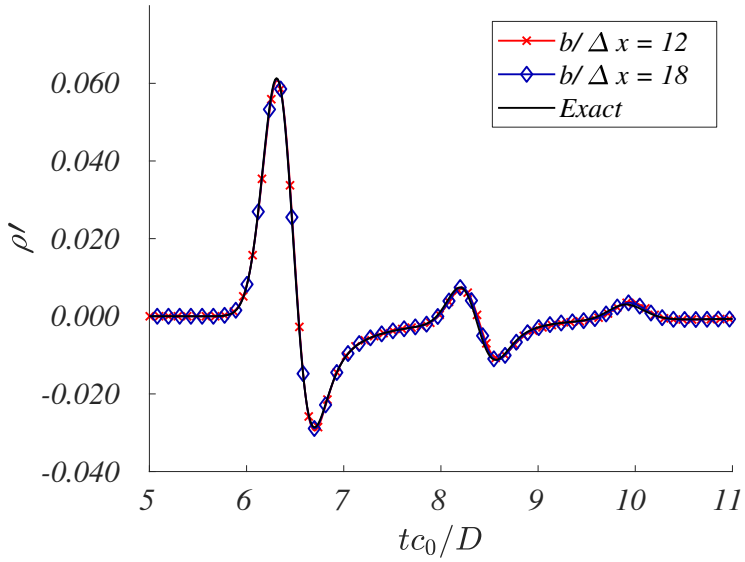


Fig. 4 Fluctuating density ρ' at probe location $(x, y) = (0, 10R)$ for dimensionless time $\tau = tc_0/D$.

C. Data normalization and energy-correction

Neural networks benefit from scaling the inputs so that the mean and the standard deviation of their inputs are close to 0 and 1 respectively [22], in order to improve the stability of the neural network training. However, scaling all the inputs with their statistical moments generates a loss of physical information for the neural network, most importantly in time-dependent problems, as the network may no longer capture the relative changes of amplitude between different time-steps, accounting for phenomena such as diffusion or in the present case, energy decay (e.g. the acoustic signals leave the computational domain after some time). Thus, two types of input normalizations are proposed in this work, in order to evaluate the performance of both strategies. Furthermore, an a-posteriori correction based on the conservation of acoustic energy is also presented to decrease the error accumulation over time of the learned model.

Global normalization : The first normalization, used in a U-Net network dubbed as **GlobUnet**, used a global scaling to normalized all inputs by $1/\varepsilon$ where ϵ is the initial amplitude of one Gaussian pulse, such that:

$$\tilde{\rho}_i = \frac{\rho_i - \rho_0}{\epsilon} \tag{13}$$

This scaling intends to preserve the scale of the original dataset, which may be important to account for time-decaying phenomena.

Local normalization : The second normalization, denominated **LocUnet**, employs an input-to-input scaling. In particular, each input is normalized by $\sigma(\rho^{i-k}$ where σ is the standard deviation of first frame for each input:

$$\tilde{\rho}_i = \frac{\rho_i - \rho_0}{\sigma(\rho_{i-k})} \tag{14}$$

Because four frames ($k = 3$) are used at each input, the network is presented with information about amplitude decay between all the different snapshots, although more locally than for the global normalization. The disadvantage of this method is that some of the network inputs at late simulations time (e.g. when the acoustic signal has mostly left the computational domain) may have a small amplitude compared with the initial time-steps, with greater amplitude. Thus, the network may struggle when processing such kind of low-signal inputs, and focus only on high-energy ones. In order to study such claims, both normalizations are employed for all test cases of Sections IV and V.

Energy preserving correction: In [5], authors found the the network predictions could be improved by using an *a posteriori* acoustic energy-preserving correction, tailored to the boundary conditions of that study, namely computational domains closed by reflecting walls. The acoustic intensity fluxes being nil, the acoustic energy is preserved over time in the inviscid case with no mean flow, thus a spatially uniform correction ε yielded:

$$\epsilon = P_0 - \hat{P}_i \tag{15}$$

where

$$\begin{aligned} \hat{P}_i &= \frac{\int_\Omega \hat{\rho}_i dx}{\int_\Omega dx} \\ P_0 &= \frac{\int_\Omega \rho_0 dx}{\int_\Omega dx}. \end{aligned} \tag{16}$$

and $\hat{\rho}(t)$ is the prediction of neural network at time t.

In the case of radiative boundaries (e.g. numerically treated as non-reflecting boundary conditions), the previous assumptions no longer holds, as the acoustic intensity fluxes may no longer be nil, as shown by the acoustic energy equation in conservation form:

$$\frac{d}{dt}\int_\Omega E dx + \int_{\partial\Omega} \mathbf{I}\cdot\mathbf{n} d\sigma = 0 \tag{17}$$

where E represents the acoustic energy density and $\mathbf{I}$ the acoustic intensity:

$$\begin{aligned} E &= \frac{p^2}{2\rho_0 c_0^2} + \frac{\rho_0 v^2}{2} \\ \mathbf{I} &= p\mathbf{v}. \end{aligned} \tag{18}$$

Here, a modified version of the energy correction for open domain cases is discussed. Here $dE/dt \simeq 0$, and thus the correction yields:

$$\epsilon = \hat{P}_i - \hat{P}_{i+1} \tag{19}$$

where

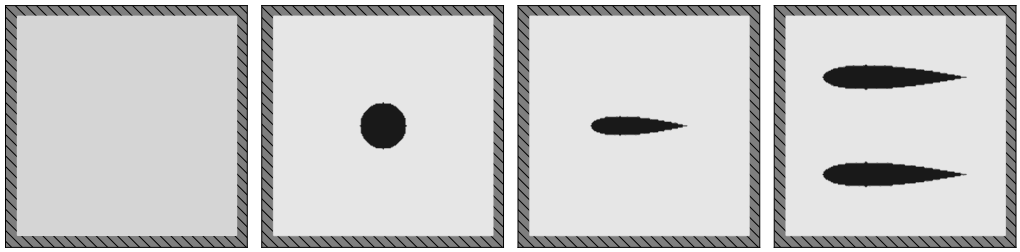
$$\hat{P}_i = \frac{\int_\Omega \hat{\rho}_i dx}{\int_\Omega dx}. \tag{20}$$

This correction only remains valid for small time steps or low frequency acoustic sources.

In total, four strategies are tested by training the presented U-Net CNN on the dataset $D1$, and using the two normalization strategies described above as well as the energy preserving correction. The four models are summarised in table 1.

Name	Dataset	Normalization	EPC
GlobUnetD1	D1	ϵ	No
GlobUnetD1+C	D1	ϵ	Yes
LocUnetD1	D1	σ	No
LocUnetD1+C	D1	σ	Yes

Table 1 Summary of the different strategies for testing the neural network



(a) **(b)** **(c)** **(d)**

Fig. 5 Schematic view of test setups, for cases of (a) free-field propagation, (b) scattering with cylinder, (c) scattering with NACA0012 airfoil and (d) proapgation inside nacelle and scattering into far field. Gray hashed zones represent non-reflecting boundary conditions, while black zones represent reflecting waves.

D. Training the neural network

The supervised training of the neural network G is performed over the dataset composed of N_s pairs of input-target samples $(X^{i+1}, \rho^{i+1}))_{\leq i \leq N_s}$.

Equation (3) is solved by defining the loss function $\mathcal{L}$ used to train the neural network, which is computed at the output of G, and reads:

$$\mathcal{L} = \frac{1}{N} \sum_{k=1}^{N} ||\hat{\rho}_k^{t+1}, \rho_k^{t+1}||_2^2 \tag{21}$$

where the summation is done over the total number N of data samples of prediction-reference pairs.

The loss minimizes the L2-norm mean-square error (MSE) of the prediction $\hat{\rho_k^{t+1}}$ with respect to the reference field ρ_k^{t+1}.

In practice, the optimization of the neural network is performed using a variant of the classical mini-batch stochastic gradient descent algorithm, the Adam optimizer [23], with an initial learning rate of 4.10^{-4} and a learning rate schedule which decreases its value by 15% each time the network validation loss reaches a plateau for more than 10 optimizer passes on the full dataset (called epochs). The Pytorch open-sourced framework [24] is employed to perform the training operations. The algorithm is stopped when the loss converges to a plateau for 100 consecutive epochs, which is achieved after a total of 1500 epochs. The total training time in a NVIDIA Tesla V100 GPU takes 48 wall-clock hours, with a batch size of 32.

IV. Density-field prediction in unbounded test cases

The autoregressive methodology is tested in a series of unbounded cases to demonstrate the ability of the framework to predict acoustic propagation for several configurations with a quiescent mean flow. The objective is to demonstrate that a single neural network, trained on a dataset such as the one described in section III, can perform accurately on all the previous test scenarios, without the need to build a tailored database for each case. The interest of such approach is to avoid the repetition of the costly training process for each new configuration, and to demonstrate that the employed neural network is able to perform well on new cases not seen during the training process.

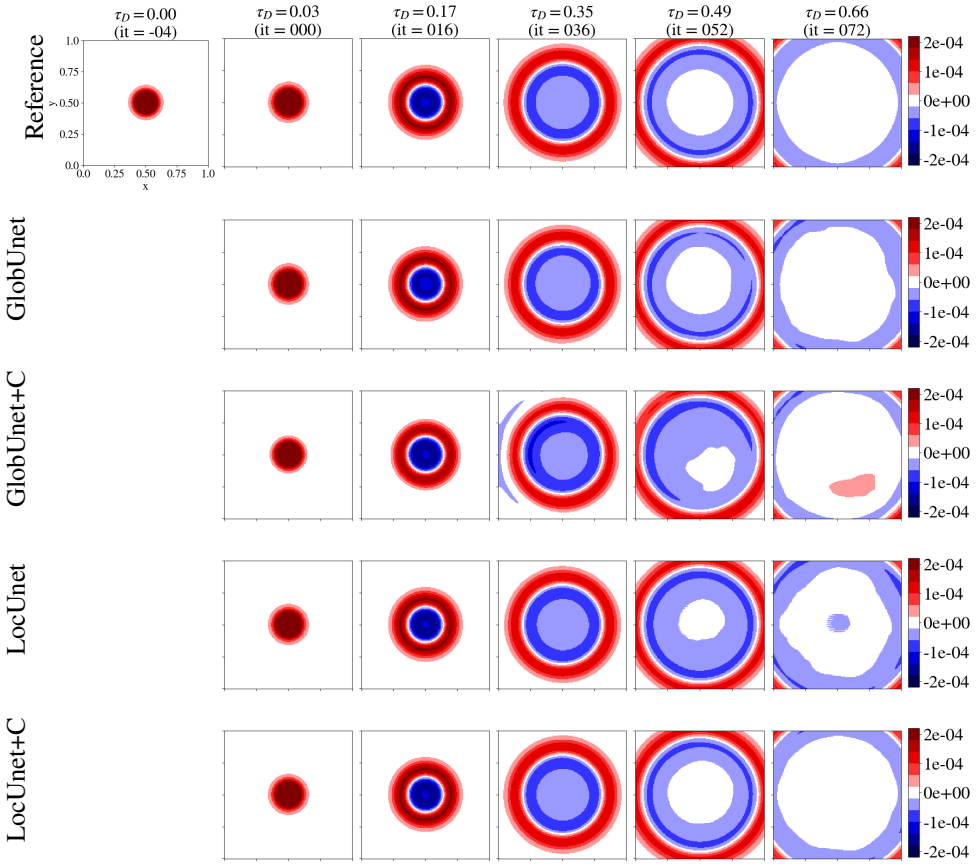


Fig. 6 Acoustic density prediction for test case 1, for the propagation of a Gaussian pulse in free-field.

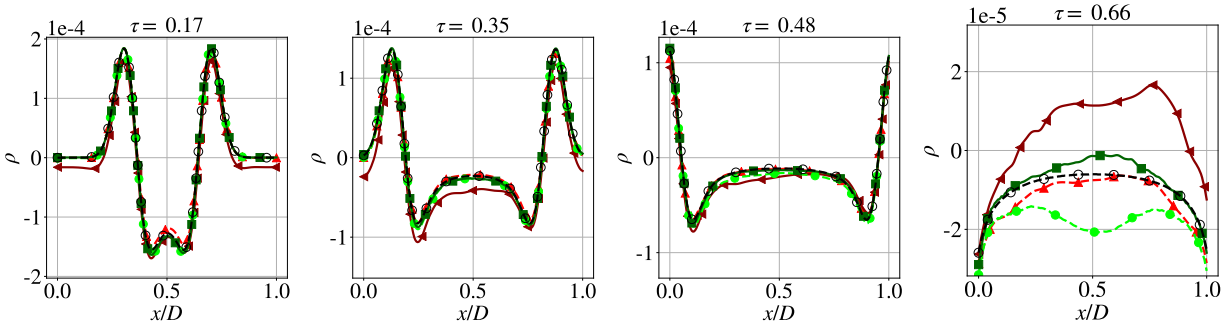


Fig. 7 Slices of density fields at $y/D = 0.5$, for different times and networks: (–○–) LBM reference, (–▲–) GlobUnet, (–◀–) GlobUnet+C, (–●–) LocUnet, and (–■–) LocUnet+C. The scale changes between time-steps for improved visualization.

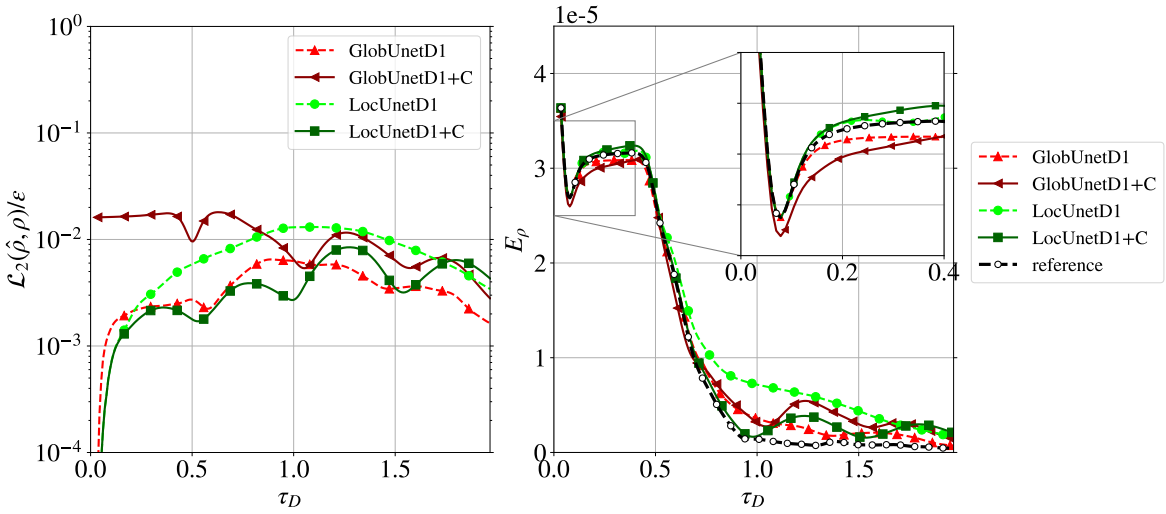


Fig. 8 Evolution of (left) mean-square error over time and (right) energy evolution for test case 1 (free-field propagation).

A. Free-field propagation

First, the network is tested on the free-field propagation case, for a domain of size $D \times D$ without any obstacles and with non-reflecting boundary conditions (Fig. 5a). The initial condition corresponds to a Gaussian pulse of half-width $b/D = 0.06$, located in the center of the domain. The neural network is provided with the first four frames ($k = 3$) of the simulation, issued from a LBM simulation with the same boundary and initial conditions, and the auto-regressive method unrolls subsequently the complete prediction, until the non-dimensional time $\tau = 2$ is reached.

Snapshots at several times are shown in Fig. 6, for the four studied networks and compared with the reference LBM method. Results show that all neural network perform well during the first phase of the propagation, before the pulse reaches the domain outer boundaries. Once the pulse interacts with the non-reflective boundary, some networks (GlobUnet and LocUnet without correction) seem to struggle keeping the symmetry of the pulse while LocUnet with correction manages to keep this symmetry much better, even when interacting with corners.

More quantitative results are shown in Fig. 7, where a slice at constant $y/D = 0.5$ is extracted. The energy preserving correction seems to benefit the LocUnet method (green curves) whereas it hampers the GlobUnet one (red curves). This can be attributed to the lack of uniform drift in the GlobUnet case, thus the hypothesis for correction by some constant value no longer holds (see section **??**). However, in the LocUnet case, such a correction seems to improve clearly the accuracy of the results. This can been for example in Fig. 8, where the time evolution of two global metrics is shown, namely the evolution of the Mean Squared Error $\mathcal{L}_2$ (relative to the initial pulse amplitude ε) and the evolution of the acoustic pressure energy over time. The use of the energy preserving correction for the local-normalized U-Net manages to mitigate the error accumulation over time, typical of auto-regressive methods. However it does the contrary for the globally scaled U-Net, suggesting that no error drift is occurs for such model. In fact, the uncorrected version of GlobUnet behaves very similarly in terms of L2 error evolution than the corrected version of the LocUnet method. Furthermore, it can be observed from the energy plot that for the uncorrected version, both networks show similar trends in the constant energy regime ($\tau < 0.5$) corresponding to times before the pulse reaches the domain boundaries. However, the sudden change of regime towards an energy-decaying one after $\tau = 0.5$ seems to be better captured by the global normalization. The addition of the EPC makes the local normalization the best performing method, both in terms of error and energy fit.

This first test highlights one of the key challenges of such data-driven methods: the change from energy-preserving to energy-decaying state with very low residual levels in the domain and decreasing signal-to-noise levels, even though the physics behavior remains identical (i.e. the propagation of waves in free-field).

B. Scattering by a cylinder

Next, as in the validation case of the dataset in section III.B, the scattering of a Gaussian pulse by a cylinder is studied. The computational domain is sketched in Fig. 5b: the initial condition is an initial Gaussian pulse of half-width

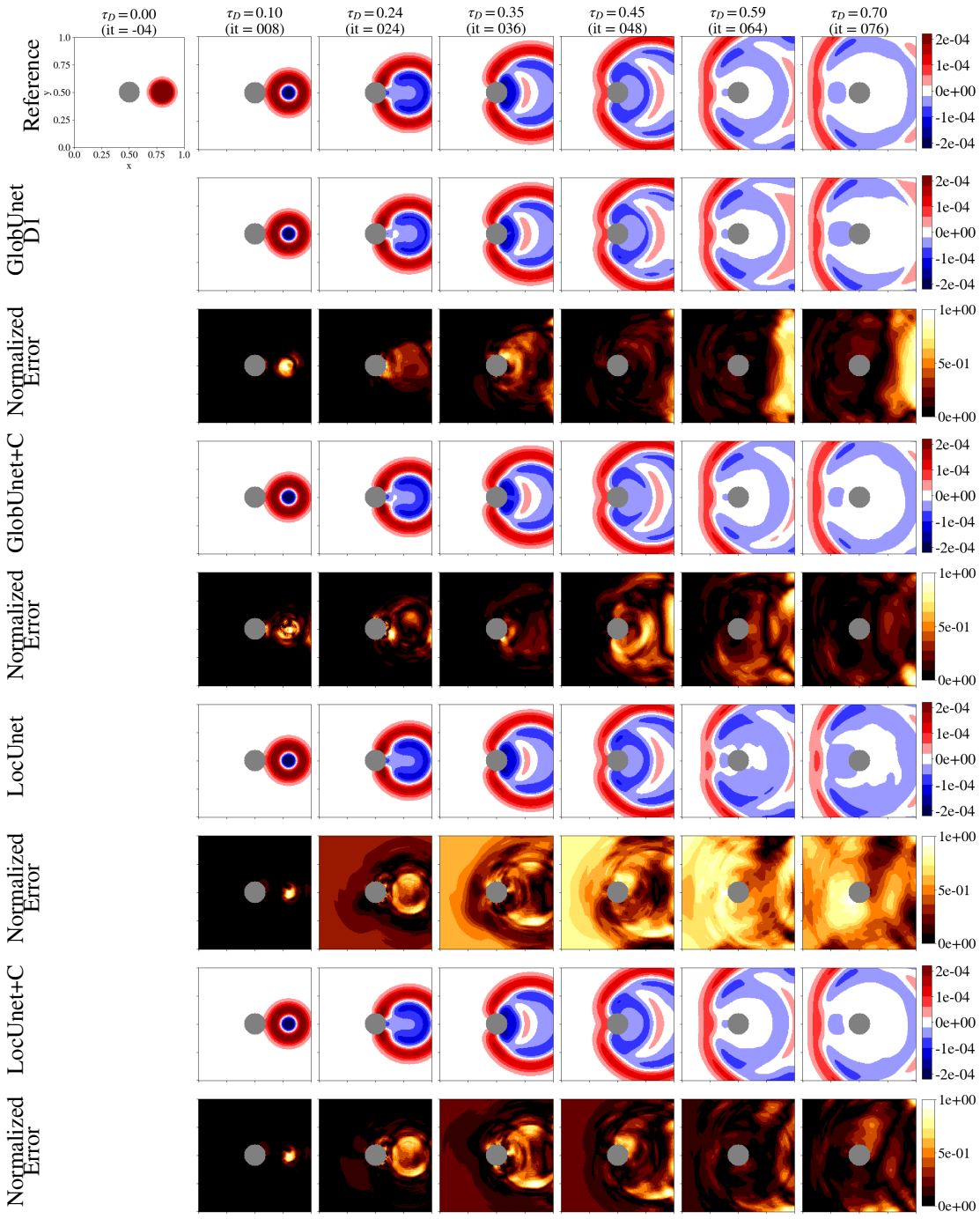


Fig. 9 Acoustic density prediction for test case 2 with cylindrical obstacles for different times, compared with reference LBM target (first row) and normalized spatial errors.

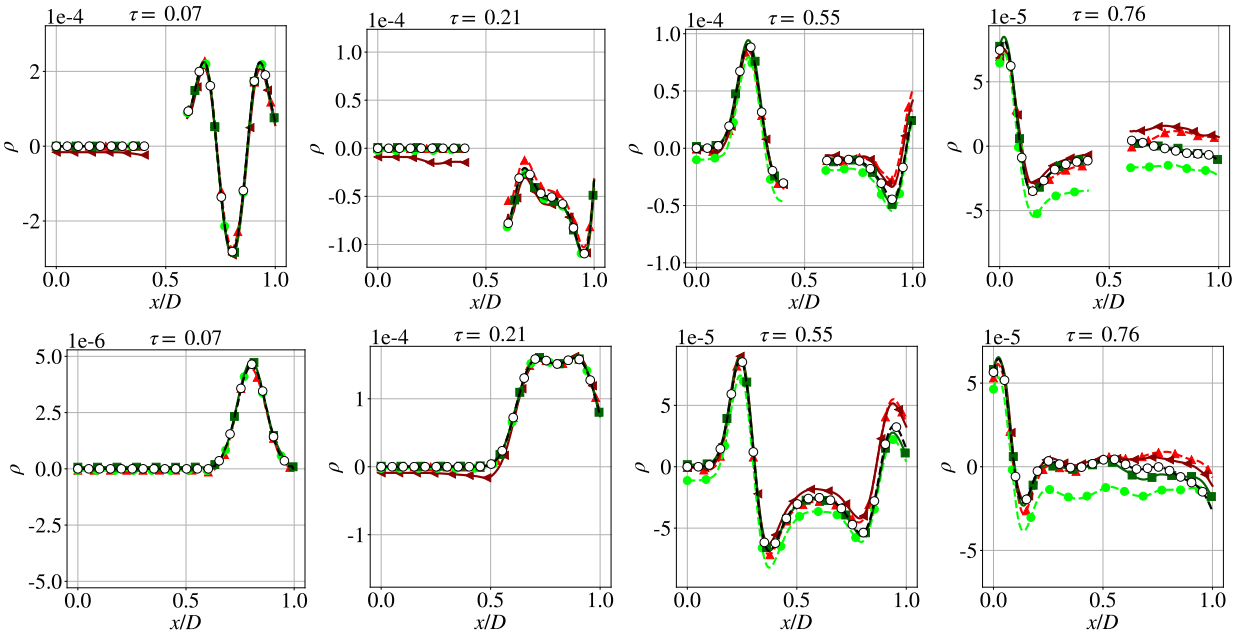


Fig. 10 Slice of density fields at different times (top) at $y/D = 0.5$ and (bottom) $y/D = 0.25$ for networks: (–○–) LBM reference, (–▲–) GlobUnet, (–◀–) GlobUnet+C, (–●–) LocUnet and, (–■–) LocUnet+C for test case 2.

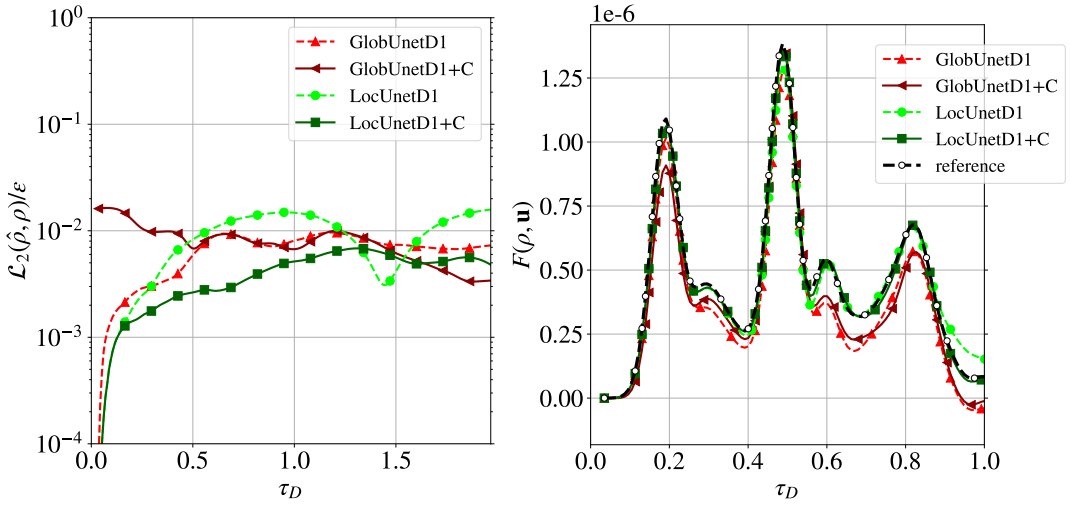


Fig. 11 Evolution of (left) mean-square error over time and (right) evolution of boundary fluxes for test case 2 (scattering with cylinder).

$b/D = 0.06$, located at a distance $x/D = 0.25$ from the cylinder of radius $R/D = 0.075$, located in the center of the domain, while the four boundaries are left as non-reflecting ones.

Results for such a test-case are shown in Fig. 9. Density fields of both the reference LBM simulation and the neural network predictions are compared, along with the local error normalized by the L_∞ norm of the error, in order to show where the error is concentrated. The challenge for the neural network is to correctly predict the scattering of the pulse by the cylinder, and also to correctly model the non-reflecting boundary conditions, as already discussed for the free-field propagation case.

Slices of density fields at two positions $y/D = 0.5$ and $y/D = 0.25$ are shown in Fig. 10, as well as evolution of two global quantities in Fig. 11, namely the mean-square error between the LBM target and the network predictions, and the temporal evolution of the total fluxes integrated over the four domain outer boundaries over time, namely:

$$F(\rho, \boldsymbol{u}) = \iint_{\partial\Omega} \rho \boldsymbol{u} dS \tag{22}$$

In order to calculate such fluxes, the acoustic velocity has been inferred through the linearized conservation of the Euler equation, which yields:

$$\boldsymbol{u}(t) = \int_0^t \nabla p(s) ds \tag{23}$$

It can be seen that for all four tested networks, a similar behaviour is observed as in the previous test: the error grows initially next to the cylinder wall, while at later times the regions of high error are concentrated near the boundaries. The LocUnet+C (i.e. normalized by the standard deviation and with energy preserving correction) achieves a lower error than the rest of methods. Here the GlobUnet encounters more difficulties, typically for times after $\tau = 0.5$. In particular, it can be seen that the network does not manage to correctly predict the absorption by the rightmost domain boundary at $x/D = 1$ of the wave reflected by the cylinder. The error is locally higher in that region near the boundary and the EPC is unable to correct such a behavior (e.g. for $\tau = 0.76$). On the contrary, the energy-corrected LocUnet manages to closely follow the signal oscillations even for such long time horizons. The evolution of the fluxes in time confirms this observation: the GlobUnet strategy only manages to capture the high amplitude fluxes at early simulation times, but fails for the lower amplitude peaks at $\tau \simeq 0.6$ and $\tau \simeq 0.8$. This suggests that the global scaling forces the network to focus on high-amplitude signals, while lower amplitude ones have been neglected during the training process. Thus, it demonstrates the superiority of the local normalization, which can adapt to varying signal amplitudes. In combination with the energy correction, this local normalization is able to capture accurately the energy-decaying dynamics.

C. Scattering with NACA0012 airfoil

The third test case corresponds to the scattering of a Gaussian pulse wave with a NACA0012 airfoil. The computational domain is sketched in Fig. 5c. The airfoil leading edge is located at coordinate $(x, y) = (0.3D, 0.5D)$ while the chord and thickness are $0.4D$ and $0.08D$ respectively. The initial acoustic source is a Gaussian pulse located at $(x, y) = (0.5D, 0.8D)$, above the airfoil trailing edge. This is the first test case where the network encounters a geometry which was not present in the database. The purpose is to study the ability of the network to predict the strong scattering typical of sharp edges, as in the airfoil trailing edge. Since the corrected GlobUnet and the uncorrected LocUnet have been found ineffective on the two previous cases, only the two relevant networks (GlobUnet and LocUnet+C) are shown in the following.

Results in Figures 12 and 13 show that for low times of propagation ($\tau < 0.4$) the error concentrates at the trailing edge region: as seen for example in the slices for $\tau = 0.28$ and $\tau = 0.35$ both GlobUnet and LocUnet+C do not fit closely the density levels at the trailing edge ($x/D = 0.7$), indicating the difficulty of the network to handle sharp edges, which it was not trained for. The leading edge is on the contrary well predicted. This offset in density creates some dispersion phenomena visible in the error fields of Fig. 12: the back-scattered wave from the trailing edge is not in phase with the reference simulation. Interestingly, the prediction error near the sharp trailing edge has a pattern similar to a dipolar source term. Note that the GlobUnet strategies exhibits artifacts close to the trailing edge, even at early times (e.g. $\tau = 0.10$). A better understanding of this localized error and possible corrections are left for future work.

D. Scattering by a 2D Nacelle

In order to demonstrate the capabilities of the presented framework in an industrial-relevant case, a simplified duct-nacelle configuration is tested, as shown in Fig. 5d. Each side of the nacelle is modeled as a symmetric NACA0012

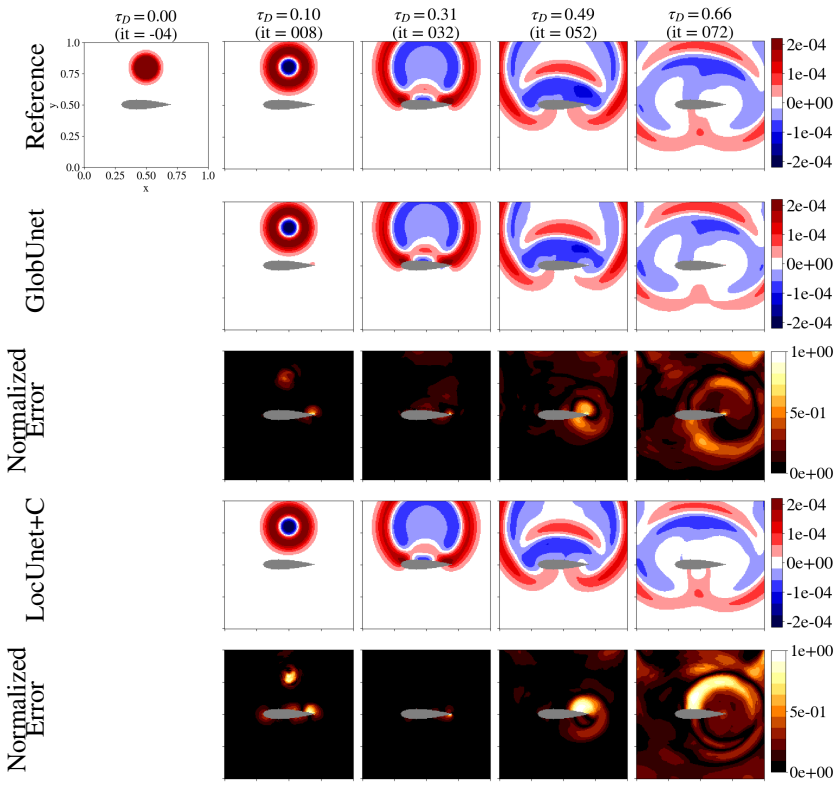


Fig. 12 Acoustic density prediction for test case 3 (scattering with NACA0012) profile.

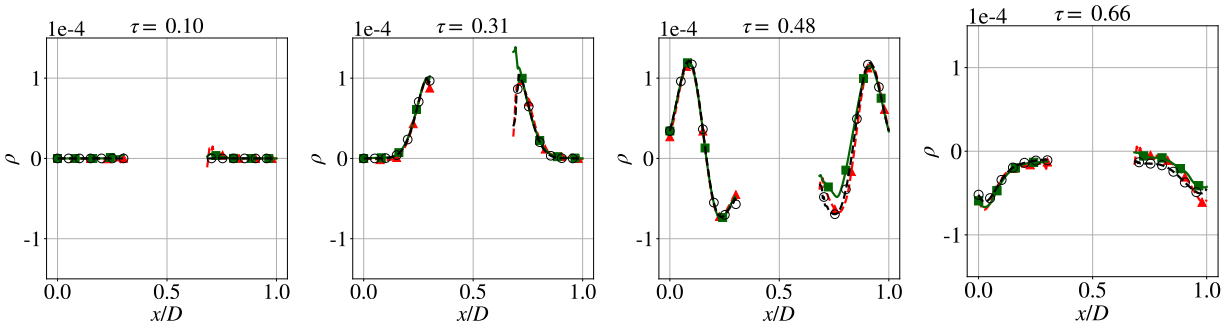


Fig. 13 Slices of density fields at $y/D = 0.5$ for test case 3, for different times and networks (–○–) LBM reference, (–▲–) GlobUnet, and (–■–) LocUnet+C.

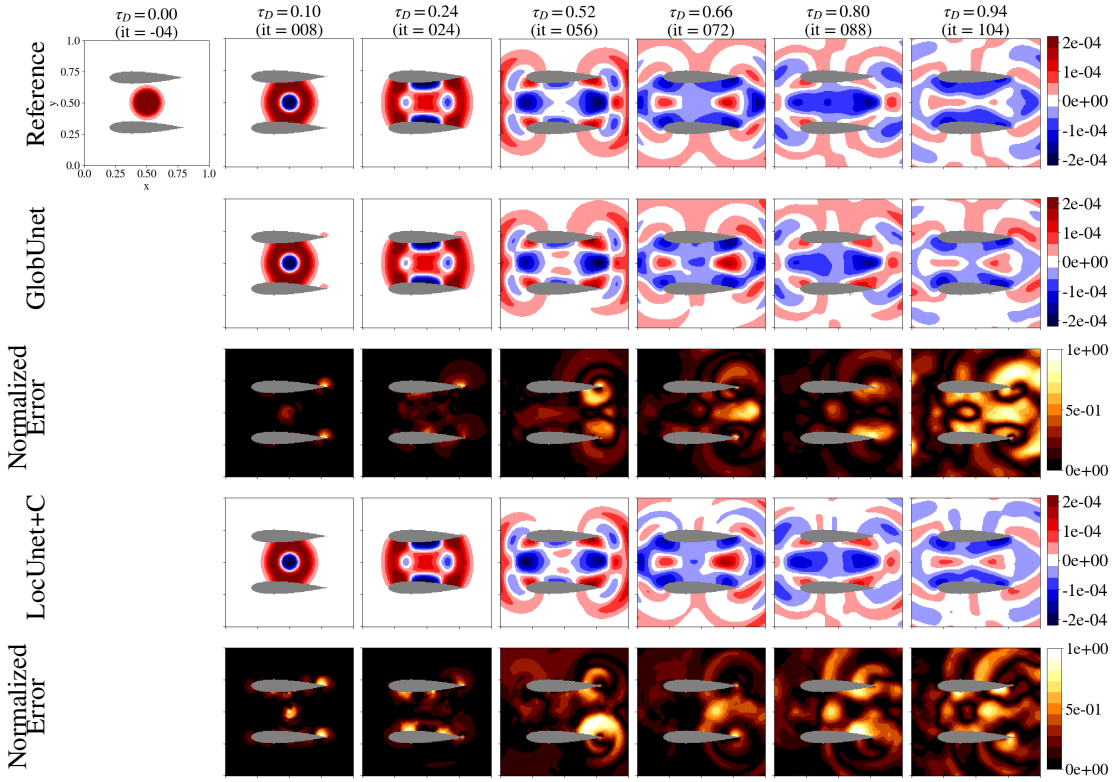


Fig. 14 Acoustic density prediction for test case 4 with two NACA0012 airfoils forming a nacelle.

profile at zero angle of attack with respect to the horizontal direction. Both airfoil leading edges are located at coordinates $(x, y) = (0.2D, 0.3D)$ and $(x, y) = (0.2D, 0.7D)$ while the chord and thickness are set at $0.6D$ and $0.1D$ respectively.

A Gaussian pulse source is initially located inside the duct at $(x, y) = (0.5D, 0.5D)$, and the neural network is unrolled to propagate the wave through the duct and into the far-field.

This case contains most of the physics studied in this work: wave reflection, duct acoustics and scattering of waves at both ends of the duct. Results shown in Fig. 14 reveal the complex patterns that appear in such a case: axial-traveling waves propagate both upstream and downstream of the duct, while transverse waves remains trapped inside the duct, forcing the appearance of duct-like modes. Both GlobUnet and LocUnet+C manage to predict accurately the density patterns appearing in such a scenario. A slice along the axis of the nacelle ($y/D = 0.5$) (Fig. 15) shows the overall fitting of the LBM reference, even after the trapped modes become established after few iterations ($\tau > 0.24$).

However, as was seen in section IV.C, the scattering by both trailing edges creates a local error accumulation, which creates a phase shift in the subsequent propagation. Again, early artifacts appear for GlobUnet at the sharp trailing edges (e.g. $\tau = 0.1$), which then propagate and pollute the rest of the time predictions. This can be seen in the plot of density. After the in-duct modes haven been established ($\tau > 0.5$), this phase shift becomes clear.

Furthermore, the evolution of acoustic energy and fluxes in the domain, shown in Fig. 16, evidences the presence of this in-duct modes through the periodic oscillation of energy and fluxes. Both neural networks show signs of phase shifts as they do not follow the oscillatory patterns after $\tau > 0.5$. However, LocUnetD1+C manages to follow the global trend of oscillating energy, while the GlobUnet diverges faster from the reference simulation.

Thus, this test highlights that the studied method is able to accurately predict complex acoustic propagation cases which differ significantly from the training database. It also shows two of the main difficulties encountered by the surrogate model: the scattering with sharp edges and the difficulty to predict in-duct propagation, because of the presence of trapped modes. The scattering seems to be the cause of the in-duct propagation discrepancy as it generates dispersion errors.

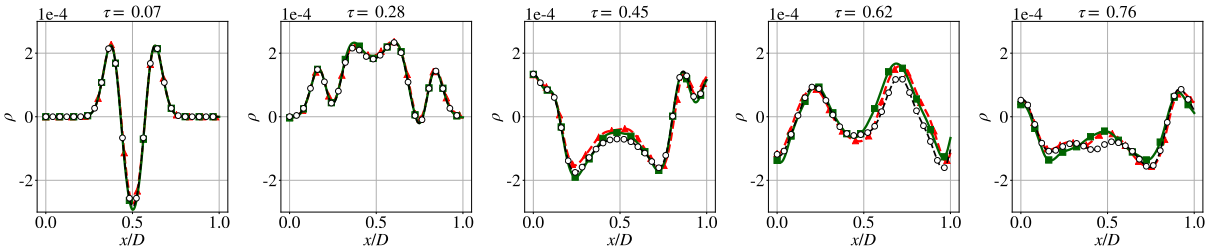


Fig. 15 Slices of density fields at $y/D = 0.5$ for test case 4, for different times and networks (–○–) LBM reference, (–▲–) GlobUnet, and (–■–) LocUnet+C.

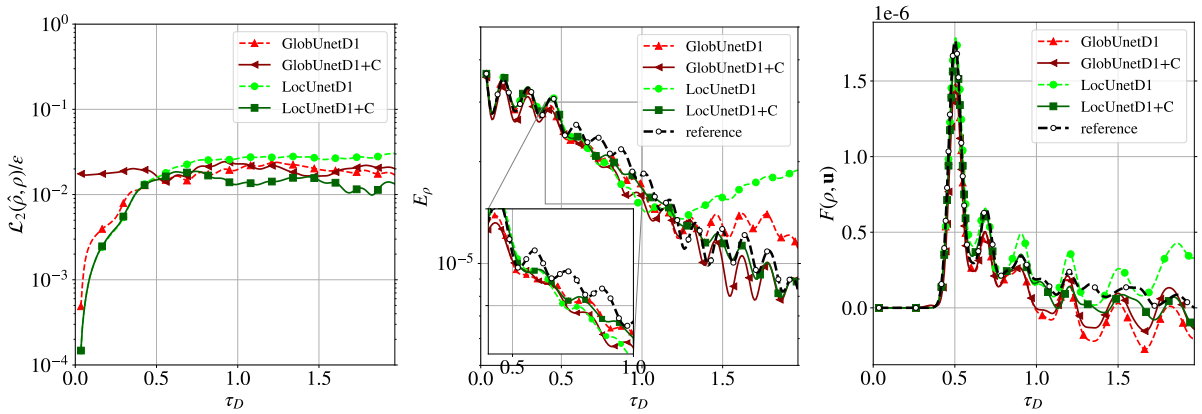


Fig. 16 Time evolution of (left) mean-square error over time, (center) acoustic energy and (right) boundary fluxes for test case 4 (nacelle).

V. Acoustic prediction in bounded domain

A. In-duct propagation

In order to complete the observations made in the previous case regarding the difficulty of neural networks to tackle the prediction of in-duct wave propagation, a fifth test case is presented, where the propagation of a Gaussian pulse inside a two-dimensional duct of constant diameter is considered. It allows the study of the long-time propagation by neural network of trapped modes, without taking into account difficulties due to sharp edges

Three configurations are studied as shown in Fig. 17. The size of the domain is kept constant at $D \times D$ while 3 diameter h of the duct are investigated: $h/D = 0.4, 0.6$ and 0.8.

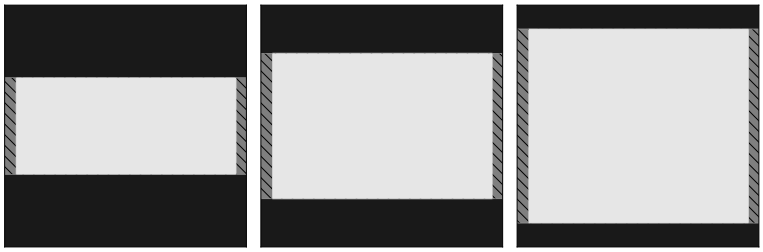


Fig. 17 Schematic view of test setups, for cases in-duct wave propagation for different diameter-to-domain length ratios h/D = (a) 0.4, (b) 0.6, (c) 0.8. . Gray hashed zones represent non-reflecting boundary conditions, while black zones represent reflecting waves.

Figure 18 compares the results for both GlobUnet and LocUnet+C which were identified as the best performing methods in the previous tests. The initial bouncing on the duct walls is seen to be well predicted ($\tau = 0.38$). However, once the wavefronts reach simultaneously the four edges of the simulation, where the non-reflecting condition coincides with the duct wall, the error accumulates quickly in that region. This behavior is clearly seen in the GlobUnet case, which creates some spurious back-reflected waves that pollute the entire domain. While this phenomenon is less present in the LocUnet+C case, it it still visible. This mismatch creates a dispersion error that can be observed in the energy evolution for the three studied cases (Fig. 19). In the three cases, the LocUnet+C method demonstrates its superiority as it manages to track the energy evolution much closer than the globally-scaled network. However, the phase dispersion is also visible in both cases, which is an effect of the error created by the edges of the simulation. These can be attributed to the fact that such a type of junction between non-reflecting and wall boundary conditions is never seen during training. Thus it is hard for the network to infer the behavior in such points. Here, it is hypothesised that even if the duct walls are of infinite length in the axial direction, the network has always learned on finite-length obstacles. Thus, it may be possible that it sees the duct-end as an open-end, thus creating some reflected waves because of the wave scattering at a duct opening. In order to mitigate this behavior, the dataset could be augmented with some examples of infinite length ducts.

Furthermore, the analysis of the energy change over time in Fig. 19 reveals that the GlobUnet is not robust with respect to an increase of the duct aspect-ratio. For values $h/D > 0.6$, such a normalization does not seem capable of following the changes of acoustic energy over time. On the other hand, the LocUnet+C method fits much closer the reference data, although the dispersion phenomena is clearly visible in such results.

B. Closed box propagation

The last test case corresponds to the same type of boundary condition studied in [6], namely a hard reflecting wall on the four boundaries of the computational domain. The initial condition corresponds to a Gausian pulse centered at $(x, y) = (0.5D, 0.5D)$. In practice, the neural network is given a geometry mask with one continuous layer of "obstacle" pixels around the computational domain. Results in Fig. 20 show the evolution of the neural network predictions for both tested networks (GlobUnet and LocUnet+C). In both cases, both networks manage to reproduce the highly symmetric patterns until times around $\tau = 0.6$. For later times, a slight phase shift, originating again from the corners, unbalances the symmetry in both cases. As the energy content of this problem remains constant, it is a hard problem for networks that have only seen decaying problems in the training database. The hypothesis is made that the network may need further temporal context, i.e. having access to longer histories of previous snapshots at the network input, or training on a database of longer simulations, in order to accurately capture the long temporal dependencies of the problem. The latter is tested by training GlobUnet on a new dataset **D2**, identical to D1 except that the LBM simulations are stopped at later times ($\tau^{D2}_{\text{training}} = 1.44$ instead of $\tau^{D1}_{\text{training}} = 0.87$). Results in Fig. 20 show that this new training is able to improve significantly the results for this particular test case. However, all the previous tests were performed with this longer dataset but no improvement was found for the other cases. The explanantion for such a phenomena remains an open question.

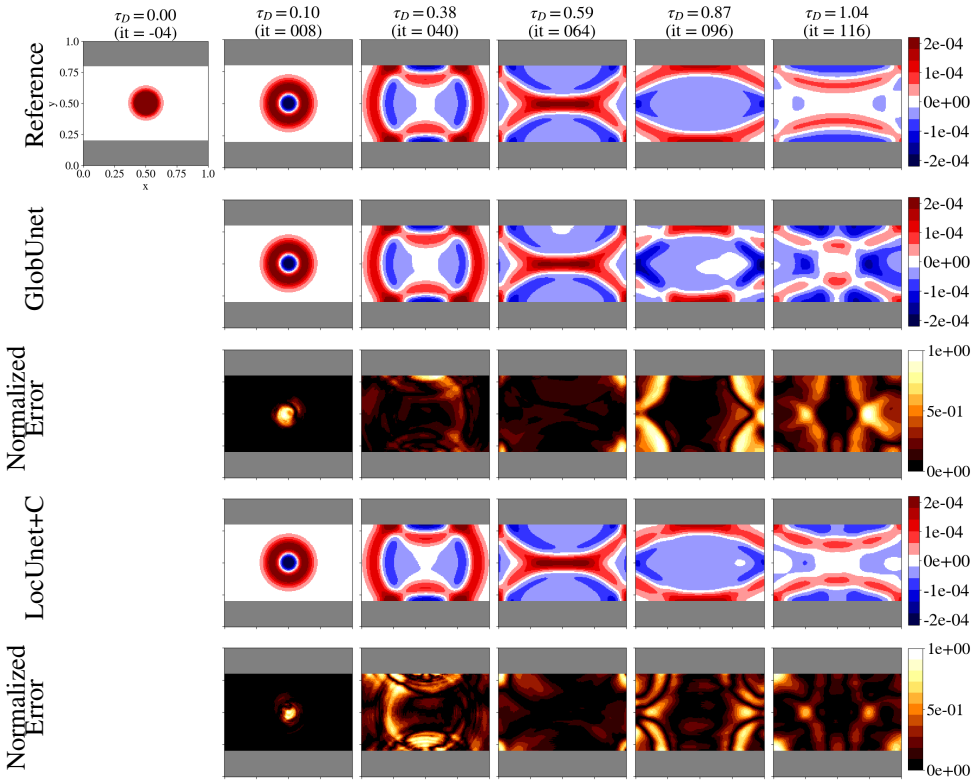


Fig. 18 Acoustic density prediction for test case 5, of in-duct wave propagation with $h/D = 0.6$.

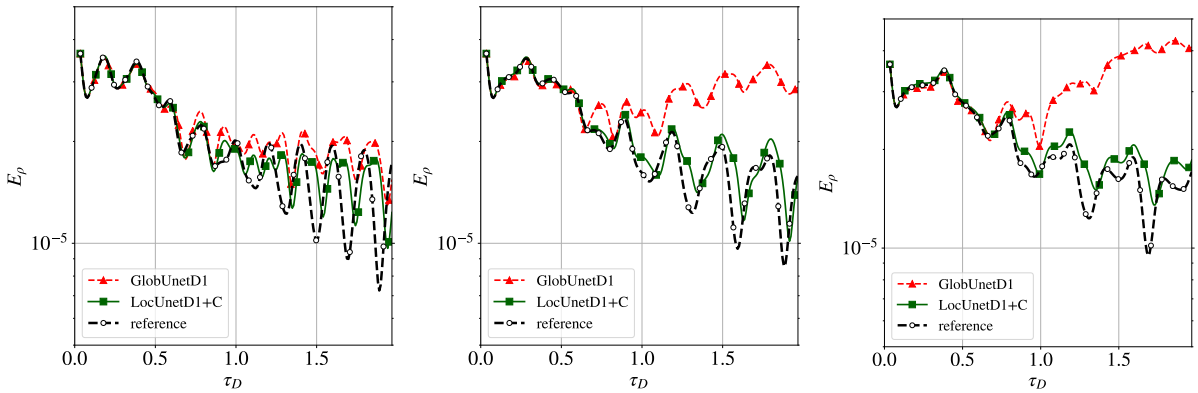


Fig. 19 Time evolution of acoustic energy for cases (left) $h/D = 0.4$, (center) $h/D = 0.6$ (right) $h/D = 0.8$ for test case 5 (in-duct propagation).

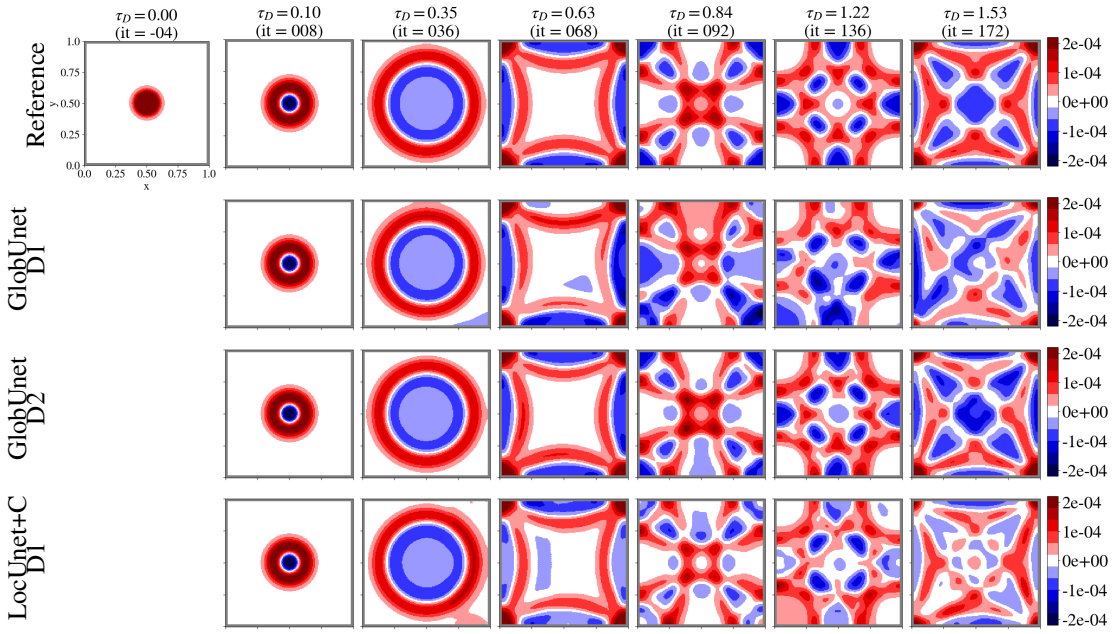


Fig. 20 Acoustic density prediction for test case 6, of closed-box wave propagation.

VI. Conclusion

In this work, a method to treat a variety of acoustic propagation cases on quiescent media with a deep learning surrogate model is presented. The neural networks are trained on a single database of high-fidelity LBM simulations which contains examples of acoustic wave propagation, reflection and scattering with obstacles, as well as free-field propagation. Multiple test cases demonstrate that a convolutional neural network trained on such a database, with some specific normalization (namely local normalization by the input standard deviation) manages to reproduce closely the results of the LBM reference. Accurate results are obtained even on boundary conditions and scattering configurations not seen during the learning phase. The combination of a locally-normalized training with an a posteriori correction, which imposes the conservation of acoustic energy, significantly improves the results and reduces the long-term error accumulation of the autoregressive method. The different examples also highlight some of the challenges encountered by the network, that are also typical of traditional numerical solvers: the accurate treatment of non-reflective boundary conditions, the scattering of acoustic waves by sharp edges and the prediction of duct modes in confined configurations. It also demonstrates that these highly efficient non-linear networks may be capable of extrapolating to changes in the underlying statistical data distribution (e.g. energy-decaying versus energy-conserving flows).

While neural network techniques for fluid dynamics and aeroacoustics are in their early phases of development, studying their capability in more complex cases remains a crucial step for their applicability to real world problems. The insights gained in this work suggests that besides the studied problem of treating the temporal integration accuracy [7, 25], care should be taken when modeling problems with complex boundary conditions. Thus, techniques from the CFD community could be employed to improve the capabilities of such data driven methods (e.g. non-reflective conditions etc).

VII. Acknowledgments

The authors wish to thank Wagner Gonçalves Pinto, from ISAE-Supaero for fruitful discussion on the topic. This work was partly supported by the French "Programme d'Investissements d'avenir" ANR-17-EURE- 0005. We acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC) through the Discovery Grant program. M. Bauerheim is also supportted by the French Direction générale de l'armement (DGA) through the AID POLA3 project. The simulations were carried out using the CALMIP supercomputing facilities (project P20035), France and Compute Canada clusters.

A. U-Net convolutional network

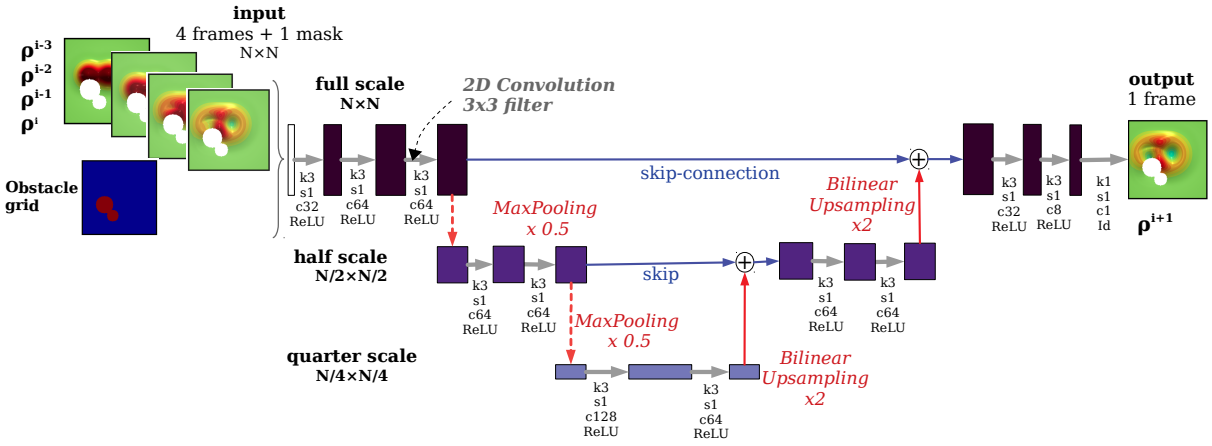


Fig. 21 Details of the employed U-Net [14] convolutional architecture.

Figure 21 shows the details of operations. The notation is follows the following convention

- $k\#$ represents the size of the 2D convolutional kernel. For instance, $k3$ is a 3×3 kernel, with 9 learnable parameters.
- $s\#$ denotes the stride of the convolution operator.
- $c\#$ is the number of output channels after each convolution.
- The non-linear activation function: either "ReLU" for Rectifying Linear Unit, or the idendity function ("Id").

References

[1] Lele, S. K., and Nichols, J. W., "A second golden age of aeroacoustics?" Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, Vol. 372, No. 2022, 2014. https://doi.org/10.1098/rsta.2013.0321.

[2] Lilley, G. M., "On the noise from jets. Noise Mechanism," AGARD-CP-131, 1974, pp. 13.1–13.12.

[3] Ffowcs Williams, J. E., and Hawkings, D. L., "Sound generation by turbulence and surfaces in arbitrary motion," Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, Vol. 264, No. 1151, 1969, pp. 321, 342. https://doi.org/10.1098/rsta.1969.0031.

[4] Posson, H., and Peake, N., "The acoustic analogy in an annular duct with swirling mean flow," Journal of Fluid Mechanics, Vol. 726, 2013, pp. 439–475. https://doi.org/10.1017/jfm.2013.210.

[5] Alguacil, A., Bauerheim, M., Jacob, M. C., and Moreau, S., "Predicting the propagation of acoustic waves using deep convolutional neural networks," Journal of Sound and Vibration, 2021, pp. 116285 (Accepted, in press). https://doi.org/10.1016/j.jsv.2021.116285.

[6] Alguacil, A., Bauerheim, M., Jacob, M. C., and Moreau, S., "Predicting the Propagation of Acoustic Waves using Deep Convolutional Neural Networks," AIAA Aviation Forum, Reston, VA, 2020, p. 2513. https://doi.org/10.2514/6.2020-2513.

[7] Geneva, N., and Zabaras, N., "Modeling the Dynamics of PDE Systems with Physics-Constrained Deep Auto-Regressive Networks," Journal of Computational Physics, Vol. 403, 2019. https://doi.org/10.1016/j.jcp.2019.109056, URL http://arxiv.org/abs/1906.05747.

[8] Thuerey, N., Weißenow, K., Prantl, L., and Hu, X., "Deep Learning Methods for Reynolds-Averaged Navier–Stokes Simulations of Airfoil Flows," AIAA Journal, Vol. 58, No. 1, 2020, pp. 25–36. https://doi.org/10.2514/1.j058291.

[9] Rüttgers, M., Koh, S. R., Jitsev, J., Schröder, W., and Lintermann, A., "Prediction of Acoustic Fields Using a Lattice-Boltzmann Method and Deep Learning," Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Vol. 12321 LNCS, Springer Science and Business Media Deutschland GmbH, 2020, pp. 81–101. https://doi.org/10.1007/978-3-030-59851-8_6, URL https://doi.org/10.1007/978-3-030-59851-8{_}6.

[10] Lino, M., Cantwell, C., Fotiadis, S., Pignatelli, E., and Bharath, A., "Simulating Surface Wave Dynamics with Convolutional Networks," arXiv:2012.00718, 2020. URL http://arxiv.org/abs/2012.00718.

[11] Raissi, M., Perdikaris, P., and Karniadakis, G. E., "Physics Informed Deep Learning (Part I): Data-driven Solutions of Nonlinear Partial Differential Equations," arXiv:1711.10561, 2017.

[12] Moseley, B., Markham, A., and Nissen-Meyer, T., "Solving the wave equation with physics-informed deep learning," arXiv:2006.11894v1, 2020.

[13] Liu, Y., Nathan Kutz, J., and Brunton, S. L., "Hierarchical Deep Learning of Multiscale Differential Equation Time-Steppers," arXiv preprint arXiv:2008.09768, 2020. URL https://github.com/luckystarufo/multiscale{_}HiTS.

[14] Ronneberger, O., Fischer, P., and Brox, T., "U-Net: Convolutional Networks for Biomedical Image Segmentation," Medical Image Computing and Computer-Assisted Intervention (MICCAI), Vol. 9351, 2015, pp. 234–241. https://doi.org/10.1007/978-3-319-24574-4-28.

[15] Luo, W., Li, Y., Urtasun, R., and Zemel, R., "Understanding the Effective Receptive Field in Deep Convolutional Neural Networks," Advances in Neural Information Processing Systems 29, Barcelona, Spain, 2016, pp. 4898–4906.

[16] Ajuria Illarramendi, E., Alguacil, A., Bauerheim, M., Misdariis, A., Cuenot, B., and Benazera, E., "Towards an hybrid computational strategy based on Deep Learning for incompressible flows," AIAA Aviation Forum, American Institute of Aeronautics and Astronautics (AIAA), Reston, Virginia, 2020, p. 3058. https://doi.org/10.2514/6.2020-3058.

[17] Brogi, F., Malaspinas, O., Chopard, B., and Bonadonna, C., "Hermite regularization of the lattice Boltzmann method for open source computational aeroacoustics," The Journal of the Acoustical Society of America, Vol. 142, No. 4, 2017, pp. 2332–2345. https://doi.org/10.1121/1.5006900.

[18] Marié, S., Ricot, D., and Sagaut, P., "Comparison between lattice Boltzmann method and Navier-Stokes high order schemes for computational aeroacoustics," Journal of Computational Physics, Vol. 228, 2008, pp. 1056–1070. https://doi.org/10.1016/j.jcp.2008.10.021.

[19] LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D., "Backpropagation Applied to Handwritten Zip Code Recognition," Neural Computation, Vol. 1, No. 4, 1989, pp. 541–551. https://doi.org/10.1162/neco.1989.1.4.541.

[20] Malaspinas, O., "Increasing stability and accuracy of the lattice Boltzmann scheme: recursivity and regularization," arXiv:1505.06900, 2015. URL https://arxiv.org/abs/1505.06900.

[21] Tam, C. K., Computational aeroacoustics: A wave number approach, Cambridge Aerospace Series, Cambridge: Cambridge University Press, 2012. https://doi.org/10.1017/CBO9780511802065.

[22] Goodfellow, I., Bengio, Y., and Courville, A., Deep Learning, MIT Press, 2016.

[23] Kingma, D. P., and Ba, J. L., "Adam: A method for stochastic optimization," 3rd International Conference on Learning Representations - ICLR, 2015. URL https://hdl.handle.net/11245/1.505367.

[24] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., Devito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S., "PyTorch: An Imperative Style, High-Performance Deep Learning Library," Advances in Neural Information Processing Systems 32, Vancouver, Canada, 2019, pp. 8024–8035. URL https://papers.nips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.

[25] Azencot, O., Benjamin Erichson, N., Lin, V., and Mahoney, M. W., "Forecasting sequential data using consistent koopman autoencoders," Proceedings of the 37th International Conference on Machine Learning, Virtual Event, 2020, pp. 119:475–485.